ML Init: A Comprehensive ML Framework for Data Wrangling and Exploratory Analysis

Ajith Kumar K , Jamsheed M P, Shilpa Singh, Sourabh Gothe Vasant
Course Project for DA 203-O ICAIML
M. Tech, DSBA
IISc, Bangalore
{*ajithkumark, jamsheedm, shilpasingh, sourabhgothe*}*@iisc.ac.in*

***Abstract*—Data preparation is the vital step of the machine learning process cycle. Data wrangling and exploratory data analysis, which consist of loading the dataset, identifying the features and their distributions, handling missing data points, analyzing skewness, and the possibility of dimensionality reduction, is the most repetitive and time-consuming task. Even though there are some graphical tools for ML data analysis, they lack customized output based on the given data. We propose ML Init, a state-of-the-art tool for data wrangling and exploratory research on the dataset. It loads the dataset, identifies the features, plots various graphs, fills missing columns, analyses attribute distribution and skewness, and performs basic ML modeling.**

***Index Terms*—Data Wrangling, Exploratory Data Analysis, Imputation, Feature Distribution, Skewness, Kurtosis, Statistical Analysis**

I. Introduction

Exploring data through various Visualizations, statistical analysis of data, data cleansing, and data preparation is a significant step of the Machine Learning (ML) workflow. The user needs to have excellent analytical, statistical, and visualization skills to effectively perform the Exploratory Data Analysis (EDA) task. As depicted in Figure 1, Data Wrangling and Data Preparation are a part of the ML process cycle, which must be performed repeatedly.

Many tools are available to perform various plots for the features in a dataset. Orange [1] is an elegant open-source tool with many GUI-based operations that include EDA plots and basic modeling. However, it does not provide any analysis of the dataset's features. The user has to make the conclusions from the plots. Rattle [2] is another popular GUI tool for Data Mining. It restricts the operation to R language, and hence, python users cannot use it easily. [3] and [4] are also designed for data analysis. But, they fail to draw clear inferences from the dataset that the user can use readily.

We propose ML Init, a python notebook-based comprehensive data analysis framework, to mitigate the earlier challenges. The remaining of the report is arranged as follows. Section 2 gives an overview of the methodology. Section 3 depicts the various modules. Section 4 summarizes the tool's plots and analysis, and Section 5 concludes the report.

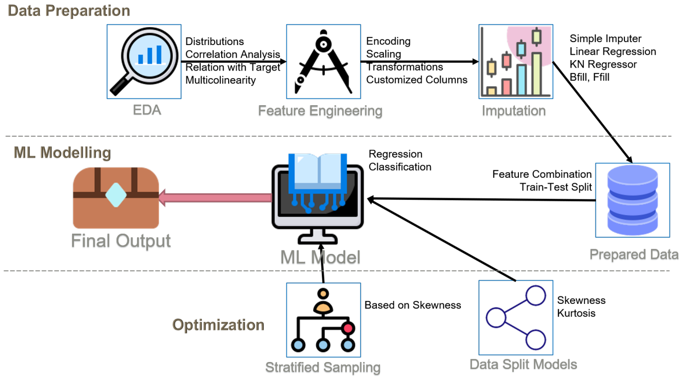


Fig. 1. Data Wrangling and Exploratory Analysis constitutes the significant part of End-to-end Machine Learning process.

II. Methodology

A. Goals and Motivation

ML Init creates an easy to use one of a kind GUI to provide great user experience. It consists of Data cleaning, Data preparation, Statistical and correlation analysis, Visualization, Stratification and a flavour of ML modeling. Following are the goals and motivations of the proposed solution.

