Car co2 emissions machine learning

Vincenzo Maria Giulio Martemucci (Matricola: 639321)

Link alla repository GitHub

Introduzione:

L'inquinamento atmosferico, con le sue conseguenze, ha raggiunto un livello critico. Gli effetti sono ormai sotto gli occhi di tutti ed anche i più scettici, potrebbero ricredersi davanti alle scene dei sempre più violenti eventi atmosferici degli ultimi tempi.

Fra le principali cause antropiche di inquinamento atmosferico, vi è il traffico veicolare. La stragrande maggioranza dei veicoli su strada utilizza motori a combustione.

Nonostante le sempre più stringenti regole antinquinamento, il problema delle emissioni è sempre attuale e non ancora risolto.

Inoltre, alcune caratteristiche delle automobili, sono molto impattanti sul risultato finale delle emissioni prodotte.

Sfruttando le conoscenze acquisite durante il corso di Ingegneria della Conoscenza, ho voluto sviluppare un sistema, che attraverso l'utilizzo di algoritmi di apprendimento supervisionato, sia capace di prevedere i valori di emissioni di un'automobile, in base ad alcuni dati in input che rappresentano determinate caratteristiche dell'automobile in questione.

In aggiunta a questo, il sistema, sul dataset utilizzato, costruisce una classificazione naturale dei dati attraverso un metodo di clustering, il K-Means, in modo da proporre all'utente del sistema, una serie di automobili con caratteristiche simili a quella le cui caratteristiche, vengono date in input al sistema.

Dataset utilizzato:

I dati utilizzati in questo progetto, sono stati ottenuti dal sito internet open.canada.ca, un sito Governativo Canadese per gli "open data" che mette a disposizione moltissimi dataset in svariati campi di interesse per cittadini o altre organizzazioni non governative, in modo tale da renderli accessibili ed analizzabili da chiunque.

Personalmente ho ritenuto opportuno affidarmi ad una fonte di dati forniti da un'Istituzione Governativa, poiché è da considerarsi affidabile ed imparziale.

Il dataset utilizzato non presentava valori nulli, spuri, rendendo così la sua manipolazione più agevole per gli scopi prefissati inizialmente.

Il dataset è organizzato nei seguenti attributi:

- **make**: marca del veicolo
- **model**: modello del veicolo
- **vehicle_class**: classe del veicolo (es. SUV, SEDAN, etc.)
- **engine_size**: cilindrata del veicolo (in cc)
- **cylinders**: numero di cilindri del veicolo
- **transmission**: tipo di trasmissione(manuale, automatico e numero di marce)
- **fuel_type**: carburante utilizzato dalla vettura (es. benzina, diesel, ibrido, etc.)
- **fuel_consumption_city**: consumo cittadino in l/100km
- **fuel_consumption_hwy**: consumo autostradale in l/100km
- **fuel_consumption_comb**: consumo combinato in l/100km
- **fuel_consumption_comb(mpg)**: consumo combinato in miglia per gallone
- **co2_emissions**: valore dichiarato di emissioni in g/km

Questi attributi rappresentano le caratteristiche tecniche di un'automobile, oltre che la loro marca e modello. L'attributo co2_emissions è l'attributo di cui si farà la previsione dopo un'opportuna costruzione di un modello di regressione.

Gli attributi che rappresentano la marca, il modello, i consumi e le emissioni invece, sono quelli mostrati per elencare le auto similari.

