IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

Car Price Prediction Using Machine Learning

Ashish Chandak[1*], Prajwal Ganorkar[2], Shyam Sharma[3], Ayushi Bagmar[4], Soumya Tiwari [5]

[1,2,3,4,5]Information Technology, Shri Ramdeobaba College of Engineering,
Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur, India

[*]*Corresponding Author: chandakav@rknec.edu, Tel.: 8237851429*

DOI: https://doi.org/10.26438/ijcse/v7i5.444450 | Available online at: www.ijcseonline.org

Accepted: 20/May/2019, Published: 31/May/2019

***Abstract*—** Because of new computing technologies, machine learning today is not like machine learning of the past. It was born from pattern recognition and the theory that computers can learn without being programmed to perform specific tasks; researchers interested in artificial intelligence wanted to see if computers could learn from data. The iterative aspect of machine learning is important because as models are exposed to new data, they are able to independently adapt. They learn from previous computations to produce reliable, repeatable decisions and results. It's a science that's not new – but one that has gained fresh momentum. While there is an end number of applications of machine learning in real life one of the most prominent application is the prediction problems. There are various topics on which the prediction can be applied. One such application is what this project is focused upon. Websites recommending items you might like based on previous purchases are using machine learning to analyze your buying history – and promote other items you'd be interested in. This ability to capture data, analyze it and use it to personalize a shopping experience (or implement a marketing campaign) is the future of retail

***Keywords*—**Environment Quality, Data Analysis, Business Intelligence, Power BI, SQL Server 2016, Air Quality, Water Quality, Tree Cover, Forest Cover, Predictions, NLP, forecasting, k-means clustering, ARIMA.

I. INTRODUCTION

From a long time since being, a continuous paradigm of transactions of commodities has been into existence. Earlier these transactions were in the form of barter system which later was translated into a monetary system. And with consideration into these, all changes that were brought about the pattern of re-selling items was affected as well. There are two ways in which the re-selling of the item is carried out. One is offline and the other being online. In offline transactions, there is a mediator present in between who is very vulnerable to being corrupt and make overly profitable transactions. The second option is online wherein there is a certain platform which lets the user find the price he might get if he goes for selling

- Kilometers traveled – We know that the number of kilometers traveled by a vehicle has a huge role to play while putting the vehicle up for sale. The more the vehicle has traveled, the older it is.

- Fiscal power – It is the power output of the vehicle. More output yields better value out of a vehicle.

- Year of registration – It is the year when the vehicle was registered with the Road Transport Authority. The newer the vehicle is; the better value it will yield. By every passing year, the value will depreciate.

- Fuel Type – There were two types of fuel types present in the dataset that we had. Petrol and Diesel. It was relatively less dominant.

It's due to the above factors that we need a system that can develop a self-learning machine learning-based system. This was the basis on which a set of objectives was supposed to be formulated. One thing that was pre-determined was that this is going to be a real-time project.

OBJECTIVE

- To build a supervised machine learning model for forecasting value of a vehicle based on multiple attributes

- The system that is being built must be feature based i.e. feature wise prediction must be possible.

- Providing graphical comparisons to provide a better view.

MOTIVATION

The automotive industry is composed of a few top global multinational players and several retailers. The multinational players are mainly manufacturers by trade whereas the retail market features players who deal in both new and used vehicles. The used car market has demonstrated a significant growth in value contributing to the larger share of the overall market. The used car market in India accounts for nearly 3.4 million vehicles per year.

FEATURES

There will be majorly two features provided in the project note that this will be not

- Re-sale platform: A centralized platform for car re-sale that will predict prices.
- Feature selection: Feature-based search and prediction.

Section I contains the introduction of our module, then objective, motivation and features of our model, Section II contains Literature Review, Section III contain the various technologies in machine learning, Section IV explains the methodology, section V describes the results and discussion, Section VI contains the conclusion and future work.

II. LITERATURE REVIEW

In this chapter, we discuss various applications and methods which inspired us to build our project. We did a background survey regarding the basic ideas of our project and used those ideas for the collection of information like the technological stack, algorithms, and shortcomings of our project which led us to build a better project.

