Journal of Advanced Zoology

ISSN: 0253-7214
*Volume **44** Issue **S-6 Year 2023** Page **546:550***

Web Based Data Visualization and Data Preprocessing Tool

Sahil Chachra[1*], Resham Sundar Kumar[2], Santosh D Kolur[3], Rohit K[4], Priyanka Bharti[5]

[1,2,3,4,5]*School of Computer Science & Engineering, REVA University*
resham.sundar@gmail.com[2]
santoshdkolur5@gmail.com[3]

**Corresponding author's E-mail: sahil.chachra3@gmail.com*

Article History

Received: 06 June 2023
Revised: 05 Sept 2023
Accepted: 29 Nov 2023

Abstract

The escalating adoption of Machine Learning techniques has given a bigger picture to newbies trying to explore the usage of it. Our tool deals with the idea of helping them, in order to make their lives easier. Various visualizations and algorithms have been developed to help Machine Learning enthusiasts decide upon the best model. Deciding the perfect model needs enough time which now can be reduced by the solution provided in this tool. This interactive technique helps users with some expertise to explore and validate predictive as well as classification models. Once the user provides the dataset, the visualization techniques discussed in this tool lets the user decide to select the features that are most suitable for training the model. It allows one to decide upon the importance of a particular feature and know the dataset predictions across various algorithms used for regression and classification. The accuracy percentage or the precision, recall score for regression and classification models respectively can be seen in order to know the best model.

Keywords: *Data visualization, Data preprocessing, Model selecting, Machine Learning, Model Evaluation.*

1. Introduction

Machine Learning is a concept which allows the machine to learn from the given examples and its experience without being explicitly programmed. In this, instead of writing the code, we just feed the data to the generic model, and the model builds the logic based on the given data. These programs or algorithms are designed in such a way that they learn and improve over time when exposed to new data. The objective of our tool is to make the process generic enough to enable even a person with scarce knowledge about computing, be able to utilise the immense potential that machine learning algorithms have to offer depending upon the data given. It also gives an idea to the individual about which machine learning algorithm best suits his data.

Our project lets the individual choose which feature is more efficient for them to build the model. The data could also be visualised to get an insight and attain additional information about itself which is usually hard for a person to grasp just by looking at those numbers. We also intend to give them the preprocessed data at the end which is custom made for their problem type, because what is the point of training a model online with all the processing if the person cannot use it for custom inputs later right? This code is not hard-coded and we have tried to keep the accuracy produced by the model to be close to the ones we could obtain from hard coded scripts for the model.

Literature Survey

In [1] (James Wexler, Mahima Pushkarna), WIT, is a tool which is open sourced. WIT(What-If-Tool), a part of TensorBoard and is also available as an extension of Jupyter notebook. It enables visualization and analysis of machine learning models with minimal code. The work aims at providing hypotheses of the model with less code, and provides visualization of the model to better understand it. It allows the user to load and explore the data points to understand the data he/she is working with. The work also aims at evaluating performance of multiple models by providing a wide range of performance metrics choice for different models. The main goal is to build a tool to visualize data and model and train the model with less coding.

In [2] (Yao Ming, Huamin Qu), the authors propose a visualization tool to help those who have good domain knowledge but little knowledge on machine learning. The tool aims at enabling such people to understand and explore the machine learning models. The work is done by creating a pipeline which is rule driven and the visual explanation consists of three steps - Rule Induction, Filtering, and Visualization. They have also proposed two algorithms, one for Rule Induction and other for Estimate Distribution. Current version of the work can visualise 100 rules with around 30 characteristics but it has been tested only with 60 rules and 20 characteristics.

In [3], (İbrahim PERÇİN, Fatma Hilal YAĞIN), the authors have proposed a web based tool to perform classification task. The whole tool is written in Java and performs data preprocessing and training of a classifier. The whole tool was tested on a cervical cancer dataset.The tool was an attempt to make features of WEKA available online so that people who want instant results or who want to perform small tasks can perform it efficiently online. Tasks such as Attribute type conversion, Normalisation, training models such as Naive Bayes and Random Forest and cross validation can be performed.

In [4], (Devashree Vaishnav, B. Rama Rao), the authors have proposed a single software which provides pre-processing of data, visualization of the data and training of the model all in one. Cross validation has been used to evaluate performance of the model. To determine the model's performance, accuracy and precision score is being used. Apart from core algorithms such as Random Forest Classifier, Decision tree, Neural Networks have been also used. The tool performs two sets of evaluations, both using different hyperparameters. Generally the second evaluation has shown better results for almost all algorithms. For each algorithm, the tool shows the confusion matrix which enables the user to choose the best algorithm.

Objectives

Various types of graphs will be available - univariate and bivariate graphs

Users can preprocess the dataset using the proposed system and export it.

Users can choose/drop any feature from the dataset and train models and evaluate it.

Users can train the model for reference and save it if required.

