Uso de técnicas de Aprendizaje de Maquina para la estimación de emisión de CO_2 de vehículos en Canadá

Joan Sebastián Bonilla Guerra[1], Lorena Patricia Mora Hernandez[1] y Robinsson Yesid Sánchez Deantonio[1]

[1] Pontificia Universidad Javeriana, Bogotá CO 110231, Colombia

1 Introducción

El aumento significativo de las emisiones de gases de efecto invernadero está impactando de manera considerable, negativa y no reversible el planeta y bienestar de las personas. Este efecto intensifica el cambio climático que según la NASA (2022) está ocasionando temperaturas más elevadas, mayor cantidad de inundaciones, erosiones, sequias, huracanes, extinción de especies y otros efectos perjudiciales. Estas consecuencias repercuten en determinantes de salud relacionados con aire limpio, agua potable, alimentos suficientes y vivienda segura según la OMS (2021). Estos problemas han impulsado a las instituciones gubernamentales y privadas a evaluar la situación actual de la emisión de gases de efecto invernadero.

Uno de los principales gases en el efecto invernadero es el Dióxido de Carbono (CO_2). La principal fuente de generación de este gas es la quema de combustibles fósiles desde el sector de transporte según un informe de la EPA (2022). Según datos de Our World in Data (2022) la emisión de CO_2 por combustibles fósiles se ha incrementado 7.6 veces en los últimos 74 años. Según la IPCC (2019) este gas tiene un potencial de calentamiento global (GWP) de 100 años que puede durar en la atmosfera miles de años, por lo cual se considera importante el uso de la Inteligencia Artificial (IA) para pronosticar la emisión de este gas y proponer recomendaciones que disminuyan el impacto ambiental desde el transporte.

En ese sentido el uso de técnicas de aprendizaje de maquina guardan un gran potencial para predecir variables basándose en un conjunto de atributos que guardan en cierta medida un grado de correlación o correspondencia con ésta, de igual manera, se busca predecir la cantidad de gramos por kilómetro que emiten los vehículos, de esta forma se puede conocer el grado de contaminación generada por este medio de transporte y pueden ser útiles como medida de comparación con las emisiones reales generadas por un vehículo, de esta forma se puede evidenciar la existencia de ineficiencias en el proceso de combustión o fallas mecánicas que pueden verse representadas en un aumentos de agentes contaminantes. Otra aplicación de un modelo con la capacidad de predecir las emisiones se puede ver a nivel macro en donde se puede hacer uso de las predicciones para conocer la cantidad de emisiones que se dan por la movilidad vehicular y conocer su comportamiento a través del tiempo, así como conocer el aporte de emisiones por parte de este sector.

2 Estado del Arte

2.1 Modelos de Aprendizaje de Maquina

En estudios anteriores se han utilizado técnicas de aprendizaje de máquina pa-ra predecir emisiones de CO2 de vehículos. Dentro de estas técnicas se han utili-zado regresiones lineales, algoritmos Lasso y Ridge, K-vecino más cercano, arbo-les de decisión, máquina de soporte vector (SVM), entre otros han sido utilizados para la predicción de emisiones de combustible de vehículos.

Debido a la alta población y demanda por infraestructura de transporte en In-dia, el impacto de emisiones de efecto invernadero por parte del país ha sido alto. Kangralkar y Khanai (2021) utilizaron las técnicas (1) K-vecino más cercano, (2) SVM, (3) Regresión linear, (4) Arboles de decisión y (5) Arboles aleatorios. El coeficiente de precisión para cada uno de los modelos fue de 28%, 80%, 82.5%, 96%, y 99% respectivamente.

Song y Cha (2022), realizaron la predicción de emisiones de CO2 a partir de variables públicas, es decir, variables disponibles en los manuales de los vehícu-los o en el rendimiento de estos. Esta predicción fue formulada a partir de regre-siones lineales, gracias a que demuestran una relación clara y directamente pro-porcional entre el consumo de combustible, el poder del motor y las emisiones de CO2. El error obtenido de los experimentos con el modelo propuesto varía entre 6.1% y 17.9%.

Aliramezani et al, (2020) usaron SVM para predecir el NOx en vehículos die-sel. También Mohammad et al. (2022) utilizaron un algoritmo Lasso junto con SVM para alimentar un modelo de predicción que usa redes neuronales para es-timar la cantidad de emisiones de NOx, CO y HC ocasionada por motores de combustión interna. El uso del algoritmo Lasso mejoro la precisión del modelo.