CARS24

Cars24 is a web platform where seller can sell their used car. It is an Indian Start-up with a simplified user interface which asks seller parameters like car model, kilometers traveled, year of registration and vehicle type (petrol, diesel)[1]. These allow the web model to run certain algorithms on given parameters and predict the price.

GET VEHICLE PRICE

Get Vehicle Price is an android app which works on similar parameters as of Cars24. This app predicts vehicle prices on various parameter like Fiscal power, horsepower, kilometers traveled. This app uses a machine learning approach to predict the price of a car, bike, electric vehicle and hybrid vehicle. This app can predict the price of any vehicle because of the smartly optimized algorithm.

CARWALE

CarWale app is one of the top-rated car apps in India for new and used car research. It provides accurate on-road prices of cars, genuine user and expert reviews. It can also compare different cars with the car comparison tool. this app also helps you to connect with your nearest car dealers for the best offers available.

CARTRADE

CarTrade is web and Android platform where user can research New Cars in India by exploring Car Prices, Car Specs, Images, Mileage, Reviews, and Car Comparisons. On this app one can Sell Used Car to genuine buyers with ease. One can list their used car for sale along with the details like image, model, and year of purchase and kilometers so that it is displayed to lakhs of interested car buyers in their city. User can read user reviews and expert car reviews with images that help in finalizing a new car buying decision

III. TECHNOLOGY USED

Python was the major technology used for the implementation of machine learning concepts the reason being that there are numerous inbuilt methods in the form of packaged libraries present in python. Following are prominent libraries/tools we used in our project.

NUMPY

NumPy is a general-purpose array-processing package[1]. it provides a high-performance multidimensional array object and tools for working with these arrays. It is the fundamental package for scientific computing with Python. Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

SCIPY

SciPy is a free and open-source Python library used for scientific computing and technical computing. SciPy contains modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal and image processing, ODE solvers and other tasks common in science and engineering.

SciPy builds on the NumPy array object and is part of the NumPy stack which includes tools like Matplotlib, pandas,

and SymPy, and an expanding set of scientific computing libraries. This NumPy stack has similar users to other applications such as MATLAB, GNU Octave, and Scilab. The NumPy stack is also sometimes referred to as the SciPy stack[2]. The SciPy library is currently distributed under the BSD license, and its development is sponsored and supported by an open community of developers. It is also supported by NumFOCUS, a community foundation for supporting reproducible and accessible science.

SCIKIT-LEARN

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. The library is built

JUPYTER NOTEBOOK

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. It includes data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text[3]. It includes data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

ENTHOUGHT CANAOPY

Enthought Canopy is a Python for scientific and analytic computing distribution and analysis environment, this package manager uses jypter notebook as a presentation layer. Anaconda tries to solve the dependency hell in python where different projects have different dependency versions, so as to not make different project dependencies require different versions, which may interfere with each other.

IV. METHODOLOGY

In this chapter, we discuss various algorithms and the required dataset that were implemented to build this module. A dataset containing more than 3 lakh tuples will be used for training the model. Attributes such as kilometers traveled, year of registration, fuel type and fiscal power determine the worth of an automobile. Since this is a classification problem, we have implemented two algorithms – K Nearest Neighbour (KNN) and Classification and Regression Trees (CART) and compared the two on different models of vehicles.

To implement these algorithms we use Enthought Canopy. Enthought Canopy is a Python for scientific and analytic computing distribution and analysis environment, this package manager uses the juypter notebook as a presentation layer [4]. Anaconda tries to solve the dependency hell in python where different projects have different dependency versions, so as to not make different project dependencies require different versions, which may interfere with each other.