2. Materials And Methods

Our tool mainly focuses on Data visualization and Data preprocessing. By providing a no-code option like interface, we focus on saving time spent by the user to type long same code to simply perform Exploratory Data Analysis. The user can also plot various graphs such as Univariate plots and Bivariate plots. After the user has visualized the dataset then he/she can move on to Feature Engineering. In Feature Engineering, our tools allow the user to first handle missing values, be it numerical or categorical type. After missing values are handled then the user can opt to see the correlation graph to determine the correlation between target variable and independent features. According to this graph, the user can drop irrelevant independent features. Then finally the user can select the target variable and can select a task such as regression or classification. On selecting the task, the user can choose from a set of algorithms to train the model to just get a rough idea about the performance of the model. All this task requires users just to select from drop down menus or check boxes. This interface is beginner friendly and can help people save time from writing long codes to perform basic feature engineering. Also the tool allows the user to download the pre-processed dataset so that the user can train models on his/her system directly instead of replicating the preprocessing steps.

The below block diagram shows the system design for the project. Some of the important features from the diagram are explained right after the image.

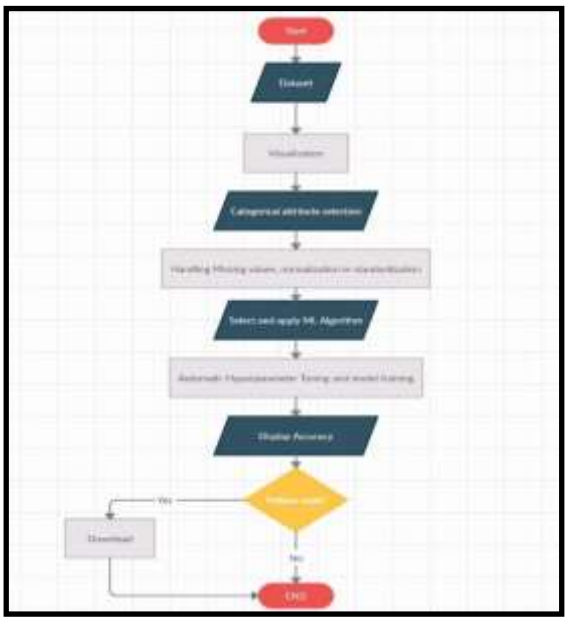

Figure 1.1

Univariate Plots: User gets the option to plot graphs such as Box plot and Count plot. With help of Box plot the user will get know if the column has outliers and with Count plot, the user can visualize the values of categorical type data. The user also gets an option to plot graphs to visualize Skewness of each column.

Bivariate Plots: This type of plot helps in understanding pairwise relationships. Users get the option to plot Barplot or Scatter plot and then the user can choose columns for X and Y axes.

Filling Missing Values: Users will be given the option to fill in missing values. Filling up missing values can help in proper interpretation of the plots and better means to feature selection. One way of doing away with missing values is by directly dropping the rows and the columns.Ways to fill in missing values for integer or float type columns are by filling all of them with a constant value provided by the user. Other ways are by replacing them with the value in the previous or next rows. Imputing values may also include methods like mean and median of the values of that particular feature. Filling in values when it comes to categorical features is by using a constant value provided by the user or by using the mode of the values of that particular feature.

Selecting Features: Users can visualize the heat map which represents the correlation between the target variable and the independent features. According to the correlation the user can drop the features with least correlation and then again can visualize the correlation. This iterative process can go on till the user is satisfied with the correlation map. Then the user can move ahead with model training.

Training model: The data after preprocessing can be used to train a desired model. To proceed to the training part the user will have to make some selections which will lead to the selection of the right model for the uploaded data. The portal will have an option to select between classification and regression. The supported algorithms will be available post the selection of a category. The selected algorithm will have a model trained with the given data with few of the hyperparameters tuned automatically. Once the training is done the accuracy of the trained model is displayed on the screen and the user can download the trained model as a pkl file along with a readme.txt file which explains how to use the downloaded model.

Apart from this the user will also be able to download the python code for the respective algorithm that he/she has chosen. The downloaded code will only contain the basic preprocessing script. This code should not be mistaken for the code used on the website as the one on the website offers a lot more functionality and can handle raw data. The downloaded code should just be used for reference or to train a basic model [7-12].

3. Results and Discussion

Figure 2.1, shows the platform where we upload the dataset. The platform supports csv files. The uploaded data is stored on the server and further operations are performed.

Figure 2.1 platform supports csv files

The uploaded data can be visualised through various plots that are available refer figure 2.2. Here we show a boxplot for the feature oldpeak. This could be used to better understand the distribution and also identify some outliers.

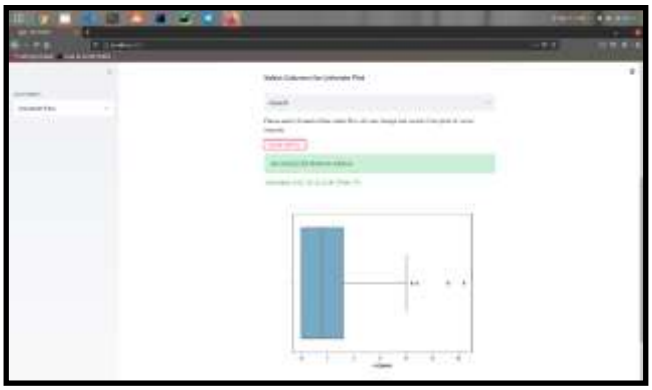

Figure 2.2 platform supports csv files (Raw data)

The raw data can then be processed using the various options that the platform provides. Missing values, categorical values and unnecessary features are some of the items that can be handled. Figure 2.3 (used some other dataset as this dataset has no missing values) shows in case of missing values, the user will get options like this to fill the missing values. Also, if categorical features are found, it is automatically encoded to numeric type using Label encoder.