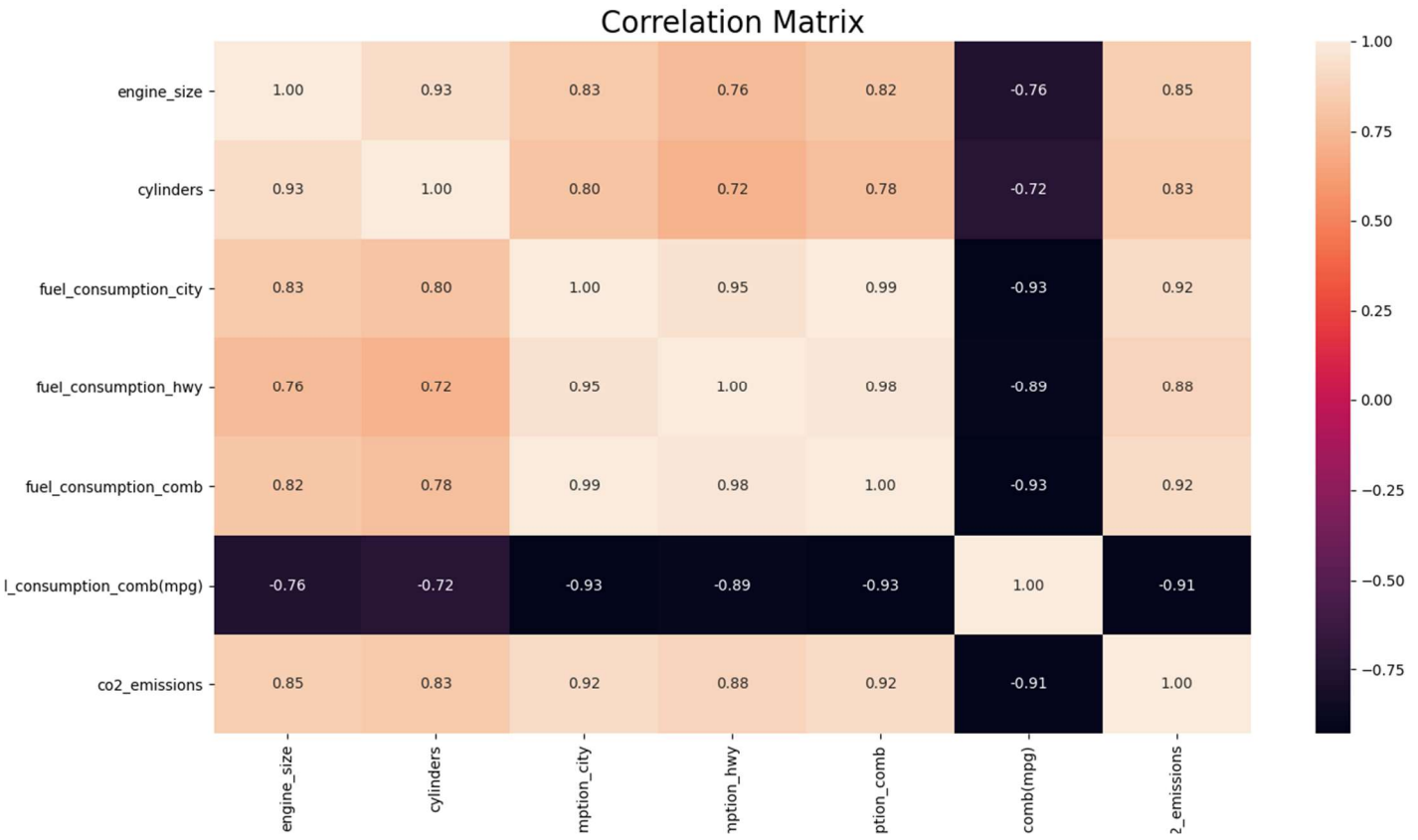
Non tutti gli attributi sono stati utilizzati per la costruzione del modello e per la creazione della lista di elementi simili, ma solamente **engine_size**, **cylinders**, e i tre valori di consumi espressi in l/100km.

Una possibile espansione futura del sistema può prevedere l'utilizzo di questi attributi.

Come premesso nell'introduzione, alcune caratteristiche delle vetture, hanno più impatto rispetto ad altre, sul risultato finale in termini di emissioni.

Ad esempio, un motore di grossa cilindrata produrrà più emissioni rispetto ad un motore di cilindrata minore.

Un'analisi del dataset, ci mostra quali siano le caratteristiche più impattanti. Un aumento della cilindrata corrisponde ad un aumento delle emissioni. Così come è proporzionale l'aumento delle emissioni in relazione all'aumento dei consumi.