K-MEANS ALGORITHM

K-means clustering is one of the simplest and popular unsupervised machine learning algorithms. Typically, unsupervised algorithms make inferences from datasets using only input vectors without referring to known or labelled outcomes. A cluster refers to a collection of data points aggregated together because of certain similarities. You'll define a target number k [5], which refers to the number of centroids you need in the dataset. A centroid is the imaginary or reallocation representing the center of the cluster. Every data point is allocated to each of the clusters by reducing the in-cluster sum of squares. In other words, the K-means algorithm identifies k number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible. The ‘means' in the K-means refers to averaging of the data; that is, finding the centroid. To process the learning data, the K-means algorithm in data mining starts with a first group of randomly selected centroids, which are used as the beginning points for every cluster, and then performs iterative (repetitive) calculations to optimize the positions of the centroids It halts creating and optimizing clusters when either: The centroids have stabilized—there is no change in their values because the clustering has been successful. The defined number of iterations has been achieved.

DECISION TREE REGRESSION

Decision tree learning uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). It is one of the predictive modeling approaches used in statistics, data mining and machine learning. Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. The goal is to create a model that predicts the value of a target variable based on several input variables decision tree learning is a method commonly used in data mining[6]. The goal is to create a model that predicts the

value of a target variable based on several input variables. An example is shown in the diagram at right. Each interior node corresponds to one of the input variables; there are edges to children for each of the possible values of that input variable. Each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf. A decision tree is a simple representation for classifying examples [7] [8].

V. IMPLEMENTATION

In this chapter, we discuss the steps and implemented methods used in our module, it includes the statistical analysis of our dataset through various scattered graphs, violin graph, comparison charts, and bar graph to study the best algorithm.

We first perform pre-processing and data cleaning on our dataset. We found that 15% of the tuples had null values and we pruned those tuples. We built a heat map comparing kilometers traveled, year of registration, price and fiscal power.

The dataset was split into 80% for training and 20% for testing. Using the Scikit learn library in python, we build the KNN (k = 7) and CART models for predicting the value of a vehicle. The value for the desired k was not directly decided rather we try to run the prediction model while assuming different values for k and compare them amongst themselves. The year of registration was slightly more dominant.

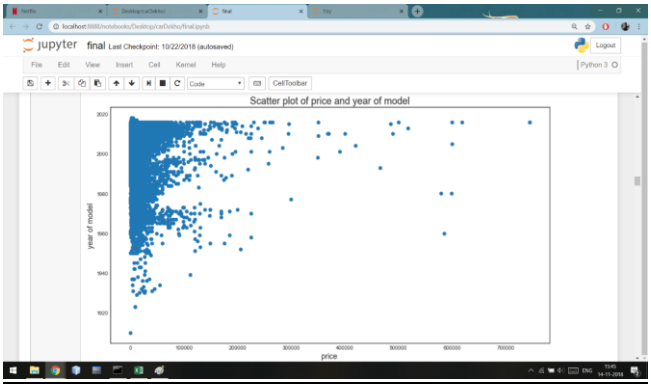

Figure 4.1: Scatter Plot of price and year of model

Figure 4.1 represents a scatter plot of price and year model, it is observed that the price between 0 to 20000 contains the most used car between the years 1990 to 2010.

-

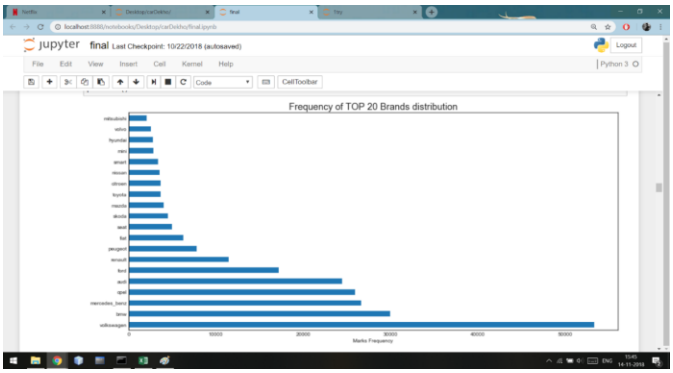

Figure 4.2: Frequency of TOP 20 Brands Distribution

Figure 4.2 represents the frequency of top 20 brand distribution in our dataset, it is observed that the Volkswagen brand contains the most cars in the database followed by BMW, Mercedes Benz, and Audi.