- Create an easy to use GUI Interface to input the data, label and for customizing data visualizations
- Provide automatic and user-driven insights.
- Provide statistical and correlational analysis of data with findings/insights.
- Prepare data with various methods of imputation like mean, median, mode, *ffill*, *bfill*, *KNN regressor*.
- Perform data type conversions for time, categorical data types
- Stunning visualizations of data based on target and correlation analysis
- Data split based on stratification(feature selection for stratification with least error)

B. Overview

We propose a tool for performing a comprehensive analysis of the given dataset to create meaningful and impactful ML modeling suggestions. The approach requires understanding and utilizing the dataset with mathematical and statistical

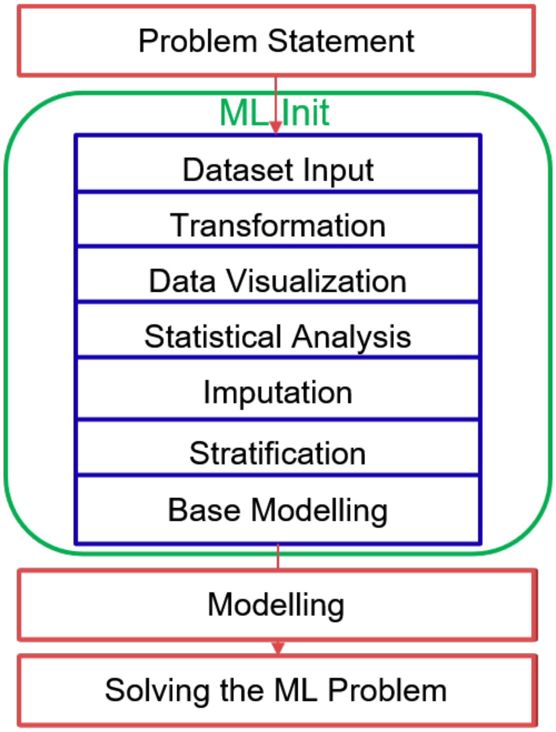


Fig. 2. ML Init acts as a shim layer between real world problem statement to a Machine Learning-based solution

rationales. ML Init communicates the insights and findings efficiently and clearly with the user using data cleaning, processing, and statistical/correlation analysis. The proposed solution enables the user to effortlessly interprets the EDA graphs/plots and enhance the domain knowledge.

III. Architecture

ML Init is an easy-to-access python notebook developed using Google Colab that provides an interactive user interface (UI), ideal for ML experts and beginners, as depicted in Figure 2. ML Init consists of 6 modules: User Input, Data Visualization, Statistical Analysis, Transformation, Imputation, Stratification, and Modelling. This section provides a brief explanation of each module.

A. User Input

The User Input module enables users to upload or use already uploaded datasets of various formats. Further, it prompts the user to enter the target column. ML Init analyses each feature in the file and create a report consisting of the type of ML problem (Regression or Classification) and the type of each attribute (Numerical or Categorical). It employs enhanced logic for detecting the kind of features instead of choosing based on the actual data type (int, float, or string).

IV. Methodology

A. Transformations

After classifying the numerical and categorical features, the dataset may need further cleaning based on particular attributes such as date, currencies, and Email. Usually, pandas' data frames may fail to identify date columns with inconsistencies. Also, currencies fields may have currency symbols that lead to classification as categorical. The transformation module with Regex determines such cases and handles them effectively.

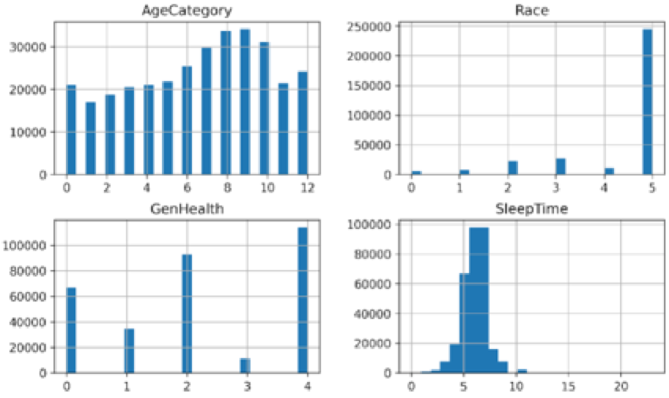


Fig. 3. Histogram of various features

B. Data Visualization

Data visualization module plots various graphs for visualizing the features of the dataset. The module uses various python libraries such as matplot, seaborn and yellow bricks. Currently, ML Init shows the following plots.