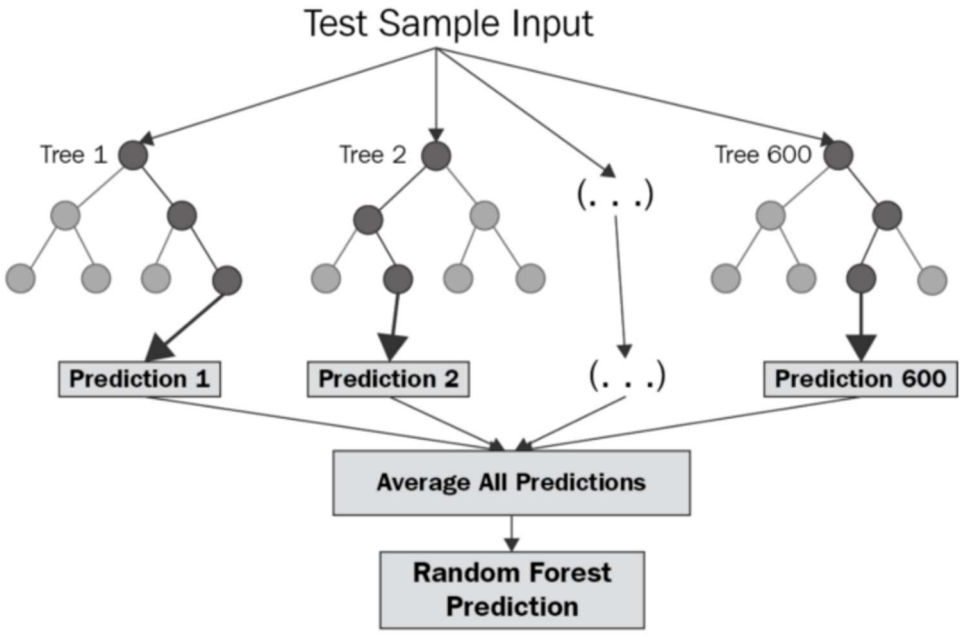


Modelli utilizzati:

-Regressione:

L'algoritmo di regressione scelto per prevedere le emissioni, è il ***RandomForestRegressor***, contenuto nella libreria ***sklearn*** di Python, dedicata al machine learning, pressoché fondamentale per operazioni come Classificazione, Regressione, Clustering, etc.

La scelta è ricaduta sull'algoritmo Random Forest poiché il dataset a disposizione è di grandi dimensioni e restituisce una media dei risultati della predizione ogni albero. Questo è un comportamento che migliora l'accuratezza e riduce l'overfitting rispetto ad un singolo albero di decisione.



Inoltre, si è dimostrato non particolarmente complicato da utilizzare, e nel complesso ha offerto risultati migliori rispetto ad altri algoritmi di regressione (es. decision tree o linearSVR).

Per migliorare il comportamento del *Random Forest Regressor* ed evitare l'overfitting, si sono precedentemente determinati i valori ottimali attraverso un tuning degli iperparametri, a tal fine è stata utilizzata la libreria *GridSearchCV*, e la libreria Yellowbrick, appartenente sempre a *sklearn*.

Questa libreria produce i possibili iperparametri in maniera automatica, valutando automaticamente svariate combinazioni possibili e scegliendo la migliore.

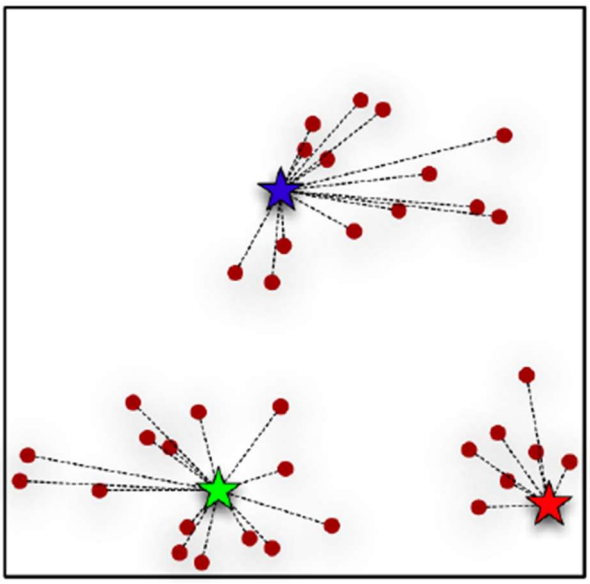
Sulla base dei risultati del GridSearchCV, il numero degli estimatori, che rappresenta il numero di alberi nella foresta è stato quindi impostato a 100 e la profondità massima, che rappresenta la profondità massima dell'albero a 7.

L'algoritmo Random Forest, di contro, presenta un problema che generalmente non è di poco conto: non ha la capacità di estrapolare dati al di fuori del training set, poiché mediando sulle predizioni, non potrà mai uscire al di fuori del range del valore più basso o più alto contenuto nel dataset.
Nel mio caso, ho ritenuto di utilizzare comunque l'algoritmo così com'è, senza far uso di modelli lineari al suo posto o una versione modificata del Random Forest (ad esempio come secondo step a seguito dell'esecuzione dell'algoritmo Lasso), poiché il dataset in oggetto contiene diverse istanze di automobili i cui valori minimi e massimi di emissioni, rappresentano già di per sé valori di emissioni limite per le automobili dotate di un'unità motrice a combustione interna attualmente in commercio.

