



UNIVERSIDAD TÉCNICA DEL NORTE

**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN MANTENIMIENTO
AUTOMOTRIZ**

**TEMA: PREDICCIÓN DE EMISIONES INSTANTÁNEAS USANDO
MACHINE LEARNING PARA LA ESTIMACIÓN PRECISA DE
GASES CO₂, A PARTIR DE LAS VARIABLES DE OPERACIÓN DE
UN VEHÍCULO LIVIANO.**

AUTORES:

HENRY DAVID CACUANGO TÚQUERREZ

LEONARDO ANDRÉS CAIZA IRUA

DIRECTOR: ING. Víctor Javier Montenegro Simancas MSc.

Ibarra, 2023

CERTIFICADO

ACEPTACIÓN DEL DIRECTOR

En mi calidad de director del plan de trabajo de grado, previo a la obtención del título de Ingeniería en Mantenimiento Automotriz, nombrado por el Honorable Consejo Directivo de la Facultad de Ingeniería en Ciencias Aplicadas.

CERTIFICO:

Que una vez analizado el plan de grado cuyo título es "Predicción de emisiones instantáneas usando machine learning para la estimación precisa de gases CO₂ a partir de las variables de operación de un vehículo liviano" presentado por el señor(es): Henry David Cacuango Túquerrez con número de cédula 1003769427 y Leonardo Andrés Caiza Irua con número de cédula 0401806401 doy fe que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a presentación pública y evaluación por parte de los señores integrantes del jurado examinador que se designe.

Atentamente

Victor Montenegro S.

ING. VÍCTOR JAVIER MONTENEGRO SIMANCAS MSC.
DIRECTOR DEL TRABAJO DE GRADO



**UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA**

**AUTORIZACIÓN DE USO Y PUBLICACIÓN
A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL
NORTE**

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA IDENTIDAD:	DE	0401806401	
APELLIDOS NOMBRES:	Y	Caiza Irua Leonardo Andrés	
DIRECCIÓN:	El Olivo		
EMAIL:	lacaizai@utn.edu.ec		
TELÉFONO FIJO:		TELÉFONO MÓVIL:	0998702524
CÉDULA IDENTIDAD:	DE	1003769427	
APELLIDOS NOMBRES:	Y	Cacuango Túquerrez Henry David	
DIRECCIÓN:	Caranqui		
EMAIL:	hdcacuangot@utn.edu.ec		
TELÉFONO FIJO:	062-651-772	TELÉFONO MÓVIL:	0961215754

DATOS DE LA OBRA	
TÍTULO:	Predicción de emisiones instantáneas usando machine learning para la estimación precisa de gases CO ₂ a partir de las variables de operación de un vehículo liviano.
AUTOR (ES):	Leonardo Andrés Caiza Irua, Henry David Cacuango Túquerrez
FECHA: DD/MM/AAAA	14 de diciembre del 2023
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TÍTULO POR EL QUE OPTA:	INGENIERÍA EN MANTENIMIENTO AUTOMOTRIZ
ASESOR /DIRECTOR:	Ing. Víctor Javier Montenegro Simancas MSc.

2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 14 días del mes de diciembre 2023

AUTOR (ES):



Leonardo Andrés Caiza Irua

0401806401



Henry David Cacuango Túquerrez

1003769427

DEDICATORIA

El presente proyecto de titulación se la dedico a mis padres María Elena y Juan quienes con tanto esfuerzo y paciencia me motivaron a conseguir este tan anhelado sueño. A mis hermanos Geovanny, Lenin quienes son en inspiración y ejemplo, los cuales de hoy en adelante me formo como un profesional como ellos, a mi hermana Gissela quien espero sea mucho mejor que todos nosotros en su vida personal como profesional.

Dedico también a familiares y amigos que de cierta forma estuvieron presentes en mi vida tanto al inicio como ahora en la culminación de esta carrera universitaria, siendo estas influentes en mis decisiones, actos y sobre todo en mi personalidad que de hoy en adelante se formó como profesional.

Henry Cacuango

Este proyecto a recorrido largos caminos, pero siempre demostrará la constancia no solo de los autores sino de las familias que los acompañaron en todo momento, sin más que agregar le dedico este esfuerzo a mis padres que nunca me abandonaron y que su corazón a demostrado una gran fortaleza para observar la culminación de una carrera universitaria.

Leonardo Caiza

AGRADECIMIENTO

Primeramente, como persona católica agradezco a Dios por todo este tiempo, por la vida, por la experiencia y por las bendiciones que en todo el transcurso de esta etapa universitaria se ha hecho presentes en mí y en toda mi familia.

Agradecer a mi padre Juan Elías Cacuango quien con su carácter me ha enseñado que, con trabajo duro, dedicación y sobre todo tener gusto por lo que se hace se puede conseguir muchas cosas, a mi madre María Elena Túquerrez, por ser quien ha formado mi carácter, humildad, responsabilidad y la forma de ver las cosas, para poder desenvolverme en esta nueva etapa de la vida tanto personal como profesional.

También quiero agradecer a mi compañero de proyecto Leonardo Caiza quien a pesar de todo este tiempo ha demostrado su dedicación y paciencia para poder culminar y lograr este tan anhelado título que por distintas razones nos ha llevado a formar una gran amistad en todo este proceso.

Agradeciendo también a la Universidad técnica del norte por todo lo aprendido en esta etapa, de igual manera a los docentes quienes impartieron sus conocimientos para formarnos como profesionales y poder ser personas óptimas para la sociedad.

Henry Cacuango

Se dice que los logros de una persona le pertenecen solo a esa persona; nada más alejado de la verdad, pues detrás de cada uno de los esfuerzos que he realizado se lo debo a la gran perseverancia y amor que me dio mi bisabuela Rosa que a pesar de no estar ya a mi lado nunca deja de cuidarme y protegerme en cada paso y decisión que doy; la fortaleza y el cariño de mi madre, el coraje de mi padre y su comprensión han sido la inspiración de todas mis metas propuestas.

Mi agradecimiento también está dirigido a mis amigos que me han apoyado con sus locuras y consejos; a mi compañero de tesis el cual no solo forjamos un equipo sino una amistad y algún día ser colegas de trabajo, en esencia mis logros no son solo míos, sino de las personas que han estado a lo largo de mi vida.

Leonardo Caiza

ÍNDICE

	PÁGINA
RESÚMEN	xv
ABSTRACT	xvi
INTRODUCCIÓN	xvii
CAPÍTULO I	1
1. REVISIÓN BIBLIOGRÁFICA	1
1.1 Antecedentes	1
1.2 Situación Actual	2
1.3 Planteamiento Del Problema	3
1.4 Objetivos	4
1.4.1 Objetivo General	4
1.4.2 Objetivos Específicos	4
1.5 Alcance	4
1.6 Justificación	5
1.7 Emisiones	6
1.7.1 Contaminación Atmosférica	6
1.7.2 Gases Contaminantes	7
1.7.3 Gases De Efecto Invernadero (GEI)	8
1.7.4 Dióxido De Carbono (CO ₂)	9
1.7.4.1 Co ₂ Impacto Del Transporte	10
1.7.5 Óxido Nitroso (N ₂ O)	11
1.7.6 Metano (CH ₄)	11
1.8 Límites Estrablecidos De Emisiones CO ₂ Vehículos Livianos	12
1.9 Machine Learning	13
1.9.1 Aplicaciones Y Funcionamiento De Machine Learning	14
1.9.2 Tipos De Algoritmos De Machine Learning	14
1.9.2.1 Aprendizaje Supervisado	15
1.9.2.2 Aprendizaje No Supervisado	15
1.9.2.3 Aprendizaje Por Refuerzo	15
1.10 Estimación Del Consumo De Combustible Y Emisiones CO ₂	16

1.11 Normativa Rde (Real Driving Emissions)	17
1.11.1 Requisitos De La Normativa Rde	17
CAPÍTULO II	19
2. MATERIALES Y MÉTODOS	19
2.1 Metodología De La Investigación	19
2.1.1 Enfoque Investigativo	19
2.1.2 Tipos De Investigación	20
2.1.2.1 Investigación De Análisis Comparativo	20
2.1.2.2 Investigación De Análisis Estadístico	20
2.1.2.3 Investigación Tecnológica	20
2.2 Materiales & Equipos	21
2.2.1 Datalogger (Canedge2)	21
2.2.2 CanmodGPS	22
2.2.3. Vehículo De Pruebas	23
2.3 Métodos	25
2.3.1 Uso De Una Ruta Según Normativa RDE	26
2.3.1.1 Prueba De Conducción En Zona Urbana	27
2.3.1.2 Prueba De Conducción En Zona Rural	28
2.3.1.3 Prueba De Conducción En Zona De Autopista	29
2.3.2 Obtención De Datos	31
2.3.2.1 Cálculo Del Flujo Másico Del Aire (MAF)	36
2.3.2.2 Cálculo Del Consumo De Combustible En Ruta Rde	38
2.3.3 Cálculo De La Estimación De Emisiones CO ₂ En Ruta Rde	39
2.3.4 Desarrollo De Modelos De Predicción	40
2.3.4.1 Importación De Librerías	41
2.3.4.2 Importación De Datos De La Ruta Rde	43
2.3.4.3 División De Datos A Variables Dependientes E Independientes	44
2.3.4.4 Correlación De Los Datos Ingresados	45
2.3.4.5 Entrenamiento De Los Modelos De Predicción	49
2.3.4.6 Entrenamiento De Regresión Lineal Simple	50
2.3.4.7 Entrenamiento De Redes Neuronales Multicapa	51
2.3.4.8 Entrenamiento De Árbol De Decisión Regressor	54
2.3.4.9 Entrenamiento De Randomforest Regressor	55
2.3.4.10 Entrenamiento De Máquinas De Vectores De Soporte Regresión	56
2.3.5 PREDICCIÓN DE LOS MODELOS	57
2.3.5.1 Predicción De La Regresión Lineal Simple	57
2.3.5.2 Predicción De La Red Neuronal Multicapa	58
2.3.5.3 Predicción Del Árbol De Decisión Regressor	59
2.3.5.4 Predicción De Randomforest (Bosque Aleatorio)	60

2.3.5.5 Predicción De Máquina De Vectores De Soporte	60
CAPÍTULO III	62
3. RESULTADOS Y DISCUSIÓN	62
3.1 Estimación De La Emisión Co ₂ Y Consumo De Combustible	62
3.1.1 Consumo De Combustible En La Zona Urbana	63
3.1.2 Consumo De Combustible En La Zona Rural	63
3.1.3 Consumo De Combustible En La Zona De Autopista	63
3.1.4 Estimación De La Emisión Co ₂ En Ruta Zona Urbana	64
3.1.5 Estimación De La Emisión Co ₂ En Ruta Zona Rural	64
3.1.6 Estimación De La Emisión Co ₂ En Ruta Zona De Autopista	65
3.2 Resultados Del Modelo Regresión Lineal Simple	65
3.3 Resultados Del Modelo Redes Neuronales Multicapa	68
3.4 Resultados Del Modelo Árbol De Decisión De Regresión	72
3.5 Resultados Del Modelo Random Forest Regressor	74
3.6 Resultados Del Modelo Svm De Regresión	76
3.7 Resultados Obtenidos En Cada Modelo De Predicción	78
3.7.1 Comparación De Los Valores De Predicción Del Vehículo	79
3.7.2 Predicción De Emisiones Usando Diversos Vehículos	80
3.7.2.1 Resultados De Modelos De Predicción Vs Valores Teóricos	82
3.8 Velocidad Vs Consumo De Combustible & Emisión Co ₂	84
3.8.1 Velocidad Vs C.C. & FECO ₂ De Ruta Zona Urbana(Inst. 400-520seg)	84
3.8.2 Velocidad Vs C.C. & FECO ₂ De Ruta Zona Rural (Inst.5314-5484seg)	85
3.8.3 Velocidad Vs C.C. & FECO ₂ De Ruta Zona Autopista (Inst.6050-6264seg)	86
CAPÍTULO IV	87
4. CONCLUSIONES & RECOMENDACIONES	87
4.1 Conclusiones	87
4.2 Recomendaciones	88
Bibliografía	89
Anexos	94

ÍNDICE DE FIGURAS

FIGURA NUM.	PÁGINA
1.1 Gases y Compuestos de Efecto Invernadero	9
1.2 Comparación de CO ₂ año 2012 vs objetivos para el periodo 2015-2022	13
2.1 Datalogger CSS electronics CANedge2	21
2.2 GPS CANmodGPS	23
2.3 Vehículo Cherry Arrizo 5 año 2022	24
2.4 Metodología para la estimación CO ₂ y Desarrollo de Modelo de Predicción	25
2.5 Perfiles de velocidad, tiempo y altitud según la normativa RDE	26
2.6 Zona Urbana	28
2.7 Zona Rural	29
2.8 Zona de Autopista	30
2.9 Ruta desarrollada bajo la normativa RDE	31
2.10 Flujograma de Obtención de PID'S	32
2.11 Datos sin procesar obtenidos del vehículo	33
2.12 Decodificadores del programa Asammdf	34
2.13 Página para la obtención de Decodificadores	35
2.14 PID'S decodificados obtenidos del vehículo	35
2.15 Importación de librerías	42
2.16 Importación de datos de la Ruta RDE	43
2.17 Conteo de datos de la Ruta RDE	44
2.18 Selección de variables dependientes e independientes	44
2.19 Identificación de Datos relevantes	45
2.20 Correlación de datos relevantes	45
2.21 Comportamiento y dispersión de CO ₂ vs Consumo Combustible	46
2.22 Mapa de Calor y coeficientes de correlación de los datos obtenidos	47
2.23 Coeficientes de correlación del mapa del calor de las variables	48
2.24 División de datos Entrenamiento/Test	50
2.25 Entrenamiento de Regresión Lineal	50
2.26 Estructura de la Red Neuronal Artificial Multicapa	52
2.27 Compilación de la Red Neuronal Multicapa	53
2.28 Entrenamiento de la Red Neuronal Multicapa	54

2.29	Entrenamiento de Árbol de Decisión	55
2.30	Entrenamiento del modelo RandomForest	56
2.31	Entrenamiento de Máquina de Vectores de Soporte	57
2.32	Predicción de Regresión Lineal en la Ruta RDE	58
2.33	Predicción de la Red Neuronal Multicapa	58
2.34	Predicción del Árbol de Decisión Regressor	59
2.35	Predicción del modelo Random Forest	60
2.36	Predicción de Máquina de Vectores de Soporte	61
3.1	Factor de Emisión y Consumo de Combustible Cherry Arrizo 5 Pro	62
3.2	Coefficiente de determinación de Regresión Lineal Simple	66
3.3	Comparación de los valores obtenidos y la muestra registrada	66
3.4	Gráfico de comparación de datos obtenidos	67
3.5	Errores presentados en el modelo Regresión Lineal Simple	67
3.6	Ejecución de predicción con valores aleatorios	68
3.7	Coefficiente de determinación de Red Neuronal Multicapa	69
3.8	Validación de la Red Neuronal Multicapa	69
3.9	Pérdidas y Entrenamiento a lo largo de las Epochs	70
3.10	Magnitud de Pérdidas de la Red Neuronal Multicapa	70
3.11	Errores presentados en la Red Neuronal Multicapa	71
3.12	Predicción con datos aleatorios Cherry Arrizo 5 Pro 2022	72
3.13	Coefficiente de determinación del Árbol de Decisión	72
3.14	Errores presentados en Árbol de Decisión Regressor	73
3.15	Árbol de Decisión Regressor	73
3.16	Factor de Emisión en Árbol de Decisión Regressor	74
3.17	Coefficiente de determinación de RandomForest Regressor	74
3.18	Errores presentados en RandomForest Regressor	75
3.19	Factor de emisión con RandomForest Regressor	75
3.20	Coefficiente de determinación de SVM Regressor	76
3.21	Errores presentados en SVM Regressor	76
3.22	Gráfico de comparación entre datos registrados vs predicción	77
3.23	Factor de emisión generado por SVRM	78
3.24	Factor de emisión y consumo de combustible de diversos vehículos	81
3.25	Factor de emisión y consumo de combustible del SWM	81

3.26	Estilos de Conducción en el instante 400-520 seg Zona Urbana	84
3.27	Estilos de Conducción en el instante 5314-5484 seg Zona Rural	85
3.28	Estilos de Conducción en el instante 6050-6264 seg Zona Autopista	86

ÍNDICE DE TABLAS

TABLA NUM.	PÁGINA
1.1 Factores de emisión CO ₂ emitido por medios de transporte	10
1.2 Reducción de emisiones CO ₂ para la flota de vehículos ligeros	12
1.3 Constantes del Consumo de Combustible y Flujo Másico del Aire	16
1.4 Requisitos de la normativa RDE para vehículos livianos	18
2.1 Especificaciones Técnicas Datalogger CANedge2	22
2.2 Datos específicos del vehículo Cherry Arrizo 5 Pro	24
2.3 Puntos de la Ruta RDE en Zona Urbana	27
2.4 Puntos de la Ruta RDE en Zona Rural	29
2.7 Rangos de Correlación en mapas de calor	48
3.1 Comparación de resultados del Consumo de Combustible	64
3.2 Emisión de CO ₂ Producida por Cherry Arrizo 5 Pro	65
3.3 Valores obtenidos en los modelos de predicción	78
3.4 Valores de Predicción de los Modelos en Cherry Arrizo 5 PRO	80
3.5 Valores de Predicción de los Modelos vs Valores Teóricos	83

ÍNDICE DE ANEXOS

ANEXO NUM.	PÁGINA
1. Instalación del equipo Datalogger en la entrada OBDII	95
2. Instalación del equipo GPS	95
3. Datos Técnicos del Equipo DataLogger	96
4. PID'S y Posicionamiento del Vehículo	97
5. PID'S Decodificados y procesados del Vehículo	98
6. Código de Programación Redes Neuronales Artificiales Multicapa RNAM	102
7. Código de Programación Máquina de Vectores de Soporte Regressor SVMR	104
8. FECO ₂ , Consumo Combustible vs Velocidad en Zona Urbana	105
9. FECO ₂ , Consumo Combustible vs Velocidad en Zona Rural	106
10. FECO ₂ , Consumo Combustible vs Velocidad en Zona de Autopista	107

RESÚMEN

La presente investigación realiza un análisis de la emisión del gas de efecto invernadero CO₂ provocada por un vehículo liviano en la ciudad de Ibarra, que ejecutó una ruta específica que consta de una zona urbana, zona rural y zona de autopista; para ello se realizó una recolección de información sobre el aprendizaje de máquina o machine learning y datos de emisiones en el campo automotriz. Esta investigación también utilizó equipos portátiles como Datalogger y un GPS CanEdge2 de la marca CSSELETRONICS que permitió la recolección, el almacenamiento de datos y el posicionamiento en tiempo real del vehículo respectivamente en una ruta que se rige bajo la normativa RDE (Real Driving Emissions) que recorre cumpliendo los requisitos de dicha normativa en un formato MDF4, que posteriormente fueron transformados a un formato principal CSV(Excel) usado como la entrada de datos para la ejecución de los modelos de predicción adecuados. En el desarrollo, compilación y ejecución de los diferentes modelos de predicción se utilizó Google Colaboratory; un servicio de cuaderno que se relaciona a la perfección con el lenguaje de programación Python y RStudio semejante a Jupyter Notebook que posee una interfaz y una configuración mucho más rápida y sencilla de manejo; Google Colab permitió que los usuarios de esta investigación desarrolle un código con una importación de librerías extensa que dio paso a la realización de múltiples modelos relacionados con las regresiones lineales, redes neuronales artificiales, árboles de decisión regressor, bosques aleatorios y máquinas de vectores de soporte regressor debido a la naturaleza de los datos numéricos, por ende, la selección del modelo se ejecutó con el que mejor se acople o interaccione con la base de datos obtenidos en la ruta mencionada. Los resultados obtenidos en la presente investigación se manifestaron 2 modelos de predicción que se acoplaron de forma favorable a la naturaleza de los datos ingresados los cuales fueron la red neuronal artificial multicapa (RNAM) y máquina de vectores de soporte regressor (SVMR) que se encuentran en los rangos de fiabilidad aceptable y que dieron paso al cumplimiento de los objetivos de este proyecto de titulación. No obstante, el uso de los equipos mencionados y los modelos desarrollados pueden efectuarse para investigaciones no solo de la flota de vehículos livianos sino también usarse para los vehículos con motores de encendido por compresión o MEC y usar las señales de los sensores no solo para encontrar emisiones CO₂ sino también, CO, NO_x, HC, etc.

ABSTRACT

The present investigation carries out an analysis of the emission of the greenhouse gas CO₂ caused by a light vehicle in the city of Ibarra, which executed a specific route that consists of an urban area, a rural area, and a highway area. For this purpose, information was collected on machine learning and emissions data in the automotive field.

This research also used portable equipment such as Datalogger and a CanEdge2 GPS from the CSSELETRONICS brand that allowed the collection, storage of data and real-time positioning of the vehicle respectively on a route that is governed by the RDE (Real Driving Emissions) regulations that runs complying with the requirements of said regulations in an MDF4 format, which were subsequently transformed into a main CSV (Excel) format used as data entry for the execution of proper prediction models. In the development, compilation and execution of the different prediction models, Google Colaboratory was used. This is a notebook service that is perfectly related to the Python and RStudio programming language, similar to Jupyter Notebook, which has a much faster and easier to use interface and configuration; Google Colab allowed the users of this research to develop code with an extensive library import that led to the creation of multiple models related to linear regressions, artificial neural networks, regressor decision trees, random forests and support vector machines regressor due to the nature of the numerical data. Therefore, the selection of the model was executed with the one that best fits or interacts with the database obtained in the mentioned route. The results obtained in the present investigation showed 2 prediction models that were favorably coupled to the nature of the data entered, which were the multilayer artificial neural network (MLP) and support vector machine regressor (SVMR) that are found in the ranges of acceptable reliability and that led to the fulfillment of the objectives of this degree project. However, the use of the mentioned equipment and the developed models can be carried out for investigations not only of the light vehicle fleet but also used for vehicles with compression ignition engines or MEC and use the sensor signals not only to find CO₂ emissions but also CO, NO_x, HC, etc.

INTRODUCCIÓN

El presente proyecto de titulación desarrolla un modelo de predicción de emisiones usando machine learning para la estimación de gases CO₂, a partir de las variables de operación de un vehículo liviano; la ciudad de Ibarra posee un 87,7% de emisiones de Dióxido de Carbono provocados por el tráfico vehicular, a pesar de ser una ciudad con menos contaminación a comparación de otras ciudades del país en los últimos años ha tenido un crecimiento del 10,6% anual sobre todo por el sector automotriz. Ecuador posee un combustible con una disminución de octanaje alarmante lo que provoca un desgaste muy prematuro en motores ya sean a Diésel o gasolina que dan como resultado una ineficiencia en los mismos, tomando en cuenta que esto emite gases contaminantes, gases de efecto invernadero y materiales particulados que ocasionan problemas respiratorios y contaminación ambiental.

El presente trabajo de grado se enfoca principalmente en obtener datos de emisiones CO₂ a partir de las variables de operación de un vehículo que realiza una ruta específica que se rige en una normativa RDE en la ciudad de Ibarra que la ejecute como lo exige dicha normativa; con esto desarrollar un modelo de predicción que use herramientas tecnológicas en este caso machine learning, un campo de la Inteligencia Artificial que permite diagnosticar emisiones instantáneas. Esta investigación pretende encajar con las políticas de cuidado del medio ambiente desarrollado por el plan nacional “Toda una Vida”; este plan llevará a cabo propuestas de uso de instrumentos ya sean tecnológicos, de economía y regulación que frenen las emisiones de gases efecto invernadero (GEI).

Esta investigación provee una base de conocimiento con respecto a machine learning o aprendizaje de máquina y el desarrollo de modelos de predicción los cuales para su ejecución se usa equipos portátiles como PEMS, DataLogger y GPS que ayuden al posicionamiento en tiempo real del vehículo, con esto se puede obtener, almacenar y desarrollar una base de datos los cuales serán de suma importancia en el entrenamiento de la máquina y alcanzar los rangos de fiabilidad en los resultados adquiridos; el funcionamiento y los valores obtenidos de los modelos de predicción puedan informar la emisión provocada por un vehículo liviano en condiciones reales.

CAPÍTULO I

1. REVISIÓN BIBLIOGRÁFICA

1.1 ANTECEDENTES

La emisión de gases contaminantes es uno de los problemas actuales que no solo afectan al medio ambiente, sino a la salud del hombre; a través de los años los resultados de estas emisiones se ven reflejados en los diferentes problemas respiratorios y sobre todo en los cambios climáticos que se han presentado en el planeta.

Una de las principales causas de la de emisiones de gases contaminantes se debe a la aparición de la Revolución Industrial, según Eric Howbsbawm, un cambio notorio fue a finales del siglo XVIII y a principios del siglo XIX en Europa Occidental, es decir, entre el año 1750 y 1820 surgió la primera fase de la revolución industrial en Inglaterra, dando un cambio económico y también en la calidad de vida de la humanidad y este cambio provocó el uso de diversos materiales como el hierro, acero y carbón que darían fuentes de energías y fuerza motriz a las primeras máquinas de vapor como trenes y barcos (Hobsbawm, 1962,p.170), por ende, debido a estos inventos de transporte y máquinas de producción fueron los primeros responsables de emitir gases contaminantes a la atmósfera como el CO, SO₂ y CO₂.

En la actualidad el uso del motor de combustión interna tecnología del siglo XIX ha sido el principal protagonista con respecto a emisiones de gases. “Para generar potencia en los motores de combustión interna se transforma la energía química del combustible en energía calórica por medio de la combustión, luego el calor generado es transformado en energía mecánica gracias a la expansión de la mezcla dentro del cilindro” (Bosch, 1996,p.130).

Según la investigación de Gabriel Villalba Pintado una persona puede respirar un volumen aproximado de 12000 litros de aire o 12 m³, el cual se puede asegurar que la calidad no es satisfactoria y este a su vez se encuentra contaminado por los procesos industriales, tráfico de transporte y diferentes actividades (Villalba, 2019,p.3). Debido a esta problemática en

diversas partes del mundo se ha intentado mitigar el tema de contaminación ambiental con la utilización de herramientas tecnológicas que ayuden a la predicción de emisión de gases de efecto invernadero, herramientas como Neural Networks o Machine Learning utilizan redes neuronales que pueden ayudar a generar un modelo de predicción. Machine Learning es una herramienta de predicción que puede utilizar modelos matemáticos utilizando metodología de regresión de parámetros como también utiliza la interacción por medio de redes neuronales.

En Bogotá se utilizó un modelo de predicción viable de la contaminación del aire que utilizó una red neuronal artificial para obtener un modelo computacional que pueda realizar una predicción que se aproxime con respecto a una variable de entorno en este caso la contaminación atmosférica (Pedraza, 2019,p.23).

Los modelos de predicción en Ecuador solo se han aplicado en proyectos de financiación y no en temas de contaminación ambiental. En Ecuador según un análisis de la OMS (Organización Mundial de Salud) 2012-2013 las ciudades como Milagro, Quito, Santo Domingo, Latacunga, Portoviejo han registrado niveles de contaminación muy elevados aproximadamente 33 ug/m^3 ($\text{PM}_{2,5}$) es decir tres veces más que lo recomendado por la OMS (10 ug/m^3), en la ciudad de Ibarra es una de las ciudades menos contaminadas con un valor aproximado de 9 ug/ m^3 que no supera los niveles de contaminación de $\text{PM}_{2,5}$ y se mantiene en un valor de PM_{10} (Sorgato, 2016), por lo cual un modelo de predicción ayudaría a conocer la realidad de las emisiones y en un futuro poder llegar a la mitigación del problema de contaminación ambiental.

En esta ciudad no se cuenta con un modelo de predicción de emisiones de gases; específicamente de CO_2 , lo cual se puede realizar obteniendo estimaciones de emisiones vehiculares gracias al uso de sensores y DataLogger utilizando las variables de operación de un auto como velocidad, aceleración y potencia.

1.2 SITUACIÓN ACTUAL

En la actualidad la reducción de emisiones es un caso de suma importancia ya que no es posible eliminarlo completamente, pero si mitigarlo. En Ecuador en el año 2016 llegaron 65000 unidades que cumplen con la norma Euro 3 pero según el presidente de la asociación de empresas automotrices del Ecuador Genaro Baldeón en el país se cuenta con gasolina de una calidad menor, lo cual comentó que es necesario tener gasolina de un valor mínimo de 95 octanos; el gerente general de Petroecuador Pedro Merizalde comentó que se está realizando un “Plan de mejoramiento de combustible” ya que la calidad de la gasolina depende de dos puntos como la planta donde se procesa y el tipo de crudo que se utiliza, en la refinería de Esmeraldas-Ecuador en los años 70, refinaba crudos que poseían 30 grados API es decir a mayor grado se obtiene mejor calidad con un porcentaje de azufre del 0,6%; en el presente el país se produce crudos pesados que tienen entre 2 y 2,6% de azufre además Merizalde mencionó que existen cambios al colocar un producto que no cumplen con la especificación necesaria para una planta diseñada para una condición establecida (Moreta, 2017,p.2).

1.3 PLANTEAMIENTO DEL PROBLEMA

Según los datos obtenidos por la investigación de Omar Arciniegas el 87,7% de emisiones de CO₂(bióxido de carbono) son resultados del tráfico vehicular en la ciudad de Ibarra específicamente por transporte escolar e institucional, también tomando en cuenta los datos del Ministerio del ambiente, en la ciudad de Ibarra las emisiones son NO_x(óxido nitroso), CO(monóxido de carbono), COV (compuestos orgánicos volátiles) que son ocasionados por el tráfico vehicular (Arciniegas, 2018,p.22).

Al existir normativas de emisiones no existe una línea base que controle los porcentajes en los transportes de la ciudad siendo escaso y extremadamente altos, tomando en cuenta que el país cuenta con un combustible de baja calidad permite que la contaminación ambiental sea de una forma alarmante.

En Ecuador las empresas como Petroecuador no efectúa limitaciones de emisión debido a un cambio en la norma INEN, y como resultado el octanaje de la gasolina súper tuvo una disminución de 92 a 90 octanos y la gasolina extra de 87 a 85 octanos provocando un combustible pobre que da como resultado un desgaste prematuro en los motores, pero sobre

todo la ineficiencia en los mismos y los problemas respiratorios por la emisión de CO₂ y materiales particulados más pesados. A pesar de una contaminación menor con relación a otras ciudades del país, según la investigación de Sonia Nathaly Chiriboga Bustamante las emisiones de CO₂ han tenido un crecimiento alarmante entre el año 2007 al año 2017 del 6.54% en la ciudad de Ibarra teniendo una correlación directa con el sector automotriz que ha tenido un crecimiento del 10.6% por año lo cual también el aumento de la población influye en la emisión de gases contaminantes o efecto invernadero (Chiriboga, 2018,p.37).