3 Metodología

La metodología consistió en realizar modelos de aprendizaje de maquina como Regresión Lineal, Arboles de decisión, Regularizaciones L1 y L2 y métodos de ensambles como es el caso de Bagging y Boosting, para cada uno de los diferentes algoritmos se utilizó optimización de hiper-parámetros a través de iteración por grilla como GridSearch y usando cross validation , posteriormente se compararon los diferentes algoritmos optimizados para comprobar los resultados y verificar cuales tuvieron un mejor ajuste en los resultados, se siguieron metodologías de análisis como CRISP-DM a partir de la cual se hace un estudio de los objetivos del análisis y entendimiento de los datos disponibles hasta el procesamiento, modelado y validación de los resultados con el fin de extraer ideas relevantes que pueden transformarse en un conjunto de acciones a fin de satisfacer los objetivos inicialmente planteados.

4 Modelos de Aprendizaje de Maquina

Los Algoritmos de ML son utilizados ampliamente en problemas cotidianos dado que son sencillos de implementar y se obtienen buenas métricas de ajustes, además se cuenta con una amplia gama de algoritmos que pueden ser aplicados tanto para modelos de no supervisados y supervisados como es al caso de regresiones y clasificaciones,

A continuación, se mostrarán los algoritmos que se utilizaron para obtener un modelo que tenga las mejores métricas en la validación.

4.1 Regresión y Regularización

La Regresión Linear es un tipo de análisis predictivo básico y de amplia utilización, la idea general es poder predecir una variable objetivo a través de un conjunto de variables predictoras, de esta forma, se obtiene una ecuación de la forma **Y** = a + b**X**, donde **X** es el conjunto de variables predictoras y **Y** es la variable objetivo, donde b hace referencia a los pesos de cada variable y a es el valor cuando todas las variables son igual a cero.

Por otro lado, *la regularización* es una forma de regresión que penaliza o reduce las estimaciones de los coeficientes de las variables predictoras a cero, de esta forma se desalienta el aprendizaje de un modelo mas complejo para mejorar la explicabilidad y reducir el riego de sobreajuste, todo esto por medio de una ecuación de costo que castiga los coeficientes evitando que estos crezcan demasiado e incluso volviéndolos igual a cero, para este caso se aplicó regularización L1 o Lasso como se observa en la Ecuación 1 y L2 o Ridge que se observa en la Ecuación 2, la principal diferencia de estas técnicas es que Lasso reduce el coeficiente de la característica menos importante a cero, eliminando así variables predictoras por completo, en ese sentido Lasso funciona bien en la selección de características cuando se tiene una gran cantidad de estas.

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| = \text{suma residuos cuadrados} + \lambda\sum_{j=1}^{p}|\beta_j| \qquad (1)$$

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 = \text{suma residuos cuadrados} + \lambda\sum_{j=1}^{p}\beta_j^2 \qquad (2)$$

Por último, existe una combinación de ambas que se conoce como redes elásticas o Elastic Net, esta regularización combina linealmente las penalizaciones L1 y L2

$$\frac{\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2}{2n} + \lambda\left(\alpha\sum_{j=1}^{p}|\beta_j| + \frac{1-\alpha}{2}\sum_{j=1}^{p}\beta_j^2\right) \qquad (3)$$

4.2 Arboles de decisión

Los árboles de decisiones son algoritmos ampliamente utilizados para clasificación y predicción dadas sus buenos ajustes y altas precisiones en modelos a comparación de otros, este tiene una estructura similar a un diagrama de flujo donde cada nodo denota una regla y cada hoja representa el resultado de esa prueba, todo esto se fundamenta bajo un proceso de división binaria que de forma recursiva va generando diferentes divisiones utilizando una función de costo.

Dentro de sus ventajas se encuentra que son fáciles de leer e interpretar y requieren menor preparación de datos, dado que este algoritmo no se ve afectado por presencia de atípicos y no es necesario dumificar variables categóricas, por otro lado, los arboles tienen una naturaleza inestable debido a la alta dependencia de los datos de entrenamiento, un pequeño cambio en estos datos puede generar un cambio importante en la estructura del árbol y son menos eficaces en la predicción de variables continuas.