-Clustering:

Per raggruppare automobili simili ho fatto uso di un modello di clustering chiamato K-Means, un algoritmo di apprendimento non supervisionato. Gli algoritmi di clustering provano a rilevare dei pattern all'interno dei dati. Nello specifico questo algoritmo ha l'obiettivo di partizionare il dataset in *k* cluster, ogni cluster raggruppa oggetti che condividono delle similarità, questi oggetti raggruppati un cluster, prendono il nome di **data points**.

Ogni cluster avrà un **centroide**, ovvero un punto che si trova al centro di un cluster.



Tre centroidi rappresentati dalle stelle

L’algoritmo fisserà dei centroidi iniziali e assegnerà i data points al centroide più vicino. Segue una fase di calcolo della distanza euclidea tra ogni data point e ogni centroide.

A questo punto un data point viene assegnato ad il centroide con il quale avrà la minore distanza.

Si ricalcola quindi la posizione media dei centroidi poiché è possibile che si siano formati nuovi cluster.

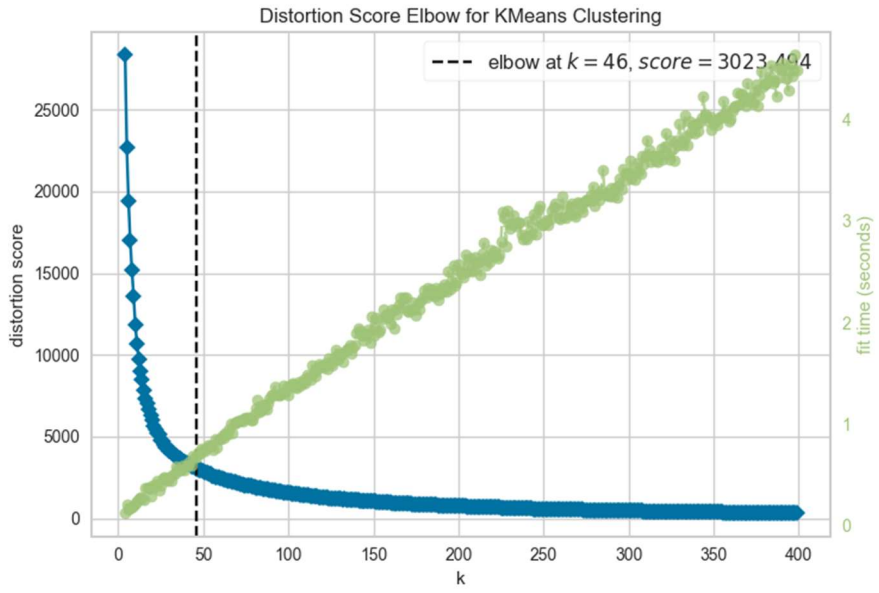
L’algoritmo proseguirà iterativamente finché i centroidi non subiscono più modifiche o viene raggiunto un numero massimo di iterazioni fissato.

L’algoritmo K-Means raggruppa i dati in un numero di cluster k specificato dall’utente, questo però non è assolutamente una garanzia che il numero k specificato dall’utente sia il numero ottimale di cluster nel dataset.

Per determinare il numero k ottimale, vi sono diversi metodi, ad esempio, l’Elbow Method o il Silhouette Method.

In questo progetto mi sono avvalso dell’**elbow method**, implementato tramite la libreria *Yellowbrick*, che permette di visualizzare il risultato del metodo elbow, specificando il numero k ottimale di cluster.

Nel nostro caso, considerando la grandezza del dataset e l’eterogeneità delle vetture in esso presenti, il numero di cluster ottenuto come risultato dell’elbow method, è di 46 cluster.