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

- Desarrollar un modelo de predicción de emisiones instantáneas usando machine learning para la estimación precisa de gases CO₂, a partir de las variables de operación de un vehículo liviano.

1.4.2 OBJETIVOS ESPECÍFICOS

- Seleccionar una ruta que cumpla con la normativa RDE para el registro de datos usando un vehículo liviano.
- Registrar una muestra de datos mediante los equipos DataLogger, GPS en un vehículo liviano compatible, para el entrenamiento de máquina en la ruta seleccionada.
- Diseñar modelos de predicción de emisión CO₂ usando machine learning mediante redes neuronales y regresiones lineales para la obtención de la fiabilidad de los modelos.

1.5 ALCANCE

En el presente proyecto se va a desarrollar una base de datos la cual contenga información con respecto a las emisiones de gases contaminantes CO₂, con ello se puede determinar la contaminación que genera un vehículo pequeño y cómo afecta al medio ambiente y los

problemas respiratorios provocados al ser humano, estos datos se podrán obtener en un vehículo liviano marca Cherry Arrizo 5 Pro año 2022 que circulará por un determinado sector de la ciudad de Ibarra, lo cual permitirá realizar un modelo de predicción con la herramienta tecnológica machine learning.

Este proyecto tiene como finalidad seguir una ruta específica para el modelo de predicción que deben seguir una normativa RDE en la ciudad de Ibarra, esta ruta se conforma de tres tramos diferentes que son la zona urbana, rural y de autopista que cruce de norte a sur y de este a oeste.

Estos tramos son protagonistas de la recolección de datos para la predicción, con puntos importantes que muestren un tráfico vehicular elevado y sobre todo estos puntos pasan por sectores donde se pueden encontrar instituciones públicas, instituciones de salud, instituciones financieras y con mayor flujo peatonal.

1.6 JUSTIFICACIÓN

Este proyecto tiene como enfoque principal obtener datos de estimaciones de gases CO₂ a partir de las variables de operación de un vehículo liviano que realice una o varias rutas de ensayo, el objetivo principal del proyecto es desarrollar un modelo de predicción que permita pronosticar las emisiones instantáneas en un auto usando herramientas tecnológicas como machine learning que es un campo de la inteligencia artificial donde utiliza las interacciones de redes neuronales.

Esta investigación pretende encajar en las políticas de cuidado del medio ambiente por ende, según la creación nacional de planificación “*Plan de creación de oportunidades 2021-2025*” Ecuador pretende ser un referente regional en la agenda ambiental global proactiva hacia el campo de las relaciones internacionales; es decir, de una manera más directa y sobre todo de forma práctica se llevará a cabo propuestas de instrumentos de economía y regulación que frene las emisiones de gases de efecto invernadero en los países emisores. Según el plan de creación, las políticas del objetivo n°12 que dice “*Fomentar modelos de desarrollo sostenible aplicando medidas de adaptación y mitigación al cambio climático*”; el apéndice 12.1 que promueve “*Fortalecer las acciones de mitigación y*

adaptación al cambio climático” que conjuntamente al apéndice 12.3 que promueve de igual forma la *“Implementación de mejores prácticas ambientales con responsabilidad social y económica”* que fomenten la concientización producción y consumo sostenible desde investigaciones, innovaciones y trasposos a nuevas tecnologías. Es así como las metas a cumplir en el año 2025 según el apéndice 12.3.3 pretende aumentar de 21,6 a 50,5 millones el ahorro de combustible en barriles de petróleo que permita una mejor optimización de generar energía eléctrica y sobre todo la eficiencia de los hidrocarburos. No obstante complementarse con los lineamientos de territorio que desea aplicar esquemas de gestiones integrales de objetos contaminantes como desechos sólidos, descargas de líquidos contaminantes tóxicos en zonas urbanas y rurales del país, ciudad o cantón (Merchán, Lemus, Armijos, Moráles, Poveda, Sulizza, Lasso, Correa, Álava, Herrera, Bermeo, Bernal, Prado, Paredes, Maino, Cabrera, Bravo, Macas, Cueva, Vélez, 2021, p.87, p.89).

Con respecto a lo mencionado anteriormente el presente proyecto proporciona una base de conocimiento con respecto al uso de herramientas tecnológicas y de predicción que en un futuro se pueda implementar estos modelos en las diferentes estaciones de revisión técnica vehicular para controlar y mitigar las emisiones y así poder mejorar la calidad del aire para reducir los problemas de contaminación ambiental y sobre todo los problemas respiratorios que presentan las personas debido a la emisión de gases.

1.7 EMISIONES

Las emisiones se han presentado de forma artificial y natural a lo largo de la historia, son fluidos que se presentan de forma gaseosa, puras o que están asociadas con sustancias en suspensión; no obstante, estas emisiones se dividen en gases contaminantes y gases de efecto invernadero provocando la contaminación atmosférica y de esta manera reduciendo de forma considerable la capa de ozono.

1.7.1 CONTAMINACIÓN ATMOSFÉRICA

En la actualidad la contaminación atmosférica no solo afecta al medio ambiente, sino también a la salud del ser humano, presentándose en problemas respiratorios y cambios climáticos abruptos que perjudican al planeta. Según la investigación de Milagros Galbiatti

comentó sobre lo establecido por Eric Howbsbawm, un cambio significativo que tuvo lugar a finales del siglo XVIII y comienzos del siglo XIX en Europa Occidental entre el año 1750 y 1820, donde comenzó la primera fase de la revolución industrial en Inglaterra, y el resultado tuvo un cambio en el aspecto económico y la calidad de vida de los seres humanos y debido a esta transición se logró utilizar materiales como hierro, acero y carbón que generan energía y fuerzas motrices a las primeras máquinas de vapor como trenes y barcos, por ende, estas fueron las principales responsables de emisiones de gases contaminantes a la atmósfera como CO, SO₂ y CO₂ (Galbiatti, 2013,p.1).

La aparición de la revolución tuvo un protagonismo principal con el tema respecto a emisiones que tuvo consecuencias muy evidentes en el planeta, pero sobre todo en la calidad del aire, que provoca problemas no solo respiratorios, sino también problemas dermatológicos como psoriasis y envejecimiento prematuro en el ser humano.

Además, según la investigación de Castañeda, Céspedes y Madrigal señalan que el aumento de las concentraciones de los gases de efecto invernadero (GEI), especialmente del CO₂, son concentraciones del resultado de la propagación del uso de combustibles fósiles y producción de cemento que han liberado 375 Pg (1Pg=10¹⁵ g) de carbono a la atmósfera (Arteaga, 2017,p.1), por ende, la causa principal de emisiones de CO₂, N₂O y CO son el resultado del transporte de turismo y las motocicletas que emiten principalmente HC (hidrocarburos), tomando en cuenta que los automóviles pequeños y pesados son aquellos que generan en gran aumento material particulado (Ong,Mahlia,Masjuki,2011,p.4).

En conclusión, estas emisiones han provocado que la calidad del aire disminuya debido al aumento continuo de vehículos y de la industrialización que utilizan combustibles fósiles y recursos no renovables, que permiten la fragmentación y disminución de la capa de ozono por lo cual en la actualidad se pretende mitigar este problema con la utilización de herramientas tecnológicas y concientización en las grandes industrias, pero sobre todo concientizar a la sociedad.

1.7.2 GASES CONTAMINANTES

En el campo automotriz los vehículos que usan combustibles fósiles como gasolina o diésel son autores principales de contaminación ambiental, no obstante, la realidad es incierta con respecto a los gases que emiten los tubos de escape producto de la combustión, por ejemplo, un vehículo que funciona con gasolina emite monóxidos de carbono, hidrocarburos, compuestos de plomo que son los causantes de perjudicar la salud del hombre (Olmos, 2019). Entre los gases más contaminantes que emiten los tubos de escape tenemos:

- **Monóxido de Carbono (CO):** resultado de una combustión incompleta.
- **Hidrocarburos (HC):** son procedentes del petróleo.
- **Benzopirenos:** son procedentes de combustibles menos refinados.
- **Dióxido de azufre (SO₂):** se origina por la combustión de compuestos fósiles que poseen azufre.
- **Óxidos de nitrógeno (NO_x):** son resultado de la combustión que producen ácidos.
- **Material particulado (PM):** son provenientes de reacciones químicas como el SO₂ y el NO_x.
- **Compuestos orgánicos volátiles (COV):** son el producto de la quema de combustibles y otros materiales como maderas y gases naturales.

1.7.3 GASES DE EFECTO INVERNADERO (GEI)

Los GEI o gases de efecto invernadero son compuestos gaseosos de la atmósfera los cuales pueden ser de forma natural o de forma antropogénica, estos gases absorben y a la vez emiten radiación de una determinada longitud de ondas del espectro de radiación infrarroja que es emitida por la Tierra, nubes y la atmósfera, cabe recalcar que los principales gases de efecto invernadero son el vapor de agua, el dióxido de carbono, el óxido nitroso, el metano y ozono: no obstante algunos GEI son causados por el ser humano y algunos animales como el ganado vacuno y bobino, algunos de estos gases contienen halocarbonos y sustancias que muchas veces en su composición contienen cloro y bromo (Aristizábal, 2007,p.22).



Figura 1.1 Gases y Compuestos de Efecto Invernadero
(Instituto Nacional de Ecología y Cambio Climático, 2019)

En la Figura 1.1 se muestra los gases generados por la industria automotriz ya sea por los vehículos livianos o pesados por el consumo de combustibles fósiles, sin olvidar que los seres vivos también son protagonistas al generar estos gases, ya sea por expansión de territorio sobre todo en el ganado (vacuno y bobino) o por los gases naturales que son emitidos por el ser humano.

1.7.4 DIÓXIDO DE CARBONO (CO₂)

El CO₂ o dióxido de carbono también conocido como Gas de Efecto Invernadero directo es un compuesto que no tiene olor ni color y su formación se debe principalmente a procesos que tienen combustión de elementos o sustancias que posean carbono (Berenguer & Bernal, 2000, p.1). Según (EINECS, 2019) este gas se compone principalmente por una molécula de 1 átomo de carbono que se une con 2 átomos de oxígeno de manera lineal (O=C=O).

Las fuentes de emisión de dióxido de carbono son los automotores; entre los que se mencionan, a los motores de gasolina que por cada litro de combustible consumido emite 2,37 kg de CO₂ a diferencia de los motores diésel que por cada litro de combustible consumido emiten 2,65 kg de CO₂, por ende, cabe recalcar que las emisiones de este gas dependen directamente del consumo y no de la potencia del motor (Sarango, 2016, p.14),

es decir, estas emisiones provocan una disminución notable en la calidad del aire lo que aumenta la temperatura que conlleva a un incremento brusco del calentamiento global. En base a estos conceptos las emisiones provocadas por los automóviles y las grandes industrias no son los únicos responsables del calentamiento global, anteriormente se ha mencionado que la descomposición de materia orgánica muerta, la fotosíntesis producida por las plantas incluso la respiración de los seres humanos produce este gas lo cual también se deben considerar como fuentes de emisión.

1.7.4.1 CO₂ Impacto del Transporte

En la industria automotriz el impacto del transporte ha sido uno de los principales autores de emisión de GEI sobre todo del CO₂, en la Tabla 1.1 se muestra los diferentes valores del factor de emisión de este gas que cada medio de transporte que produce mientras circula:

Tabla 1.1 Factores de emisión CO₂ emitido por medios de transporte

VEHÍCULO	DISTANCIA (km)	VOLUMEN CO ₂ (g)
Automóvil convencional	1	150
Avión	1	180
Autobús	1	30
Ferrocarril	1	35

Fuente: (Benito, 2016,p.6).

Estos valores demuestran que los autobuses y ferrocarriles son medios de transporte más eficientes, es decir, que pueden trasladar de un punto a otro a más personas con un nivel de emisiones mucho menor, cabe recalcar que todo transporte al sumar las cantidades de miles de individuos que viajan de un lugar a otro por todo el mundo, este valor es multiplicado por la cantidad de kilómetros que recorren dando como resultado un enorme tonelaje de CO₂ que se emiten a la atmosfera de la tierra por parte de estos medios de transporte.

Según Yolanda Benito comenta que el cambio de combustibles que contengan una proporción menor de carbono-hidrogeno, ya sea un ejemplo del paso del carbón al petróleo o este dar el cambio al gas natural se notaría una disminución en las emisiones; una muestra

clara es el gas natural que genera una emisión de CO₂ menor por cada energía de los demás combustibles fósiles, es decir con 15 kgC/GJ, el petróleo por otro lado genera 20 kgC/GJ y por ultimo aparece el carbón con 25 kgC/GJ, si bien dar el paso del carbón al gas natural y se mantiene una eficiencia semejante de conversión de combustible-electricidad las emisiones disminuirían en aproximadamente 40% (Benito, 2016,p.6).

1.7.5 ÓXIDO NITROSO (N₂O)

El óxido nitroso es un gas que interviene principalmente con los cambios abruptos del clima, ya que reposa principalmente en la estratósfera provoca una disminución rápida y violenta de la capa de ozono dando como resultado una capa más delgada por así decirlo, el N₂O es causante del 5% del efecto invernadero en el último siglo (Núñez,García,Caridad,Asencio,2021,p.2).

El óxido nitroso se compone de nitrógeno y oxígeno; el nitrógeno es un gas completamente inerte y forma parte del aire que todo ser vivo terrestre respira, la concentración que tiene es del 79% y la del oxígeno no supera el 21%; cabe recalcar que el oxígeno es un gas importante para la combustión, las altas temperaturas que se provocan en el interior del motor hacen que el nitrógeno se oxide, dando como resultado el gas de efecto invernadero (N₂O) (Pérez, 2018,p.24).

Concluyendo el N₂O forma parte del calentamiento global en gran parte provocado por los vehículos y las grandes industrias, pero también los animales como los bovinos que son los causantes principales de la disminución de la capa de ozono en los últimos 100 años.

1.7.6 METANO (CH₄)

El metano es un gas de efecto invernadero el cual los animales bovinos o vacunos contribuyen con un 18% del total que se produce en el planeta (Ramírez, 2018,p.23), este gas es un hidrocarburo que no posee olor, es inflamable y tiene un peso más ligero que el aire; se produce por la descomposición de la materia sobre todo orgánica y en la industria automotriz el metano se destaca por su gran número de octano con un valor de 135/122,

por ende, este gas se adapta a elevadas relaciones de compresión que se presentan sobre todo en los motores Diésel sin ocasionar problemas de detonación (González, 2017,p.5).

El metano al igual que el óxido nitroso y el dióxido de carbono son los principales autores de los cambios climáticos que se han dado en la actualidad ya sean provenientes de la combustión o de forma natural provocan una disminución perjudicial en la calidad del aire dando como resultado un sin número de afecciones respiratorias que no solo afectan al ser humano sino también en la reducción de flora y fauna debido al calentamiento global.

1.8 LÍMITES ESTABLECIDOS DE EMISIONES CO₂ VEHÍCULOS LIVIANOS

En el campo automotriz las emisiones se establecen por medio de las normas EURO, que en vehículos ligeros se establecen un con un numero arábigo, por ejemplo: EURO 1, EURO 2, EURO 3, etc., por otro lado para vehículos pesados se establecen con un numero romano EURO I, EURO II, EURO III, etc.; las cuales permiten un mejor control con respecto al cuidado del medio ambiente y sobre todo permiten mitigar en gran proporción el problema de la contaminación ambiental por medio de límites de emisiones en particular del Dióxido de Carbono o CO₂. Actualmente los vehículos livianos y turismos nuevos están comprometidos a seguir los límites establecidos por el REGLAMENTO (CE) N°443/2009 que se muestran en la Tabla 1.2, cabe recalcar que estas normativas tienen un solo objetivo que es la contaminación cero:

Tabla 1.2 Reducción de emisiones CO₂ para la flota de vehículos ligeros

Límites establecidos y porcentajes de emisiones CO₂ para la flota de vehículos livianos		
AÑO	CO₂(g CO₂/km)	OBJETIVO DE REDUCCIÓN DE CO₂ (%)
2008-2009	130	65% en 2012
2012-2014	120	75% en 2013
2015-2020	95	80% en 2014
		100% en 2018 en adelante

Fuente: (Necas & Pottering, 2009).

Con base a estos datos algunas empresas fabricantes de vehículos ya han alcanzado el objetivo de reducción del 130 gCO₂/km y por ende, se requiere para el período 2015-2020

y efectivamente también para el año 2021 es del 27% para toda la industria automotriz, es decir; 3,8 litros por cada 100 km recorridos (consumo combustible), cabe recalcar que las emisiones medias dadas en 2012 fueron de 180 gCO₂/Km y para 2017 su reducción fue notable llegando a 178 gCO₂/km un valor con varios años de antelación (Mock, 2014,p.5).

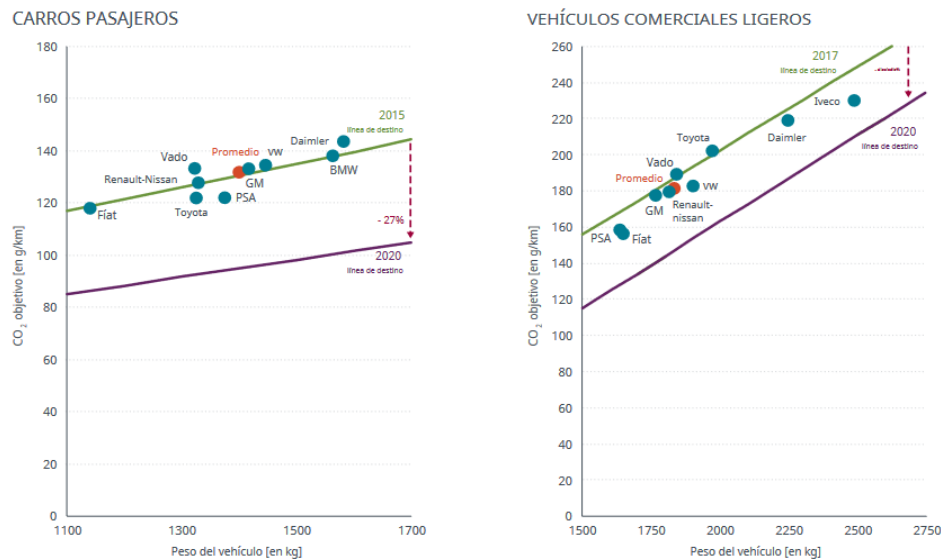


Figura 1.2 Comparación de CO₂ año 2012 vs objetivos para el periodo 2015-2022 (Mock, 2014,p.5).

En la Figura 1.2 se muestra el desempeño y los objetivos de reducción de emisiones CO₂ en vehículos ligeros y de turismos propuestos para el año 2020 en adelante.

1.9 MACHINE LEARNING

Machine learning (aprendizaje automático) o también conocido como aprendizaje de máquina es una rama o subdisciplina de la inteligencia artificial, que permite que los ordenadores aprendan automáticamente a partir de datos específicos, en el año de 1959 Arthur Samuel definió machine learning como la capacidad de los computadores de aprender sin una programación explícita (Géron, 2019,p.19).

El aprendizaje de máquina permite utilizarlo como una herramienta tecnológica la cual está presente en diferentes campos como la medicina y la robótica dando como resultado

máquinas que pueden salvar vidas y realicen tareas específicas facilitando las actividades realizadas por el hombre.

1.9.1 APLICACIONES Y FUNCIONAMIENTO DE MACHINE LEARNING

El funcionamiento de machine learning o aprendizaje de máquina consta de la recolección de datos que generaliza la información para ejecutar predicciones; la asimilación de datos o también conocidos como big data es muy extensa, pero es importante recalcar que el aprendizaje de máquina los percibe como un gran listado de ejemplos prácticos y no como datos (Maisueche, 2019,p.61).

Machine learning ha permitido a la humanidad tener una calidad de vida mucho más práctica, pero sobre todo brindar facilidades en diversos campos que se desempeña el hombre. Los usos del aprendizaje de máquina se adaptan a múltiples industrias entre las más destacadas se aplican en(Calvo,Guzmán,Ramos,2018,p.15):

- **Ámbitos educativos:** en este ámbito machine learning posee sistemas que funcionan como compañeros de aprendizaje para toda la carrera estudiantil de un alumno.
- **Finanzas:** en este ámbito machine learning usa algoritmos que permite la detección de fraudes, mediciones de alto riesgo financiero, trading y REGTECH que es un crecimiento el cual utiliza el aprendizaje de máquina para regular y controlar los diferentes conjuntos de compañías financieras.
- **Salud:** en este sector machine learning mejora los diagnósticos entregados por imagen, donde recolecta información con todo lo relacionado en medicina para su estudio, pero sobre todo también lograr adecuadas cirugías robóticas.
- **Industrias o logísticas:** en el sector industrial el aprendizaje automático se usa en el mantenimiento predictivo utilizando señales de sensores para solucionar y mejorar el mantenimiento de la maquinaria.

1.9.2 TIPOS DE ALGORÍTMOS DE MACHINE LEARNING

En machine learning existen diversos algoritmos que permiten la ejecución de modelos de predicción que ayuden a los diferentes sectores mencionados anteriormente entre los algoritmos más destacados se tiene:

- Aprendizaje Supervisado
- Aprendizaje no Supervisado
- Aprendizaje por Refuerzo

A continuación, se describe cada uno de estos algoritmos más importantes en machine learning.

1.9.2.1 Aprendizaje Supervisado

Este tipo de aprendizaje se basa en la comparación de una salida calculada y una salida esperada, por ende, este proceso calcula el error y se ajusta para obtener un resultado deseado, es decir, que cada muestra de los datos recolectados tiene una variable que indica lo sucedido (Das, Dey, Pal, Roy,2015,p.31).

Para entender este tipo de aprendizaje se considera ejemplos sencillos como la clasificación en correos electrónicos ya sean correos basura o no, otro ejemplo más preciso es la predicción de precios de objetos dependiendo de sus particularidades.

1.9.2.2 Aprendizaje no Supervisado

También denominado algoritmo de agrupación el cual opera en las agrupaciones de datos no etiquetados, el objetivo es revelar las características y propiedades de los datos recolectados un claro ejemplo del aprendizaje no supervisado es GOOGLE NEWS que realiza agrupamientos de historias nuevas en la web y las ubica en las secciones de noticias (Das et al.,2015,p.32). En conclusión, este proceso de aprendizaje puede tener una entrada, pero no una salida, en estas variables es necesario buscar patrones y para la realización de la predicción se puede encontrar estructuras que estén sobre los datos.

1.9.2.3 Aprendizaje por Refuerzo

En este aprendizaje para obtener un resultado se basa en elegir secuencias de decisiones, es decir, se toman las acciones correctas para perfeccionar destrezas en el momento de ejecutar actividades definitivas tales como las comunicaciones, robótica y otras (Arango, 2019,p.4).

De acuerdo con este criterio el aprendizaje por refuerzo se aplica en acciones que requieren una toma de decisiones en los que el individuo interactúa en un entorno tomando acciones que son secuenciales y los resultados obtenidos son en base de observaciones, aportaciones etc., mientras recibe retroalimentación sobre cada acción elegida.

1.10 ESTIMACIÓN DEL CONSUMO DE COMBUSTIBLE Y EMISIONES CO₂

La estimación del consumo de combustible se puede obtener con diversos parámetros y coeficientes que ya se encuentran previamente calculados exclusivamente para los vehículos livianos como se muestra en la tabla 1.3, tomando en cuenta estos valores pueden ser usados en las fórmulas descritas en la sección de métodos del capítulo III de este proyecto, este cálculo se realiza con el objetivo de observar si existe una relación con la emisión de CO₂ para la ejecución de los modelos de predicción.

En la estimación del CO₂ se considera también las variables como el gas quemado por litro de combustible, densidad del combustible, relaciones de combustión, constantes generales de los gases, masas moleculares en este caso específicamente del aire y masas atómicas que dan paso a la información de la composición de la emisión CO₂. Todos estos datos específicos se observan en la Tabla 1.3 que se muestra a continuación:

Tabla 1.3 Constantes del Consumo de Combustible y Flujo Másico del Aire

Datos Específicos (Constantes)	
CO ₂ quemado por litro de combustible [Gasolina]	2,28 (kg/l) ⁽¹⁾
Masa del Vehículo Masa de ocupantes del vehículo (Piloto, Copiloto, Pasajero)	1655 kg ⁽²⁾ 225kg (75 kg c/u)

g = Gravedad	9.81 m/s ²
$\rho_{combustible}$ = Densidad del combustible [Gasolina]	720 g/l ⁽³⁾
AFR = Relación de Aire Combustible [Gasolina]	14,6:1
Poder Calorífico de la Gasolina	45.50 MJ/kg ⁽⁴⁾
Constante General de los Gases	8,314 ($\frac{J}{K \cdot mol}$) ⁽⁵⁾
Fuentes: ⁽¹⁾ (Mosquera, Fernández, Mosquera, 2010,p.3) ⁽²⁾ (Cherry,2022, p.3) ⁽³⁾ (Narváez & Narváez,2023, p.120) ⁽⁴⁾ (RECOPE, 2018) ⁽⁵⁾ (Levine,2004,p.18)	

Estos datos son necesarios para encontrar la estimación de emisión CO₂ provocada por un vehículo liviano, de esta manera el modelo matemático para encontrar dicha emisión no se tornará inconclusa.

1.11 NORMATIVA RDE (REAL DRIVING EMISSIONS)

La normativa RDE fue introducida por el reglamento (UE) 2016/427 de la comisión; provee requisitos o datos cuantitativos que permite dar límites a las emisiones del tubo de escape de un vehículo en condiciones reales de conducción y reflejar de una mejor manera las emisiones generadas en carretera gracias al uso de equipos PEMS (Juncker, 2016,p.24). Siguiendo el Reglamento UE 2017/1151 para el desarrollo del ciclo de conducción se toma en cuenta los datos recolectados y almacenados se logró gracias a los equipos que se mencionaran posteriormente los cuales ayudaron a encontrar las variables correspondientes para encontrar las emisiones de CO₂.

1.11.1 REQUISITOS DE LA NORMATIVA RDE

La normativa RDE permite una conducción más dinámica, es decir, consiste en la ejecución de pruebas en una zona urbana, zona rural y una zona de autopista, esta normativa incluye temperaturas, altitudes, velocidades (bajas, medias y altas); este ensayo sigue protocolos de velocidades, tiempo y distancia; en este proyecto la ruta no estará conformada por sectores, sino más bien se basará en perfiles de velocidad y el número de paradas sobre

todo las calles que atraviesan de norte a sur y de este a oeste, como se muestra en la Tabla 1.4 a continuación:

Tabla 1.4 Requisitos de la normativa RDE para vehículos livianos

NORMATIVA RDE			
Composición de la ruta	Zona Urbana	Zona Rural	Zona Autopista
Velocidad	Entre 15 km/h y 40 km/h	Entre 60 km/h y 90 km/h	≥ a 90 km/h
Porcentaje de distancia	34%	33%	33%
Porcentaje de paradas	6-30% del total del tiempo		
Tiempo	Variación entre 90 y 120 minutos en toda la ruta		
Distancia	Distancia mínima 16 km		
Distancia Total	>48 km		
Altura	-0 a 700 m.s.n.m. con exigencia moderada -De 700 a 1300 m.s.n.m. con exigencia extendida. -A 100 m de diferencia de alturas		

Fuente: (Juncker, 2017,p.203,p.204)

CAPÍTULO II

2. MATERIALES Y MÉTODOS

En este capítulo se procede a describir los materiales y la metodología usada en este trabajo de grado para lograr con el cumplimiento de los objetivos planteados mencionados anteriormente. No obstante, se detalla los equipos que se usaron para la recolección de datos, el modelo matemático para la obtención del resultado de emisiones de gases CO₂ a partir de las variables de operación del vehículo, el programa y el lenguaje de programación que fue utilizado para la obtención del modelo de predicción.

2.1 METODOLOGÍA DE LA INVESTIGACIÓN

El presente capítulo muestra la metodología correspondiente para la ejecución del tema propuesto sobre la predicción de emisiones usando machine learning para la estimación de gases CO₂ a partir de las variables de operación de un vehículo liviano para el cumplimiento de los objetivos planteados. Haciendo uso de la investigación documental se obtuvo información científica y auténtica con respecto a los temas que se relacionan directamente con el proyecto propuesto.

Las metodologías desarrolladas secuencialmente logró obtener los resultados deseados por medio de una metodología cuantitativa que a su vez llevo a cabo un análisis comparativo y un análisis estadístico de los datos recolectados con los equipos y materiales que se explicaran posteriormente.

2.1.1 ENFOQUE INVESTIGATIVO

El objetivo principal de este trabajo de grado es desarrollar un modelo de predicción de emisiones usando machine learning, lo cual fue necesario adquirir conocimiento sobre el aprendizaje de máquina, los aprendizajes correspondientes que lo conforman y los campos que hacen uso de machine learning sobre todo en el campo automotriz.

2.1.2 TIPOS DE INVESTIGACIÓN

La ejecución del enfoque investigativo permitió tener una idea más clara con respecto a los tipos de investigación que conforman la metodología empleada en este proyecto las cuales permitieron un entendimiento más concreto y facilitaron la forma de llegar a los resultados deseados; a continuación, se describe los tipos de investigación ejecutados en el presente trabajo.

2.1.2.1 Investigación de Análisis Comparativo

Este tipo de investigación permitió realizar la comparación entre los resultados de fiabilidad obtenidos en los modelos de predicción ejecutados y la comparación de los valores reales de la estimación de emisiones CO₂.

2.1.2.2 Investigación de Análisis Estadístico

Esta investigación permitió un análisis de la correlación lineal o regresiones que están presentes en las variables numéricas, es decir, las variables de operación del vehículo. El presente trabajo de grado provee variables de tipo numérico lo que conlleva a una dispersión de datos en forma de regresión entre dichas operaciones y la variable dependiente que es la emisión de CO₂ provocada por el automóvil en una ruta específica que se rige bajo una normativa.

2.1.2.3 Investigación Tecnológica

El trabajo de grado se ejecutó con los equipos Datalogger CSS electronics CANedge2 y para el registro y almacenamiento de datos y el GPS CANmodGPS de la misma marca para el posicionamiento en tiempo real del vehículo y su sistema de software que ayudó a obtener una visualización de las gráficas de comportamiento de las operaciones del vehículo en una ruta específica.