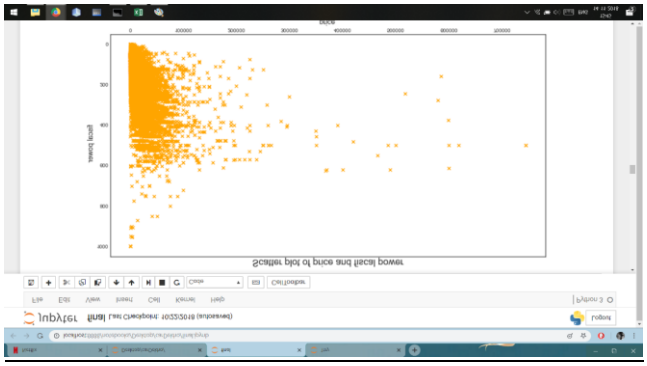

Figure 4.3: Scatter Plot of price and fiscal power

Figure 4.3 represents the Scatter Plot of Price and Fiscal power, it is observed that the car price ranging between 0 to 20000 gives a fiscal power of 100 to 600.

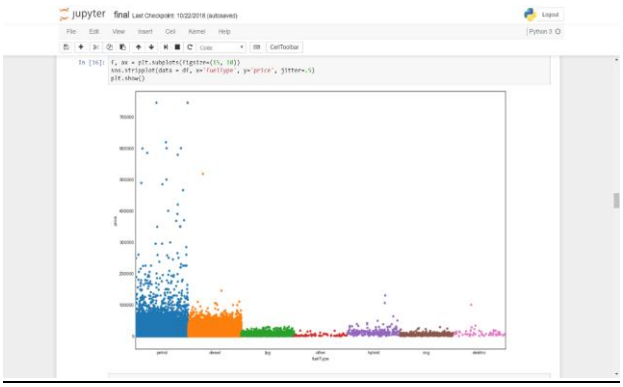

Figure 4.4: Strip Plot of price versus fuel type

Figure 4.4 represents the strip plot of price versus fuel type, it is observed that our dataset contains mostly petrol and

diesel cars ranging price till 20000 followed by LPG and hybrid cars.

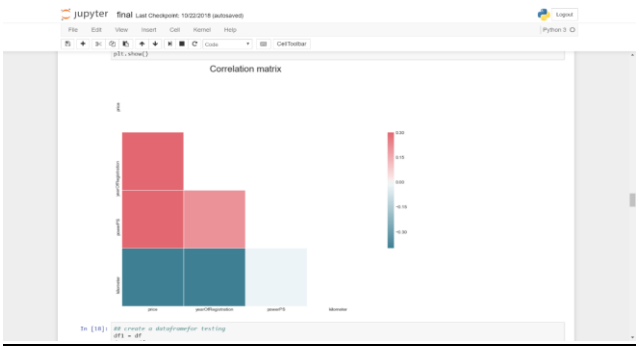

Figure 4.5: Correlation Matrix

Figure 4.5 represents a correlation matrix between different attributes of the dataset. Positive relationships exist between the year of registration and price and an inverse relation between price and kilometer traveled.

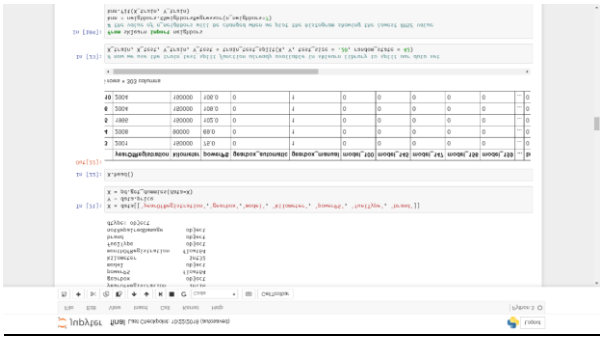

Figure 4.6: Snapshot of the dataset header

Figure 4.6 represents a snapshot of dataset header which has attributes after pre-processing which means data cleaning has been performed and it does not contain any null value in the data set.