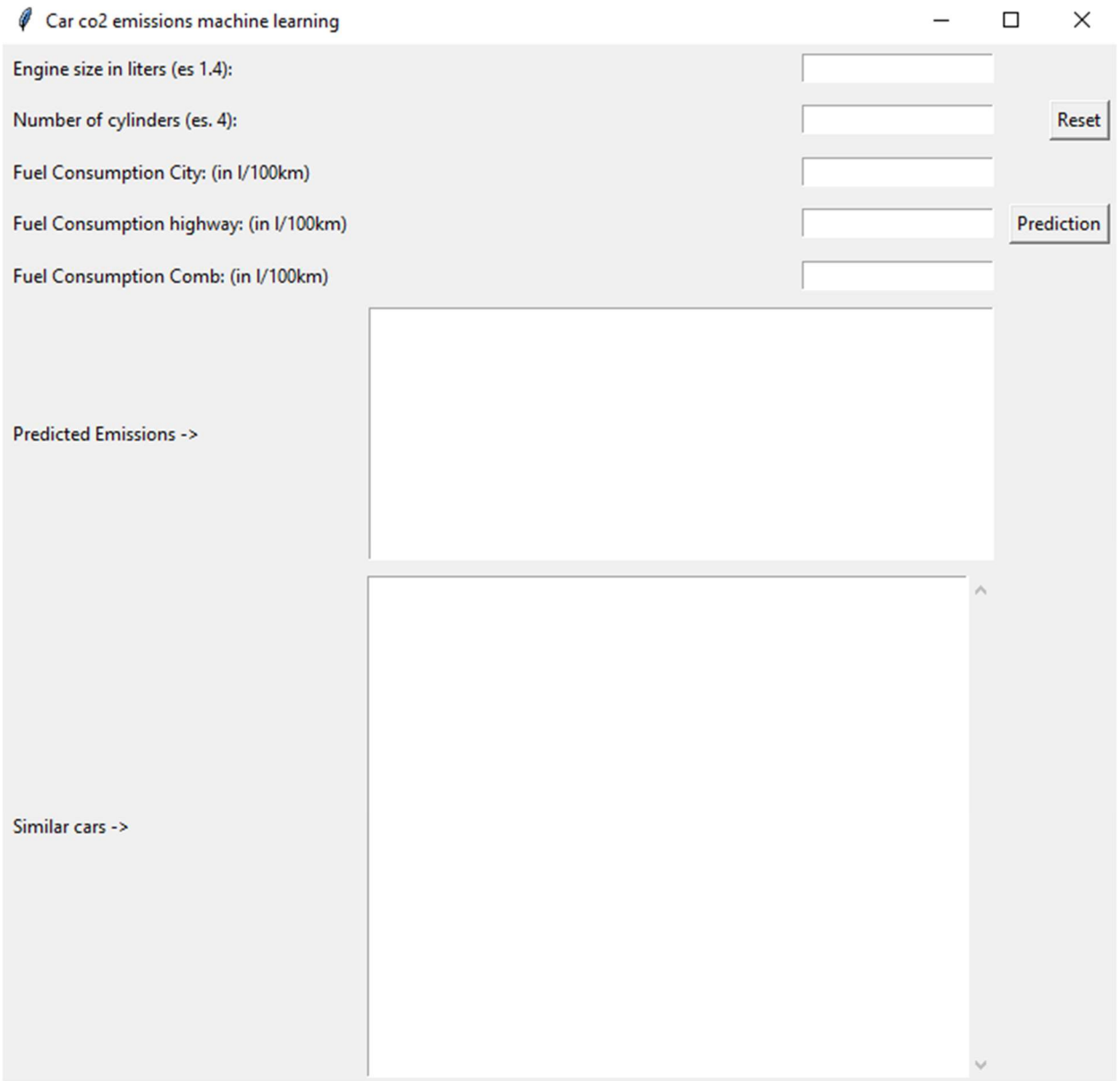


Interfaccia Grafica e funzionamento del programma:

L'interfaccia, progettata grazie alla libreria *tkinter* è schematica ed essenziale.

Presenta cinque campi di input per le caratteristiche dell'auto di cui si vogliono predire le emissioni e con le quali mostrare le automobili simili presenti nel dataset utilizzato.

Il pulsante «Reset» pulisce tutti i campi, mentre il pulsante «Prediction» dà il via al calcolo e mostra le emissioni predette e le auto simili.