4.3 Boosting

Es una de las categorías en los que se puede dividir los métodos de ensamble, la característica de este método es que busca reducir la reducción del sesgo, de esta forma, algoritmos sencillos son utilizados secuencialmente, así, el rendimiento general puede mejorarse haciendo que un modelo posterior de más importancia a los errores cometidos en el modelo previo, a diferencia del bagging los algoritmos no se entrenan independientemente, si no que se ponderan de errores anteriores.

Para este caso se seleccionaron los algoritmos XGBoost, ADABoost, Lightgbm y CATBoost, en especial los 2 primeros son ampliamente conocidos por sus precisiones en los modelos, sin embargo, en diferentes pruebas se ha demostrado que Lightgbm y CATBoost pueden tener mejores rendimientos en los resultados.

5 Protocolo experimental

Para poder hacer una compasión de los modelos que sea relevante para realizar un análisis de las precisiones de cada uno de los modelos propuestos es necesario realizar un optimización de hiper-parámetros, de esta forma se busca obtener los mejores resultados que pueda llegar a darnos cada algoritmo, de esta forma se usaron métodos de validación cruzada y grilla de parámetros para poder ajustar iterativamente parámetros y donde fueron seleccionados a partir de los resultados de las métricas en los datos de validación

Dentro del proceso experimental se procuró analizar los resultados obtenidos en términos de la velocidad y su eficiencia al momento de predecir los valores de un conjunto de validación, para cada modelo se hizo 10 pruebas con el fin de obtener un promedio de los resultados, dado que este protocolo experimental busca conocer cuales algoritmos tienen un mejor desempeño no se realizó iteraciones adicionales en los modelos optimizados dado que sus características y parámetros no son similares.

Para la optimización general de hiper-parámetros se utilizo GridSearch apoyado una validación cruzada de 5 particiones,

5.1 Optimización de Modelos

5.1.1 Regresión y Regularización

El modelo generado por Regresión Linear no tuvo la necesidad de generar optimizaciones dado que es un modelo que por su poca complejidad no utiliza parámetros que permitan mejorar su rendimiento, por otro lado, los métodos de regularización utilizan variables que pueden ser cambiadas y que pueden entregar un mejor desempeño del modelo, en el caso de **Ridge, Lasso** y **ElasticNet** el parámetro de mayor relevancia es *alfa*, en donde valores grandes de alfa especifican una regularización más fuerte, de esta forma mejora el condicionamiento del problema y reduce la varianza, adicionalmente **ElasticNet,** cuenta con una parámetro adicional *l1_ratio* que permite cambiar la distribución de la penalización, es decir, si *l1_ratio* es igual a cero la penalización será de tipo L2 o Ridge, caso contrario, si *l1_ratio* es igual a 1 la penalización será L1 o Lasso.

Cuando se utiliza regularización, es útil evaluar cómo se aproximan a cero los coeficientes a medida que se incrementa el valor de alpha.