- *Histograms* - For all features
- *Countplot* - For categorical features
- *Box Plot* - For numerical columns
- *Distribution vs Target* - Distribution of categorical feature classes with respect to target feature
- *Correlation Analysis* - Pearson method for numerical columns and Chi Square test for categorical columns
- *Correlation with Target* - How the features are correlated to target using Pearson correlation method
- *QQPlot* - Quantile Quantile plot
- *Mutual Information* - Measure of mutual dependance

C. Statistical Analysis

One of the unique selling points of the proposed ML Init is the Statistical Analysis section. Most of the well-established Data Wrangling tool lacks an analysis part that a user can understand easily. We have added the following observatory sections in the proposed solution to ease the effort of end-users.

Skewness Detection module uses skewness and kurtosis functionalities and provides inferences to the user. Skewness essentially measures the symmetry of the distribution, while kurtosis determines the heaviness of the distribution tails. Based on the standard threshold values for each, the test classifies the features into Positive/Negative Skewes, Approximately Symmetric, Too Flat, and Too Peak.

Multi-Collinearity Test: This module uses the Variance Inflation Factor (VIF) for the test. VIF is used to detect the severity of multicollinearity using the ordinary least square

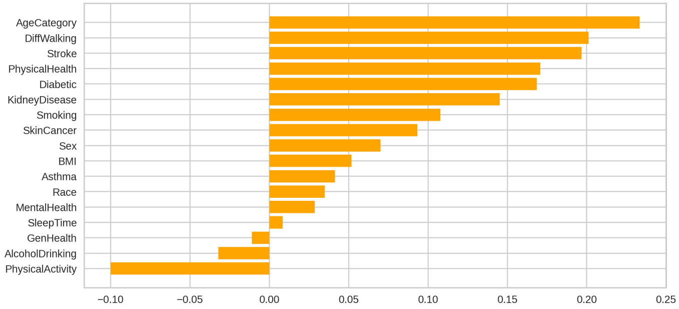


Fig. 4. Pearson Correlation

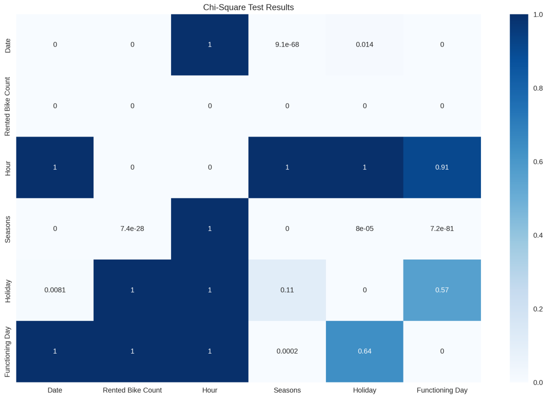


Fig. 5. Chi Square Test for Categorical Features

(OLS) regression analysis. Based on the standard threshold for the VIF Test, ML Init provides the list of features with high multicollinearity to the user.

Dimensionality Reduction Test: ML Init uses principle component analysis (PCA) for dimensionality reduction. It fills the missing data points and scales the features for performing PCA. This module shows the explained variance from the PCA and asks the user to choose the amount of variance needed. ML Init will decide the number of features to drop/retain based on the user-chosen value.

Missing Datapoints Report summarises the features that have the missing values and their ratio of empty points to the total data points. Users can perform the Imputation task based on the blank datapoint report.

D. Imputation

Imputation is the process of filling the missing values of each feature. There are many methods available for performing imputations for numerical and categorical columns. In addition to that, the user can choose to drop the data points with missing values. Following are the techniques employed in ML Init.

Numerical Feature: KNearest Neighbor Imputer, Iterative Imputer, Replace with Mean, Replace with Median, and Window.

