International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 11 ǁ November 2022 ǁ PP. 304-308

Data Pre-Processing Framework for Supervised Machine Learning

Naga Jahnavi Kommareddy
[*] *B.Tech Student, CSE, 2017-2021, CVR College of Engineering, Hyderabad, India*
Corresponding Author: knj192000@gmail.com

Abstract
Many factors contribute to the successful modelling of Machine Learning (ML) problems. The first and foremost step involves pre-processing of data instances and their quality presentation to get better accuracy in Machine Learning modelling. The data should be reliable, relevant and should not have redundancy and outliers. The data preparation stage consumes good amount of time in solving ML problems. The various steps involved in pre-processing of data are: cleaning, replacing the missing values, treatment of outliers, identification of less correlated features with dependant variable, feature reduction, data standardization/normalization, addressing the dataset imbalance issues etc. It is evident that no single algorithm can address all these steps in one go in pre-processing phase. The selection of pre-processing steps and methods are presented in this paper. The discussed pre-processing steps are applied in developing supervised ML model on weather Australia dataset. The dataset consists of 145460 records with missing values, outliers and imbalanced classes. The dataset consists of various properties of atmospheric conditions like, humidity, pressure, wind direction, location, rain occurrence etc. The objective is to predict whether rainfall will occur tomorrow based on the instances provided in the dataset. The ensemble methods are seen to predict with better accuracy of 93% and above. The model accuracy is found to be almost same with mean /mode values replacing the missing values as well as imputing with kNN imputer.

Keywords: *Data Pre-Processing, Supervised Machine Learning, Imputation of missing values, treatment of outliers, data standardization/normalization, model accuracy*

Date of Submission: 14-11-2022 Date of acceptance: 28-11-2022

I. INTRODUCTION

The importance of data-preprocessing as mentioned by various researchers in literature is discussed in this section. **S. Banumathi and Dr. A. Aloysius (2019)** presented An Enhanced Preprocessing Algorithms and Accuracy Prediction of Machine Learning Algortihms. The ML supervised algorithms performances have been checked with selected features with cross validation. Their analysis indicated that enhanced feature selection with supervised learning for data set resulted in accurately predicting the target value. They also discussed the normalization score and enhanced relief algorithm feature selections, k-fold cross validation methods.

Carlos Vladimiro and Gonzalez Zelaya (2019) discussed about Effects of Data Preprocessing on Machine Learning. They explored metrics to quantify the effect of some of these steps. They defined a simple metric called volatility, to measure the effect of including/excluding a specific step on predictions made by the resulting model. Using training set rebalancing as a concrete example, they reported on on measuring volatility in two public benchmark datasets, Students' Academic Performance and German Credit, with the goal of identifying the predictors for volatility that are independent of the dataset and of the specific preprocessing.

Stamatios-Aggelos N et al (2019) published about Data preprocessing in predictive data mining. In this work they presented the most well-known and widely used up-to-date algorithms for each step of data preprocessing in the predictive data mining framework. They demonstrated that the methods of daqta preprocessing have strong influence on the the performance of a classifier. **Changming Zhu and Daqi Gao (2016)** studied the influence of data preprocessing. In this work, they researched the influence of data preprocessing and came up with a conclusion that using different preprocessing methods leads to different classification performances. Furthermore, they experimented with some algorithms with different preprocessing methods also confirmed that preprocessing had a great influence on the performance of a classifier.

Theodoros Iliou et al (2015) proposed a Novel Machine Learning Data Preprocessing Method for Enhancing Classification Algorithms Performance. In this work, a novel data preprocessing method is proposed and evaluated in three difficult classification data sets of the well known UCI Repository, in which various classifiers have average performance lower than 75%. The three UCI repository datasets that have been used were the Mammographic masses, Indian Liver and Contraceptive Method. The performance of their proposed

data preprocessing method and Principal Component Analysis preprocessing method were evaluated using the 10-fold cross validation method assessing five classification algorithms, Nearest-Neighbour classifier (IB1), C4.5 algorithm implementation (J48), Random Forest, Multilayer Perceptron and Rotation Forest, respectively. The classification results were presented and compared analytically. The results indicated that the generated features after proposed preprocessing method implementation to the original dataset markedly improved the performance of the classification algorithms.

Nazri Mohd Nawi et al (2013) presented about The Effect of Data Pre-Processing on Optimized Training of Artificial Neural Networks. Their main focus was to improve the accuracy of ANN models by using three selected pre-processing techniques. In this study, They proposed to reduce the computational cost of ANN training by introducing pre-processing techniques (such as; Min-Max, Z-Score and Decimal Scaling Normalization). For that, four variations of well-known gradient descent methods were used. The simulations results showed that the use of pre-processing techniques increased the accuracy of the ANN classifier by at least more than 95%.