No obstante, para el desarrollo de los modelos de predicción ejecutados posteriormente se usó un cuaderno de programación Python (Google Colab) que detallaremos más adelante en este capítulo y permitió la entrada de los datos registrados.

2.2 MATERIALES & EQUIPOS

En esta sección del capítulo II se muestra los respectivos equipos y su definición que fueron usados para la ejecución del presente trabajo de grado y el desarrollo de los modelos de predicción, estos equipos permitieron el registro, el almacenamiento y el posicionamiento en tiempo real del vehículo.

2.2.1 DATALOGGER (CANedge2)

Para desarrollar el presente trabajo de grado se usó un datalogger CANedge2, el cual facilitó la recolección de datos en las rutas específicas, este equipo posee la factibilidad de configuración de su frecuencia de recolección y almacenamiento de datos, es decir, su tasa de bits puede llegar a un valor máximo de hasta 4 tomas de datos por segundo para cada sensor con diferente frecuencia; este equipo posee un tamaño relativamente pequeño tal como se muestra a continuación



Figura 2.1 Datalogger CSS electronics CANedge2
(CSSELECTRONICS, 2021a)

En la Figura 2.1 muestra el datalogger CANedge2 que posee un software de código abierto y facilita el almacenamiento de datos registrados en formato MDF4 y permite la

interoperabilidad entre las herramientas CAN. No obstante, la frecuencia usada en este trabajo fue de 1 Hz, sin embargo, este equipo y este software también permite convertir estos archivos en diversos formatos como ASC (vectores), TRC(Peak) y CSV(Excel), este último formato es el más recomendable para el desarrollo del modelo de predicción. A continuación, en la Tabla 2.1 se muestra las especificaciones técnicas de este equipo:

Tabla 2.1 Especificaciones Técnicas Datalogger CANedge2

Especificaciones Técnicas	
Seguridad	Certificación CE. FCC. IC.
Pruebas de voltaje	Transitorios ISO
Software	Gratuito
Documentación	En línea
Registro y almacenamiento de datos	
Tarjeta SD	Micro SDHC industrial de 8Gb
Reloj de tiempo real	Incluido batería de respaldo
Formato de archivo	MDF4 procesa/convierte fácilmente
Mecanismo/suministro	
Conectores	2x DB9
Suministro de entrada	+7V a +32v DC a través del can DB9
Dimensiones	52,5x81,4x24,5(largo x ancho x alto)
Led	5 leds

Fuente: (CSSELECTRONICS, 2021)

2.2.2 CANmodGPS

Este dispositivo es un módulo de sensor de posición (GNSS) inercial 3D(IMU), este equipo se complementa perfectamente con el Datalogger CANedge2 de la misma marca CSS electronics y fue usado para proporcionar 6 lecturas importantes para la ejecución del modelo de predicción, es decir, 3 de aceleración y 3 lecturas de giroscopio(IMU=3 aceleraciones [x, y, z]; GIROSCOPIO=balanceo, cabeceo, guiñada) las cuales su frecuencia de trabajo es de 100 Hz, no obstante, cabe recalcar que el CANmodGPS también proporcionó señales altitud, cuentakilómetros, velocidad, posición y por ultimo información de satélite que trabajan a una frecuencia de 1Hz.



Figura 2.2 GPS CANmodGPS
(CSSELECTRONICS, 2021b).

En la Figura 2.2 muestra el dispositivo CANmodGPS que fue usado para el posicionamiento del vehículo en tiempo real, esta posición fue lograda gracias a los datos de longitud y latitud otorgada por el equipo.

2.2.3. VEHÍCULO DE PRUEBAS

Los vehículos usados y sobre todo compatibles con el equipo pasaron por una prueba de conducción en una ruta corta para confirmar la obtención de datos; entre 7 vehículos usados para las pruebas las marcas más compatibles con el equipo CANEGDE2 CSS ELECTRONICS son:

- Kia Soluto
- Hyundai Accent
- Hyundai Verna
- Ford Escape
- JAC
- SWM
- Cherry Arrizo 5 Pro

El vehículo para ejecutar este proyecto se usó la marca Cherry Arrizo 5 Pro que se muestra a continuación:



Figura 2.3 Vehículo Cherry Arrizo 5 año 2022

En la Figura 2.3 muestra el vehículo utilizado en las zonas de la ruta RDE y los datos específicos ya mencionados para la ejecución del modelo de predicción; este vehículo fue compatible con el equipo Datalogger para la recolección y almacenamiento de estos por medio de la entrada OBD II. La Tabla 2.2 proporcionó información importante con respecto al consumo de combustible en diferentes tramos que menciona su ficha técnica, estos datos también hicieron referencia con sus medidas, potencia, torque, y también su emisión de CO₂ promedio que se muestra a continuación:

Tabla 2.2 Datos específicos del vehículo Cherry Arrizo 5 Pro

Datos Técnicos del vehículo Cherry Arrizo 5 Pro	
Ancho del vehículo	1825 mm
Altura del vehículo	1487 mm
Longitud del vehículo	4532 mm
Relación Peso/Potencia	9,6 kg/105,4hp
N° de cilindros	4
Cilindrada	1499 cm ³
Potencia	115 hp/6150 RPM/125kW
Torque	141 N.m/3800 RPM
Consumo de Combustible Extraurbano	6,2 L/100 km
Consumo de Combustible Urbano	7,9 L/100 km
Consumo de Combustible Combinado	7,0 L/km

Fuente: (Cherry Arrizo 5 Pro, 2021,p.2)

2.3 MÉTODOS

La figura que se muestra a continuación ayudó al desarrollo del presente trabajo de grado el cual se resume el proceso que se debe seguir para dar cumplimiento a los objetivos planteados.

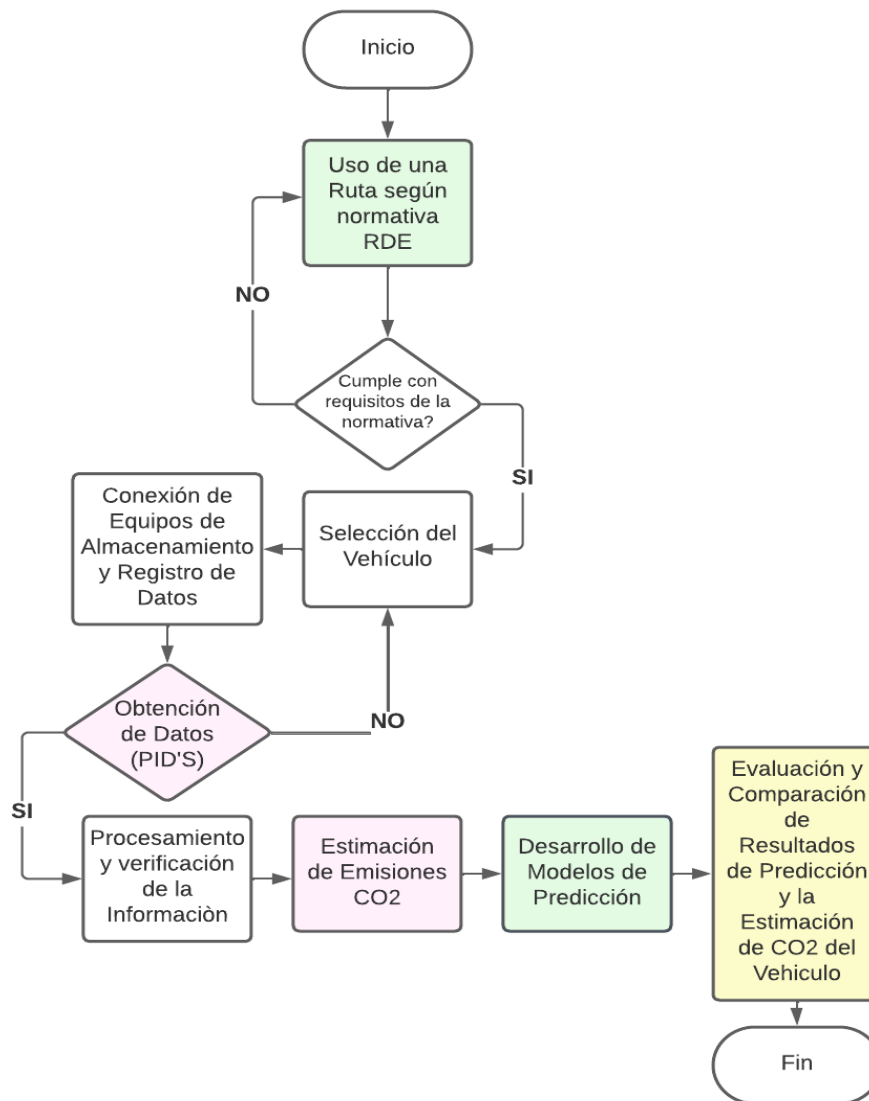


Figura 2.4 Metodología para la estimación CO₂ y Desarrollo de Modelo de Predicción

La Figura 2.4 muestra el proceso para la obtención de datos y llegar a la estimación CO₂ del vehículo de prueba para desarrollar diversos modelos de predicción y ejecutar una comparación, de esta forma lograr una selección del o los modelos que mejor se acerquen a la emisión generada por el vehículo.

2.3.1 USO DE UNA RUTA SEGÚN NORMATIVA RDE

La normativa RDE (Real Driving Emissions) fue implementada en una ruta en la ciudad de Ibarra desarrollada por el grupo de estudiantes conformado por Narváez Espinoza Kevin Andrés y Narváez Sosa Anderson Miguel el cual su tema de trabajo de grado fue “Desarrollo de Ruta de Ensayo RDE en el Cantón Ibarra”; con el vehículo que se tomó como referencia y el desarrollo de la ruta se llegó a obtener un modelo de predicción de emisiones CO₂ gracias a los datos almacenados y recolectados en el equipo Datalogger. La ruta se conforma por 3 tramos diferentes, un tramo en zona urbana, un tramo en zona rural y un tramo en zona de autopista, esta ruta no se basa en la sectorización, pues la ciudad de Ibarra cuenta con una geografía muy sinuosa donde no permite cumplir con ciertos requisitos, por ende, la ruta ejecutada para este trabajo de grado se enfocó con perfiles de velocidad, tiempo, número de paradas, calles y avenidas que atraviesan la ciudad de norte a sur y de este a oeste y sobre todo en cumplir con el porcentaje correcto con las distancias mencionadas en la Tabla 1.4.

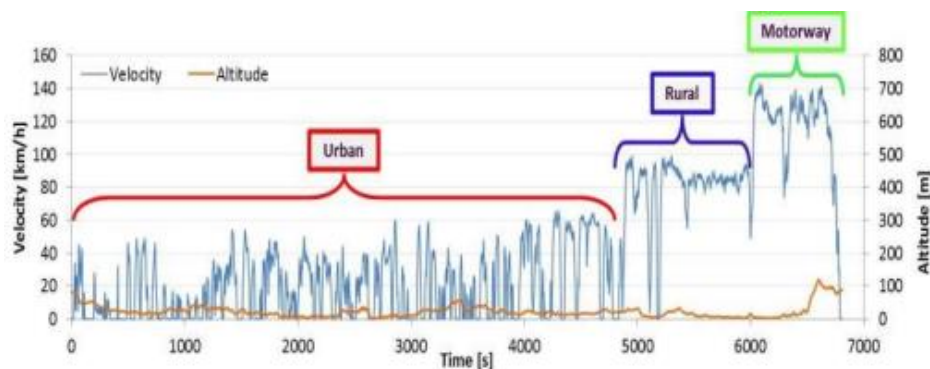


Figura 2.5 Perfiles de velocidad, tiempo y altitud según la normativa RDE
(Triantafyllopoulos, 2019)

En la Figura 2.5 muestra los perfiles de velocidad, tiempo y de altitud que rige la normativa RDE para la ejecución de pruebas de conducción en las zonas diferentes que conforman dicha normativa, con base a estos perfiles se logró elegir las calles y avenidas correspondientes de la ruta seleccionada.

2.3.1.1 Prueba de conducción en Zona Urbana

La ejecución de la ruta en zona urbana se eligió lugares tomando como partida la UTN y tomando las calles que conectan al centro de la ciudad de Ibarra; la tabla 2.3 muestra los puntos de la ciudad por donde se ejecutó la ruta, en esta tabla se observa el inicio de la calle hasta donde llegó para futuras referencias en la selección de una ruta RDE que se mencionan a continuación:

Tabla 2.3 Puntos de la Ruta RDE en Zona Urbana

INICIO	HASTA
Universidad Técnica del Norte (Entrada Principal)	Av. Padre Aurelio Polit
Redondel Universidad Católica del Ecuador (Redondel Universidad Católica)	Av. 17 de Julio
Av. 17 de Julio	Juan José Flores
Juan José Flores	Juan Montalvo
Juan Montalvo	Pedro Moncayo
Pedro Moncayo	Juan de Salinas
Juan de Salinas	German Grijalva
German Grijalva	Pedro Vicente Maldonado
Pedro Vicente Maldonado	Pedro Moncayo
Pedro Moncayo	Vicente Rocafuerte
Vicente Rocafuerte	Germán Grijalva
Germán Grijalva	Antonio José de Sucre
Pedro Moncayo	Sánchez y Cifuentes
Sánchez y Cifuentes	Rafael Troya
Rafael Troya	Av. Víctor Manuel Peñaherrera
Av. Víctor Manuel Peñaherrera	Av. Jaime Roldós Aguilera
Av. Jaime Roldós Aguilera	Av. Cristóbal de Troya (Redondel de la policía)
Av. Cristóbal de Troya (Redondel de la policía)	Av. Mariano Acosta (Redondel de la madre)
Av. Mariano Acosta (Redondel de la madre)	Eugenio Espejo (Obelisco)
Eugenio Espejo (Obelisco)	Alfredo Pérez Guerrero
Alfredo Pérez Guerrero	Antonio José de Sucre
Antonio José de Sucre	Av. Teodoro Gómez de la Torre
Av. Teodoro Gómez de la Torre	Av. El retorno
Av. El retorno	Hernán Gonzales de Saa
Hernán Gonzales de Saa	Av. Atahualpa
Av. Atahualpa	Av. Teodoro Gómez de la Torre
Av. Teodoro Gómez de la Torre	Av. Eleodoro Ayala
Av. Eleodoro Ayala	Av. José Tobar y Tobar
Av. José Tobar y Tobar	Av. Camilo Ponce Enríquez
Av. Camilo Ponce Enríquez	Av. Mariano Acosta (Redondel de la madre)
Av. Mariano Acosta (Redondel de la madre)	Av. Mariano Acosta
Av. Mariano Acosta	Agustín Rosales (semáforo la Florida)

Fuente:(Narváez & Narváez, 2023,p.189)

A continuación, se muestra la zona urbana y las calles que recorrió el vehículo.

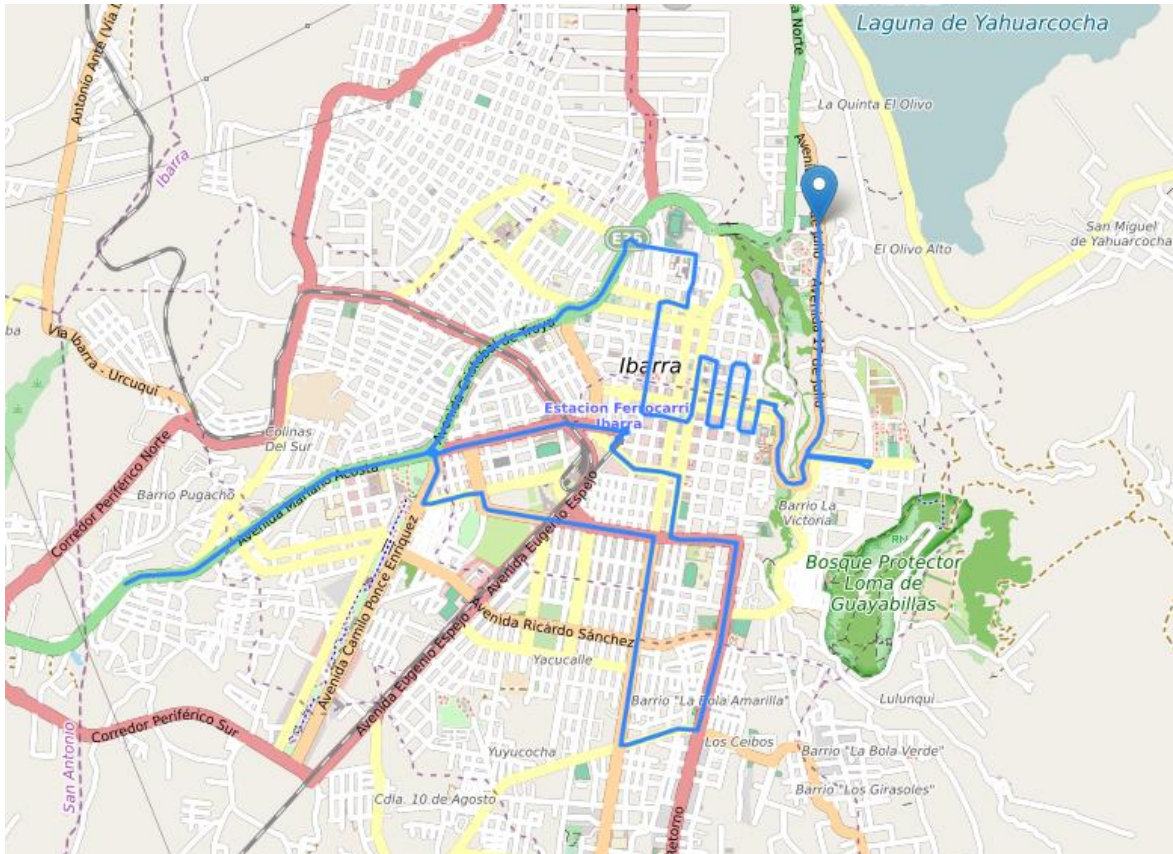


Figura 2.6 Ruta Zona Urbana

En la Figura 2.6 se muestra el recorrido de la zona urbana realizada en la ciudad de Ibarra gracias al equipo GPS donde muestra una longitud de 19,401 Km con un tiempo total de conducción de 3879 segundos o 1hora:07 minutos y llegando a una velocidad máxima de 60,60 km/h (Velocidad Promedio en la ruta= 20,5 km/h) como lo establece el reglamento ya antes mencionado (entre 50 y 60 km/h).

2.3.1.2 Prueba de conducción en Zona Rural

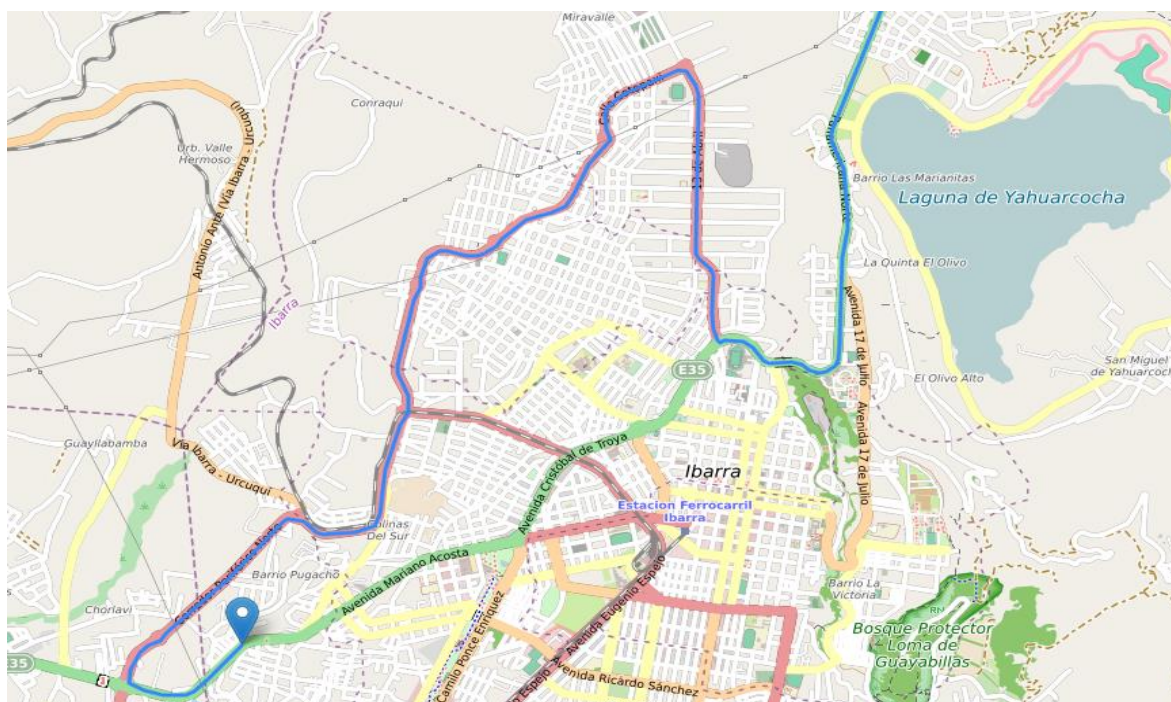
En esta zona se caracteriza por estar conformada por parroquias rurales que se encuentran a las afueras de la ciudad principal en este caso la ciudad de Ibarra. Parte de estas rutas tienen articulaciones viales que conectan con la zona urbana y de autopista que forman la ruta completa siguiendo la normativa RDE; la tabla 2.4 también muestra los puntos o las calles las cuales conectan la zona urbana con la zona rural que se mencionan a continuación:

Tabla 2.4 Puntos de la Ruta RDE en Zona Rural

INICIO	HASTA
Agustín Rosales (semáforo de la florida)	Av. Mariano Acosta
Av. Mariano Acosta	Corredor periférico sur
Corredor periférico sur	Av. Fray Vacas Galindo
Av. Fray Vacas Galindo	Av. Monseñor Leónidas Proaño
Av. Monseñor Leónidas Proaño	Corredor Periférico Norte
Corredor Periférico Norte	Av. 13 de abril
Av. 13 de abril	Obispo Jesús Yerovi
Obispo Jesús Yerovi	Panamericana E35
Panamericana E35	Calle Mojanda (Semáforo priorato)

Fuente: (Narváez & Narváez, 2023,p.194)

A continuación, se muestran el mapa de la zona rural que recorrió el vehículo.

**Figura 2.7** Ruta Zona Rural

En la Figura 2.7 muestra la conducción que obtuvo una duración de 1282 segundos o 21,38 minutos con una longitud de 16,431 km con una velocidad máxima comprendida de 75 km/h (Velocidad Promedio en la ruta= 40 km/h) según lo estipulado por la normativa RDE.

2.3.1.3 Prueba de conducción en Zona de Autopista

Esta zona se caracteriza por alcanzar altas velocidades, ya que no posee paradas constantes debido a semáforos o rompe velocidades, según la normativa RDE la zona de autopista debe estar conformada en su mayoría por rectas.

Según la normativa RDE se debe mantener por un lapso de 5 minutos o 300 segundos con una velocidad constante de 100 km/h; en la ciudad de Ibarra este requisito fue ejecutado en la vía Panamericana E35 y en la vía dirigida Ibarra-San Lorenzo o E10; esta zona de autopista está conformada por los siguientes puntos:

- Punto 1(Partida): Panamericana E35 Calle Mojanda (Semáforo priorato)
- Punto 2(Final): Fin de la vía 4 carriles sector los cañaverales

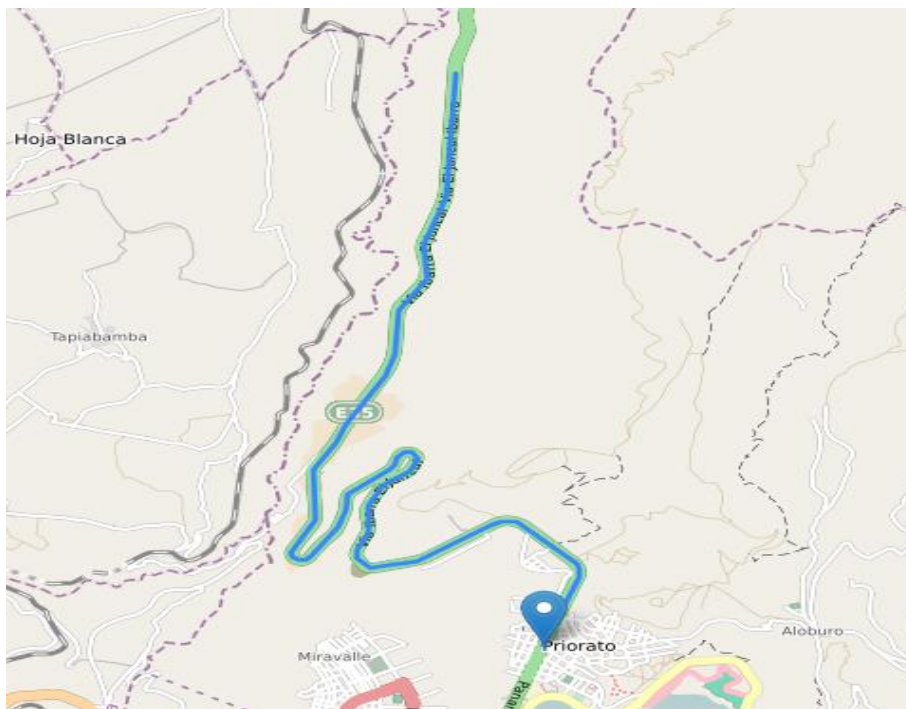


Figura 2.8 Ruta Zona de Autopista

En la Figura 2.8 se muestra el tramo de autopista donde la velocidad máxima alcanzada en esta zona fue de 105 km/h llegando a una velocidad promedio de 85 km/h; esta ruta en zona de autopista tiene una longitud de 16,031 km con una duración de 778 segundos o 13,36 minutos. A continuación, se observa la ruta completa conformada por la zona urbana, rural y la zona de autopista, esta figura representa la unión de dichos tramos y se detalla la duración y distancia total.

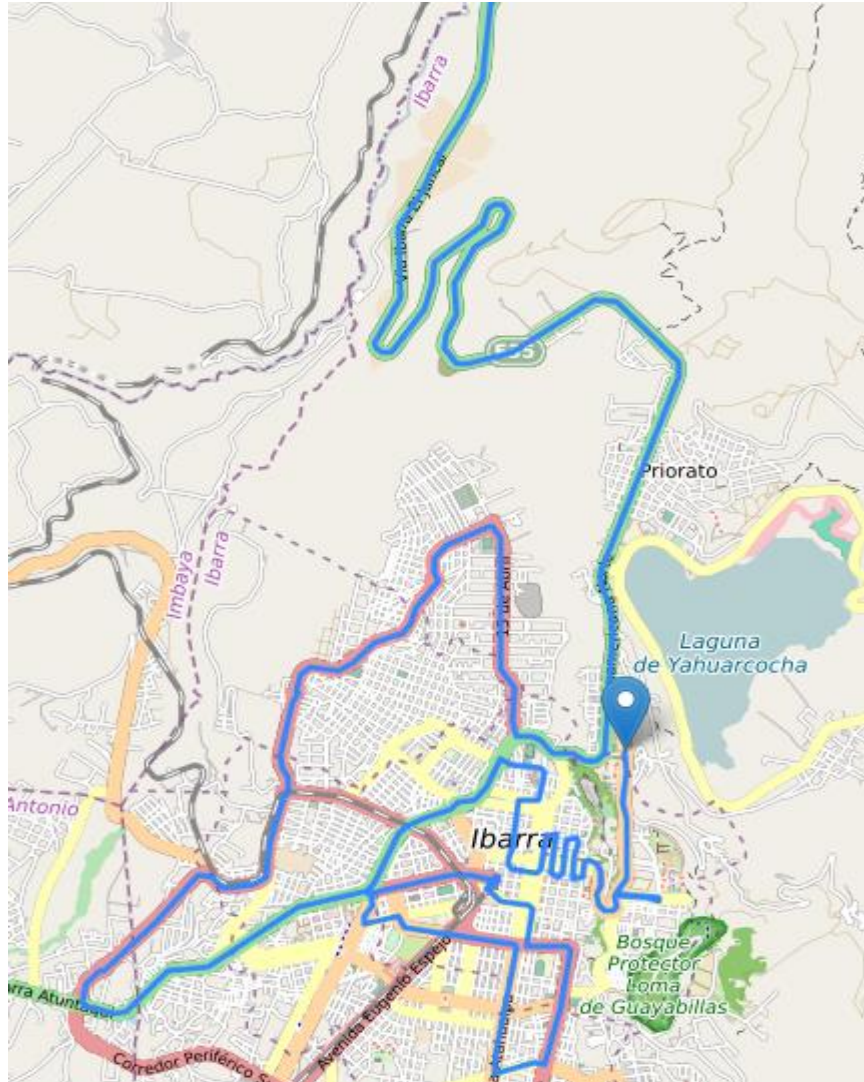


Figura 2.9 Ruta desarrollada bajo la normativa RDE

En la Figura 2.9 se muestra la ruta completa; esta ruta de ensayo de emisiones CO₂ conformada por una zona urbana, rural y de autopista en la ciudad de Ibarra siguiendo la normativa RDE tiene una longitud total de 51,863 km con una duración de tiempo equivalente a 5980 segundos o de 2 horas y 06 minutos; estos datos son la suma total de distancias de los tramos ya mencionados y el tiempo que se ejecutó en cada uno.