Figure 2.3 preprocessed data

The user can plot correlation heatmap to see which feature is contributing the least and then that feature can be dropped from the dataset before using the data for training. The preprocessed data can then be used to train the model of choice depending on the target variable. There are several algorithms that are available and the desired one can be selected to train the model and check the accuracy that can be obtained. The accuracy is displayed after performing some of the hyperparameter tuning. The trained model can be downloaded and used offline as required. Refer figure 2.4.

Figure 2.4 hyperparameter tuning.

4. Conclusion

The output of our project could shed light on the appropriate algorithm and the features that could be used by any individual with less expertise while designing their model. By using this, it becomes very easy for the individual to decide which is the efficient algorithm that they can use to build their own model. The visualisation in the tool will ease up the process of feature selection, hence lead to better understanding and faster development of models.

Acknowledgment

We would like to thank our college, REVA University for providing us this opportunity to present our idea as a project. We would also like to thank the Director of our department, Dr. Sunilkumar Manvi for inspiring us to come up with innovative ideas and implementing them. We would also like to express our gratitude to our guide, Prof. Priyanka Bharti who has helped us in every way possible.

References:

[1] James Wexler, Mahima Pushkarna, Tolga Bolukbasi, Martin Wattenberg, Fernanda Viegas, and Jimbo Wilson , “The What-If Tool: Interactive Probing of Machine Learning Models”, IEEE Transactions on Visualization & Computer Graphics, Jan. 2020, pp. 56-65, vol. 26

[2] Yao Ming, Huamin Qu, Enrico Bertini, “RuleMatrix: Visualizing and Understanding Classifiers with Rules”, IEEE Transactions on Visualization and Computer Graphics (Volume: 25 , Issue: 1 , Jan. 2019)

[3] İbrahim PERÇİN, Fatma Hilal YAĞIN, A. Kadir ARSLAN and Cemil ÇOLAK, “An Interactive Web Tool for Classification Problems Based on Machine Learning Algorithms Using Java Programming Language: Data Classification Software”, 2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), published by IEEE

[4] Devashree Vaishnav, B. Rama Rao, “Comparison of Machine Learning Algorithms and Fruit Classification using Orange Data Mining Tool ”, International Conference on Inventive Computation Technologies (ICICT-2018) IEEE Xplore Part Number: CFP18F70-ART; ISBN:978-1-5386-4985-5

[5] Dr. O. Obulesu, M. Mahendra, M. ThrilokRedd, “Machine Learning Techniques and Tools: A Survey”, International Conference on Inventive Research in Computing Applications (ICIRCA 2018) IEEE Xplore Compliant Part Number:CFP18N67-ART; ISBN:978-1-5386-2456-2

[6] Josua Krause, Adam Perer, Kenney Ng.”Interacting with Predictions: Visual Inspection of Black-box Machine Learning Models“, CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, May 2016 Pages 5686–5697

[7] Sudhan Murugan Bhagavathi, Anitha Thavasimuthu, Aruna Murugesan, Charlyn Pushpa Latha George Rajendran, A Vijay, Raja Laxmi, Rajendran Thavasimuthu, Weather forecasting and prediction using hybrid C5.0 machine learning algorithm International Journal of Communication Systems, Vol. 34, Issue. 10, Pp. e4805, 2021.

[8] PM Surendra, S Manimurugan, A New Modified Recurrent Extreme Learning with PSO Machine Based on Feature Fusion with CNN Deep Features for Breast Cancer Detection, Journal of Computational Science and Intelligent Technologies, Vol. 1, Issue. 3, Pp. 15-21, 2020.

[9] PK Sadineni, Comparative Study on Query Processing and Indexing Techniques in Big Data, 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), pp. 933-939, 2020.

[10] AH Omar Baabood, Prajoona Valsalan, Tariq Ahmed Barham Baomar, IoT Based Health Monitoring System, Journal of Critical Reviews , Vol. 7, Issue. 4, pp. 739-743, 2020.

[11] Sajay KR, Suvanam Sasidhar Babu, Vijayalakshmi Yellepeddi, Enhancing The Security Of Cloud Data Using Hybrid Encryption Algorithm, Journal of Ambient Intelligence and Humanized Computing, 2019. https://doi.org/10.1007/s12652-019-01403-1

[12] Bindhia K Francis, Suvanam Sasidhar Babu, Predicting academic performance of students using a hybrid data mining approach, Journal of Medical Systems, 43:162, 2019. https://doi.org/10.1007/s10916-019-1295-4