Análisis: Emisión de CO2 de Vehículos en Canadá

Jhon Camilo Baron Berdugo, Manuel Alejandro Pontón Rico
Estudiantes de VI Semestre Tecnología en Desarrollo de Software
Fundación Universitaria Tecnológico Comfenalco
Cartagena, Colombia

INTRODUCCION

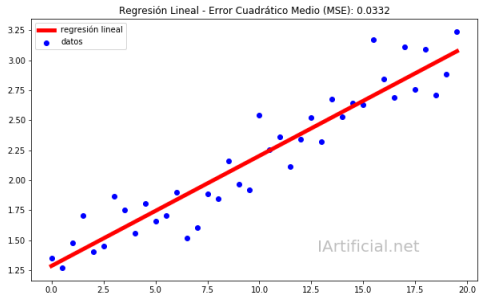
En el siguiente informe, lograremos identificar el paso a paso que se llevó a cabo para poder realizar un análisis a un conjunto de datos obtenidos a partir de la página oficial de datos abiertos canadiense con respecto a la emisión de gases que producen los vehículos en este país.

PREPARACIÓN

Para poder comenzar cualquier tipo de investigación es necesario de que el grupo de trabajo o investigador se cuestione así mismo y pueda responder *¿qué es lo que quiero y con qué herramientas puedo resolverlo?*; en este orden de ideas, este grupo de trabajo quiere predecir las posibles emisiones de dióxido de carbono (CO2) de vehículos en Canadá a partir de la implementación de la recolección de datos abiertos en páginas oficiales del gobierno Canadiense, visualización de la información recolectada y tecnologías que agilicen estos procesos con un algoritmo que permita la predicción de estas emisiones como lo sería: la regresión linear. El aprendizaje automático (Machine Learning) contiene una gran cantidad de aplicaciones en los diferentes campos profesionales, y en el siguiente análisis se adentrará, una vez más, en pro de contribuir en un factor que contribuye de forma directa a fenómenos como el calentamiento global.

FUNDAMENTOS: ¿QUÉ ES UNA REGRESIÓN LINEAL?

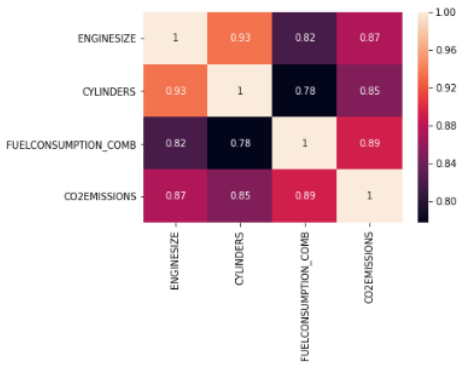
La regresión permite, determinar el grado de dependencia de las series de valores X e Y, prediciendo el valor y estimado que se obtendría para un valor x que no esté en la distribución.



Regresión Lineal.

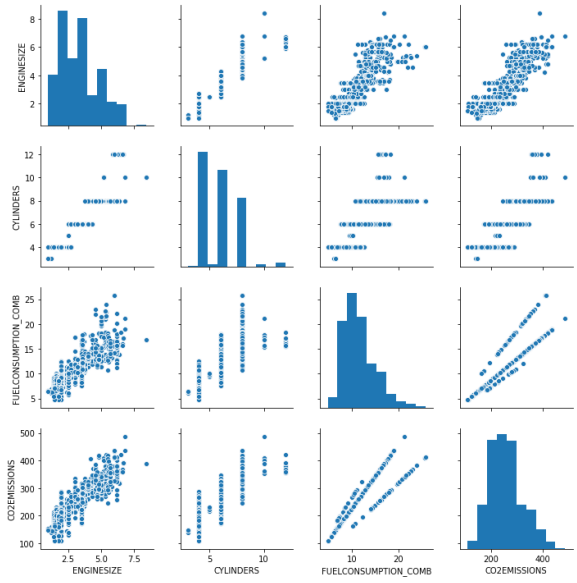
PREPROCESAMIENTO DE DATOS

Luego de haber almacenado los datos recolectados en un *DataFrame* se logró identificar cuáles eran las variables que mayor importancia tenían para poder ser incluidas en el modelo de regresión linear simple a partir de la correlación que existía entre las variables independientes (ENGINESIZE, CYLINDERS, UELCONSUMPTION_COMB) con la dependiente (*CO2EMISSIONS)*; gracias a la librería **Seaborn** logramos identificar sus coeficientes de correlación y visualizarlos en un mapa de calor.