2.3.2 OBTENCIÓN DE DATOS

En la obtención de datos PID'S se utilizó el siguiente esquema como se muestra a continuación:

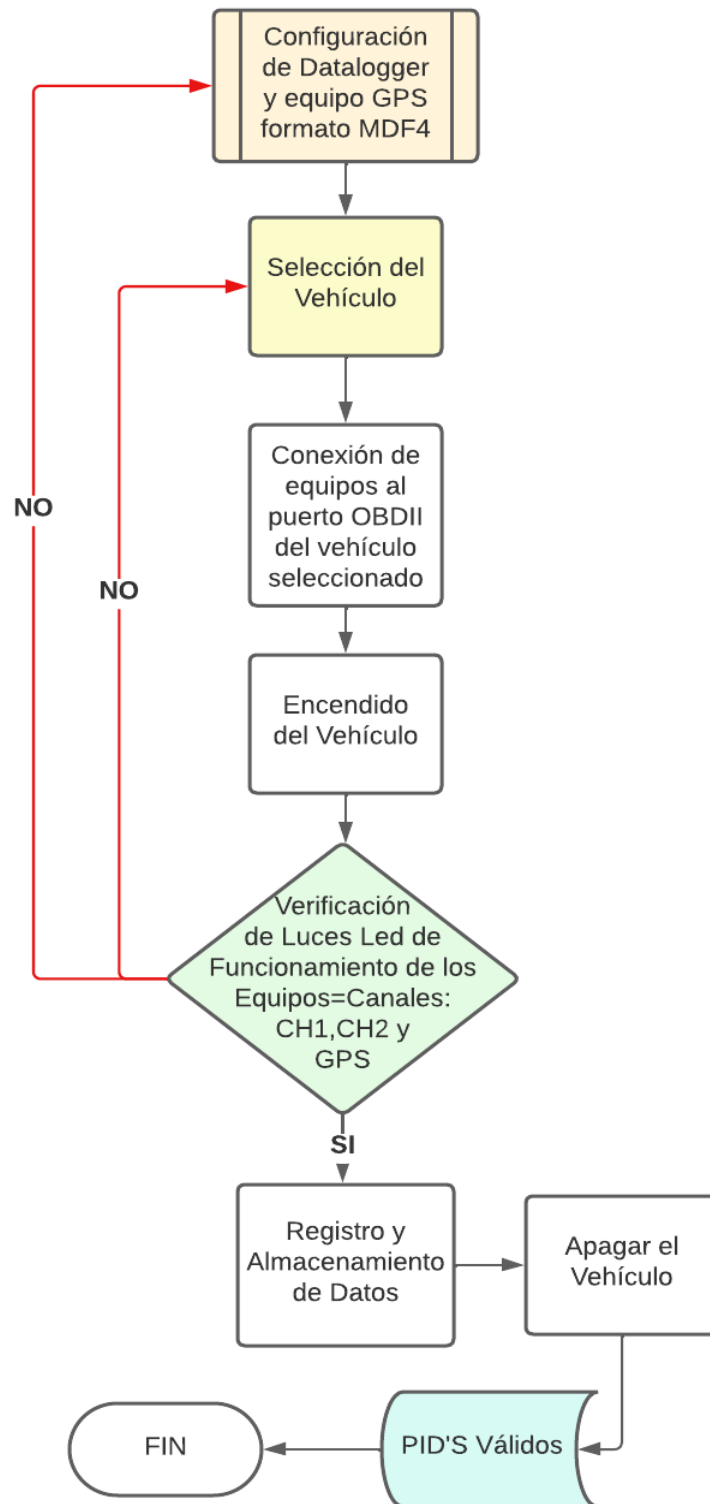


Figura 2.10 Flujograma de Obtención de PID'S

En la Figura 2.10 muestra el esquema por el cual se logró obtener los primeros datos que son la velocidad, altitud, tiempo, latitud, longitud; sensores del vehículo como la Presión

absoluta en el colector de admisión (MAP), Régimen del motor (RPM), temperatura (IAT); con estos resultados se pudo obtener el consumo de combustible generado en la ruta y conseguir la estimación de emisión CO₂.

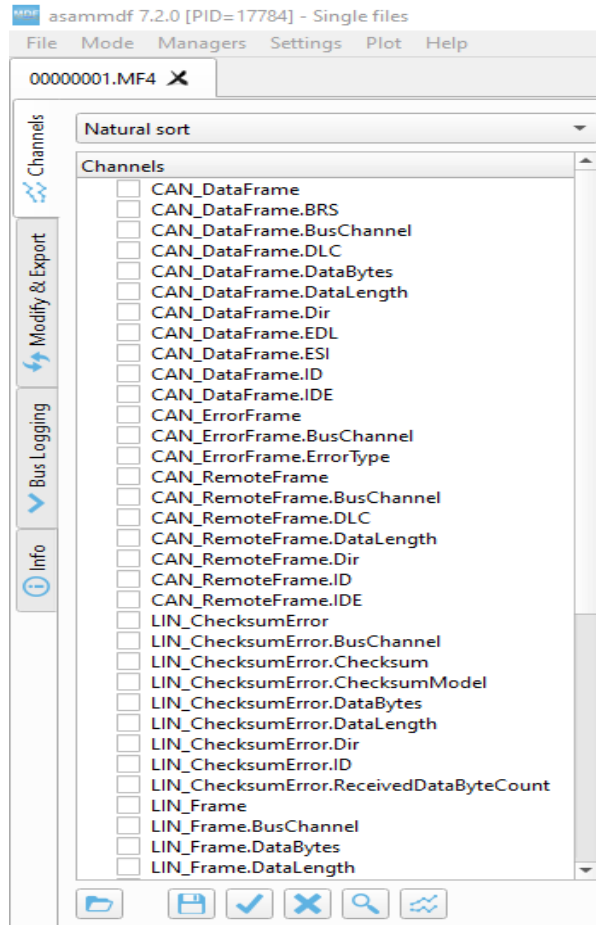


Figura 2.11 Datos sin procesar obtenidos del vehículo

En la Figura 2.11 muestra los PID'S sin procesar del vehículo de pruebas que se logró obtener en el recorrido de la ruta RDE, para poder observar los datos de una manera más concisa y clara se debe colocar los decodificadores que se muestran a continuación:

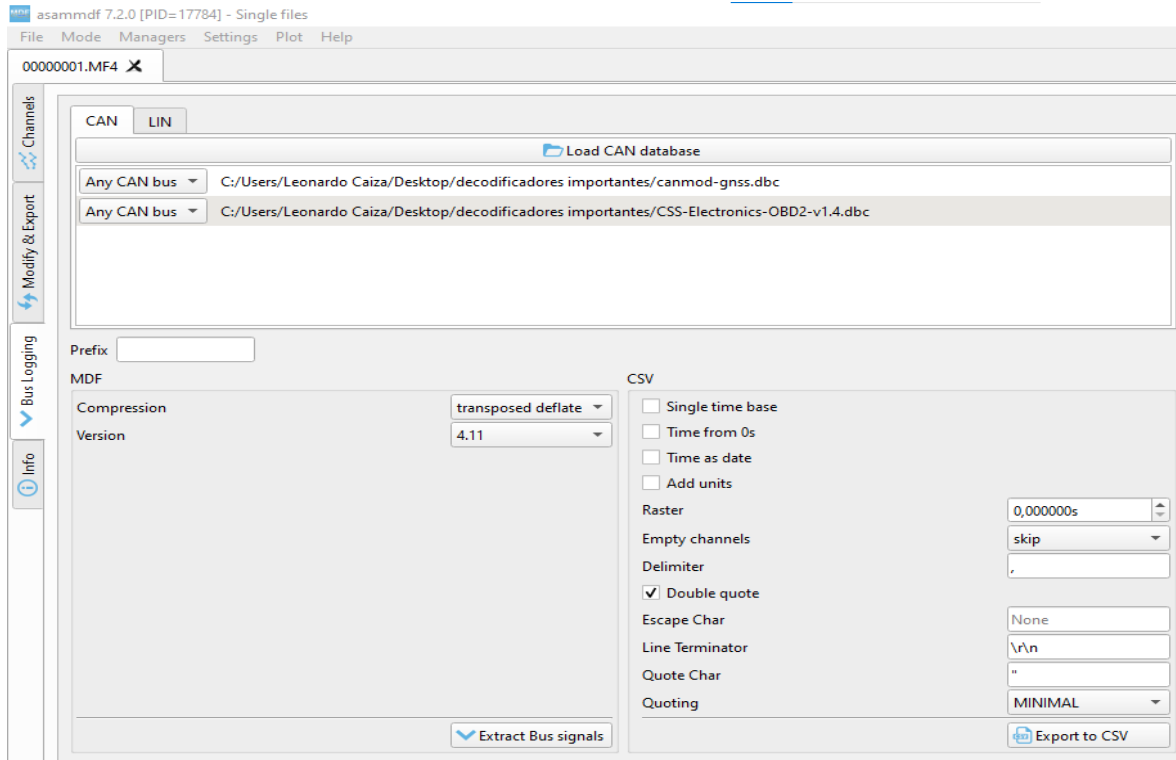


Figura 2.12 Decodificadores del programa Asammdf

En la Figura 2.12 se muestra los decodificadores correspondientes del módulo GPS y datalogger para obtener los PID'S necesarios para calcular las variables importantes para lograr la estimación de CO₂.

La obtención de los diferentes decodificadores que ofrece la página CSS ELECTRONICS son sumamente variados, pues tienen una variedad de no más de 10 vehículos de alta gama y estos se los pudo encontrar en el siguiente enlace: <https://www.csselectronics.com/pages/obd2-data-pack-car-dbc>; tal como se muestra a continuación:

OBD2 Data Pack [PDF, DBC Files & Sample Data]

Join our newsletter to get your 'OBD2 data pack'!

The package contains the following:

- Our OBD2 intro + OBD2 logger guides [PDF]
- The OBD2 DBC file [150+ PIDs]
- 25+ car DBC files (reverse engineered)
- 100+ MB of OBD2 log files across 10+ cars



- 2 Intros to OBD2 [PDF]
- OBD2 DBC
- 25+ car DBCs (reverse engineered)
- 100+ MB of data (10+ cars)

Figura 2.13 Página para la obtención de Decodificadores

En la Figura 2.13 muestra el sitio web el cual se logró obtener los decodificadores importantes para encontrar los PID'S del vehículo, este sitio posee la forma de obtención de más de 150 ID y esto se puede enviar los decodificadores al correo personal sin ningún contratiempo ni publicidad.

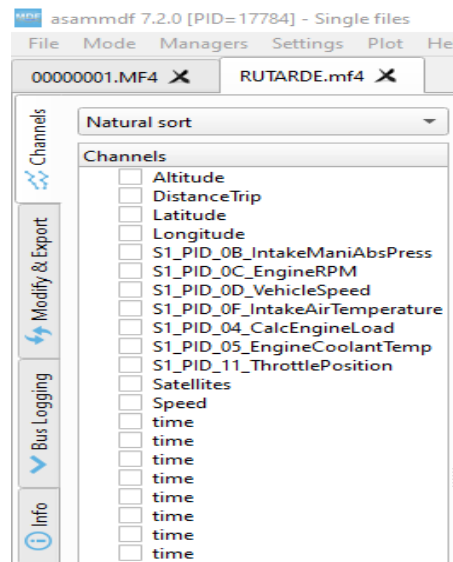


Figura 2.14 PID'S decodificados obtenidos del vehículo

En la Figura 2.14 se muestra todas las señales de los sensores obtenidos del vehículo Cherry Arrizo 5 Pro, gracias a estos sensores se logró tener resultados favorables.

2.3.2.1 Cálculo del Flujo Másico del Aire (MAF)

Los PID'S recolectados del vehículo permitieron encontrar el flujo másico del aire usando el sensor MAP e IAT; con base a estos datos se logró usar la ecuación del gas ideal [2.1] que se muestra a continuación:

$$PV=nRT \quad [2.1]$$

Donde:

P : Presión Absoluta $\left[Pa \rightarrow \frac{J}{m^3}\right]$

V: Volumen del gas [metros cúbicos $\rightarrow m^3$]

n : número de moles [moles]

T : Temperatura del gas [$^{\circ}k$]

R : Constante del gas ideal $\left[8,314 \frac{J}{^{\circ}kMol}\right]$

Con base a esta ecuación se logró calcular la constante ideal del gas específico, de esta forma se evita trabajar con los números de moles de cada gas en particular que se desea trabajar, por ende, en este trabajo de grado se hizo uso del aire para admisión y el combustible que en este caso fue la gasolina como se muestra a continuación:

$$8,314 \frac{J}{^{\circ}kMol} * \frac{1Mol}{0,02897kg}$$

$$R_{aire} = 286,9865 \left[\frac{J}{kg^{\circ}k} \right]$$

La señal del sensor MAF también hizo uso de una variable sintética que incluye los datos RPM, los datos del sensor MAP y por último los datos del sensor IAT que otorgó los equipos como se muestra en la ecuación [2.2].

$$IMAP = \frac{RPM * MAP}{IAT} \quad [2.2]$$

Donde:

RPM: Régimen del motor

MAP: Sensor de Presión Absoluta del Múltiple [Kilopascascales $\rightarrow kPa$]

IAT : Sensor de Temperatura del Aire de Entrada [Grados Kelvin $\rightarrow ^{\circ}k$]

IMAP: Variable Sintética $\left[\frac{kPa}{^{\circ}k} \right]$

Esta variable permitió resumir de forma concisa la introducción de las señales de los sensores MAP e IAT y sobre todo el régimen del motor lo cual para el cálculo del flujo másico se logró obtener de forma más rápida y fácil.

El siguiente paso fue encontrar la variable VE (Eficiencia Volumétrica) en este caso de la mezcla como se muestra en la ecuación [2.3] como se muestra a continuación:

$$VE = \frac{\text{MassMol}}{\rho_{\text{Entrada}} * V_c} \quad [2.3]$$

Donde:

MassMol: Masa Molar de la mezcla $\left[\frac{\text{kg}}{\text{Mol}} \right]$

V_c : Volúmen del cilindro [Metros Cúbicos \rightarrow m^3]

ρ_{Entrada} : Densidad de Aire de Entrada a cada cilindro $\left[\frac{\text{kg}}{\text{m}^3} \right]$

La densidad de entrada se representa en la ecuación [2.4]; esta ecuación también hace uso de los sensores de temperatura en (°Kelvin) y de la presión medida en (Pascales) como se muestra a continuación:

$$\rho_{\text{Entrada}} = \frac{P}{R_{\text{Especifico}} * T} \quad [2.4]$$

Donde:

P: Presión de Admisión otorgado por el sensor MAP [Pa]

$R_{\text{Especifico}}$: Constante Específica del Aire $\left[\frac{\text{J}}{\text{kg}^\circ\text{k}} \right]$

T: Temperatura de Admisión otorgada por el sensor IAT [Grados Kelvin \rightarrow $^\circ\text{k}$]

Las variables calculadas dieron paso al cálculo del flujo másico del aire como se muestra en la ecuación [2.5] que se observa a continuación:

$$\text{MAF} = \left(\frac{\text{IMAP}}{120} \right) * VE * V_{\text{CM}} * R_{\text{aire}} \quad [2.5]$$

Donde:

MAF: Flujo Másico del Aire $\left[\frac{\text{gramos}}{\text{segundo}} \rightarrow \frac{\text{g}}{\text{s}} \right]$

V_{CM} : Volúmen del cilindro [m^3]

VE: Eficiencia Volumétrica [%]

IMAP: Variable Sintética $\left[\frac{\text{kPa}}{^\circ\text{k}} \right]$

R_{aire} : Constante Ideal Específica del Aire $\left[\frac{\text{J}}{\text{kg}^\circ\text{k}} \right]$

En la ecuación [2.5] el resultado encontrado del flujo másico del aire dio paso al cálculo del consumo de combustible en unidades de Litros/segundo.

2.3.2.2 Cálculo del Consumo de Combustible en Ruta RDE

En la obtención del consumo de combustible se hizo uso del resultado encontrado del flujo másico del aire entre el producto de la relación estequiométrica y la densidad del combustible usado para esta investigación tal como se muestra en la ecuación [2.6].

$$FC = \frac{MAF}{AFR * \rho_{gasolina}} \quad [2.6]$$

Donde:

FC: Flujo de Combustible $\left[\frac{\text{litros}}{\text{segundo}} \rightarrow \frac{1}{s} \right]$

MAF: Flujo másico de aire $\left[\frac{\text{gramos}}{\text{segundo}} \rightarrow \frac{g}{s} \right]$

$\rho_{gasolina}$: Densidad del combustible $\left[\frac{\text{gramos}}{\text{litros}} \rightarrow \frac{g}{l} \right]$

AFR: Relación de Mezcla aire/combustible [sin unidades]

En base a esta ecuación se logró obtener el consumo de combustible por una unidad recorrida que, según (Meseguer, 2016,p.5), el CC o consumo de combustible es el cociente o relación entre el combustible que se consume por una distancia recorrida, es decir, este resultado puede obtenerse en litros por cada 100 km; en la ecuación [2.6] al tener el valor del flujo de combustible se relaciona con la distancia en la que recorre el vehículo de la ruta predeterminada medida en km como se muestra a continuación en la ecuación [2.7].

$$CC = \frac{FC}{D_r} \quad [2.7]$$

Donde:

CC: Consumo de Combustible $\left[\frac{\text{litros}}{\text{kilómetros}} \rightarrow \frac{1}{km} \right]$

FC: Flujo de Combustible $\left[\frac{\text{litros}}{\text{segundo}} \rightarrow \frac{1}{s} \right]$

D_r : Distancia recorrida [km]

Tomando en cuenta las ecuaciones mencionadas se procedió al cálculo del consumo de combustible en el tramo urbano, rural y de autopista para encontrar los valores promedios

del consumo y observar en que zona se consumió más combustible y por ende la emisión de CO₂ generada en la ruta.

2.3.3 CÁLCULO DE LA ESTIMACIÓN DE EMISIONES CO₂ EN RUTA RDE

La obtención de emisión CO₂ no solo proviene de los datos basados en los sensores de las operaciones básicas del vehículo; este cálculo posee variables específicas ya conocidas con respecto a la cantidad de carbono, cantidad de combustible vista en la sección anterior de este capítulo en la ruta RDE elegida; de esta forma la emisión de CO₂ en el vehículo de prueba escogido se demuestra con la ecuación [2.8] que se muestra a continuación:

$$CO_2 = CC * CO_{2pl} \quad [2.8]$$

Donde:

CO₂: Dióxido de Carbono $\left[\frac{\text{gramos}}{\text{kilómetros}} \rightarrow \frac{\text{g}}{\text{km}} \right]$

CC: Consumo de Combustible $\left[\frac{\text{litros}}{\text{kilómetros}} \rightarrow \frac{\text{l}}{\text{km}} \right]$

CO_{2pl}: Gramos de CO₂ quemado por cada litro de combustible $\left[\frac{\text{gramos}}{\text{litros}} \rightarrow \frac{\text{g}}{\text{l}} \right]$

La ecuación [2.8] fue usada en los 3 tramos correspondientes de la ruta RDE, esta ecuación permitió la medición instantánea en dicha ruta; como resultado los datos obtenidos dieron paso a la ejecución del modelo de predicción que se describe más detalladamente en la siguiente subsección de este capítulo. En la ecuación [2.9] la variable CO_{2pl} representa los gramos de CO₂ emitido por cada litro de combustible quemado, este resultado se logró obtener con la ecuación [2.9] que se muestra a continuación:

$$\frac{\text{KgCO}_2}{\text{KgFuel}} = \frac{44}{12 + \frac{H}{C}} \quad [2.9]$$

Donde:

H: Átomos de Hidrógeno

C: Átomos de Carbono

En la ecuación [2.10] se logró obtener los kilogramos de CO₂ emitidos por cada kilogramo de combustible quemado, en este caso la fórmula molecular de la gasolina es de C_{7,2}H₁₄ perteneciente a Gasolina Extra de 80 octanos y una densidad de 720 g/l (Narváz & Narváz, 2023,p.120); de la misma forma y gracias a este resultado se logró calcular los litros correspondientes a 1 kg de gasolina.

$$\frac{\text{KgCO}_2}{\text{KgFuel}} = \frac{44}{12 + \frac{14}{7,2}}$$

$$1\text{KgFuel} = 3,1553\text{KgCO}_2$$

$$V = \frac{\text{kg}_{\text{fuel}}}{\rho_{\text{fuel}}}$$

$$V = 1,388 \text{ litros}$$

$$1\text{KgFuel} = 1,388 \text{ litros}$$

El resultado representa los litros que se posee en un 1 kg de gasolina; este valor fue de mucha ayuda pues gracias a esto al multiplicar con la ecuación [2.9] se obtuvo los kilogramos de emisión CO₂ por cada litro quemado de combustible como se observa a continuación:

$$\frac{3,1553\text{KgCO}_2}{1\text{KgFuel}} * \frac{1\text{KgFuel}}{1,388 \text{ litros}}$$

$$\text{CO}_{2\text{pl}} = 2,272 \frac{\text{KgCO}_2}{\text{litro}}$$

El valor de 2,272 kgCO₂/l al multiplicarlo con el consumo de combustible como se mencionó en la ecuación [2.8] se logró obtener el factor de emisión de CO₂ en [gramos/kilometro].

2.3.4 DESARROLLO DE MODELOS DE PREDICCIÓN

Para el desarrollo de este trabajo de grado se optó por el uso de Google Colab, el cual es una plataforma rápida y sencilla que tiene una configuración muy fácil de uso; es un servicio de cuaderno de programación que se relaciona con Python con características en modo de ejecución GPU; Google Colab ayudó a desarrollar el modelo de predicción gracias a que se relaciona perfectamente con el lenguaje de programación ya mencionado y BIG DATA.

En esta sección se presenta los diversos modelos de predicción usados en este trabajo de grado, los modelos de predicción se pueden dividir para datos categóricos y datos numéricos. La naturaleza de la información recolectada en este proyecto es numérica por

lo cual se eligió todo lo relacionado con regresiones lineales; con este principio se escogieron los siguientes:

- **Regresión Lineal Simple:** este modelo interactúa de forma lineal donde involucra una variable dependiente (Y =resultado) y una o varias variables independientes ($X_0+X_1+X_2+X_3+\dots X_n$ =Causas) de los datos ingresados de esta manera logra predecir el comportamiento y el estado al que se desea llegar.
- **Redes Neuronales Multicapa:** este proceso se asemeja o intenta simular las redes neuronales biológicas, en este trabajo se usó redes neuronales multicapa las cuales poseen capas de entrada(sensoriales), capas ocultas(procesamiento) y una capa de salida(resultado).
- **Random Forest Regresión:** también llamado Bosque Aleatorios de Regresión estos bosques son algoritmos provenientes del aprendizaje supervisado, es decir este modelo usó varias predicciones de algoritmos antiguos de machine learning para dar como resultado una predicción más exacta.
- **Árbol de Decisión Regresión:** en este modelo la variable dependiente se caracteriza por ser cuantitativa, es decir, en base a diversas reglas que tienen sucesión pueden llegar a la toma de una decisión.
- **Máquina de Vectores de Soporte Regresión:** este modelo es un aprendizaje supervisado, el cual los datos de entrenamiento y validación se encuentran dispersados en un hiperplano y los clasifica en de manera que desarrolla un sub modelo interno para predecir una clasificación de nuevos datos numéricos.

2.3.4.1 Importación de Librerías

Para el desarrollo de los modelos de predicción se necesita la importación de diversas librerías que fueron fundamentales para la ejecución de este trabajo y que se muestra a continuación.

IMPORTACIÓN DE LIBRERIAS

```

import pandas as pd
import numpy as np
import pylab as pl
import tensorflow as tf
import matplotlib.pyplot as plt
import seaborn as sns
import math
from sklearn.model_selection import train_test_split
#REGRESIÓN LINEAL
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
from sklearn.metrics import mean_squared_error
from sklearn.metrics import confusion_matrix
from sklearn.preprocessing import StandardScaler
#MAQUINA DE VECTORES
from sklearn.svm import LinearSVC
from sklearn.svm import SVC
from sklearn.svm import SVR
#RANDOM FOREST
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
#REDES NEURONALES
from tensorflow import keras
from keras import models
from keras import layers
from keras import optimizers
from keras import callbacks

```

Figura 2.15 Importación de librerías

En la Figura 2.15 muestra las librerías más importantes como pandas, tensorflow, numpy, matplotlib y keras, que ayudaron tanto al entrenamiento como al aprendizaje de máquina de esta forma en un solo cuaderno se pudo ejecutar varios modelos que nos permiten elegir el más adecuado para la prueba realizada. Cada una de estas librerías asume una acción como se describe brevemente a continuación:

- **Librería Pandas (pd):** es un paquete de estructura de datos, esta librería permitió una lectura y escritura de ficheros de los datos ingresados en formato CSV, también permitió el acceso de estos por medio de índices o nombres para las columnas y filas y poder trabajar con cadenas o llamadas comúnmente series temporales.
- **Librería Numpy (np):** esta librería se especializa en cálculos matemáticos, gracias a esta librería permitió el cálculo y análisis de los datos ingresados registrados por el datalogger lo que generó un gran volumen de estos, esta librería incorpora “arrays” que es una estructura de datos que se organiza en forma de tabla o cuadrículas, esto ayudó a una manipulación más eficiente de los datos ingresados al cuaderno de programación en Google Colab.

- **Librería Tensorflow (tf):** es un código abierto, esto permitió un marco general para la ejecución de operaciones matemáticas que puede ser de forma paralela y de manera distributiva como son las regresiones lineales, debido al registro de datos con un volumen amplio mostró la distribución en un plano bidimensional la cual se pudo apreciar las variables dependientes e independientes para los modelos de predicción.
- **Librería Keras:** es una librería que se utilizó específicamente para redes neuronales, al ser un código abierto provee de “*frontend*” que sirvió para la modelización de sistemas neuronales y la clase “*backend*” que se usó para el entrenamiento de estos modelos.
- **Librería Matplotlib (plt):** esta librería permitió la ejecución de gráficos mostrados en 2 dimensiones de los datos registrados en diferentes clases como dispersión o puntos, líneas, o mapas de calor.

2.3.4.2 Importación de Datos de la Ruta RDE

En la importación de datos se recomienda guardar los datos en la plataforma Google drive para obtener una importación más rápida y precisa de los datos, esto es posible gracias a la librería pandas sin olvidar que los datos CSV deben estar separados por un “;” como se muestra a continuación:

IMPORTACIÓN DE DATOS

```
[ ] from google.colab import drive
    drive.mount('/content/drive')

[ ] dataset = pd.read_csv ("/content/drive/MyDrive/DATOSCSV/RUTARDECOMPLETAFICHERO.csv",
                           sep=";", encoding="UTF-8")
    dataset.head()
```

Figura 2.16 Importación de datos de la Ruta RDE

La Figura 2.16 muestra el código y la respectiva importación de datos correspondiente a la ruta ejecutada en la ciudad de Ibarra siguiendo una normativa RDE, previamente con las variables ya calculadas para la estimación del CO₂. A continuación, se muestra las variables de operación del vehículo ingresados a la plataforma gracias a la librería pandas.

A continuación, se muestra el conteo exacto de las variables gracias al código “*dataset.describe()*”.

	TIEMPO	ALTITUD	RPM	CC(L/h)	CC(L/km)	CO2(g/km)
count	5979.000000	5979.000000	5979.000000	5979.000000	5979.000000	5979.000000
mean	2990.000000	2210.206289	1491.035110	1.648108	0.096384	218.997334
std	1726.132961	105.847474	671.130952	0.973202	0.063684	144.660593
min	1.000000	1736.500000	644.390000	0.640000	0.033000	75.190000
25%	1495.500000	2231.350000	774.780000	0.760000	0.043000	97.960000
50%	2990.000000	2236.750000	1415.920000	1.370000	0.077000	175.070000
75%	4484.500000	2248.200000	1991.065000	2.250000	0.129000	293.790000
max	5979.000000	2333.800000	3381.120000	5.180000	0.358000	814.310000

Figura 2.17 Conteo de datos de la Ruta RDE

En la Figura 2.17 muestra el número total del registro de datos de esta forma ayudó a corroborar que la importación del registro en formato CSV esté completo.

2.3.4.3 División de datos a Variables Dependientes e Independientes

Para los diversos modelos de predicción usados en esta investigación fue importante la realización de división de datos, es decir, en las variables dependientes y las variables independientes como se muestra a continuación.

ENTRENAMIENTO

```
[32] x = df.iloc[:, :-1].to_numpy(dtype=float)
      y = df.iloc[:, -1].to_numpy(dtype=float)
```

Figura 2.18 Selección de variables dependientes e independientes

La Figura 2.18 muestra la selección de las variables independientes y dependientes usando “*iloc*” que es un localizador de índices con el cuál se pudo seleccionar todas las columnas

de nuestros datos exceptuando la última donde se encuentra nuestra variable a predecir, del mismo modo se hizo uso del localizador para las variables dependientes. Usando el comando “*print ()*” tanto para “X” & “y” se puede visualizar; esta división se usó en los modelos descritos anteriormente como la regresión lineal, redes neuronales multicapa, árbol de decisiones, random forest y máquina de vectores de soporte de regresión lineal.

2.3.4.4 Correlación de los Datos Ingresados

La correlación de los datos ingresados se utilizó la librería “*matplotlib.pyplot*” que permitió una observación de cómo se relacionan los datos unos a otros, no obstante, en los datos ingresados existen coeficientes y variables constantes que no son necesarias y ralentizan la ejecución del código, por ende, se hizo uso de otra variable de paso para seleccionar variables relevantes.

```
datos = dataset[["TIEMPO", "ALTITUD", "MAP(kPa)", "RPM", "V(km/h)", "IAT(°k)",
                "TPS", "CC(L/s)", "CC(L/km)", "DensidadEntrada(kg/m3)",
                "MAF(g/s)", "RendimientoVolumetrico", "CC(L/h)", "CO2(g/km)"]]
datos.head()
```

Figura 2.19 Identificación de Datos relevantes

En la Figura 2.19 se observa la variable de paso utilizada para encontrar las correlaciones de los datos más relevantes en este proyecto, al hacer uso de otra lista se logró descartar ciertas variables donde sus datos no ofrecían valores numéricos ni caracteres importantes.

```
corr_val = datos[["TIEMPO", "ALTITUD", "MAP(kPa)", "RPM",
                 "V(km/h)", "IAT(°k)", "TPS",
                 "DensidadEntrada(kg/m3)", "MAF(g/s)",
                 "RendimientoVolumetrico", "CC(L/h)", "CO2(g/km)"]]
sns.pairplot(corr_val , kind="scatter")
plt.show()
```

Figura 2.20 Correlación de datos relevantes

La Figura 2.20 muestra la opción “*sns.pairplot*” y se logró correlacionar la variable a predecir que es la emisión CO₂ (variable dependiente) con las variables independientes

como las RPMS, velocidad, consumo de combustible, Flujo másico del Aire, temperatura del aire de admisión, presión del múltiple de admisión y altitud.

Estos códigos proporcionaron como resultado, imágenes del comportamiento de los datos unos con otros y su correlación como se observa a continuación.