Figure 4.7: Accuracy mapping for different k-values

Figure 4.7 represents accuracy mapping for different k-values against root mean square. For different neighbors k value, it is observed that they do not deviate a lot from each other.

Figure 4.8: Code Snippet

Figure 4.8 represents a code of KNN algorithm, here we set the value of k = 7 which results in the selection of seven neighbors for learning the dataset and predict the value.

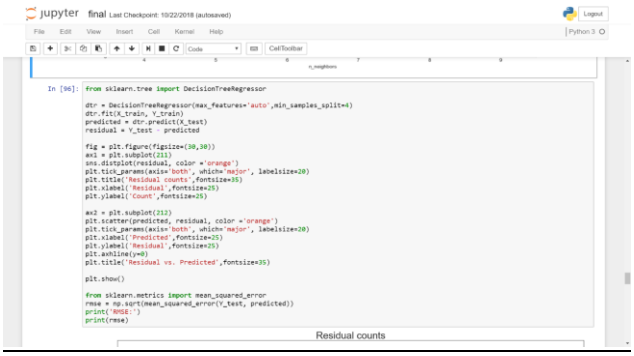

Figure 4.9: Code Snippet 2

Figure 4.9 represents code for Decision tree regression algorithm, here we use Scikit-learn library to implement these algorithms. This code plots a graph for a predicted value and residual value. It learns the dataset and then applies it to predict values.

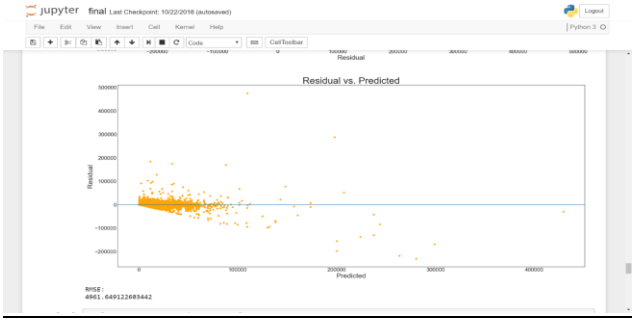

Figure 4.10: Residual vs Predicted comparison graph

Figure 4.10 represents a comparison graph between Residual value vs predicted value, the dotted points from 0^{th} line represent the deviation from actual against predicted for decision regression algorithm.

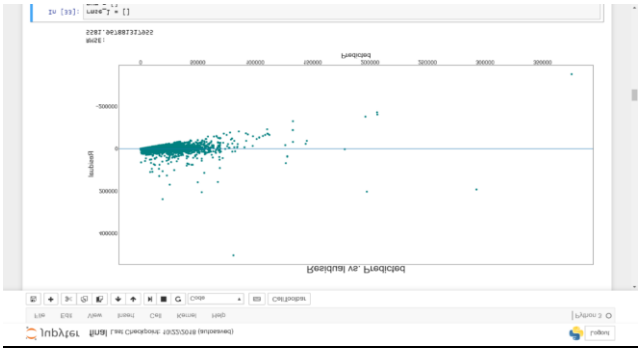

Figure 4.11: Residual vs Predicted graph 2

Figure 4.11 represents a comparison graph between Residual value vs predicted value, the dotted points from 0th line represent the deviation from actual against predicted for KNN algorithm.

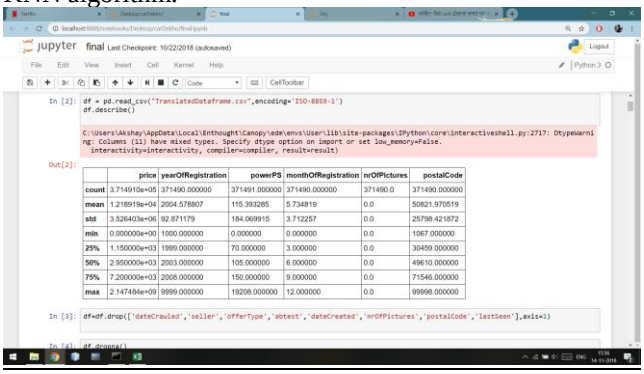

Figure 4.12: Different statistical aggregations in dataset

Figure 4.12 represents a statistical aggregation in the dataset, describing the mean, count, standard deviation, and the min percentage of the data