Mapa de Calor utilizando Seaborn

Como podemos darnos cuenta, la variable FUELCONSUMPTION_COMB tiene un mayor coeficiente de correlación a comparación de las otras, pero en realidad al realizar una matriz con gráficos relacionados logramos nuevamente identificar 'anomalías' en la distribución de los datos en dichas graficas.



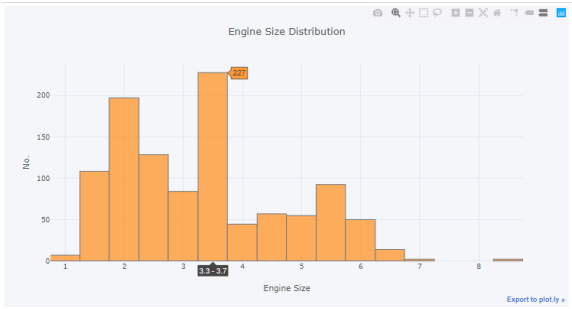
Matriz Relacional de Variables utilizando Seaborn

De la anterior imagen, nos ubicamos en la relación entre FUELCONSUMBTION_COMB y CO2EMISSIONS; nos damos cuenta que en el momento de trazar una línea recta que logre ajustarse a los datos de este par de variables, vamos a obtener un error sumamente alto, es por esto que para el modelo se decidió por seleccionar al ENGINESIZE como variable independiente ya que esta sí tiene una mejor distribución lineal con respecto al CO2EMISSIONS.

ENGINESIZE & CO2EMISSIONS

Utilizando la librería **Cufflinks** logramos realizar algunas visualizaciones interactivas y obtener resultados de interés como, por ejemplo:

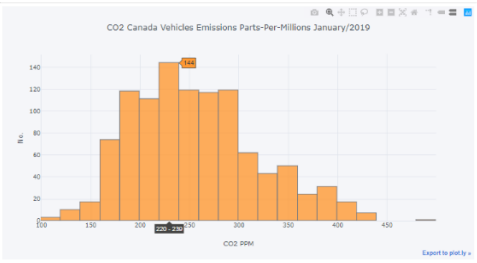
- Existen 227 motores dentro del *DataSet* que se encuentran entre 3.3 y 3.7 centímetros cúbicos.



Distribución: Tamaños de Motor

- El mayor número de Partículas por Millón (PPM) de dióxido de carbono emitidos por los vehículos de Canadá

en Enero/2019 fue de 480 a 490, pero 144 veces se dieron casos de 220 a 239 PPM, dando un resultado aproximado de 32832 PPM.



Distribución de Dióxido de Carbono en PPM

Luego de haber identificado nuestras variables necesarias para construir nuestro modelo de regresión linear, importamos de la librería *Scikit-Learn* las funciones necesarias para poder realizar llevar a cabo el entrenamiento del modelo y su implementación para realizar predicciones.

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
```

Importando a partir de Scikit-Learn

Con la implementación de *train_test_split* lograremos dividir nuestros datos en valores de entrenamiento y valores de prueba, definir un porcentaje de estos para entrenamiento y el resto para 'test' y establecer una semilla aleatoria para que se distribuyan los datos de forma aleatoria. Se establece el modelo como una *LinearRegression* y se le ingresa en los parámetros aquellas variables con los datos de entrenamiento. Finalmente, se realizan predicciones con valores que el modelo-algoritmo no ha visto, es decir, con la que no fue entrenada y obtener un resultado para poder, a través de métricas, medir qué tan efectivo es nuestro modelo.

Data Split & Linear Model

```
In [11]: x_train, x_test, y_train, y_test = train_test_split(x, y, train_size=0.8, random_state=42)
In [12]: model = LinearRegression()
In [13]: model.fit(x_train, y_train)
Out[13]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

Entrenamiento del Modelo.