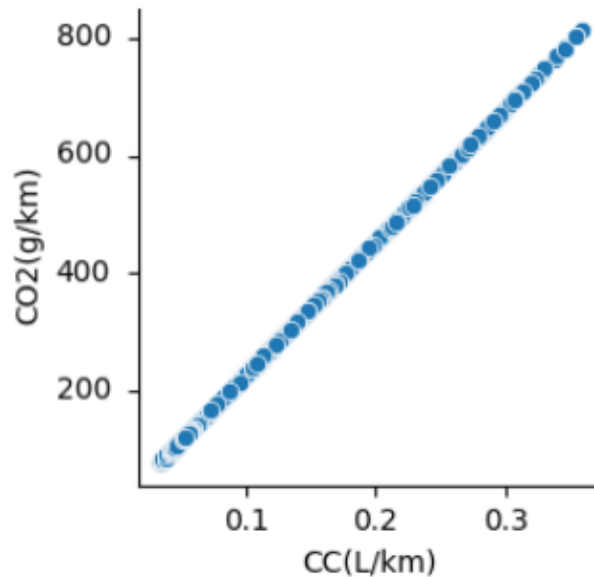


Figura 2.21 Comportamiento y dispersión de CO₂ vs Consumo Combustible

En la Figura 2.21 muestra un comportamiento lineal (pendiente positiva) que la dispersión de datos entre la emisión CO₂ y la variable del consumo de combustible se relaciona de manera directamente proporcional, en conclusión, la emisión aumenta conforme el consumo de combustible se eleva. Es importante recordar que se logró obtener el comportamiento de la emisión CO₂ con todas la variables restantes, no obstante, la relación CO₂(g/km) vs CC(l/km) es la que mayor relevancia tiene en este proyecto.

En la correlación de datos también se optó por utilizar los mapas de calor y los coeficientes de correlación los cuales permitieron un mejor entendimiento en el comportamiento de los datos unos a otros como se muestra a continuación.

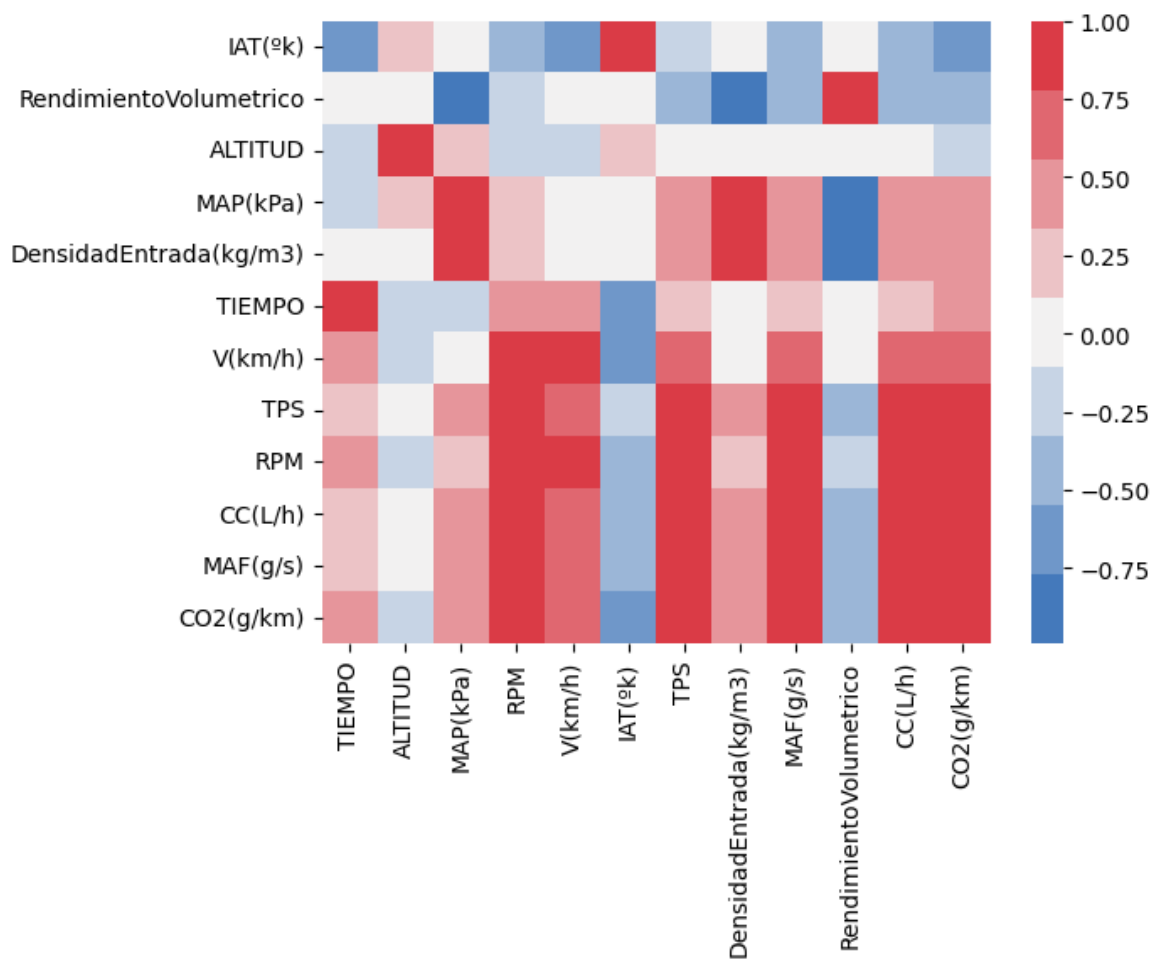


Figura 2.22 Mapa de Calor y coeficientes de correlación de los datos obtenidos

En la Figura 2.22 se puede visualizar un mapa de calor que hizo uso de la opción “*sns.heatmap*” y permitió conocer una correlación de variables por medio de colores y de valores específicos; en este mapa se encuentra las variables de operación más relevantes y por supuesto la emisión de CO₂.

Para la interpretación correspondiente de los mapas de calor realizados en la ruta RDE se toman en cuenta los valores de correlación dados en la tabla 2.7 que se muestra a continuación; la interpretación de los datos obtenidos en la Figura 2.22 se puede visualizar la correlación de la variable dependiente en este caso la emisión de CO₂ y las variables de operación del vehículo y gracias al comando “*corr()*” se usó el coeficiente de correlación “*spearman*”.

Tabla 2.7 Rangos de Correlación en mapas de calor

RANGO	CRITERIO
0,7 a 1,00	Correlación Positiva Fuerte
0,5 a 0,7	Correlación Positiva Moderada
0,2 a 0,5	Correlación Positiva Baja
-0,2 a 0,2	Correlación (Positiva o Negativa) Débil o Nula
-0,2 a -0,5	Correlación Negativa Moderada
-0,5 a -0,7	Correlación Negativa Moderada
-0,7 a -1,00	Correlación Negativa Fuerte

Fuente: (UTECH, 2021)

El coeficiente “*spearman*” hace que las variables de operación y las variables calculadas poseen una distribución poblacional de conjunto normal, es decir, que posee grados de agrupación lineal entre las variables que son objeto de estudio ya que estas siguen un comportamiento normal, el coeficiente ya mencionado se basa en rangos, de esta forma los valores de la variable de emisión “ $CO_2(g/km)$ ” al tener una correlación lineal y este coeficiente al ser independiente puede ser usada para cualquier incógnita lo cual es una ventaja (Morales & Rodríguez, 2016,p.6).

A continuación, muestra los rangos de relación de cada una de las variables ya mencionadas en base al mapa de calor ya visualizado.

```
corr_val.corr("spearman")['CO2(g/km)']
```

TIEMPO	0.436689
ALTITUD	-0.162143
MAP(kPa)	0.398486
RPM	0.925169
V(km/h)	0.760655
IAT(°k)	-0.585269
TPS	0.864858
DensidadEntrada(kg/m3)	0.430759
MAF(g/s)	0.977373
RendimientoVolumetrico	-0.420234
CC(L/h)	0.977119
CO2(g/km)	1.000000
Name: CO2(g/km), dtype: float64	

Figura 2.23 Coeficientes de correlación del mapa del calor de las variables

La Figura 2.23 tomó en cuenta la variable principal que es la emisión CO₂ con respecto a las variables más relevantes como las RPM, Velocidad, datos del cálculo del sensor MAF, datos del sensor MAP, datos del sensor IAT, Consumo de Combustible, Altitud y Tiempo.

La relación entre los cálculos de la estimación de emisiones CO₂ se puede observar una “*Correlación Positiva Fuerte*” entre los datos del sensor MAF y el Consumo de Combustible, pues estas variables fueron autores principales para las emisiones, por ende, se esperaba una pendiente positiva y una compatibilidad fuerte.

No obstante, en la Figura 2.23 también se muestra la correlación con las variables restantes como la velocidad, datos del sensor MAP y RPM que están dentro de un rango comprendido entre 0,5 a 0,7 considerándose una “*Correlación Positiva Moderada*”; es decir, una correlación indirecta que indicó que los datos de estas variables no se utilizaron para la estimación de gases, pero fueron de suma importancia para los datos del sensor MAF y del Consumo de Combustible.

Por otro lado, tenemos “*Correlaciones Negativas o Nulas*” entre la emisión CO₂ y los datos de sensor IAT, Altitud y Tiempo donde los datos no mostraron un patrón de comportamiento lineal, en otras palabras, la emisión no depende de estas variables pues se observaría algo ilógico que las emisiones provocadas por un vehículo liviano aumentarían o disminuirían por la altitud o tiempo directamente. Estos argumentos resumen de forma más sencilla los mapas de calor y los rangos de correlación entre los datos obtenidos de un vehículo liviano.

2.3.4.5 Entrenamiento de los Modelos de Predicción

El proceso desarrollado para cualquier modelo de predicción constó de 3 pasos fundamentales que fue el entrenamiento, el aprendizaje y por último la predicción; para la ejecución de este trabajo los entrenamientos en los modelos de predicción mencionados como Regresión Lineal, Redes Neuronales Multicapa, Árbol de Decisión, Random Forest y Máquina de Vectores de Soporte fue importante el uso de la división de datos para el entrenamiento y testeo como se muestra a continuación.

```
x_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.2,random_state=1)
print ("División de datos en Entrenamiento/Test Completa")
```

División de datos en Entrenamiento/Test Completa

Figura 2.24 División de datos Entrenamiento/Test

La Figura 2.24 corresponde a la división de datos donde el “*random_state*” inicia con un numero entero igual a 1 y este puede estar comprendido entre 0 y 42, este es un hiper parámetro que sirve para el estado aleatorio de datos involucrada en los modelos de aprendizaje y predicción, es decir, un punto de partida para obtener una aleatoriedad secuencial para tener el mismo resultado cada vez que se ejecutó el código.

La división de los datos en este trabajo de grado se optó por un porcentaje del 80% para el entrenamiento y un 20% para testeo o validación de los datos ingresados, estos porcentajes fueron usados para todos los modelos desarrollados en este proyecto para obtener un nivel de fiabilidad más aceptable.

2.3.4.6 Entrenamiento de Regresión Lineal Simple

En el entrenamiento de la regresión lineal se utilizó el método “*fit*” una vez desarrollada la división de los datos y definir su respectivo “*random_state*”; de esta manera dentro del conjunto de datos de este método se ingresa las variables tanto independientes que en este caso fue “*X_train*” (Variables de operación) y la variable dependiente que fue “*y_train*” (emisión de CO₂) como se muestra a continuación.

```
regresion = LinearRegression(fit_intercept = True,n_jobs= 1 )
regresion.fit(X_train, y_train, sample_weight = None)
```

▼ LinearRegression
LinearRegression(n_jobs=1)

Figura 2.25 Entrenamiento de Regresión Lineal

La Figura 2.25 muestra el entrenamiento en regresión lineal gracias a la importación de la biblioteca del lenguaje Python `“sklearn.linear_model”` y se logró integrar la librería `“LinearRegression”`; de esta forma el método `“fit”` ejecutó las correlaciones correspondientes y el cuadro azul de la figura informó que el entrenamiento se ejecutó adecuadamente.

2.3.4.7 Entrenamiento de Redes Neuronales Multicapa

En el desarrollo de una red neuronal artificial el primer paso fue la construcción de la red la cual ya se definió cuáles son las variables independientes y la variable dependiente.

El segundo paso para la construcción de la estructura de la red fue la división de datos la cual se tomó un 80% para entrenamiento y 20% para su validación tomando un `“random_state=1”`.

El siguiente paso de la estructura de red que se compone de las capas de entrada, las capas ocultas y las capas de salida se describió el número de neuronas que conforman dichas capas que servirán para el análisis mismo de los datos, la correlación que tienen unos a otros y la forma de obtener un resultado de predicción que se acerque al valor real de la muestra registrada.

CAPA DE ENTRADA (PRIMERA CAPA OCULTA)

```
[120] red = tf.keras.Sequential()
      red.add(layers.Dense(units=100,activation = "relu",use_bias=True,
                           bias_initializer="ones"))
```

SEGUNDA CAPA OCULTA

```
[121] red.add(layers.Dense(units=100,activation = "relu",use_bias=True,
                           bias_initializer="ones"))
```

TERCER CAPA OCULTA

```
[122] red.add(layers.Dense(units=100,activation = "relu",use_bias=True,
                           bias_initializer="ones"))
```

CUARTA CAPA OCULTA

```
[123] red.add(layers.Dense(units=100,activation = "relu",use_bias=True,
                           bias_initializer="ones"))
```

CAPA DE SALIDA

```
[124] red.add(layers.Dense(units=1,use_bias=True,
                           bias_initializer="ones"))
```

Figura 2.26 Estructura de la Red Neuronal Artificial Multicapa

En la Figura 2.26 muestra la red neuronal que se conformó de 4 capas y fueron creadas gracias a las librerías “*tensorflow*” y “*keras*” de esta forma la red realizó un aprendizaje profundo e igualmente se añadió la clase “*Sequential()*” del módulo “*models*” y los métodos necesarios como la clase “*Dense*” y el módulo “*layers*” que ayudó a implementar los parámetros que añaden las neuronas en cada capa con la clase “*unit*”.

La red neuronal hizo uso de 100 neuronas en cada una de las capas ocultas obteniendo que los modelos sean estables ya que no tienen un sobre entramiento y disminución de probabilidades. El segundo parámetro es el activador “*activation*”, que nos indica la composición de las funciones, en nuestro caso el activador es un “*relu*”, (figura 2.26). La última capa (salida) es una salida binaria (1 o 0) solo se debe optar por una neurona ya que se necesita un solo valor, con estos antecedentes la estructura de la red neuronal solo se conformó de una sola neurona para lograr un resultado de manera binaria que en este caso fue igual a 1. En la fase de estructura también se encuentra la compilación de red este es

un paso muy importante que se realizó con el método “*compile*” que usó 3 parámetros que se describen a continuación:

- Optimizador = “*Optimizer*”
- Pérdidas = “*Loss*”
- Métricas = “*metrics*”

La optimización “*Optimizer*” utilizó el método “*Adam*” ya que es muy eficiente sobre todo para datos de naturaleza lineal, la muestra registrada en este proyecto tiene la dispersión de datos en su mayoría la forma “*lineal*” y esto permitió que el método se integre mejor a la estructura.

Las pérdidas se obtienen gracias a “*Loss*” que utilizó el método “*mean_squared_error*”, que permitió medir la diferencia entre los datos que se pudo predecir con los datos reales, es decir la suma de todas las pérdidas del modelo. Este método también fue usado en la regresión lineal y este es parte importante de la biblioteca “*sklearn.metrics*”. La métrica “*metrics*” define la precisión utilizando el método “*mse*” es decir; la afinación del modelo que se realizó, todos estos parámetros se pueden observar a continuación en la siguiente figura.

COMPILACIÓN DE LA RED

```
[126] from keras.metrics import accuracy
      red.compile(optimizer=tf.keras.optimizers.Adam(0.05), loss=["mean_squared_error"],metrics = ["mse"])
```

Figura 2.27 Compilación de la Red Neuronal Multicapa

En la Figura 2.27 muestra la compilación de la red donde se hace uso de los parámetros ya descritos para obtener una correlación de los datos y de la red misma, de esta forma una vez concluida la compilación dar paso al entrenamiento.

El entrenamiento se realizó con el método “*fit*” en donde sus parámetros se componen de datos de entrenamiento de las variables independientes y dependientes “ (X_{train}, y_{train}) ”, y “*epochs*” que son las corridas o interacciones en los datos implementados, es decir que generó una barrida de todo para llegar a un valor de pérdida y retroalimentar al modelo y

la obtención de un resultado óptimo, cada interacción o “*epochs*” es diferente para cada dato, acoplándose a cada uno como se observa a continuación.

ENTRENAMIENTO

```
[127] entrenamiento = red.fit(X_train, y_train, epochs=20, verbose = "auto", validation_data=(X_train,y_train))

Epoch 1/20
150/150 [=====] - 3s 10ms/step - loss: 17155.8496 - mse: 17155.8496 - val_loss:
Epoch 2/20
150/150 [=====] - 1s 8ms/step - loss: 59.4371 - mse: 59.4371 - val_loss: 78.3267
```

Figura 2.28 Entrenamiento de la Red Neuronal Multicapa

En la Figura 2.28 se muestra el entrenamiento que usó la interacción o intervalo de 20 “*epochs*” (“*épocas*”) estas son las vueltas que dio el algoritmo para usar los datos; con el fin de evitar el sobre entrenamiento de la máquina y la sobre limitación, además con este valor las pérdidas fueron diminutas incluso llegando a ser menores que 0 lo que aumentó las probabilidades de poseer una fiabilidad de predicción favorable.

2.3.4.8 Entrenamiento de Árbol de Decisión Regressor

Este algoritmo fue usado debido a su aprendizaje supervisado que no es paramétrico, con los datos ingresados de la muestra registrada en la ruta RDE se pudo realizar nodos de raíz, ramas, hojas y por ende los nodos internos. La similitud de los códigos de los demás modelos fue utilizada para el árbol de decisión el cual se tomó el 80% para entrenamiento y 20% para su validación y un “*random_state*” igual a 1; de la misma forma se utilizó el método “*fit*” y la librería “*DecisionTreeRegressor*” para la ejecución del entrenamiento como se muestra a continuación.

```

arbolregresion = DecisionTreeRegressor(criterion = 'squared_error', splitter = 'best',
                                     min_samples_split=5,max_depth = 10,
                                     random_state=1, min_samples_leaf=5,
                                     min_weight_fraction_leaf = 0.0,
                                     max_features=10, max_leaf_nodes = 4,
                                     min_impurity_decrease=0.0, ccp_alpha = 0.0)
arbolregresion.fit(X_train, y_train)

```

DecisionTreeRegressor

DecisionTreeRegressor(max_depth=10, max_features=10, max_leaf_nodes=4,
min_samples_leaf=5, min_samples_split=5, random_state=1)

Figura 2.29 Entrenamiento de Árbol de Decisión

En la Figura 2.29 muestra el entrenamiento del modelo Árbol de Decisión usando los métodos y librerías correspondientes usando 4 nodos en el árbol.

2.3.4.9 Entrenamiento de RandomForest Regressor

Los modelos ya antes mencionados se ha observado la definición de las variables tanto independientes y dependientes; de esa forma se pudo usar los códigos para diversos modelos como Random Forest o Bosque Aleatorio para constatar cual es el mejor modelo para la estimación de CO₂. La división de datos en este trabajo de grado se eligió un 80% para entrenamiento y un 20% de validación, debido a la cantidad de datos ingresados el “*random_state*” es igual a 1 para evitar un sobre entrenamiento.

En el entrenamiento de este modelo la clase “*RandomForest Regressor*” permitió añadir los parámetros para el bosque aleatorio, en este caso se utilizó un total de 150 árboles los cuales dieron un resultado de predicción favorable para el presente trabajo como se muestra a continuación.


```

modelrf = RandomForestRegressor(n_estimators=150, criterion = 'squared_error',
                               max_depth =5, min_samples_split = 5,
                               min_samples_leaf = 1,random_state=1,
                               max_leaf_nodes = 3,n_jobs = 1,verbose = True)
modelrf.fit(X_train, y_train,sample_weight=1)

[Parallel(n_jobs=1)]: Done 49 tasks      | elapsed:    0.1s
┆
┆      RandomForestRegressor
┆
┆ RandomForestRegressor(max_depth=5, max_leaf_nodes=3, min_samples_split=5,
┆                       n_estimators=150, n_jobs=1, random_state=1, verbose=True)

```

Figura 2.30 Entrenamiento del modelo RandomForest

En la Figura 2.30 muestra la clase y el número de árboles correspondientes para desarrollar el bosque aleatorio, este número de árboles evitó el sobre entrenamiento y la sobre limitación de la predicción; se hizo pruebas en un rango entre 100 a 300 árboles y al ser menor al número estimado se observó una sobre limitación, es decir el código tuvo una duración más de lo normal, y de 100 a 150 árboles pues el bosque ejecutó su código con una duración más corta; al añadir el parámetro “*criterion*” con su igualdad a “*squared_error*” el resultado fue un entrenamiento adecuado pues el criterio permitió encontrar la modelación de los datos de conteo entre las variables independientes con la variable dependiente para su posterior predicción.

2.3.4.10 Entrenamiento de Máquinas de Vectores de Soporte Regresión

Este modelo representa una agrupación de algoritmos que tienen un aprendizaje supervisado que en este caso están relacionados con una regresión lineal donde los datos divididos en entrenamiento y testeo están en un hiperplano donde las etiquetas de clasificación la máquina se entrena para desarrollar un modelo donde predice una clase de datos nuevos.

De la misma forma donde los modelos anteriores se dividen los datos en un 80% de entrenamiento y un 20% de testeo; este modelo uso la misma división con un valor de “*random_state*” igual a 1.

```
[139] regresionsvr = SVR(kernel="rbf", C=40, gamma=0.1)
      regresionsvr.fit(X_train, y_train)
```

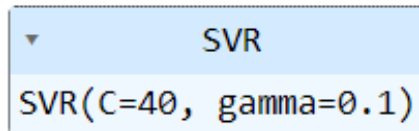


Figura 2.31 Entrenamiento de Máquina de Vectores de Soporte

En la Figura 2.31 muestra el modelo que usó el método “*kernel*” ya que posee diversas clases, en este caso se usó la clase “*rbf*” debido a la naturaleza de los datos ingresados, por otro lado, este método posee clases como “*linear*” y “*sigmoid*” que puede ser usado para operaciones matemáticas más simples y datos clasificatorios respectivamente. El cuadro azul en la figura mostró que el entrenamiento SVR se ejecutó de forma adecuada pues el coste o “*C*” el peso que se le dio a cada variable fue estable, no obstante, entre más grande fue este peso el entrenamiento se hizo más estricto; por otro lado “*gamma*” el coeficiente de kernel que fueron las interacciones en el hiperplano fue constante con el valor integrado.

2.3.5 PREDICCIÓN DE LOS MODELOS

Las predicciones se presentan a continuación y se muestran los códigos ejecutados en cada uno de los modelos ya mencionados, los entrenamientos desarrollados permitieron obtener una comparación de los valores de predicción y los valores reales de la muestra registrada y por ende los porcentajes de fiabilidad en cada uno de ellos.

2.3.5.1 Predicción de la Regresión Lineal Simple

En la predicción del modelo Regresión Lineal Simple se utilizó la clase “*predict*”, la librería “*numpy*” y el método “*concatenate*” que nos muestran los datos que se logró predecir en el modelo realizado, el método usado permitió comparar los valores reales o de validación con los datos deseados como se muestran a continuación.

PREDICCIÓN

```
[ ] y_predr1 = regresion.predict(X_test)
    np.set_printoptions(precision=2)
    print(np.concatenate((y_predr1.reshape(-1,1), y_test.reshape(-1,1)),1))

[[100.   101.04]
 [352.14 352.72]
 [124.99 124.5 ]
 ...
 [186.32 185.95]
 [422.56 422.35]
 [270.36 270.5 ]]
```

Figura 2.32 Predicción de Regresión Lineal en la Ruta RDE

En la Figura 2.32 muestra los valores de predicción deseados y los valores de muestra registrados, con esta comparación se pudo constatar y corroborar que el modelo de regresión lineal múltiple tuvo una predicción de fiabilidad incongruente.

2.3.5.2 Predicción de la Red Neuronal Multicapa

En la predicción de los datos se usó la clase “*predict*” con los datos de validación de “*X_test*” los cuales se realizó una concatenación para que el resultado esté dado en 2 columnas las cuales en el lado izquierdo estaba conformada por los datos que se lograron predecir y se comparan con la columna derecha que fueron los datos reales registrados como se muestra a continuación.

```
38/38 [=====] - 0s 2ms/step - loss: 0.5930 - mse: 0.5930
[0.5930136442184448, 0.5930136442184448]
```

```
y_predrn = red.predict(X_test)
np.set_printoptions(precision=2)
print("El valor del CO2 instantaneo es:",
      np.concatenate((y_predrn.reshape(len(y_predrn),1), y_test.reshape(len(y_predrn),1)),1))
```

```
38/38 [=====] - 0s 2ms/step
El valor del CO2 instantaneo es: [[ 99.7  101.04]
 [351.67 352.72]
 [124.67 124.5 ]]
```

Figura 2.33 Predicción de la Red Neuronal Multicapa

En la Figura 2.33 se pudo observar una mejor organización de los valores que predice la máquina y los valores previamente calculados en este capítulo, el entrenamiento y el aprendizaje se ejecutaron de forma correcta pues los datos observados en la figura se acercan a la muestra. En esta figura también se observa la evaluación al ejecutar su entrenamiento para dar paso a su predicción pues en términos generales se observa que el tiempo de ejecución de cada capa y de datos es de 2 ms con una pérdida diminuta de 0,59.

2.3.5.3 Predicción del Árbol de Decisión Regressor

De la misma forma que se ha realizado la predicción en los diversos modelos se utilizó la clase “*predict*” y su respectiva concatenación para obtener los datos que se logró predecir y compararlos con los datos de validación o reales como se observa a continuación.

```
y_predad = arbolregresion.predict(x_test)
np.set_printoptions(precision=2)
print(np.concatenate((y_predad.reshape(-1,1), y_test.reshape(-1,1)),1))
```

```
[[ 94.03 101.04]
 [339.13 352.72]
 [ 94.03 124.5 ]
 ...
 [170.95 185.95]
 [516.94 422.35]
 [258.09 270.5 ]]
```

Figura 2.34 Predicción del Árbol de Decisión Regressor

En la Figura 2.34 muestra los valores que se logró predecir; sin embargo, este modelo también puede ser usado para datos clasificatorios pues de la misma forma que en la naturaleza el árbol posee una raíz, nodos o ramas y hojas y este modelo sigue la misma estrategia ya que busca el mejor camino para llegar a un resultado concreto, pues como se observa en la figura los valores de predicción (izquierda) se acercaron de inmediato a los valores registrados(derecha).

2.3.5.4 Predicción de RandomForest (Bosque Aleatorio)

La predicción de RandomForest Regressor se usó el mismo modo de operación que se ha observado en los modelos anteriores se utilizó la clase “*predict*” y la opción “*concatenate*” para separar los datos que se logró predecir con los datos reales.

```
y_predrf = modelrf.predict(x_test)
np.set_printoptions(precision=2)
print(np.concatenate((y_predrf.reshape(-1,1), y_test.reshape(-1,1)),1))
```

```
[[105.24 101.04]
 [343.52 352.72]
 [105.24 124.5 ]
 ...
 [222.53 185.95]
 [353.18 422.35]
 [241.74 270.5 ]]
```

```
[Parallel(n_jobs=2)]: Using backend ThreadingBackend with 2 concurrent workers.
[Parallel(n_jobs=2)]: Done 46 tasks      | elapsed: 0.0s
[Parallel(n_jobs=2)]: Done 150 out of 150 | elapsed: 0.1s finished
```

Figura 2.35 Predicción del modelo Random Forest

En la Figura 2.35 muestra los resultados de predicción del modelo RandomForest y se logró una comparación con la muestra registrada. En este caso RandomForest también puede ser utilizado para casos clasificatorios y en este proyecto al tener una naturaleza de datos numéricos lineales se usó de forma de regresión lo cual la predicción se logró de manera adecuada, pues en este modelo incluso una variable puede constar como un árbol del bosque lo cual su entrenamiento y aprendizaje es más profundizado que el modelo árbol de decisión lo que permitió que la ejecución de código sea mucho más eficaz y rápida.

2.3.5.5 Predicción de Máquina de Vectores de Soporte

El modelo máquinas de vectores de soporte como se describió anteriormente utiliza un hiperplano de separación, en este proyecto de regresión este modelo estaba formado por un vector de características numéricas lineales. En este caso los datos los clasificó en diversos hiperplanos de separación y halló uno que maximizó la distancia o margen entre este y el dato más cercano de cada clase; el vector por otro lado poseía la ponderación de cada

atributo o cada característica de cada dato que ayudó a una aportación en los procesos de clasificación.

```
y_predmv = regresionsvr.predict(X_test)
np.set_printoptions(precision=2)
print(np.concatenate((y_predmv.reshape(-1,1), y_test.reshape(-1,1)),1))
```



```
[[117.83 101.04]
 [310.13 352.72]
 [136.87 124.5 ]
 ...
 [183.63 185.95]
 [363.8  422.35]
 [247.75 270.5  ]]
```

Figura 2.36 Predicción de Máquina de Vectores de Soporte

En la Figura 2.36 muestra los valores de predicción vs los valores reales; este modelo es uno de los más confiables incluso su nivel de fiabilidad es igual a las Redes Neuronales Artificiales ya que su aprendizaje y entrenamiento permite una correcta clasificación de la muestra; las funciones kernel permitieron la ejecución de las operaciones algebraicas de una manera eficiente y por ende el análisis de múltiples variedades para datos logró reformular los algoritmos computacionales en productos escalares haciendo los resultados más eficaces y confiables.

CAPÍTULO III

3. RESULTADOS Y DISCUSIÓN

En el siguiente capítulo se detallan los valores obtenidos en los diferentes modelos vs la muestra registrada en la ruta que se rige bajo la normativa RDE; con los procesos descritos anteriormente se logró la obtención de los porcentajes de fiabilidad que mostraron los diversos modelos y de esa forma poder compararlos entre ellos para determinar qué modelo es el más confiable. A continuación, se ejecutó una comparación del valor encontrado en esta investigación con los resultados generados por los modelos ejecutados.