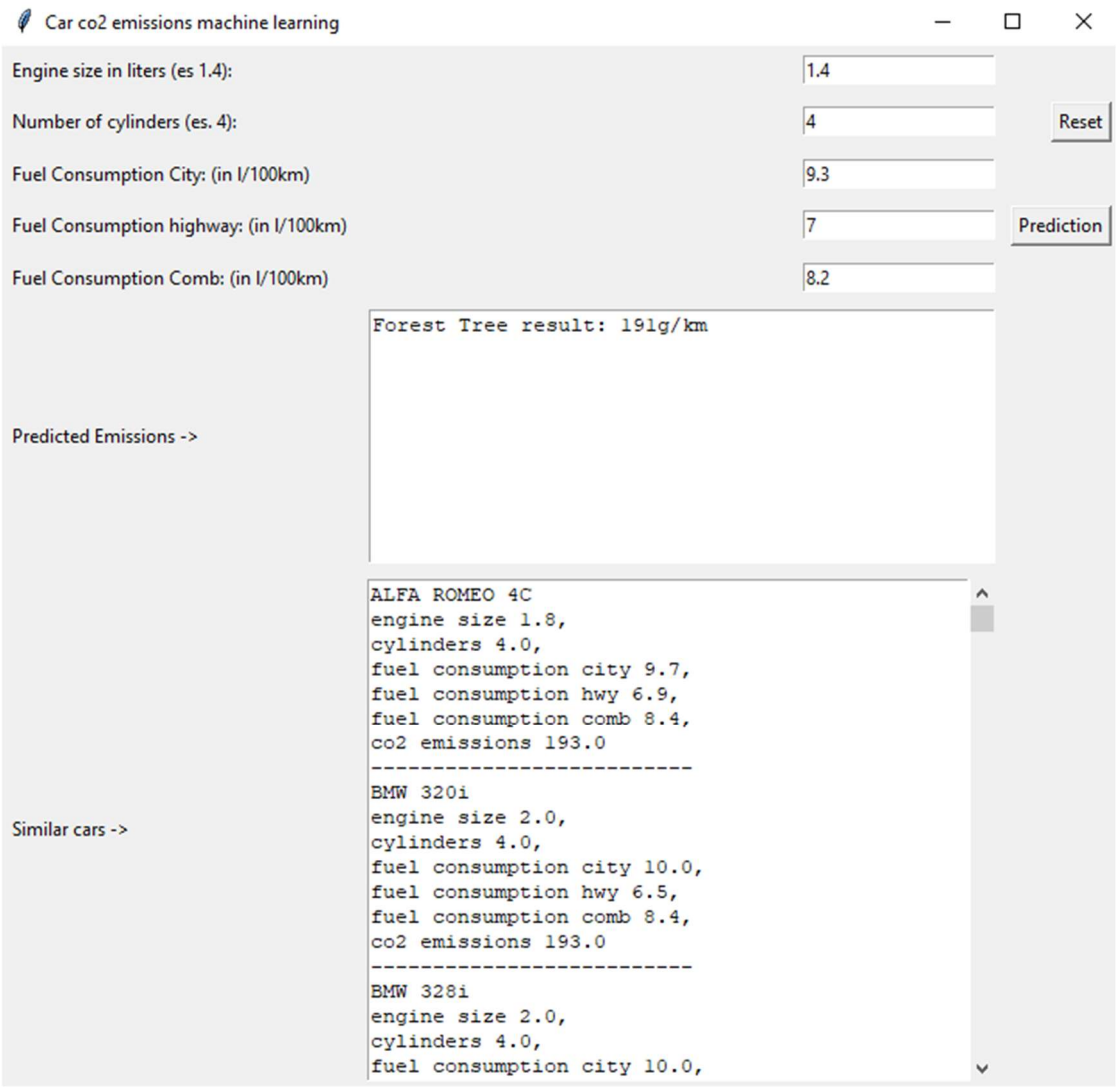


L’utente inserisce le cinque caratteristiche necessarie per il calcolo delle emissioni e delle automobili simili.

Il sistema attraverso un algoritmo di regressione, il *RandomForestRegressor*, calcola le emissioni previste per un’automobile con le caratteristiche in input.

Il sistema non accetta in input valori incompatibili con le caratteristiche tecniche di un’automobile (es: cilindrate enormi, consumi troppo ridotti, etc.)

Con gli stessi dati di input, il sistema attraverso un algoritmo di clustering, fornisce in output le vetture con caratteristiche simili presenti nel dataset.



Conclusioni e possibili sviluppi:

Nel complesso i risultati del sistema si sono rilevati soddisfacenti, ad esempio: anche presentando al sistema una serie di vetture non presenti nel dataset, si sono ottenuti risultati di predizione più che soddisfacenti.

Alcuni attributi del dataset potrebbero essere sfruttati per una clusterizzazione e per delle previsioni ancora più accurate e per offrire una lista di auto simili a quella in input (ad esempio, utilizzare l'attributo del tipo di carburante o la tipologia di cambio, manuale, automatico, etc.)

Si potrebbe ulteriormente migliorare il sistema, utilizzando un dataset con attributi ancora più specifici sulla tipologia di motore (es. aspirazione naturale, turbo, e varie tipologie di motori ibridi), nonché sui dati relativi alla potenza del motore stesso, da mettere in relazione con il dato di emissioni.