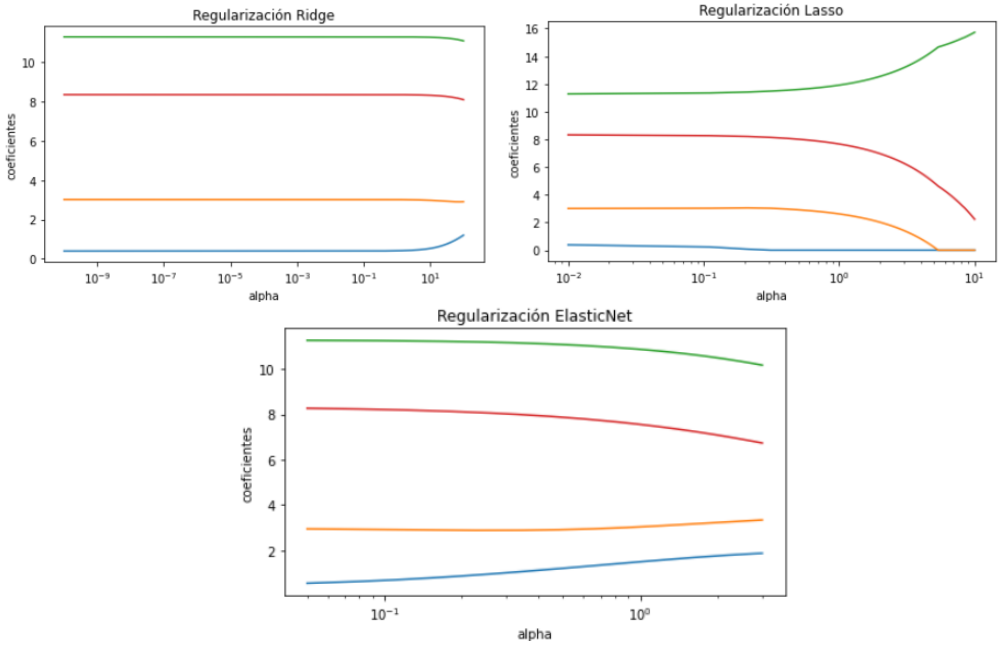

Figura 1. Coeficientes del modelo en función de alfa

5.1.2 Arboles de decisión

Dentro de los parámetros iterados en los árboles de decisión como se observa en la Tabla 1, se encuentra la estrategia utilizada para elegir la división en cada nodo, la profundidad máxima del árbol y el número de características a considerar al buscar la mejor división, el número mínimo de muestras requeridas para estar en un nodo hoja, es decir, un punto de división a cualquier profundidad solo se considerará si deja al menos una cantidad mínima de muestras en cada una de las ramas.

Table 1. Neg-RMSE de la iteración de parámetros en árboles de decisión

param_max_depth	param_max_features	param_min_samples_leaf	param_splitter	mean_test_score
None	None	0.001	random	-11.841859
None	None	0.0025	random	-12.311239
None	log2	0.001	random	-12.618990
None	auto	0.001	random	-12.657984
None	sqrt	0.001	random	-12.665345

5.1.3 Boosting

Para los métodos de ensamble por boosting que realizaron las siguientes variaciones.

Lightgbm: tipo de algoritmo para boosting, 'gbdt' traditional Gradient Boosting Decision Tree y goss' Gradient-based One-Side Sampling, máximo de hojas de árbol para las muestras base, término de regularización L1, tasa de aprendizaje y el número de árboles a entrenar.

Table 2. Neg-RMSE de la iteración de parámetros en Lightgbm

param_boosting_type	param_learning_rate	param_n_estimators	param_num_leaves	param_reg_alpha	mean_test_score
gbdt	0.05	1000	30	0.3	-12.364794
gbdt	0.04	1500	20	0.2	-12.374323
gbdt	0.03	2000	20	0	-12.376487
gbdt	0.03	2000	20	0.4	-12.377077
gbdt	0.04	1000	20	0.4	-12.379601

XGBoost: La profundidad máxima de cada árbol, número de caracteristicas (variables) utilizadas en cada árbol, el número de muestras (filas) utilizadas en cada árbol, tasa de aprendizaje utilizada para ponderar cada modelo y el número de árboles en el conjunto, a menudo aumentado hasta que no se ven más mejoras.

Table 3. Neg-RMSE de la iteración de parámetros en XGBoost

param_colsample_bytree	param_eta	param_max_depth	param_min_child_weight	param_n_estimators	param_subsample	mean_test_score
1	0.04	5	1	1500	1	-10.077030
1	0.04	5	0.5	1500	1	-10.077030
1	0.04	5	0	1500	1	-10.077030
1	0.04	5	1	2000	1	-10.081293
1	0.04	5	0.5	2000	1	-10.081293

Catboost: La profundidad máxima de cada árbol, la tasa de aprendizaje, este ajuste se utiliza para reducir el paso de gradiente. Afecta el tiempo total de entrenamiento: cuanto más pequeño es el valor, más iteraciones se requieren para el entrenamiento, cantidad de iteraciones y por último un parámetro de regularización L2.

Table 4. Neg-RMSE de la iteración de parámetros en **CatBoost**

param_depth	param_iterations	param_l2_leaf_reg	param_learning_rate	mean_test_score
5	1800	0.5	0.03	-8.941961
5	2100	0.5	0.03	-8.945898
5	1500	0.5	0.03	-8.963530
5	1500	0.5	0.05	-9.092230
5	1800	0.5	0.05	-9.102178

AdaBoost: La función de pérdida que se utilizará al actualizar los pesos después de cada iteración, tasa de aprendizaje aplicado a cada regresor en cada iteración, El número máximo de estimadores en los que finaliza el refuerzo

Table 5. Neg-RMSE de la iteración de parámetros en AdaBoost

param_learning_rate	param_loss	param_n_estimators	mean_test_score
0.3	linear	900	-15.879227
0.2	linear	900	-15.951693
0.1	linear	700	-16.085512
0.3	linear	1100	-16.199501
0.1	linear	1100	-16.249219

5.2 Análisis de resultados

Para las diferentes técnicas de aprendizaje de maquina se obtuvieron los resultados descritos en la siguiente sección. Para estos resultados se evalúan tiempos y errores como resultado de la optimización de los diferentes modelos como lo son de regresión, regularización, arboles de decisión y métodos de ensamble.