CÁLCULO DEL CO2 TOTAL GENERADO EN TODA LA RUTA RDE (g/km)

```
▶ print("EL CO2 total generado por el vehiculo es de: ",sum(df['CO2Inst']) / DTtotal,"g/km")
↳ EL CO2 total generado por el vehiculo es de: 126.96397834016226 g/km
```

CÁLCULO DEL CONSUMO DE COMBUSTIBLE

```
[11] print("EL Consumo de Combustible total del vehiculo es de: ",sum(df['ConsumoCombustible']),"L")
EL Consumo de Combustible total del vehiculo es de: 2.72926398070956 L
```

Figura 3.1 Factor de Emisión y Consumo de Combustible Cherry Arrizo 5 Pro

En la Figura 3.1 muestra el consumo de combustible y el factor de emisión del vehículo de prueba usado en este trabajo de grado el cual es el dato principal con el cual se comparó los resultados que entrega cada modelo ejecutado.

3.1 ESTIMACIÓN DE LA EMISIÓN CO₂ y CONSUMO DE COMBUSTIBLE

Las ecuaciones usadas en esta investigación permitieron encontrar los siguientes resultados en cada una de las rutas correspondientes a las zonas de la ruta RDE como se muestra a continuación.

3.1.1 CONSUMO DE COMBUSTIBLE EN LA ZONA URBANA

En base a todo lo mencionado anteriormente las ecuaciones y las especificaciones correspondientes se obtuvo resultados del consumo de combustible con un total de 1,467 litros que corresponde a la zona urbana con los siguientes parámetros:

- **Tiempo:** 4214 seg
- **Velocidad Promedio:** 20,50 km/h
- **Velocidad Máxima:** 60,60 km/h
- **Número de Paradas:** 39 paradas
- **Número de Semáforos:** 23 semáforos

3.1.2 CONSUMO DE COMBUSTIBLE EN LA ZONA RURAL

En la zona rural que conforma la ruta RDE se desarrolló el mismo procedimiento que en la zona urbana; los resultados del consumo de combustible en este tramo se obtuvo un total de 0,809 litros que se dieron con los siguientes parámetros presentados en esta zona:

- **Tiempo:** 1283 seg
- **Velocidad Promedio:** 40 km/h
- **Velocidad Máxima:** 75 km/h
- **Número de Paradas:** 7 paradas
- **Número de Semáforos:** 5 semáforos

3.1.3 CONSUMO DE COMBUSTIBLE EN LA ZONA DE AUTOPISTA

El consumo de combustible en este tramo se desarrolló de la misma forma que los tramos ya mencionados (Rural, Urbano); en esta zona se logró alcanzar una velocidad promedio entre los 90 y 100 km/h, con un consumo de combustible equivalente a 0,450 litros.

- **Tiempo:** 846 seg
- **Velocidad Promedio:** 85 km/h
- **Velocidad Máxima:** 105 km/h

- **Número de Paradas:** 0 paradas
- **Número de Semáforos:** 0 semáforos

La Tabla 3.1 muestra los valores que puede llegar consumirse en un recorrido de hasta 100 km tanto en la ruta RDE y los que se muestra en su ficha técnica, al compararlos es evidente que el mayor consumo es en la zona urbana, seguido por la zona rural y por último la zona de autopista; en esta última su consumo es menor debido a que el vehículo no ejerce un esfuerzo mecánico elevado como en la zona urbana y rural donde las aceleraciones, desaceleraciones, pendientes (+, -), cambios de marcha constantes, paradas por tráfico vehicular, rompe velocidades, incluso imprevistos como maniobras de evasión para evitar accidentes y lo más importante estilos de conducción provocan un consumo sumamente alto lo que provoca de igual manera una mayor contaminación ambiental.

Tabla 3.1 Comparación de resultados del Consumo de Combustible

Zona	C.C. Ciudad de Ibarra Normativa RDE (l/100km)	C.C. Cherry Arrizo 5 Pro (l/100 km)
Urbana	8,6	7,9
Rural	7,9	6,2
Autopista	6,3	7,0

3.1.4 ESTIMACIÓN DE LA EMISIÓN CO₂ EN RUTA ZONA URBANA

El cálculo de la emisión CO₂ en esta zona se desarrolló de forma instantánea gracias a las ecuaciones ya mencionadas; tomando en cuenta que en esta zona existieron un número considerable de paradas debido al tráfico vehicular y los semáforos encontrados con un total de 39 y 23 respectivamente en diversos puntos; el factor de emisión CO₂ de este tramo se obtuvo un valor de 173,738 g/km.

3.1.5 ESTIMACIÓN DE LA EMISIÓN CO₂ EN RUTA ZONA RURAL

En el tramo rural de igual manera que el urbano se hizo uso de las ecuaciones ya mencionadas; en esta zona al recorrer por el anillo vial no se presentó varias paradas sino con un total de 7 aunque dichas detenciones tuvieron un periodo de tiempo considerable, por otro lado el número total de semáforos fue de 5 que de la misma forma el tiempo de

detención estuvo en el mismo rango de tiempo que las paradas por tráfico, por ende el factor de emisión CO₂ fue equivalente a 115,248 g/km un tanto menor al factor de emisión de la zona urbana.

3.1.6 ESTIMACIÓN DE LA EMISIÓN CO₂ EN RUTA ZONA DE AUTOPISTA

Las emisiones de CO₂ en esta zona se realizó de la misma forma que las otras zonas ya mencionadas; en base a estos cálculos se llegó a obtener un factor de emisión CO₂ de 74,692 g/km, en este tramo careció de paradas que tomaran más de 10 seg sino más bien existieron altas velocidades y aceleraciones abruptas que se presentaron en la vía por el intento de rebasar a los vehículos pesados en la autopista. Con los resultados que se logró obtener en cada una de las zonas, la Tabla 3.2 muestra el valor promedio de 126,96 gCO₂/km que se aproxima al valor real dado por la ficha técnica donde se comprobó que el método y las ecuaciones mencionadas en esta sección de este capítulo son los adecuados para llegar a un resultado idóneo.

Tabla 3.2 Emisión de CO₂ Producida por Cherry Arrizo 5 Pro

Zona	Emisión de CO₂ Ruta RDE en la Ciudad de Ibarra (g/km)	Emisión de CO₂ Cherry Arrizo 5 Pro Ficha Técnica (g/km)
Urbana	173,738	N/D
Rural	115,258	N/D
Autopista	74,692	N/D
Valor Promedio	126,96	118

3.2 RESULTADOS DEL MODELO REGRESIÓN LINEAL SIMPLE

Para lograr el resultado deseado se utilizó el método de “R²” o coeficiente de determinación que definió el porcentaje (%) de la varianza de la variable independiente “Y”; entre más variables independientes “X”, el “R²” dio como resultado un porcentaje más aceptable y favorable como se muestra a continuación.

EVALUACIÓN

```

▶ r2r1 = r2_score(y_test, y_predr1)*100
  print("El porcentaje de la variación en la variable de respuesta (R^2)es:",r2r1,"%")

```

El porcentaje de la variación en la variable de respuesta (R^2)es: 99.77565646093981 %

Figura 3.2 Coeficiente de determinación de Regresión Lineal Simple

En la Figura 3.2 muestra el porcentaje de determinación que se obtuvo gracias a los datos del Datalogger lo cual muestra un 99,77% de fiabilidad; a pesar de tener un valor cercano al 100% este modelo no es fiable para un modelo de Regresión Lineal pues el tiempo de ejecución de la predicción duró menos de 1 seg lo cual este resultado indaga que la predicción solo usó los valores de testeo para mostrar un resultado.

```

[44] df = pd.DataFrame({"Muestra Registrada":y_test, "Predicción":y_predr1})
      df.head()

```

	Muestra Registrada	Predicción
0	1.771	1.770695
1	3.025	3.025343
2	0.837	0.836839
3	1.059	1.059074
4	5.036	5.036127

Figura 3.3 Comparación de los valores obtenidos y la muestra registrada

La Figura 3.3 haciendo uso de la librería “*pandas*” y los métodos “*DataFrame*” y “*head*” se logró obtener una comparación y visualización simple de los datos registrados y los datos de predicción los cuales se observa que los datos obtenidos son exactamente iguales a la muestra real. A continuación, se muestra la comparación ejecutada de manera gráfica:

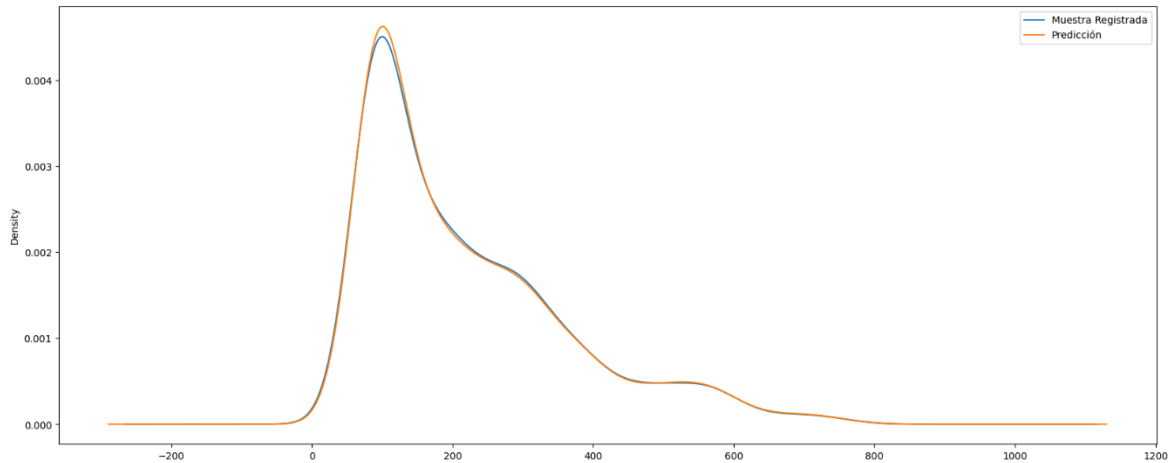


Figura 3.4 Gráfico de comparación de datos obtenidos

La Figura 3.4 muestra la comparación de los datos de manera gráfica en el modo “*density*”; los datos que se logró predecir están representados de color naranja y los datos de la muestra registrada están representados de color azul; no obstante, en el gráfico no se diferencia los datos pues están superpuestos unos a otros. En esta gráfica se observa que la densidad de los valores de predicción no es mayor ni menor a la densidad de la muestra registrada lo que implica que la distribución de los datos son exactamente iguales.

En la regresión lineal simple también se ejecutó los códigos necesarios para encontrar los errores cuadráticos y absolutos que se presentaron en el modelo como se muestra a continuación.

```
print('Error absoluto medio (MAE):', metrics.mean_absolute_error(y_test, y_predr1))
print('Error cuadrático medio(MSE):', metrics.mean_squared_error(y_test, y_predr1))
print('Raíz del Error cuadrático medio(RMSE):', np.sqrt(metrics.mean_squared_error(y_test, y_predr1)))
```

```
Error absoluto medio (MAE): 0.0024307621575274246
Error cuadrático medio(MSE): 8.24865764122509e-06
Raíz del Error cuadrático medio(RMSE): 0.0028720476390939427
```

Figura 3.5 Errores presentados en el modelo Regresión Lineal Simple

La Figura 3.5 gracias al método “*metrics*” se logró obtener el error absoluto medio (MAE) que es la cuantificación de precisión entre la predicción y la muestra registrada y el error cuadrático medio (MSE) que es el valor de error entre los datos obtenidos y los valores reales registrados.

En esta figura se muestra el MAE con un valor del 0,000243 que significa en valores porcentuales que el modelo tiene un 0,02% en comparación de los valores obtenidos y los valores de la muestra registrada; por otro lado, el MSE se logró sacar su raíz cuadrada dando como resultado el RMSE con un valor del 0,000287 que en valores porcentuales el modelo tiene un 0,02% de error entre el conjunto de datos reales y los datos que obtuvo en la predicción, en conclusión entre más pequeño sea el valor del RMSE y MAE los valores pronosticados se acercan más a los valores registrados y viceversa, no obstante estos valores son menores que 0 lo cual corrobora que el modelo funciona.

A continuación, se muestra el resultado del modelo predicción:

```
CC = [[2.729]] # [L] Consumo de Combustible en Litros
d = [[48.8]] # [km] Distancia Recorrida
i = np.array(CC) / np.array(d)
print("El Consumo de Combustible es de: ",str(i),"L/km")
prediccion = regresion.predict(i)
print("El resultado del Factor de emision es de: ",str(prediccion),"gCO2/km")
```

```
El Consumo de Combustible es de: [[0.06]] L/km
El resultado del Factor de emision es de: [125.01] gCO2/km
```

Figura 3.6 Ejecución de predicción con valores aleatorios

La Figura 3.6 muestra los valores ingresados del vehículo Cherry Arrizo; con estos datos se logró un resultado de un Factor de Emisión de 125,01 gCO₂/km comparado con el valor real que es de 126,96 gCO₂/km que proporcionó un error relativo del 1,53% y se observa que no se aleja mucho de la realidad.

3.3 RESULTADOS DEL MODELO REDES NEURONALES MULTICAPA

Los valores obtenidos en este modelo con respecto a emisiones prometen ser un candidato ideal, tomando en cuenta que posee solamente 4 capas de las cuales las 3 capas ocultas poseen 100 neuronas y la de salida que tiene solo 1 neurona. En esta ocasión se evitó el sobreajuste o sobre entrenamiento que tomaba solo datos de entrenamiento y validación

para dar un resultado, las cuatro capas permitieron un entrenamiento más rápido y resultados más confiables como se muestra a continuación.

EVALUACIÓN

```
[32] r2rn = r2_score(y_test, y_predrn)*100
      print("El porcentaje de la variación en la variable de respuesta (R^2)es:",r2rn,"%")
```

El porcentaje de la variación en la variable de respuesta (R^2)es: 93.81694515051102 %

Figura 3.7 Coeficiente de determinación de Red Neuronal Multicapa

En la Figura 3.7 se observa una confiabilidad muy elevada; los resultados que se obtuvo muestra un valor de 0,9381 que mostrada en un porcentaje sería del 93,81% esto demuestra que los valores no están sobre entrenados ya que previamente se hizo una varianza de las interacciones o “*epochs*” llegando a considerar un valor de 30 pero el valor no tuvo una variación significativa por lo cual se redujo a 20 y el resultado se mantuvo con pérdidas sumamente diminutas y con un valor semejante en todas las corridas del código.

En otra instancia también se pudo obtener de manera gráfica las validaciones del entrenamiento y las pérdidas generadas en el modelo mencionado como se muestra a continuación .

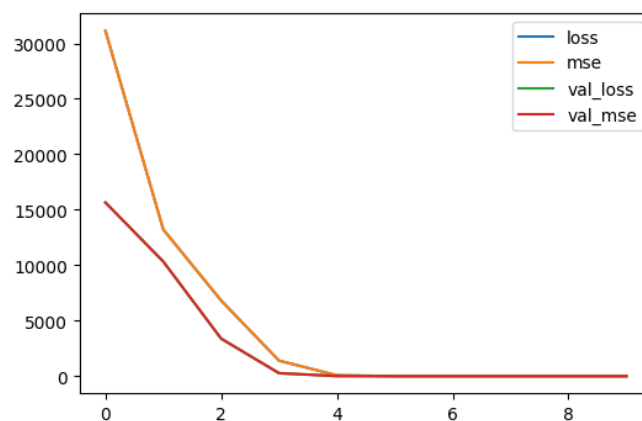


Figura 3.8 Validación de la Red Neuronal Multicapa

En la Figura 3.8 se muestra el entrenamiento y su validación representada por el color rojo y naranja a lo largo de las “*épocas*” o “*epochs*” ingresadas; de la misma forma las pérdidas y su validación representadas de color azul y verde respectivamente y se mostró una

disminución a lo largo de las “*epochs*” con una cercanía a los datos de entrenamiento favorable; esto en conclusión significa que el sobreajuste o “*overfitting*” no se presentó en la ejecución de la red neuronal multicapa. Este seguimiento se logró observar de mejor manera con el grafico de barras que se muestra a continuación:

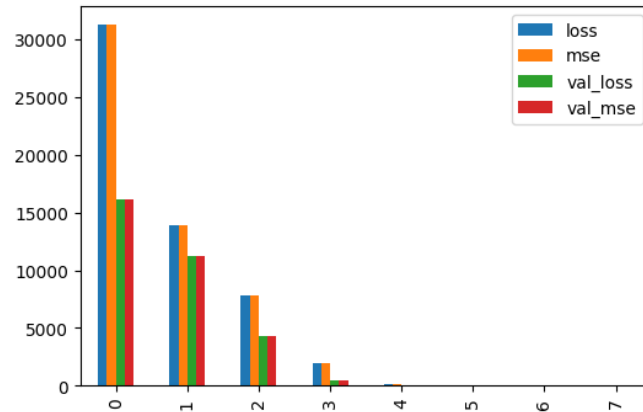


Figura 3.9 Pérdidas y Entrenamiento a lo largo de las Epochs

En la Figura 3.9 se observa de mejor manera la validación del entrenamiento, las pérdidas y sus validaciones a lo largo de las épocas lo cual las pérdidas son paralelas al entrenamiento lo que significa que el algoritmo identificó las variables correspondientes para lograr un resultado. Por otro lado, la gráfica de magnitud de pérdidas en función de las “*epochs*” o vueltas que logró dar el algoritmo se muestra a continuación.

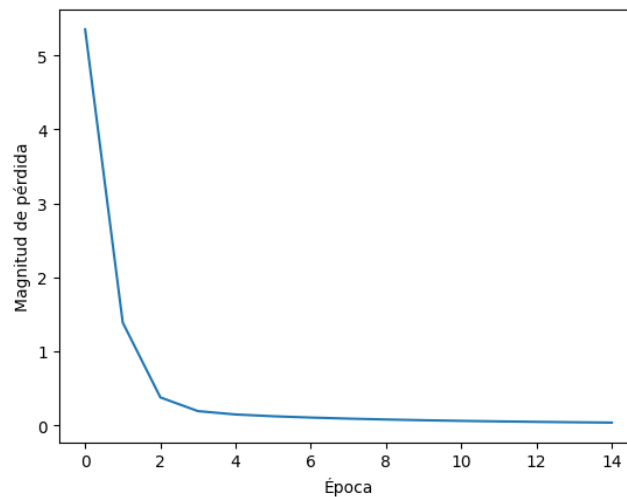


Figura 3.10 Magnitud de Pérdidas de la Red Neuronal Multicapa

En la Figura 3.10 se observa las pérdidas o “*loss*” generadas en el algoritmo de la red neuronal y su disminución que logró a lo largo de las épocas ingresadas, las “*epochs*” en un principio se optó por un rango entre 20 y 30 las cuales el modelo dejaba de aprender en la “*epochs n°25*” es decir a partir de dicha época generaba el “*overfitting*” y el resultado generado en el coeficiente de determinación o “*R²*” se presentó en un 100% e incluso sobrepasando ese límite, el cual en conclusión mostraba un resultado de fiabilidad no favorable; por ende al realizar una corrección de “*epochs*” en un rango entre 10 a 20 las cuales la gráfica presentada en la figura mostró que el modelo reducía sus pérdidas y luego se mantenían constantes a partir de la “*epochs n°20*” en adelante y se ejecutó un “*R²*” más realista y sobre todo fiable.

Por otro lado, gracias a este ajuste los valores de los errores tanto absoluto y raíz cuadrático medio mostraron un valor favorable como se muestra a continuación.

```
print('Error absoluto medio (MAE):', metrics.mean_absolute_error(y_test, y_predrn))
print('Error cuadrático medio(MSE):', metrics.mean_squared_error(y_test, y_predrn))
print('Raíz del Error cuadrático medio(RMSE):', np.sqrt(metrics.mean_squared_error(y_test, y_predrn)))
```

```
Error absoluto medio (MAE): 0.1507246583765886
Error cuadrático medio(MSE): 0.055737232954073485
Raíz del Error cuadrático medio(RMSE): 0.23608734179128174
```

Figura 3.11 Errores presentados en la Red Neuronal Multicapa

En la Figura 3.11 muestra los valores del Error absoluto medio y de la Raíz del Error cuadrático medio mostrando porcentajes diminutos y mostrando una cercanía uno del otro; el MAE tuvo un valor de 0,15 que en porcentaje equivale al 15,07% y el valor del RMSE equivale al 0,2361 o en porcentaje igual al 23,61%, estos valores significan que entre más pequeños sean los resultados de los errores, más cercanos están los datos que se logró predecir de los datos registrados por los equipos.

A continuación, se muestra los datos ingresados de un vehículo para su predicción y corroborar si el modelo predice de forma eficiente la emisión de CO₂:


```

CC = [[2.729]] # [L] Ingrese el Valor del Consumo de Combustible en Litros
d = [[48.8]]# [km] Ingrese el Valor de la Distancia en kilómetros
i = np.array(CC) / np.array(d)
print(" El Consumo de Combustible por distancia recorrida es: ",str(i),"L/km")
prediccion = red.predict(i)
print("El resultado del Factor de emision es: ",str(prediccion),"gCO2/km")

```

```

El Consumo de Combustible por distancia recorrida es: [[0.06]] L/km
1/1 [=====] - 0s 55ms/step
El resultado del Factor de emision es: [[122.19]] gCO2/km

```

Figura 3.12 Predicción con datos aleatorios Cherry Arrizo 5 Pro 2022

De la misma forma que el modelo de Regresión Lineal Simple en la Figura 3.12 muestra los resultados de la emisión generada por el Cherry Arrizo con un consumo de combustible de 2,72 galón por cada 48.8 km, de esta forma el modelo RNM logró predecir una emisión de 122,19 gCO₂/km lo cual comparado con el factor de emisión teórico que es de 126,96 gCO₂/km presenta un error relativo del 3,75 % lo cual se consideró que está dentro del rango.

3.4 RESULTADOS DEL MODELO ÁRBOL DE DECISIÓN DE REGRESIÓN

Los resultados de este modelo mostraron un porcentaje de fiabilidad favorable, a pesar de lograr obtenido una estructura de árbol similar a la naturaleza, fue imposible comprender su modo de aprendizaje sobre todo por la naturaleza de datos ingresados que esencialmente tienen una tendencia lineal; estos resultados se observan a continuación.

```

r2ad = r2_score(y_test, y_predad)*100
print("El porcentaje de la variación en la variable de respuesta (R^2)es:",r2ad,"%")

```

```

El porcentaje de la variación en la variable de respuesta (R^2)es: 92.95132939083194 %

```

Figura 3.13 Coeficiente de determinación del Árbol de Decisión

En la Figura 3.13 muestra coeficiente de determinación en el modelo de árbol de decisión con un valor de 0,9295 que en porcentaje es el 92,95% de fiabilidad; pues este modelo emplea un aprendizaje supervisado y sobre todo una estrategia de división y resultados, es decir, ejecutó una búsqueda que identificó a los puntos más relevantes del árbol con los

datos ingresados de una forma descendente y de esta forma clasificó los resultados bajo las etiquetas de regresión que se ingresaron en un comienzo. En este modelo también se logró sacar los valores de los errores que se muestra a continuación.

```
print('Error absoluto medio (MAE):', metrics.mean_absolute_error(y_test, y_predad))
print('Error cuadrático medio(MSE):', metrics.mean_squared_error(y_test, y_predad))
print('Raíz del Error cuadrático medio(RMSE):', np.sqrt(metrics.mean_squared_error(y_test, y_predad)))
```

```
Error absoluto medio (MAE): 0.21232461429018065
Error cuadrático medio(MSE): 0.060712718647115915
Raíz del Error cuadrático medio(RMSE): 0.24639951024122575
```

Figura 3.14 Errores presentados en Árbol de Decisión Regressor

En la Figura 3.14 el valor del MAE fue de 0,2123 que en porcentaje equivale a 21,23% de comparación entre los datos registrados y los resultados que se logró predecir; no obstante, el valor del RMSE obtuvo un 0,2464 que en porcentaje es el 24,64% de error entre el conjunto de valores registrados y los datos que se logró predecir con este modelo.

Es importante recordar que los valores de los errores absoluto y raíz cuadrático medio entre más pequeños sean los resultados el modelo posee más confiabilidad.

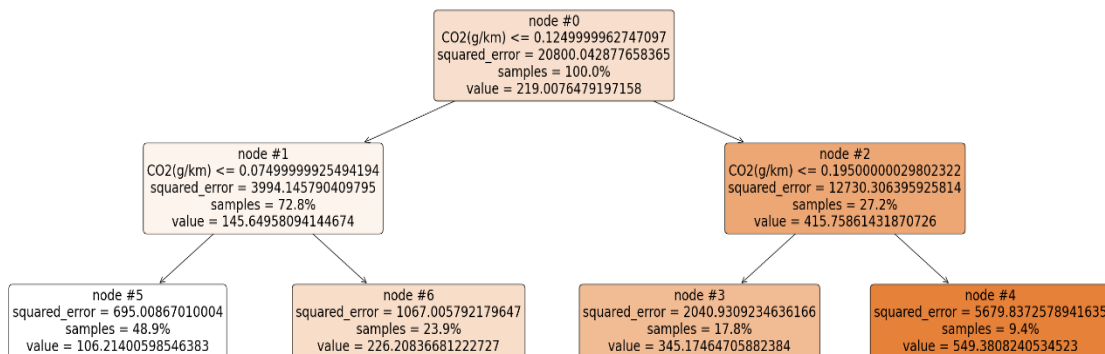


Figura 3.15 Árbol de Decisión Regressor

En la Figura 3.15 muestra el árbol de decisión generado en este modelo el lado derecho muestra las variables mostradas como un falso positivo y el izquierdo son las variables verdaderas o positivas, el primer bloque es la raíz del árbol, para cada cuadro la segunda línea es el (MSE) este se mencionó que entre más pequeño sea este resultado más fiable es la relación entre variables y resultados; la tercera línea “*samples*” que fue la cantidad de instancias que cumplen el criterio; en el caso de la primera rama a la izquierda es

“*y_predad*” este valor es del 72,80 % y la cuarta línea es “*value*” es decir, el valor promedio de la variable a predecir que en este caso es de 145,54 gCO₂/km (“*y_predad*” → “CO₂(g/km)”). No obstante, este modelo también ofreció una predicción con los datos teóricos aleatorios del vehículo mencionado en los anteriores modelos como se muestra a continuación:

```
CC = [[2.729]] # [L] Consumo de Combustible en Litros
d = [[48.8]] # [km] Distancia en kilómetros
i = np.array(CC) / np.array(d)
print(" El Consumo de Combustible por distancia recorrida es: ",str(i),"L/km")
prediccion = arbolregresion.predict(i)
print("El resultado del Factor de emision es: ",str(prediccion),"gCO2/km")
```

```
El Consumo de Combustible por distancia recorrida es: [[0.06]] L/km
El resultado del Factor de emision es: [106.21] gCO2/km
```

Figura 3.16 Factor de Emisión en Árbol de Decisión Regressor

En la Figura 3.16 muestra la emisión del Cherry Arrizo con un total de 106,21 gCO₂/km; este valor con respecto al factor de emisión teórico del vehículo el cual es de 126,92 gCO₂/km obtuvo un error relativo del 16,34% lo que se consideró un rango aceptable ya que no supera el 50 % de error.

3.5 RESULTADOS DEL MODELO RANDOM FOREST REGRESSOR

El Bosque aleatorio de regresión dio resultados favorables pues al tener un bosque conformado de 100 árboles entrego un valor prometedor como se observa a continuación.

```
r2rf = r2_score(y_test, y_predrf)*100
print("El porcentaje de la variación en la variable de respuesta (R^2)es:",r2rf,"%")
```

```
El porcentaje de la variación en la variable de respuesta (R^2)es: 84.63972092349704 %
```

Figura 3.17 Coeficiente de determinación de RandomForest Regressor

En la Figura 3.17 el bosque dio un valor del 0,8463 que en porcentaje se representaría en 84,63%, una fiabilidad que en este modelo también resulta favorable pues la relación entre todas las variables independientes puede ser tomadas como un conjunto de árboles para

llegar al resultado de la variable dependiente (emisión de CO₂) y los valores se acerquen cada vez más a los registros.

```
print('Error absoluto medio (MAE):', metrics.mean_absolute_error(y_test, y_predrf))
print('Error cuadrático medio(MSE):', metrics.mean_squared_error(y_test, y_predrf))
print('Raíz del Error cuadrático medio(RMSE):', np.sqrt(metrics.mean_squared_error(y_test, y_predrf)))
```

Error absoluto medio (MAE): 0.24945743697767173
 Error cuadrático medio(MSE): 0.0893837756887899
 Raíz del Error cuadrático medio(RMSE): 0.29897119541653155

Figura 3.18 Errores presentados en RandomForest Regressor

En la Figura 3.18 muestra los resultados de los errores presentados en el modelo; en el valor de MAE se obtuvo un total de 0,2495 que en modo porcentual es 24,95% y el valor de RMSE se logró obtener un total 0,2990 que en porcentaje representaría el 29,90%, al poseer estos valores en la predicción se acercan a los datos registrados de una forma considerable, logrando que el resultado de la fiabilidad obtenida se corrobore y brinde una confiabilidad más concisa. En el modelo RandomForest Regressor también logró una predicción con valores encontrados del Cherry Arrizo como se muestra a continuación:

```
CC = [[2.729]] # [L] Consumo de Combustible en Litros
d = [[48.8]] # [km] Distancia Recorrida
i = np.array(CC) / np.array(d)
print("El Consumo de Combustible es de: ",str(i),"L/km")
prediccion = modelrf.predict(i)
print("El resultado del Factor de emision es de: ",str(prediccion),"gCO2/km")
```

El Consumo de Combustible es de: [[0.06]] L/km
 El resultado del Factor de emision es de: [143.49] gCO2/km
 [Parallel(n_jobs=2)]: Using backend ThreadingBackend with 2 concurrent workers.
 [Parallel(n_jobs=2)]: Done 46 tasks | elapsed: 0.0s
 [Parallel(n_jobs=2)]: Done 150 out of 150 | elapsed: 0.0s finished

Figura 3.19 Factor de emisión con RandomForest Regressor

En la Figura 3.19 muestra el Factor de emisión generado con un valor total de 143,49 gCO₂/km, a pesar de que este modelo tiene un porcentaje de fiabilidad dentro de los porcentajes de confiabilidad aceptables, el valor encontrado tiene un error relativo del 13,02% con respecto al valor real que es de 126,96 gCO₂/km.

3.6 RESULTADOS DEL MODELO SVM DE REGRESIÓN

En este trabajo de grado también optó por utilizar un quinto modelo de predicción llamado SVMR o Máquina de Vectores de Soporte de Regresión, este último ofreció buenos resultados debido a la identificación de una regresión, y una suma de errores en el hiperplano diminuta; a continuación, se muestra los resultados que se obtuvieron con este quinto modelo.