VI. CONCLUSION

In this chapter, we discuss the results and observation we did while implementing this module. We successfully implemented the machine learning algorithmic paradigms using prominent algorithms from libraries in python. We first perform pre-processing and data cleaning on our dataset. We found that 15% of the tuples had null values and we pruned those tuples. The results showed that there is a positive correlation between price and kilometers traveled, year of registration and kilometers traveled and a negative correlation between price and year of registration.

Positive correlation basically relates to the concept of direct proportion whereas Negative correlation relates to the concept of inverse proportion. Three lakh tuples were used for training the model. The year of registration was slightly more dominant. K Nearest Neighbour (KNN) and Classification and Regression Trees (CART) are compared on two different models of vehicles.

We found that the root means square error for KNN with k = 7 is 5581.96 and for CART is 4961.64 and actual price was 4999.

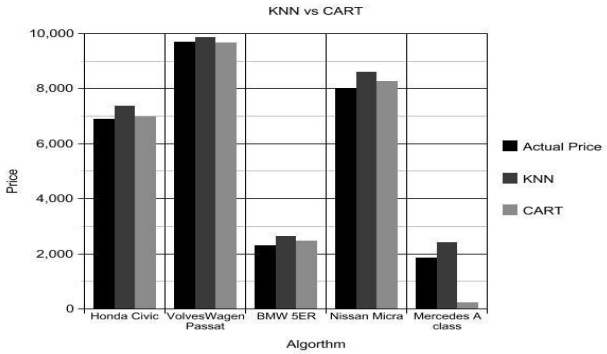

Figure 5.1 KNN vs CART vs Actual Price

FUTURE SCOPE

As a part of future work, we aim at the variable choices over the algorithms that were used in the project. We could only explore two algorithms whereas many other algorithms which exist and might be more accurate.

More specifications will be added in a system or providing more accuracy in terms of price in the system i.e.

1) Horsepower
2) Battery power
3) Suspension
4) Cylinder
5) Torque

As we know technologies are improving day by day and there is also advancement in car technology also, so our next upgrade will include hybrid cars, electric cars, and Driverless cars

REFERENCES

[1].M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou, "Understanding the mirai botnet," in Proc. of USENIX Security Symposium, 2017.

[2].Iman Sharafaldin, Arash Habibi Lashkari, and Ali A. Ghorbani, ―Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization‖, 4th International Conference on Information Systems Security and Privacy (ICISSP), Portugal, January 2018

[3].Hossein Hadian Jazi, Hugo Gonzalez, Natalia Stakhanova, and Ali A. Ghorbani. "Detecting HTTP-based Application Layer DoS attacks on Web Servers in the presence of sampling." Computer Networks, 2017

[4]. A. Shiravi, H. Shiravi, M. Tavallaee, A.A. Ghorbani, Toward developing a systematic approach to generate benchmark datasets for intrusion detection, Comput. Security 31 (3) (2012) 357–374.

[5].Z. He, T. Zhang, and R. B. Lee, ―Machine Learning Based DDoS Attack Detection from Source Side in Cloud,‖ in Proceedings of the 2017 IEEE 4th International Conference on Cyber Security and

Cloud Computing (CSCloud), pp. 114–120, New York, NY, USA, June 2017

[6].R. Doshi, N. Apthorpe and N. Feamster, "Machine Learning DDoS Detection for Consumer Internet of Things Devices," 2018 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, 2018, pp. 29-35.

[7].Jerome H. Friedman, (2002), Stochastic gradient boosting, Computational Statistics & Data Analysis, 38, (4), 367-378

[8].Friedman, Jerome H. Greedy function approximation: A gradient boosting machine. Ann. Statist. 29 (2001), no. 5, 1189--1232.