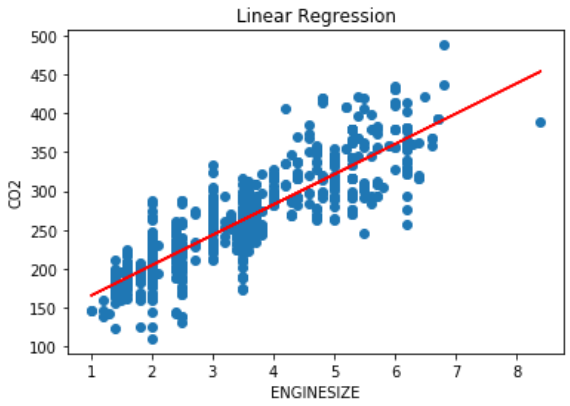
Utilizando métricas como el *r2_score* logramos obtener la correlacionalidad que existe entre los datos de prueba (y_test) y las predicciones obtenidas por el modelo (y_predict)

Performance ¶

```
In [17]: 100*(r2_score(y_test, y_predict))
Out[17]: 76.15595731934374
```

Implementación de r2_score

Por último, podemos observar la gráfica que se obtiene a partir de los datos de entrenamiento, la pendiente y el intercepto obtenidos por el modelo de regresión linear, además se anexan algunos ejemplos para determinar qué tan efectivo es este modelo de regresión linear para lograr predecir las emisiones de gases en vehículos de Canadá a partir del tamaño de su motor.



Regresión Linear

Example

```
In [25]: co2_df.head(5)
```

Out[25]:

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244

```
In [26]: model.predict([[2.1]])
Out[26]: array([[208.1749575]])

In [27]: model.predict([[2.2]])
Out[27]: array([[212.07425537]])

In [34]: model.predict([[2.3]])
Out[34]: array([[215.97355324]])

In [35]: model.predict([[2.4]])
Out[35]: array([[219.87285111]])
```

CONCLUSIÓN

Podemos concluir que el modelo logra ser asertivo a la hora de realizar predicciones con la variable que fue identificada para la construcción de su modelo, pero en casos reales se utiliza una mayor cantidad de estas ya que son diferentes los factores que pueden afectar a que un vehículo, en este caso, libere mayor cantidad de Dióxido de Carbono.

HERRAMIENTAS UTILIZADAS

NOMBRE	JUSTIFICACIÓN
- Anaconda Distribution	Es la mejor plataforma para la realización de Data Science & Machine Learning ya que trae consigo la mayoría de librerías y entornos de desarrollo necesarios para el análisis y visualización de nuestros/resultados.
- Pandas	Permite la manipulación de datos almacenados en tipos de archivo '.csv'.
- Numpy	Permite crear arreglos con mayor optimización de iteración y almacenamiento; en el proyecto fue utilizado para almacenar los valores de las variables dependiente e independiente.
- Matplolib.pyplot	Permite la visualización básica de datos.
- Seaborn	Permite una visualización de datos más estética, como el caso del mapa de calor mostrando los coeficientes de correlación entre las variables independientes con la dependiente
- Cufflinks	Robusta herramienta para la visualización de datos de una forma más interactiva; utilizado para dar a conocer la distribución del tamaño de los

	motores en el dataset.
- Scikit-Learn	Principal librería para el machine Learning, ya que contiene la mayoría de algoritmos utilizados en este proceso.
- train_test_split	Permite dividir los datos de entrenamiento y test en 4 variables previamente declaradas (x_train, x_test, y_train, y_test)
- LinearRegression	Modelo de regresión linear seleccionado para la predicción de las emisiones de gases producidas por un vehículo con característica especifica.

REFERENCIAS

[1] Open Canada, Fuel Consumption Ratings: https://open.canada.ca/data/en/dataset/98f1a129-f628-4ce4-b24d-6f16bf24dd64

[2] Canada–Emission Project Repository: https://github.com/Jhooomn/learning-ML/tree/master/College%20Class/Canada%20-%20Emission