EVALUACIÓN

```
[141] r2mv = r2_score(y_test, y_predmv)*100
      print("El porcentaje de la variación en la variable de respuesta (R^2)es:",r2mv,"%")
```

El porcentaje de la variación en la variable de respuesta (R^2)es: 95.1537062911308 %

Figura 3.20 Coeficiente de determinación de SVM Regressor

En la Figura 3.20 se observa el valor del coeficiente de determinación en este modelo que muestra un porcentaje de 95,15% de fiabilidad, pues el margen de tolerancia esta cercano al vector que minimiza el error, cabe recordar que el margen es una distancia entre vectores de 2 clases y estas son categorías que se encuentran divididas a lo largo del hiperplano; los valores de entrenamiento y los valores de testeo se muestran en dicho hiperplano y la predicción se dio en medio de estos datos como un vector de soporte.

De la misma forma que los anteriores modelos ejecutados, se procedió a ejecutar la medición de los errores presentados en el modelo como se muestra a continuación.

```
print('Error absoluto medio (MAE):', metrics.mean_absolute_error(y_test, y_predmv))
print('Error cuadrático medio(MSE):', metrics.mean_squared_error(y_test, y_predmv))
print('Raíz del Error cuadrático medio(RMSE):', np.sqrt(metrics.mean_squared_error(y_test, y_predmv)))
```

Error absoluto medio (MAE): 0.15956281728577446
 Error cuadrático medio(MSE): 0.05549356078094076
 Raíz del Error cuadrático medio(RMSE): 0.2355707129100321

Figura 3.21 Errores presentados en SVM Regressor

En la Figura 3.21 muestra el valor de MAE con un total de 0,1596 que en porcentaje equivalió a 15,96% y el RMSE con un total de 0,2356 que en valores porcentuales fue el 23,56% y poseen un resultado medial por lo cual es un total considerable, es decir, no se

acerca mucho a los valores reales, pero tampoco se aleja demasiado de los mismos. No obstante, esta aclaración también se pudo observar de manera grafica como se muestra a continuación.

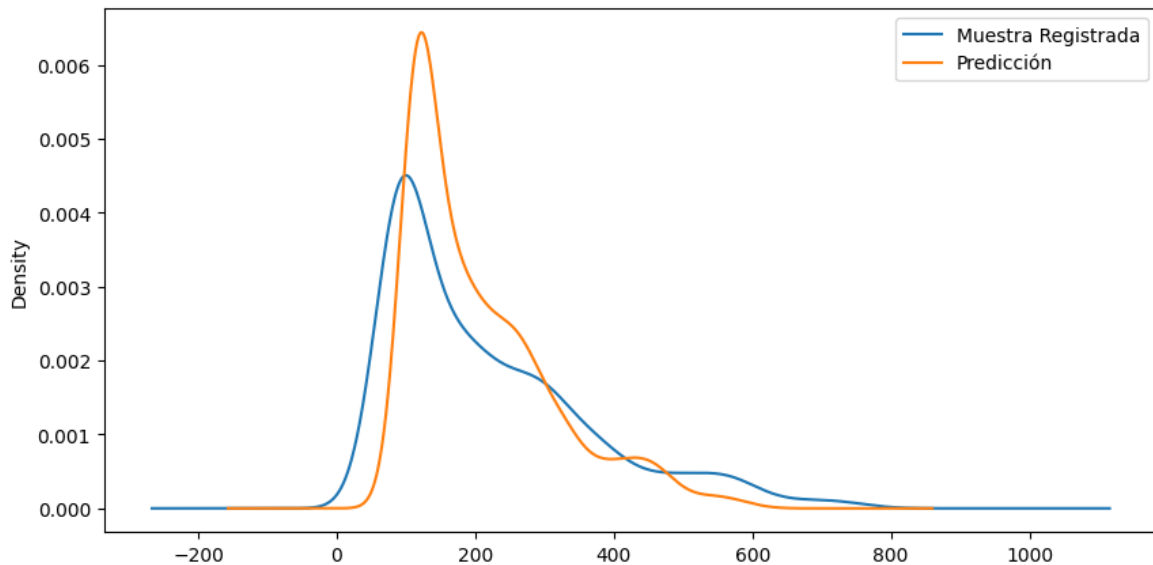


Figura 3.22 Gráfico de comparación entre datos registrados vs predicción

En la Figura 3.22 muestra un gráfico que sirvió para visualizar una mejor comparación en modo “*density*” que mostró de forma coherente la comparación. La figura muestra de color azul a los datos registrados con los equipos y de color naranja a los datos que se logró predecir y al estar superpuestos uno del otro se logró concluir que los valores obtenidos por el modelo no se alejan por mucho de la muestra registrada, la densidad de los datos de predicción es mayor que la muestra registrada lo que significa que esta observación pertenezca a la distribución de datos ingresados obteniendo resultados confiables.

El modelo de SVMR también ofreció una predicción con los datos teóricos del Cherry Arizo como se ejecutó en los demás modelos mencionados y que se muestra a continuación:

```

▶ CC = [[2.729]] # [L/h] Ingrese el Consumo de Combustible en Litros
  d = [[48.8]] # [km] Ingrese la Distancia Recorrida
  i = np.array(CC) / np.array(d)
  prediccion = regresionsvr.predict(i)
  print("El Factor de emisión generado es de: ",str(prediccion),"gCO2/km")

```

↳ El Factor de emisión generado es de: [136.79] gCO₂/km

Figura 3.23 Factor de emisión generado por SVRM

En la Figura 3.23 muestra un factor de emisión de 136,79 gCO₂/km lo cual comparado con el valor encontrado este modelo tuvo un error relativo del 7,74%, es decir, que el SVRM presenta una cercanía favorable y dentro del rango permisible.

3.7 RESULTADOS OBTENIDOS EN CADA MODELO DE PREDICCIÓN

Cada modelo de predicción realizado en este trabajo de grado proporcionó resultados interesantes los cuales permitieron elegir el o los modelos adecuados que cumplan con los objetivos principales del presente trabajo, pero sobre todo el que mejor se adapte a los datos recolectados no obstante todos los modelos fueron ajustados en un rango del 90 al 95% para llegar a un resultado confiable; además estos valores se presentan en la tabla 3.3 los cuales permiten clasificar la fiabilidad de cada modelo como se muestra a continuación:

Tabla 3.3 Valores obtenidos en los modelos de predicción

MODELOS DE PREDICCIÓN						
Ruta	Modelos	Regresión Lineal	Árbol de Decisión Regresor	Random Forest	Máquina de Vectores de Soporte	Redes Neuronales Multicapa
RDE Ciudad de Ibarra		99,77%	92,95%	84,64%	95,15%	93,81%
Fiabilidad		NO FIABLE	FIABLE	FIABLE	FIABLE	FIABLE

La fiabilidad de cada modelo se clasifica en los diferentes rangos porcentuales que están comprendidos entre:

- De 10% a 40%=No Fiable

- De 55% a 60%=Fiable
- De 70% a 80%=Fiable
- De 85% a 95%=Fiable
- De 95% a 100%=No Fiable

Los valores presentados en estos modelos de predicción son fiables debido a la naturaleza de los datos ingresados, sobre todo porque la fiabilidad tiene una relación directamente proporcional entre las variables unas a otras; dicha correlación es una guía sólida como se observa en la Figura 2.23; de esta manera la fiabilidad se vuelve un resultado más concreto.

El único modelo que no presentó una fiabilidad aceptable es Regresión Lineal pues los resultados de predicción son exactamente iguales a la muestra registrada, es decir, la máquina no identificó la relación entre todas las variables y solo se limitó a presentar los valores de testeo. Cabe recalcar que los valores que se han logrado predecir con los diferentes modelos funcionan de manera diferente para cada base de datos que se ingresen ya que pueden ser confiables para diversos datos.

Los modelos de predicción prometen ser un gran apoyo sobre todo en la mitigación de emisiones de gases, el modelo de predicción más fiable usado en este proyecto es la red neuronal artificial multicapa con esto se pudo deducir que en los próximos años se comience a dar más fortaleza a machine learning y de esa forma conocer más sobre como realizan el proceso de aprendizaje con respecto a las redes neuronales.

No obstante, el otro modelo prometedor fue SVMR pues logró mantenerse constante con su entrenamiento y aprendizaje al igual que los demás modelos ejecutados acercándose por mucho a los valores registrados.

3.7.1 COMPARACIÓN DE LOS VALORES DE PREDICCIÓN DEL VEHÍCULO

En esta sección se muestra de manera resumida los valores de predicción con respecto al Factor de Emisión encontrado en el vehículo Cherry Arrizo, la Tabla 3.4 muestra los

resultados de predicción que generó cada modelo descrito; sin embargo, los modelos con un error menor al 10% que son Redes Neuronales Artificiales Multicapa y Máquina de Vectores de Soporte resultaron los más confiables pues estos se acercaron considerablemente al valor teórico.

Tabla 3.4 Valores de Predicción de los Modelos en Cherry Arrizo 5 PRO

Modelos de Predicción	Predicción (gCO ₂ /km)	Resultado en la ruta RDE (gCO ₂ /km)	Error Relativo (%)
Regresión Lineal	125,01	126,96	1,53
RedesNeuronalesMulticapa	122,19		3,75
Árbol de Decisión Regressor	106,21		16,34
Random Forest Regressor	143,49		13,02
Máquina Vectores de Soporte Regressor SVMR	136,79		7,74

3.7.2 PREDICCIÓN DE EMISIONES USANDO DIVERSOS VEHÍCULOS

Los modelos de predicción mostraron resultados favorables con respecto al valor del vehículo de pruebas en la ruta RDE, no obstante, se hizo uso de la plataforma FUELECONOMY para encontrar los consumos de combustibles y factores de emisión de diferentes marcas, tipos y modelos de vehículos y verificar que los modelos se adapten a esta información.

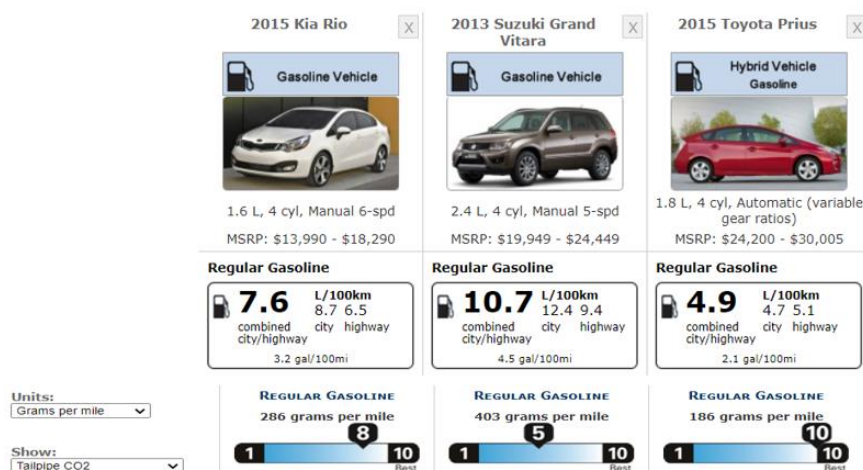


Figura 3.24 Factor de emisión y consumo de combustible de diversos vehículos (Energy, 2015)

En la Figura 3.24 se muestra la comparación entre tres vehículos tipo Sedán(Kia), SUV(Suzuki) he Híbrido(Prius) donde se observa el consumo de combustible y el factor de emisión CO₂; estos datos proporcionan información relativamente importante como el gasto de combustible para ingresar a los modelos descritos y comparar su resultado con respecto a la emisión. No obstante, también se optó por utilizar como ejemplo el vehículo más vendido en Ecuador que es el SWM como se muestra a continuación:



Prestaciones y consumos homologados	
Velocidad máxima	180 km/h
Aceleración 0-100 km/h	10 s
Consumo WLTP	
Combinado	8 l/100 km
Velocidad baja	10,9 l/100 km
Velocidad media	7,5 l/100 km
Velocidad alta	7 l/100 km
Velocidad muy alta	8,2 l/100 km
Emisiones de CO₂ WLTP	182 gr/km
Normativa de emisiones	Euro 6

Figura 3.25 Factor de emisión y consumo de combustible del SWM (SWM G01 Basic, 2021)

En la Figura 3.25 muestra los consumos de combustible combinado (zona urbana y rural) y sobre todo el factor de emisión CO₂ del vehículo tipo SUV marca SWM. A continuación, los valores de emisión y consumo de combustible combinado, se muestra en la Tabla 3.4 de manera práctica.

Con estos valores al introducirlos al cuaderno de programación Python y a los diferentes modelos, se logró encontrar los resultados aproximados a los teóricos los cuales se describen a continuación.

3.7.2.1 Resultados de Modelos de Predicción vs Valores Teóricos

Los valores encontrados en los modelos predictivos mostraron cierta cercanía con los valores teóricos como se muestra en la Tabla 3.5 con un error relativo menor al 50% en todos los modelos y para los diferentes tipos de vehículos, no obstante, los porcentajes más altos se presentaron en los modelos de Árbol de Decisión y RandomForest a pesar de que todos los modelos se reajustaron para tener una confiabilidad comprendida entre el rango del 90 a 95% estos modelos mostraron una sobre limitación de los valores reales como se observa a continuación:

Tabla 3.5 Valores de Predicción de los Modelos vs Valores Teóricos

MARCAS VEHÍCULOS & FECO ₂ (g/km) FuelEconomy		RESULTADOS DE FECO ₂ (g/km) DE MODELOS PREDICTIVOS									
		Regresión Lineal		RedesNeuronalesMulticapa		Árbol de Decisión Regressor		Random Forest Regressor		Máquina Vectores de Soporte Regressor	
		Valor Predictivo	ERelativo	Valor Predictivo	ERelativo	Valor Predictivo	ERelativo	Valor Predictivo	ERelativo	Valor Predictivo	ERelativo
SWM	182	180	1,02%	181,01	0,54%	226,21	24,29%	145,91	19,82%	180,35	0,90%
Híbrido Prius	115,59	109,16	5,56%	109,65	5,14%	106,21	8,11%	143,49	24,14%	124,27	7,50%
Kia Rio	177,75	170,97	3,81%	171,84	3,32%	226,21	27%	145,91	17,91%	173,11	2,61%
Suzuki Grand Vitara	250,46	241,91	3,42%	242,88	3,00%	226,21	9,68%	145,91	41,74%	229,23	8,47%
Cherry Arrizo 5 Pro	126,96	125,01	1,53%	122,19	3,75%	106,21	16,34%	143,49	13,02%	136,79	7,74%

No obstante, los modelos de mayor fiabilidad fueron las redes neuronales multicapa y máquina de vectores de soporte regresor que mostraron un error menor al 10%, dónde los valores encontrados se acercan considerablemente a los resultados teórico

3.8 VELOCIDAD vs CONSUMO DE COMBUSTIBLE & EMISIÓN CO₂

La relación de estas variables a lo largo de la ruta RDE permitió encontrar las condiciones de los estados de conducción, es decir, las aceleraciones, desaceleraciones, crucero y ralentí, para esto se ejecutó una división para cada una de las zonas que son urbana, rural y de autopista de las cuales se eligió un rango o instante donde se logró observar de mejor manera estas condiciones.

3.8.1 VELOCIDAD vs C.C. & FECO₂ DE RUTA ZONA URBANA(INST. 400-520seg)

En esta sección se muestra un instante de 120 seg correspondiente a la zona urbana del Anexo VII como se muestra a continuación:

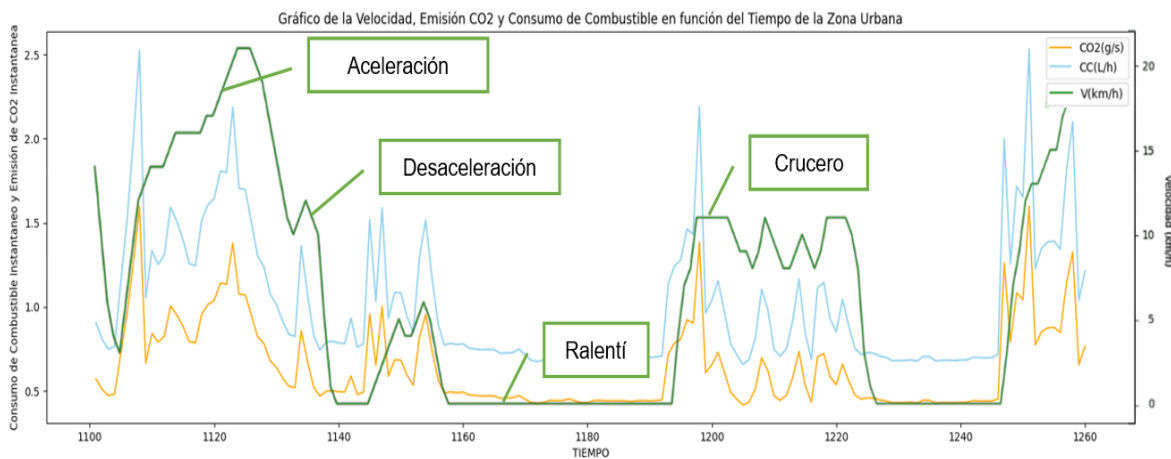


Figura 3.26 Estilos de Conducción en el instante 1100-1260 seg Zona Urbana

En la Figura 3.26 se logró observar como el vehículo a lo largo de esta zona presentó los estilos de conducción como la aceleración, desaceleración, crucero y ralentí de esta forma se obtuvo el siguiente análisis.

Ralentí = 54%
Aceleración= 24%

Desaceleración=20%
Crucero=10%

Estos resultados muestran que el estado de conducción que más porcentaje representa en la ruta es Ralentí con un 54%, esto debido a las paradas frecuentes presentes en dicho tramo ocasionadas por el tráfico vehicular, semáforos, etc. No obstante, tanto la emisión de CO₂ y consumo de combustible su comportamiento fue similar a los estados de conducción, es decir, tanto en la aceleración y desaceleración estos factores aumentaban o disminuían respectivamente, por otro lado, en el estado ralentí el CC y el FECO₂ no llegaban a cero pues estos factores se mantenían constantes lo cual demostraba que existía mayor consumo y por ende mayor emisión.

3.8.2 VELOCIDAD vs C.C. & FECO₂ DE RUTA ZONA RURAL (INST.5314-5484seg)

En esta zona al igual que el tramo urbano se optó por seleccionar un instante de 150 seg del Anexo VIII como se muestra a continuación:

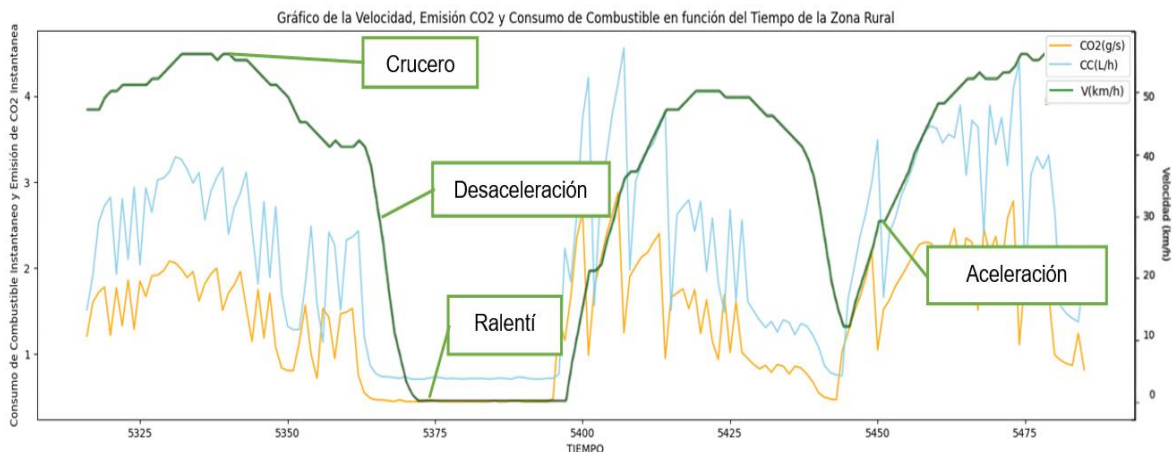


Figura 3.27 Estilos de Conducción en el instante 5325-5475 seg Zona Rural

En la Figura 3.27 se muestra los estados de conducción del tramo rural en un instante determinado de esta manera se logró obtener el siguiente análisis:

Ralentí= 16,66%	Desaceleración= 34,02%
Aceleración= 34,66%	Crucero= 14,66%

En esta zona los estados de conducción que presentaron un mayor porcentaje en el instante seleccionado son la Aceleración y Desaceleración con un 34,66% y 34,02%

respectivamente, es decir, el vehículo alcanzó en ciertos tramos de dicha zona velocidades altas que se dieron por rebasar los vehículos presentes o la disminución de la misma al entrar en redondeles o semáforos, no obstante, el estado de conducción Ralentí alcanzó un 16,66% debido a los semáforos encontrados y por otro lado el estado de conducción Crucero obtuvo un 14,66% debido a la velocidad constante sin la presencia de pendientes positivas o negativas y tráfico vehicular.

3.8.3 VELOCIDAD vs C.C. & FECO₂ DE RUTA ZONA AUTOPISTA (INST.6050-6264seg)

En esta zona de la misma forma que se ejecutó en la urbana y rural se seleccionó un instante de 200 seg del Anexo IX como se muestra a continuación:

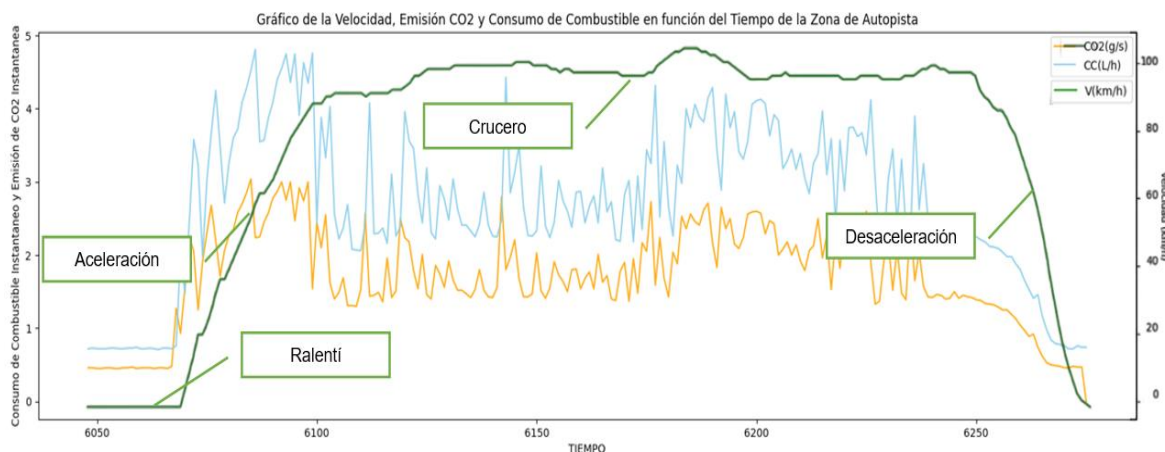


Figura 3.28 Estilos de Conducción en el instante 6050-6250seg Zona Autopista

En la Figura 3.28 se muestra los estados de conducción en la zona de autopista la cual se logró obtener el siguiente análisis:

Ralentí= 9,10%	Desaceleración= 13,63%
Aceleración= 18,18%	Crucero= 59,09 %

El instante seleccionado se muestra un 59,09% en el estado Crucero pues en este instante esta la velocidad máxima alcanzada y permitida por la normativa, no obstante, el estado de Aceleración con un 18,18% es debido a las aceleraciones abruptas al rebasar a los vehículo presentes y el estado de Desaceleración con un 13,63% debido a la disminución de la velocidad.

CAPÍTULO IV

4. CONCLUSIONES & RECOMENDACIONES

4.1 CONCLUSIONES

Al finalizar este trabajo de grado se llegó a las siguientes conclusiones:

- El presente trabajo muestra resultados para predecir los valores de emisiones CO₂ en un vehículo al aplicar las Redes Neuronales Artificiales Multicapa y Máquina de Vectores de Soporte con un error que está por debajo del 10% usando datos relevantes como el consumo de combustible y la distancia recorrida ajustados a un rango comprendido entre el 90 y 95% de fiabilidad.
- Los modelos Árbol de Decisión y RandomForest no son fiables para predecir resultados de emisiones CO₂ pues a pesar de estar en un rango de fiabilidad aceptable al igual que los demás modelos no se adaptaron a la naturaleza de datos numéricos recolectados, por lo que mostraron errores con valores del 16,34% & 13,02% respectivamente y no mostraron una correlación congruente entre las variables de entrenamiento y testeo ocasionando que el entrenamiento de máquina esté sobre limitado.
- Los códigos de Redes Neuronales Artificiales Multicapa y Máquina de Vectores de Soporte Regressor ejecutados en el cuaderno de programación Python puede y podría efectuar predicciones de gases como CO, NO_x, HC, usando las mismas señales de las variables recolectadas de RPM, MAP, MAF, IAT usando registros en formato CSV.
- Al efectuar la revisión del estado del arte, se comprobó que existen estudios previos sobre la predicción de emisiones de gases contaminantes de un motor de combustión interna o MEP usando redes neuronales; sin embargo, estos trabajos no han sido aplicados en el país por lo que este trabajo de grado simboliza una base para futuras investigaciones relacionadas con el tema de machine learning y poder aplicarlo no solo a la flota de vehículos livianos sino también a la flota de vehículos MEC.

4.2 RECOMENDACIONES

- Se recomienda usar los modelos de Redes Neuronales Artificiales Multicapa y Máquina de Vectores de Soporte Regressor en la predicción de otros gases de escape al aire incluso usando el cálculo dinámico del vehículo por lo que las señales de los sensores del vehículo en prueba son en tiempo real usando los equipos específicos en esta investigación con una frecuencia de registro mucho más alta.
- Los dispositivos utilizados en esta investigación tienen una compatibilidad más confiable y acertada en vehículos pesados por lo que se recomienda utilizar estos equipos para futuras investigaciones enfocadas a las predicciones de NOx en vehículos MEC.
- El almacenamiento de los datos debe estar calibrada a una frecuencia de 1 Hz, pues esta frecuencia es recomendable en el equipo CSS ELECTRONICS para evitar la sobre acumulación de información y sobre todo evitar que la memoria extraíble acumule más datos de la capacidad de la Micro SDHC industrial de 8Gb; no obstante, en caso de ser usados en vehículos MEC utilizar una frecuencia de registro mayor comprendida entre los 5 y 10Hz.
- El presente trabajo tuvo dificultades con los decodificadores del programa ASAMMDFGUI por lo que se recomienda usar las diferentes actualizaciones de decodificadores tanto de posicionamiento y obtención de PIDS que la empresa CSS ELECTRONICS ofrece a sus clientes al utilizar sus equipos que se encuentran en el link <https://www.csselectronics.com/pages/obd2-data-pack-car-dbc>.

Bibliografía

1. Arango, M. (2019). Introducción al Aprendizaje por Refuerzo. En *Oracle A-Team*.
2. Arciniegas, O. (2018). *ESTIMACIÓN DE EMISIONES CONTAMINANTES DE VEHÍCULOS DE TRANSPORTE ESCOLAR E INSTITUCIONAL EN LA CIUDAD DE IBARRA MEDIANTE MODELADO COMPUTACIONAL*. Universidad Técnica Del Norte.
3. Aristizábal, H., Benavides, G., & Esperanza, L. (2007). Información Técnica sobre gases de efecto invernadero y el cambio climático. En *Academia* (Vol. 08, Número Información técnica sobre gases de efecto invernadero y el cambio climático).
4. Arteaga, A., Milena, C., Madrigal, A., & Segura, H. (2017). Emisión de gases de efecto invernadero por uso de combustibles fósiles en Ibagué, Tolima (Colombia). *Corpoica Ciencia y Tecnología Agropecuaria*, 18(1), 103–112. https://doi.org/10.21930/rcta.vol18_num1_art:561
5. Benito, Y. (2016). Guía específica de trabajo sobre CO2 y cambio climático. *Investiga I+D+i*, 1, 10.
6. Berenguer, M., & Bernal, F. (2000). NTP 549 : El dióxido de carbono en la evaluación de la calidad del aire interior. En *INSHT. Instituto Nacional de Seguridad e Higiene en el Trabajo*. http://www.insht.es/InshtWeb/Contenidos/Documentacion/FichasTecnicas/NTP/Ficheros/501a600/ntp_549.pdf
7. Bosch, R. (1996). *Manual de la Técnica del Automóvil* (3ra Edición). REVERTÉ.S.A.
8. Calvo, J., Guzmán, M., & Ramos, D. (2018). Machine Learning, una pieza clave en la transformación de los modelos de negocio. *Management Solutions*, 44.
9. Cherry, A. (2022). *ARRIZO 5*.
10. Cherry Arrizo 5 Pro. (2021). *Ficha Técnica Cherry Arrizo 5 Pro* (p. 5).