Categorical Feature: KNearest Neighbor Imputer and Replace with most frequent class.

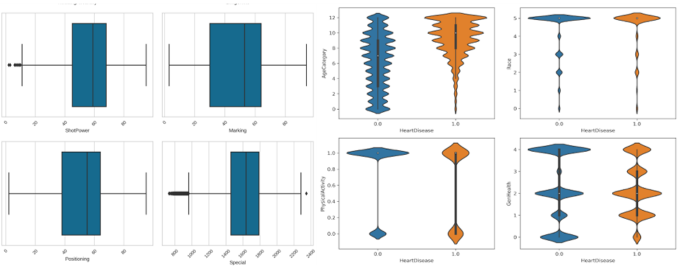


Fig. 6. Boxplot for Numerical and Violin plot for Categorical Features

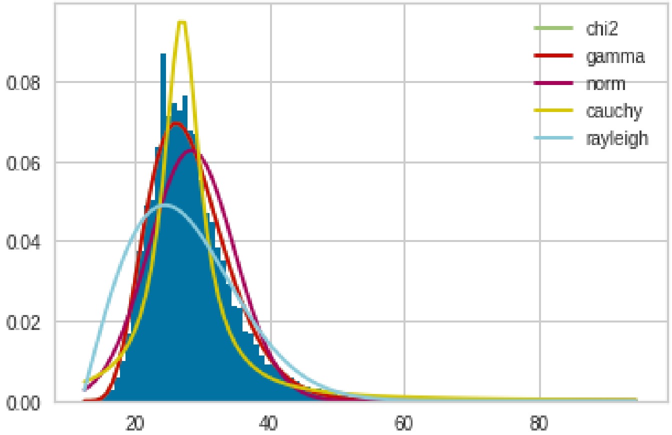


Fig. 7. Approximating a feature into various distributions

E. Stratification

For top correlated features whose correlation is more than 15% to the target, the Stratification module performs a stratified split and reports the error reduction achieved compared to a random split. In addition, ML Init enables the User to stratify the data based on any feature and compare the stratified error reduction.

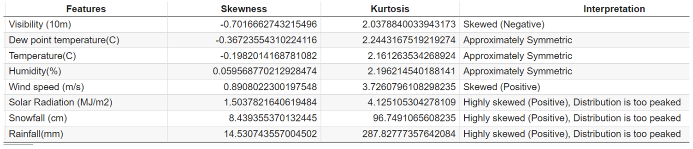
Stratification ensures that sample distribution in train/test data matches the complete data. ML Init creates a histogram and bucketed samples accordingly if a variable is continuous. On the other hand, if a variable is discrete, it uses the variable directly for bucketing samples.

F. Modelling

ML Init extends its functionality to implement various base models for regression and classification problems. As identified by the User input module, the Modelling module chooses algorithms based on the type of the problem. Additionally, It utilizes the inferences from the Statistical Analysis module to define the model pipeline consisting of scaling, encoding, and imputation schemes.

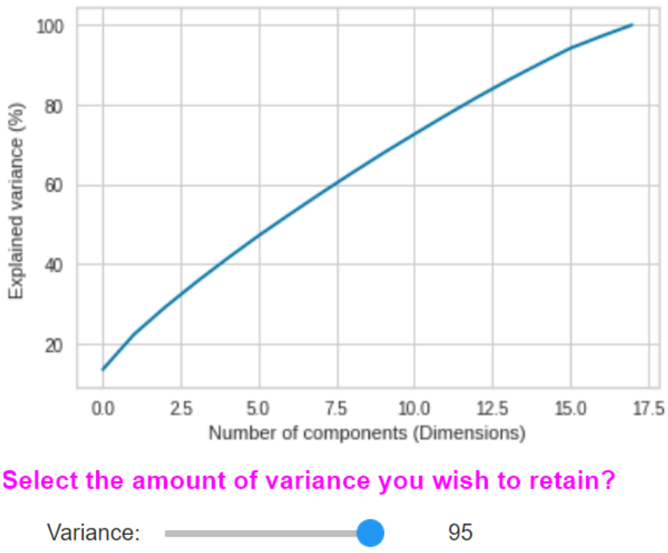
V. Result and Analysis

This section discusses the performance of the proposed system with some of the popular datasets in Kaggle [5]. We chose Heart Disease [6], and Seoul Bike Rent [7] problems,