5.2.1 Tiempos de ejecución de los modelos

Para los diferentes modelos de aprendizaje de maquina se puede observar una clara diferencia entre los tiempos de entrenamiento los algoritmos que generan solo modelo a comparación de los métodos de ensamble que generan una combinación de varios modelos para poder realizar la predicción de los resultados.

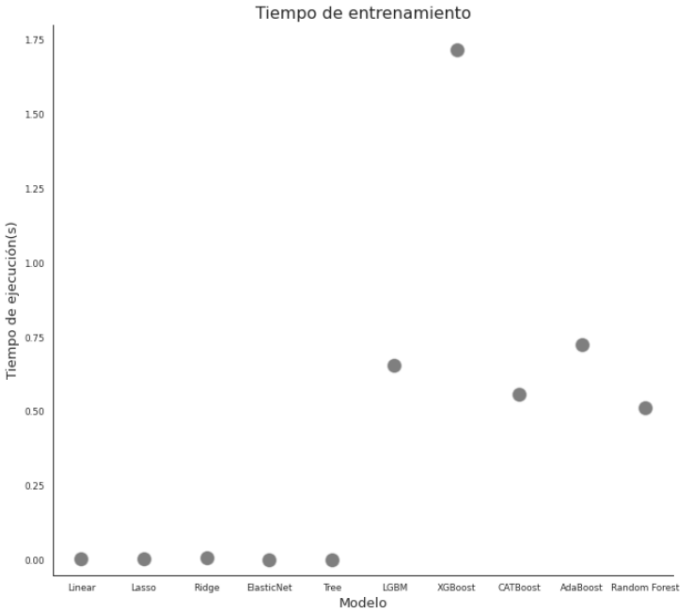

Figura 2. Tiempo de entrenamiento por cada modelo de aprendizaje de maquina

Se puede observar en la gráfica anterior que en general los métodos de regresión, las regularizaciones y los arboles de decisión tienen tiempos de entrenamiento muy bajo, alrededor de los 0.05 segundos, por otro lado, los métodos de ensamble son por lo menos 100 veces mas lento, en general los algoritmos de boosting fueron mas demorados que el de bagging, en todo caso XGBoost fue el algoritmo que mas demoro en su entrenamiento sobrepasando por mucho a los demás algoritmos.

5.2.2 Métricas de error y desempeño

Dentro de los resultados arrojados en los datos de entrenamiento que representan al 30% de los datos disponibles en la base de datos, se observa que en la mayoría de los modelos de ensambles tanto boosting como bagging se obtienen unas mejores métricas de desempeño (ver Figura 3), dado que estos modelos tienen un tiempo mayor de ejecución y una complejidad superior se esperaba que así mismo se obtuvieran métricas de desempeño mejores, en términos de neg-RMSE el modelo que mejor logra predecir datos por fuera de las muestras de entrenamiento es CatBoost con un -11.83 gr/km

seguido del método de bagging Random Forest con -13.34 gr/km, también, el algoritmo con el peor desempeño pero por muy poco es AdaBoost con 15.88 gr/km seguido del modelo de regularización ElasticNet con 15.83 gr/km, sin embargo, en general todos los modelos utilizados tuvieron buenos desempeños logrando coeficiente de determinación por R^2 de 0.945, lo que hace que cualquier modelo analizado puede ser de alta utilidad para predecir la cantidad de CO2 por kilómetro recorrido de un automóvil.