11. Chiriboga, S. (2018). *ANÁLISIS DE EMISIONES DE CO₂ DEL TRANSPORTE EN LA CIUDAD DE IBARRA*. Universidad Técnica del Norte.
12. CSSELECTRONICS. (2021a). *CANedge2: 2x CAN Bus DataLogger (SD + WiFi) – CSS Electronics*. <https://www.csselectronics.com/products/can-bus-data-logger-wifi-canedge2>
13. CSSELECTRONICS. (2021b). *CANmod.gps: GPS-to-CAN with 3D Inertial Sensor and UDR – CSS Electronics*. <https://www.csselectronics.com/products/gps-to-can-bus-gnss-imu?variant=40452971954364>
14. Das, S., Dey, A., Pal, A., & Roy, N. (2015). Applications of Artificial Intelligence in Machine Learning: Review and Prospect. *International Journal of Computer Applications*, 115(9), 31–41. <https://doi.org/10.5120/20182-2402>
15. EINECS. (2019). *CO₂ (Dióxido de carbono)*. Ministerio para la transición ecológica y Reto Demográfico. <https://prtr-es.es/co2-dioxido-de-carbono,15590,11,2007.html>
16. Energy, U. D. of. (2015). *Compare Side-by-Side*. Department website. <https://www.fueleconomy.gov/feg/Find.do?action=sbs&id=35296&id=33127>
17. Galbiatti, M. (2013). *Revolución Industrial*.
18. Géron, A. (2019). Hands On Machine Learning with Scikit-Learn, Keras, and TensorFlow. En N. Tache (Ed.), *Hands on Machine Learning with Scikit-Learn, Keras, and TensorFlow* (Second Edi). Chapman and Hall/CRC. <https://doi.org/10.1201/9780367816377>
19. Gonzáles, A. (2017). *Estudio del empleo de metano como combustible de sustitución en hibridación de motores diésel*. Universidad Politécnica de Madrid.
20. Hobsbawm, E. (1962). *The Age of the Revolution. Europe 1789-1848* (1 Edition). Planeta S.A.2014.
21. Instituto Nacional de Ecología y Cambio Climático. (2019). *Gases y Compuestos de Efecto Invernadero* (GYCEI). <https://cambioclimatico.gob.mx/estadosymunicipios/Emisiones.html>

22. Juncker, J. (2016). REGLAMENTO (UE) 2016/427. Por el que se modifica el Reglamento (CE) n.o 692/2008 en lo que concierne a las emisiones procedentes de turismos y vehículos comerciales ligeros (Euro6). *Diario Oficial de la Unión Europea*, L 82, 1–98.
23. Juncker, J. C. (2017). *Reglamento UE 2017/1151*. 4(4), 1–643.
24. Levine, I. (2004). *FISIOQUÍMICA. Quinta edición. Volumen 1* (McGraw-Hill, Ed.; 5a ed., Vol. 1).
25. Maisueche, A. (2019). Utilización Del Machine Learning En La Industria 4.0. En *Universidad de Valladolid*. <https://uvadoc.uva.es/handle/10324/37908>
26. Merchán, J., Lemus, D., Armijos, R., Morales, F., Poveda, S., Suliza, J., Lasso, G., Correa, I., Álava, P., Herrera, D., Bermeo, J., Bernal, E., Prado, J., Paredes, L., Maino, V., Cabrera, M., Bravo, C., Macas, D., Cueva, D., & Vélez, C. (2021). Plan de Creación de Oportunidades 2021-2025. En *Secretaría Nacional de Planificación*.
27. Meseguer, J., Toh, C. K., Calafate, C., Cano, J., & Manzoni, P. (2017). Drivingstyles: A mobile platform for driving styles and fuel consumption characterization. *Journal of Communications and Networks*, 19(2), 162–168. <https://doi.org/10.1109/JCN.2017.000025>
28. Mock, P. (2014). EU CO2 standards for passenger cars and light-commercial vehicles. *International Council on Clean Transportation, January*, 1–9.
29. Morales, P., & Rodríguez, L. (2016). *APLICACIÓN DE LOS COEFICIENTES CORRELACIÓN DE KENDALL Y SPEARMAN (Application of the Kendall correlation and Spearman coefficients)*. <http://www.postgradovipi.50webs.com/archivos/agrollania/2016/agro8.pdf>
30. Moreta, M. (2017). *Ajuste en los motores por regla ambiental*. El Comercio. <https://www.elcomercio.com/actualidad/ajuste-motores-reglaambiental-autos.html>
31. Mosquera, J., Fernández, S., & Mosquera, J. (2010). Análisis de emisiones de CO2 para diferentes combustibles en la población de taxis en Pereira y Dosquebradas. *Scientia*

- et Technica - Universidad Tecnológica de Pereira*, XVI(45), 141–146.
<https://doi.org/https://doi.org/10.22517/23447214.385>
32. Narváez, M., & Narváez, A. (2023). *DESARROLLO DE RUTA DE ENSAYO RDE EN EL CANTÓN IBARRA*. Universidad Técnica del Norte.
33. Necas, P., & Pottering, H. G. (2009). *REGLAMENTO (CE) N° 443/2009 DEL PARLAMENTO EUROPEO Y DEL CONSEJO de 23 de abril de 2009*. 2008(2), 1–15.
34. Núñez, P., García, G., Caridad, J., & Asencio, V. (2021). Mediciones de óxido nitroso (N₂O) en suelo manejado bajo pastoreo con bovinos de leche. *REVISTA TERRA LATINOAMERICANA*, 39, 1–12. <https://doi.org/10.28940/terra.v39i0.813>
35. Olmos, C. (2019). *Estos son todos los gases contaminantes que emite tu coche*. 06/01/2019. <https://www.clicacoches.com/gases-contaminantes-emisiones-coche/>
36. Ong, H., Mahlia, T., & Masjuki, H. (2011). A review on emissions and mitigation strategies for road transport in Malaysia. *Renewable and Sustainable Energy Reviews*, 15(8), 3516–3522. <https://doi.org/10.1016/j.rser.2011.05.006>
37. Pedraza, J. C. (2019). *Prototipo de un modelo de machine learning para la predicción de partículas de contaminación atmosférica finas, en la localidad de Kennedy en Bogotá*. Universidad Distrital Francisco José de Caldas.
38. Pérez, D. (2018). Estudio De Emisiones Contaminantes Utilizando Combustibles Locales Study of Polluting Emissions Using Local Fuels. *INNOVA Research Journal*, 2(23), 23–34. <https://doi.org/https://doi.org/10.33890/innova.v3.n3.2018.635>
39. Ramírez, J. E. (2018). Eventos Químicos-Fisiológicos Del Metano En Los Rumiantes. *Agro productividad*, 11(2), 22–33.
40. RECOPE. (2018). *Poder calórico*. <https://www.recope.go.cr/productos/calidad-y-seguridad-de-productos/poder-calorico/>

41. Sarango, K. (2016). *Determinación de los niveles de contaminación a la atmosfera por emisiones de CO2 en fuentes móviles de la ciudad de Zamora, provincia de Zamora Chinchipe*. Universidad Nacional de Loja.
42. Sorgato, V. (2016). *Seis urbes en Ecuador se exceden en contaminación ambiental, según OMS*.
43. SWM G01 Basic. (2021). *SWM G01 Basic (2021) | Precio y ficha técnica - km77.com*.
<https://www.km77.com/coches/swm/g01/2018/estandar/estandar/g01/datos>
44. Triantafyllopoulos, G., Dimaratos, A., Ntziachristos, L., Bernard, Y., Dornoff, J., & Samaras, Z. (2019). A study on the CO₂ and NO_x emissions performance of Euro 6 diesel vehicles under various chassis dynamometer and on-road conditions including latest regulatory provisions. *Science of the Total Environment*, 666, 337–346. <https://doi.org/10.1016/j.scitotenv.2019.02.144>
45. UTEC. (2021). *Eligiendo buenas variables a través de coeficientes de correlación – Machine Learning – Comunidad aiutechallenge*.
<https://aichallenge.utec.edu.uy/community/machine-learning/eligiendo-buenas-variables-a-traves-de-coeficientes-de-correlacion/>
46. Villalba, P. (2019). *Madrid mediante modelos supervisados*. Universitat Oberta de Catalunya.

ANEXOS

ANEXO I

EQUIPOS CONECTADOS EN EL VEHÍCULO



Anexo 1. Instalación del equipo Datalogger en la entrada OBDII

ANEXO II

INSTALACIÓN DEL EQUIPO GPS



Anexo 2. Instalación del equipo GPS

ANEXO III

ESPECIFICACIONES TÉCNICAS DEL EQUIPO

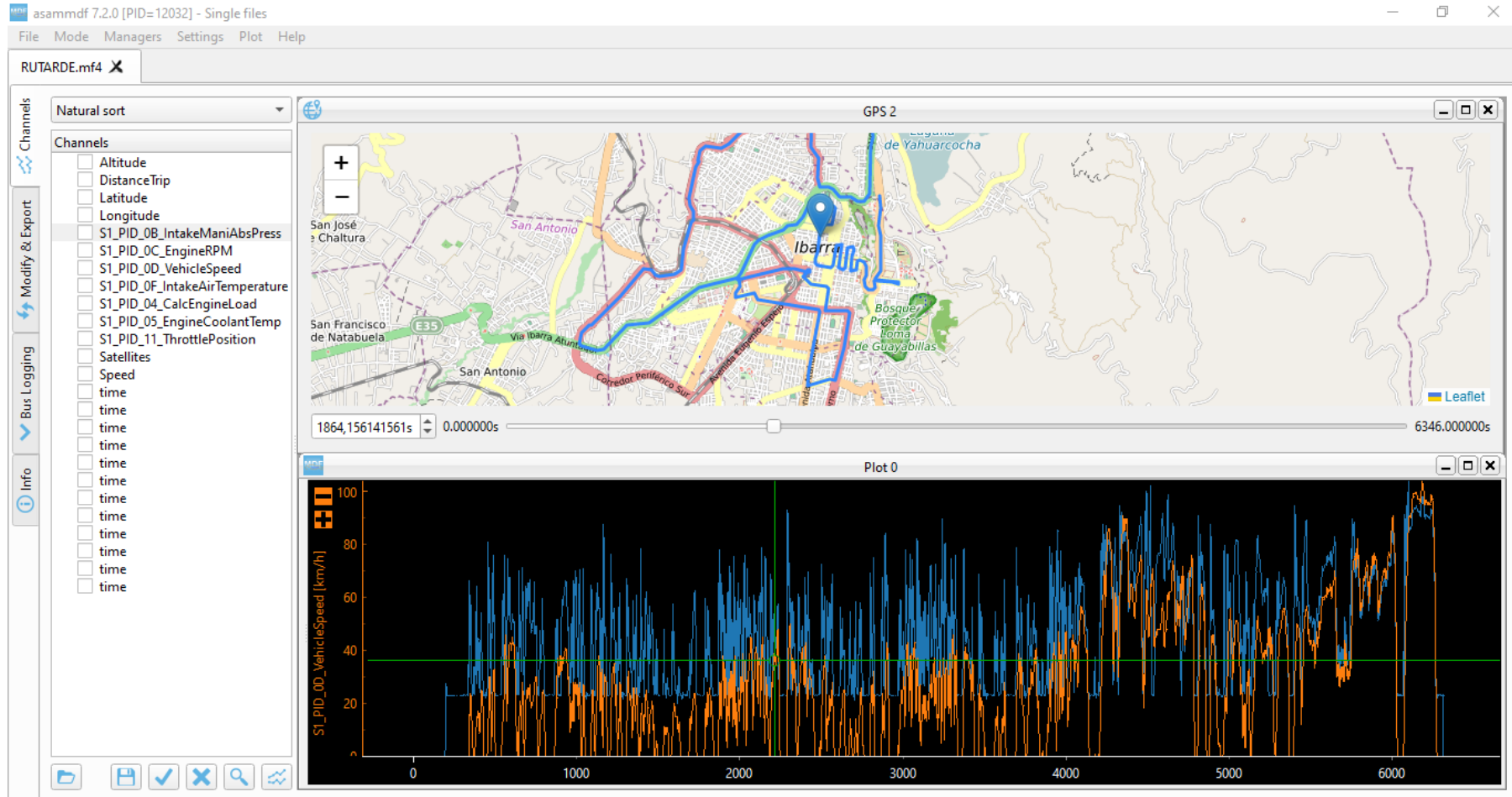
Especificaciones Técnicas	
Seguridad	Certificación CE. FCC. IC.
Pruebas de voltaje	Transitorios ISO
Software	Gratuito
Documentación	En línea
Registro y almacenamiento de datos	
Tarjeta SD	Micro SDHC industrial de 8Gb
Reloj de tiempo real	Incluido batería de respaldo
Formato de archivo	MDF4 procesa/convierte fácilmente
Formatos admisibles	CSV, ASC, TRC
Mecanismo/suministro	
Conectores	2x DB9
Suministro de entrada	+7V a +32v DC a través del can DB9
Dimensiones	52,5x81,4x24,5(largo x ancho x alto)
Led	5 leds

Fuente: (CSSELECTRONICS, 2021)

Anexo 3. Datos Técnicos del Equipo DataLogger

ANEXO IV

PROGRAMA ASAMMDFGUI CSSELECTRONICS



Anexo 4. PID'S y Posicionamiento del Vehículo

ANEXO V

DATOS DE POSICIONAMIENTO & PID'S DEL VEHÍCULO

timestamps	Altitude	Latitude	Longitude	S1_PID_0B_IntakeManiAbsPress	S1_PID_0C_EngineRPM	S1_PID_0D_VehicleSpeed	S1_PID_0F_IntakeAirTemperature	S1_PID_11_ThrottlePosition
1	2229.899999999999	0.3599984538250634	-78.1103778179417	56	971.7864354258325	3	62.0	14.90208
2	2229.959452234439	0.3599882670096312	-78.1103734309275	62	1012.0536642671436	5	62.0	15.623497535999919
3	2230.0	0.3599715759433195	-78.1103643860724	63	1459.3309330932898	8	62.0	18.740519496100752
4	2230.0	0.3599466322478079	-78.1103522144559	61	2080.7059999999991	12	62.0	21.279621215999963
5	2230.0	0.3599119986640862	-78.1103399226658	65	2164.5538607721705	16	61.168775316297484	12.94128
6	2230.058146583851	0.3598721468249968	-78.1103303457219	62	1381.6470000000118	15	61.0	13.494562431756638
7	2230.157454094236	0.3598314316129016	-78.1103235509677	66	1358.354849999992	16	61.83115844207787	19.608
8	2230.313775078323	0.3597878166882420	-78.1103175847854	66	1533.8665699419637	18	61.16889155542224	16.181384352000002
9	2230.458393408857	0.3597381994438384	-78.1103114940522	61	1680.636143071535	19	61.0	19.88061913879441
10	2230.559608941341	0.3596841947194676	-78.1103048247824	63	1844.711855927965	22	61.0	17.25504
11	2230.718843757788	0.3596263259406900	-78.1102966488911	58	1956.380802320925	23	61.0	17.615118103810588
12	2230.919675026492	0.3595654757266794	-78.1102876111728	33	1986.414089862904	24	61.0	16.250813897389836
13	2231.119942225321	0.359503795059508	-78.1102791993625	31	1983.069272291084	24	61.0	14.713457005604441
14	2231.319189000697	0.3594426260834829	-78.1102730185314	28	1914.6613895558253	24	61.0	12.94128
15	2231.517858736059	0.3593838008426205	-78.1102686414374	21	1795.940295147581	22	61.0	13.042940934841363
16	2231.659072112960	0.3593286730711031	-78.1102658178517	26	1611.2829381752758	20	61.0	12.839396832000029
17	2231.700000000000	0.3592787721068221	-78.1102638238377	26	1206.0245098039259	18	61.0	12.447421278639334
18	2231.758016216486	0.3592346189007692	-78.1102624194976	28	942.9720110055014	16	61.0	12.055194479999999
19	2231.800000000001	0.3591969678345156	-78.1102614276524	30	816.7712585034042	13	61.0	12.055194479999999
20	2231.856627587927	0.3591644346220942	-78.1102604333749	31	776.9070535267637	11	61.0	11.764800000000001
21	2231.956315999203	0.3591357852659847	-78.1102600000000	32	765.8432686537308	10	61.0	11.764800000000001
22	2232.0	0.3591106770967525	-78.1102594207433	32	762.2257427228171	9	61.0	11.764800000000001

Anexo 5. PID'S Decodificados y procesados del Vehículo

ANEXO VI

CÓDIGO DE PROGRAMACIÓN REDES NEURONALES

IMPORTACIÓN DE LIBRERIAS

```
[ ] import pandas as pd
import numpy as np
import pylab as pl
import tensorflow as tf
import matplotlib.pyplot as plt
import seaborn as sns
import math
from sklearn.model_selection import train_test_split
- - #REDES NEURONALES
from tensorflow import keras
from keras import models
from keras import layers
from keras import optimizers
from keras import callbacks
```

IMPORTACIÓN DE DATOS

```
[ ] from google.colab import drive
drive.mount('/content/drive')
```

```
[ ] dataset = pd.read_csv (""/content/drive/MyDrive/DATOSCSV/RUTARDECOMPLETAFICHERO.csv","
```

VERIFICACIÓN DE DATOS

```
[ ] dataset.shape
```

```
[ ] dataset.value_counts()
```

```
[ ] dataset.columns
```

```
[ ] dataset.describe()
```

```
[ ] datos = dataset[["TIEMPO", "ALTITUD", "MAP(kPa)", "RPM", "V(km/h)", "IAT(°k)",
                    "TPS", "CC(L/s)", "CC(L/km)", "DensidadEntrada(kg/m3)",
                    "MAF(g/s)", "RendimientoVolumetrico", "CC(L/h)", "CO2(g/km)"]]
datos.head()
```

```
[ ] corr_val = datos[["TIEMPO", "ALTITUD", "MAP(kPa)", "RPM",
                    "V(km/h)", "IAT(°k)", "TPS",
                    "DensidadEntrada(kg/m3)", "MAF(g/s)",
                    "RendimientoVolumetrico", "CC(L/h)", "CO2(g/km)"]]
```

GRÁFICAS DE CORRELACIÓN DE LOS DATOS

```
sns.pairplot(corr_val , kind="scatter")
plt.show()
```

```
[ ] sns.heatmap(corr_val.corr("spearman").sort_values(by="CO2(g/km)"),annot=False,
               cmap=sns.diverging_palette(250, 10, n=9))
```

```
[ ] corr_val.corr("spearman")['CO2(g/km)']
```

REDES NEURONALES

```
[ ] df = pd.DataFrame(datos)
    df0 = df[['CC(L/km)', 'CO2(g/km)']]
    df0
```

CONSTRUCCIÓN DE LA RED

```
[ ] X = df0.iloc[:, :-1].to_numpy(dtype=float)
    y = df0.iloc[:, -1].to_numpy(dtype=float)
```

```
[ ] X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.2,random_state=1)
    print ("División de datos en Entrenamiento/Test Completa")
```

CAPA DE ENTRADA (PRIMERA CAPA OCULTA)

```
[ ] red = tf.keras.Sequential()
    red.add(layers.Dense(units=100,activation = "relu",use_bias=True,
                        bias_initializer="ones"))
```

SEGUNDA CAPA OCULTA

```
[ ] red.add(layers.Dense(units=100,activation = "relu",use_bias=True,
                        bias_initializer="ones"))
```

CAPA DE SALIDA

```

▶ red.add(layers.Dense(units=1,use_bias=True,
                        bias_initializer="ones"))

```

COMPILACIÓN DE LA RED

```

[ ] from keras.metrics import accuracy
    red.compile(optimizer=tf.keras.optimizers.Adam(0.1), loss=["mean_squared_error"],
                ,metrics = ["mse"])

```

ENTRENAMIENTO

```

[ ] entrenamiento = red.fit(X_train, y_train, epochs=20, verbose = "auto",
                            validation_data=(X_train,y_train))

```

PREDICCIÓN

```

[ ] df2=pd.DataFrame(entrenamiento.history)
    df2.head()

```

```

[ ] df2.plot(figsize=(6, 4))

```

```

[ ] df2.plot(kind='bar',figsize=(6,4))
    plt.show()

```

```

[ ] plt.xlabel("Época")
    plt.ylabel("Magnitud de pérdida")
    plt.plot(entrenamiento.history["loss"])

```

```

▶ red.evaluate(X_test,y_test)

```

```

[ ] #len(y_predrn),len(y_predrn)
    y_predrn = red.predict(X_test)
    np.set_printoptions(precision=2)
    print("El valor del CO2 instantaneo es:",
          np.concatenate((y_predrn.reshape(-1,1), y_test.reshape(-1,1)),1))

```

EVALUACIÓN

```

[ ] r2rn = r2_score(y_test, y_predrn)*100
    print("El porcentaje de la variación en la variable de respuesta (R^2)es:",r2rn,"%")

```

PREDICCIÓN USANDO VALORES ALEATORIOS

```
[ ] CC = [[10]] # [L] Ingrese el Valor del Consumo de Combustible en Litros
d = [[48.8]]# [km] Ingrese el Valor de la Distancia en kilómetros
i = np.array(CC) / np.array(d)
print(" El Consumo de Combustible por distancia recorrida es: ",str(i),"L/km")
prediccion = red.predict(i)
print("El resultado del Factor de emision es: ",str(prediccion),"gCO2/km")
```

```
[ ] print('Error absoluto medio (MAE):', metrics.mean_absolute_error(y_test, y_predrn))
print('Error cuadrático medio(MSE):', metrics.mean_squared_error(y_test, y_predrn))
print('Raíz del Error cuadrático medio(RMSE):', np.sqrt(metrics.mean_squared_error(y_test, y_predrn)))
```

Anexo 6. Código de Programación Redes Neuronales Artificiales Multicapa RNAM**ANEXO VII****CÓDIGO DE PROGRAMACIÓN MÁQUINA DE VECTORES DE SOPORTE SVMR****IMPORTACIÓN DE LIBRERIAS**

```
[ ] import pandas as pd
import numpy as np
import pylab as pl
import tensorflow as tf
import matplotlib.pyplot as plt
import seaborn as sns
import math
from sklearn.model_selection import train_test_split
#MAQUINA DE VECTORES
from sklearn.svm import LinearSVC
from sklearn.svm import SVC
from sklearn.svm import SVR
```

IMPORTACIÓN DE DATOS

```
[ ] from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

```
[ ] dataset = pd.read_csv ("/content/drive/MyDrive/DATOSCSV/RUTARDECOMPLETAFICHERO.csv"
,sep=";",encoding="utf-8")
dataset.head()
```

VERIFICACIÓN DE DATOS

```
[ ] dataset.shape
```

```
[ ] dataset.value_counts()
```

```
[ ] dataset.columns
```

```
[ ] dataset.describe()
```

```
[ ] datos = dataset[["TIEMPO", "ALTITUD", "MAP(kPa)", "RPM", "V(km/h)", "IAT(°k)",
                    "TPS", "CC(L/s)", "CC(L/km)", "DensidadEntrada(kg/m³)",
                    "MAF(g/s)", "RendimientoVolumetrico", "CC(L/h)", "CO2(g/km)"]]
datos.head()
```

```
[ ] corr_val = datos[["TIEMPO", "ALTITUD", "MAP(kPa)", "RPM",
                    "V(km/h)", "IAT(°k)", "TPS",
                    "DensidadEntrada(kg/m³)", "MAF(g/s)",
                    "RendimientoVolumetrico", "CC(L/h)", "CO2(g/km)"]]
```

MÁQUINAS VECTORES DE SOPORTE (REGRESIÓN)

```
[ ] df0 = pd.DataFrame(datos)
df = df0[["CC(L/km)", "CO2(g/km)"]]
```

ENTRENAMIENTO

```
[ ] X = np.array(df.iloc[:, :-1], dtype=float)
y = np.array(df.iloc[:, -1], dtype=float)
```

GRÁFICAS DE CORRELACIÓN DE LOS DATOS

```
sns.pairplot(corr_val, kind="scatter")
plt.show()
```

```
[ ] sns.heatmap(corr_val.corr("spearman").sort_values(by="CO2(g/km)"), annot=False,
                cmap=sns.diverging_palette(250, 10, n=9))
```

```
[ ] corr_val.corr("spearman")['CO2(g/km)']
```


ENTRENAMIENTO

```
[ ] X = np.array(df.iloc[:, :-1], dtype=float)
    y = np.array(df.iloc[:, -1], dtype=float)
```

```
[ ] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20,
                                                    random_state=1)
    print("División de datos en Entrenamiento/Test Completa")
```

División de datos en Entrenamiento/Test Completa

```
[ ] regresionsvr = SVR(kernel="rbf", C=38, gamma=0.1)
    regresionsvr.fit(X_train, y_train)
```

PREDICCIÓN

```
[ ] y_predmv = regresionsvr.predict(X_test)
    np.set_printoptions(precision=2)
    print(np.concatenate((y_predmv.reshape(-1,1), y_test.reshape(-1,1)),1))
```

EVALUACIÓN

```
[ ] r2mv = r2_score(y_test, y_predmv)*100
    print("El porcentaje de la variación en la variable de respuesta (R^2) es: ", r2mv, "%")

print('Error absoluto medio (MAE):', metrics.mean_absolute_error(y_test, y_predmv))
print('Error cuadrático medio (MSE):', metrics.mean_squared_error(y_test, y_predmv))
print('Raíz del Error cuadrático medio (RMSE):', np.sqrt(metrics.mean_squared_error(y_test, y_predmv)))
```

PREDICCIÓN USANDO VALORES ALEATORIOS

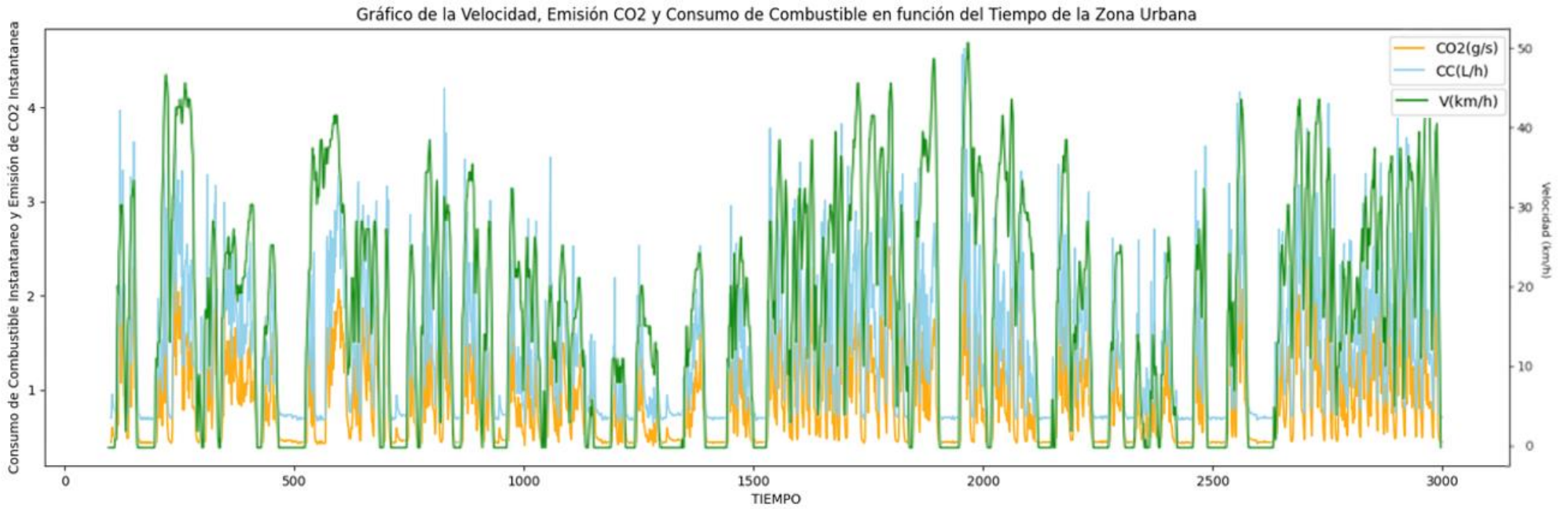
```
[ ] CC = [[2.729]] # [L/h] Ingrese el Consumo de Combustible en Litros
    d = [[48.8]] # [km] Ingrese la Distancia Recorrida
    i = np.array(CC) / np.array(d)
    prediccion = regresionsvr.predict(i)
    print("El Factor de emisión generado es de: ", str(prediccion), "gCO2/km")
```

```
[ ] dfsvm = pd.DataFrame({"Muestra Registrada": y_test, "Predicción": y_predmv})
    dfsvm.head()
```

Anexo 7. Código de Programación Máquina de Vectores de Soporte Regresor SVMR

ANEXO VIII

VELOCIDAD vs CONSUMO DE COMBUSTIBLE & EMISIÓN CO₂ A LO LARGO DE LA ZONA URBANA



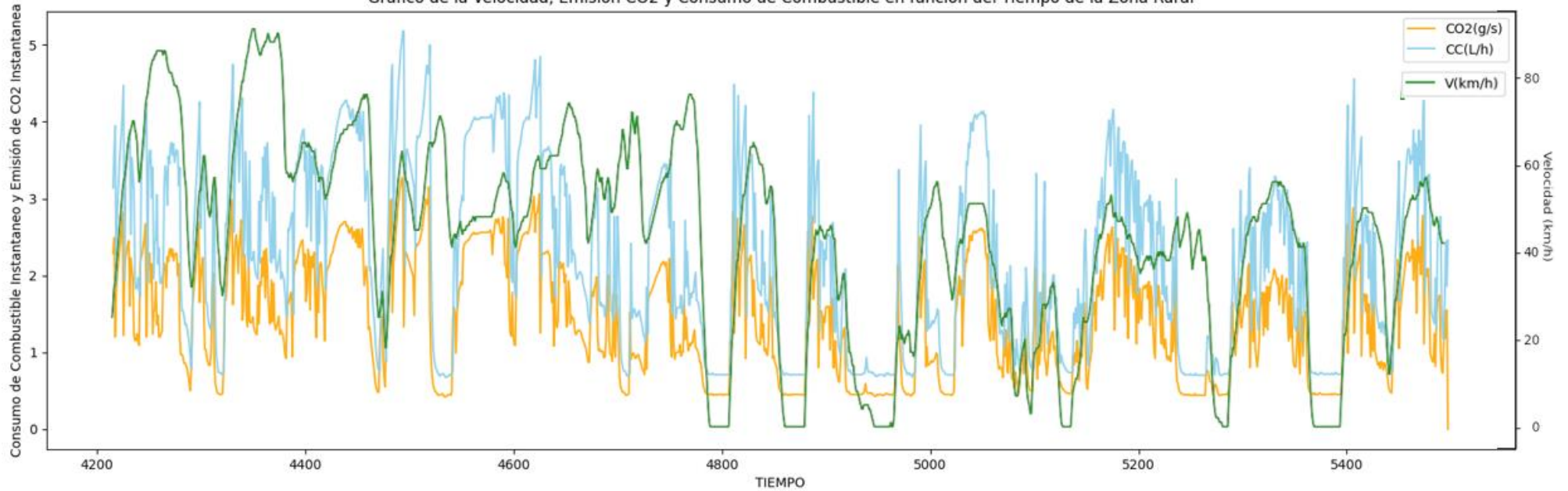
Anexo 8. FECO₂, Consumo Combustible vs Velocidad en Zona Urbana

ANEXO IX

VELOCIDAD vs CONSUMO DE COMBUSTIBLE & EMISIÓN CO₂ A LO LARGO DE LA ZONA RURAL

Anexo 9. FECO₂, Consumo Combustible vs Velocidad en Zona Rural

Gráfico de la Velocidad, Emisión CO₂ y Consumo de Combustible en función del Tiempo de la Zona Rural



ANEXO X

VELOCIDAD vs CONSUMO DE COMBUSTIBLE & EMISIÓN CO₂ A LO LARGO DE LA ZONA AUTOPISTA

Anexo 10. FECO₂, Consumo Combustible vs Velocidad en Zona de Autopista