Features	Skewness	Kurtosis	Interpretation
Visibility (10m)	-0.7016662743215496	2.0378840033943173	Skewed (Negative)
Dew point temperature(C)	-0.36723554310224116	2.2443167519219274	Approximately Symmetric
Temperature(C)	-0.1982014168781082	2.161263534268924	Approximately Symmetric
Humidity(%)	0.059568770212928474	2.196214540188141	Approximately Symmetric
Wind speed (m/s)	0.8908022300197548	3.7260796108298235	Skewed (Positive)
Solar Radiation (MJ/m2)	1.5037821640619484	4.125105304278109	Highly skewed (Positive), Distribution is too peaked
Snowfall (cm)	8.439355370132445	96.7491065608235	Highly skewed (Positive), Distribution is too peaked
Rainfall(mm)	14.530743557004502	287.82777357642084	Highly skewed (Positive), Distribution is too peaked

Fig. 8. ML Init shows easily explainable inference from Skewness and Kurtosis



Select the amount of variance you wish to retain?

Variance: 95

Fig. 9. Dimensionality Reduction Test

respectively, for regression and classification. Following are some of the observations made by the tool.

As shown in Figure 3, histograms help identify and glance at the feature distributions. This visualization will help in deciding scaling, encoding, and modeling attributes. Figure 4 and Figure 5 depict the Pearson correlation and Chi-Square test, respectively, two different approaches for correlation analysis. Box plots and Violin plots are used to describe the behavior of numerical features, respectively, as illustrated in Figure 6. Similarly, ML Init generates Countplot, Distribution vs Target, Correlation with Target and Mutual Information plots.

In the analysis section, ML Init tries to approximate the features into various distributions, as shown in Figure 7. This approximation helps in identifying the proper transformations for the column. Also, ML Init performs skewness and kurtosis tests to draw insights into the feature distribution. It attempts to make user-understandable remarks, as listed in Figure 8. In addition to that, it allows the user to choose the explained variance for performing the dimensionality reduction.

ML Init has more analysis sections that include Multicollinearity test using VIF score, Stratification of highly correlated columns, imputation of missing data points, preparing the pipeline, and showing the results with basic modeling. The ML Init is a valuable tool for creating excellent knowledge about the given dataset without taking much effort to understand it manually.

VI. Conclusion

This report proposes a comprehensive python-notebook based framework for data wrangling and exploratory analysis. EDA is one of the most vital parts of the end-to-end machine learning process. Even though many tools are available in the market, they fall short in providing analysis of the plots they create. On the other hand, ML Init uses various mathematical and statistical analysis methods to draw clear insight from the data set, which is helpful for the modeling. ML Init significantly improves the user expertise for data wrangling and hence, helps create better ML models.

References

[1] Janez Demšar et al., Orange: Data Mining Toolbox in Python, Journal of Machine Learning Research, 2013, p. 2349-2353

[2] Graham Williams, Data Mining with Rattle and R: The Art of Excavating Data for Knowledge Discovery, Springer, Use R!, 2011.

[3] Kan Nishida, Exploratory: Democratize Data Science, https://exploratory.io/, 2022.

[4] KNIME Inc., KNIME Software: End to end data science for better decision making., https://www.knime.com/software-overview, 2022.

[5] Kaggle Inc, Kaggle, https://www.kaggle.com/about/team, 2022.

[6] Heart Disease, Kaggle, https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease, 2022.

[7] Seoul Bike Rental Prediction, Kaggle, https://www.kaggle.com/c/seoul-bike-rental-prediction, 2022.