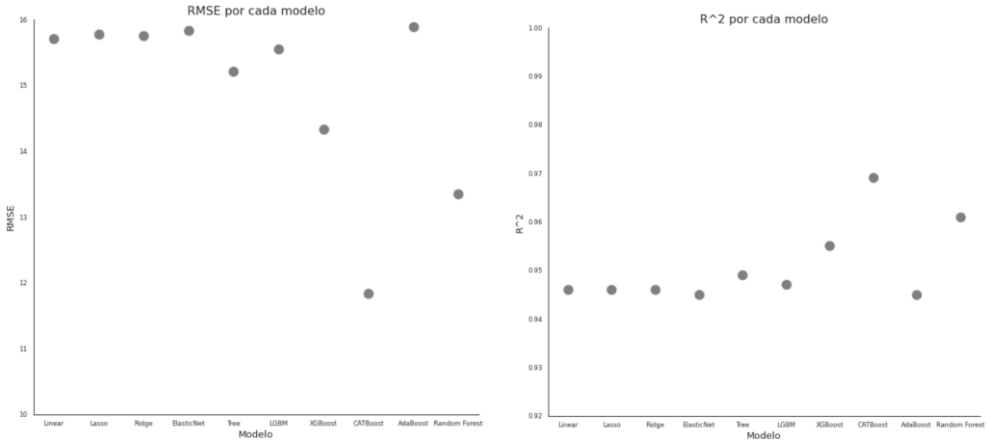

Figura 3. RMSE y coeficiente de determinación de la validación de los modelos de aprendizaje de maquina

En la siguiente tabla se muestra un resumen de estos resultados.

Table 1. Resultados estadísticos de entrenamiento y validación de los modelos

model	exec_time	RMSE_train	RMSE_test	r^2_train	r^2_test
Linear	0.004	14.784	15.704	0.946	0.946
Lasso	0.002	14.794	15.766	0.945	0.946
Ridge	0.002	14.800	15.747	0.945	0.946
ElasticNet	0.001	14.840	15.826	0.945	0.945
Tree	0.001	0.194	15.205	1.000	0.949
LGBM	0.605	5.934	15.543	0.991	0.947
XGBoost	1.636	0.436	14.332	1.000	0.955
CATBoost	0.543	1.277	11.835	1.000	0.969
AdaBoost	0.626	12.141	15.884	0.963	0.945
Random Forest	0.457	3.577	13.340	0.997	0.961

6 Conclusiones

- En estudios anteriores se han utilizado técnicas de aprendizaje de máquina para predecir la cantidad de emisiones generadas por un vehículo. Se han utilizado regresiones lineales, arboles de decisión, arboles aleatorios, SVM, algoritmos Lasso y Ridge, y otras técnicas adicionales.

- Los algoritmos de ensamble tienen un mayor tiempo de entrenamiento y por lo general tienen un mejor desempeño que los algoritmos de regresión o arboles de decisión.
- Una de las formas más eficientes para ajustar hiper-parámetros es utilizar una grilla de parámetros como GridSearch o RandomSearch para evaluar un conjunto de parámetros y obtener un conjunto de resultados de cada iteración para evaluar cual es la mejor combinación de estos

7 Referencias

Aliramezani m, Norouzi A, Koch CR. Support vector machine for a diesel engine perfor-mance and NOx emission control-oriented model. 21st IFAC World Congress in Berlin, Germany 2020.

Government of Canada. (2022). *Fuel consumption ratings.*

Intergovernmental Panel on Climate Change. (2019). *Calentamiento global de 1,5°C.*

Mohammad, A., Rezaei, R., Hayduk, C., Delebinski, T., Shahpouri, S., & Shahbakhti, M. (2022). Physical-oriented and machine learning-based emission modeling in a diesel com-pression ignition engine: Dimensionality reduction and regression. International Journal of Engine Research. https://doi.org/10.1177/14680874211070736

Organización Mundial de la Salud. (2021). *Cambio climático y salud.*

Ritchie H., Roser M. (Agosto 2020). CO_2 and Greenhouse Gas Emissions. *Our World in Data* recuperado de https://bit.ly/3hp5q5Q

S. Kangralkar and R. Khanai, "Machine Learning Application for Automotive Emission Prediction," 2021 6th International Conference for Convergence in Technology (I2CT), 2021, pp. 1-5, doi: 10.1109/I2CT51068.2021.9418152.

Shaftel H., Callery S., Jackson R., Bailey D. (23 febrero 2022). Los efectos del cambio climatico. *Earth Science Communications Team NASA's Jet Propulsion Laboratory California Institute of Technology.* Recuperado de https://go.nasa.gov/3ht4vRG

Song, Jingeun & Cha, Junepyo, 2022. "Development of prediction methodology for CO2 emissions and fuel economy of light duty vehicle," Energy, Elsevier, vol. 244(PB).

United States Environmental Protection Agency. (2022). *Overview of Greenhouse Gases.*