Data Preprocessing for Supervised Leaning was studied by **S. B. Kotsiantis, D. Kanellopoulos and P. E. Pintelas (2006).** This paper addresses issues of data pre-processing that can have a significant impact on generalization performance of a ML algorithm. Though lot of research has taken place on data preprocessing techniques still there is lot of scope for improvement in getting the data ready for efficient supervised learning due to the continuously evolving Machine Learning algorithms. Choosing relevant Machine Learning algorithm is an important task as it is not a straightforward one. Therefore it is felt that further research is required in this area of data preprocessing. Every step implemented in data preprocessing is very crucial and directly impacts the performance of Machine Learning algorithm.

II. DATA PREPROCESSING METHODS

The steps involved in data preprocessing are described in this section. The flow chart depicting the methods is shown in Figure 1.

1. **Data Visualization**: The dataset in .csv format is read into a pandas data frame. The data is displayed in the table format. The information of column data types and number of rows are extracted. The descriptive statistics of all columns are displayed and studied. The missing values are visualized in the graphical format using the python library "missingno". The number of missing values are extracted for each feature in the dataset. These details provide sufficient information regarding missing values and pave the way for further handling.
2. **Treating the missing values**: As a first step it is recommended to not to remove the columns or rows with missing values as it would result in loss of information and poor model performance. This is even more important for smaller datasets. One should exercise caution while deleting the entire column with more missing values. The correlation of the column with target variable may be checked in order that the model accuracy is not compromised. The most popular method of replacing missing values is with their column mean for continuous variables and column mode for categorical variables. The other method proposed for imputing missing values is using kNN algorithm for both continuous and categorical values. The other methods used for imputing missing values are: Multivariate Imputation by Chained Equation (MICE), Stochastic Regression Imputation, Hot-Deck Imputation, and Imputation using Deep Learning etc. The selection of suitable method for imputing missing values has been found to be problem specific rather than general suggestion.
3. **Handling the outliers**: Outliers are those data points that do not fall in the normal range of values and certain times may possess abnormal values also. The presence of outliers may occur due to large variability in the measurements or due to experimental errors. These data points are considered as noise in modeling. Care should be taken that the model does not capture the noise during training. Capturing noise may lead to over-fitting and poor generalization on unseen data. The outliers are to be either eliminated or shall be treated before using it for training. The univariate method, multivariate method and Minkowski error are some proposed methods for treating the outliers. The univariate and multivariate methods are used to detect and eliminate the outliers where as the Minkowski error method is used to reduce the impact of outliers on the model performance.
4. **Normalisation/Standardisation of features:** The data points in different columns values may not be on similar scale and may possess extremely low and extremely high values in magnitude. In such cases column data normalization or standardization is used to bring them on to similar scale. This is called feature scaling. This will help in faster convergence of Machine Learning models that are distance based in their computations. This may not impact the tree based models. Normalization is a feature scaling method in which values are adjusted so that they range between 0 and 1. This is also termed as Min-Max scaling. Standardization is another scaling method where the values are centered on the mean

with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation.

5. **Balancing the dataset:** Sometimes in the given training examples of supervised machine learning, the target variable classes can be imbalanced. The number of class 1 examples may be much higher than the class 2 examples or vice versa. This case may pose problem during training especially if the no of examples are less. For the case of larger datasets this may not pose a problem as there would be sufficient number of examples in both classes for training. For imbalanced data sets, the resampling methods can be used. The two resampling methods that are used are random oversampling and random undersampling. Random oversampling is achieved by randomly selecting examples from the minority class, with replacement, and adding them to the training dataset. Random undersampling is achieved by randomly selecting examples from the majority class and deleting them from the training dataset. Apart from resampling the popular techniques used to model imbalanced data sets are: using appropriate metrics, k-fold cross validation, clustering the abundant class etc.
6. **Feature Selection:** It is well known that all the independent features of a dataset may not equally contribute to the dependant variable in supervised machine learning. The correlation can be strong, moderate or weak. Having more features may not translate to improved accuracy of ML model. In fact the more the features present the more the complexity of model. It is wise to reduce the model complexity before building the model using feature selection methods. Filter methods such as Information Gain, Chi-square test, Fisher's test, Correlation coefficient, Variance threshold etc can be used to select best features for modeling. Wrapper methods like Forward feature selection, Backward feature elimination, Exhaustive Feature Selection, Recursive Feature Elimination are used for selecting the important featurures. Also Embedded method like Lasso Regularisation (L1) can be used to eliminate some unimportant features.

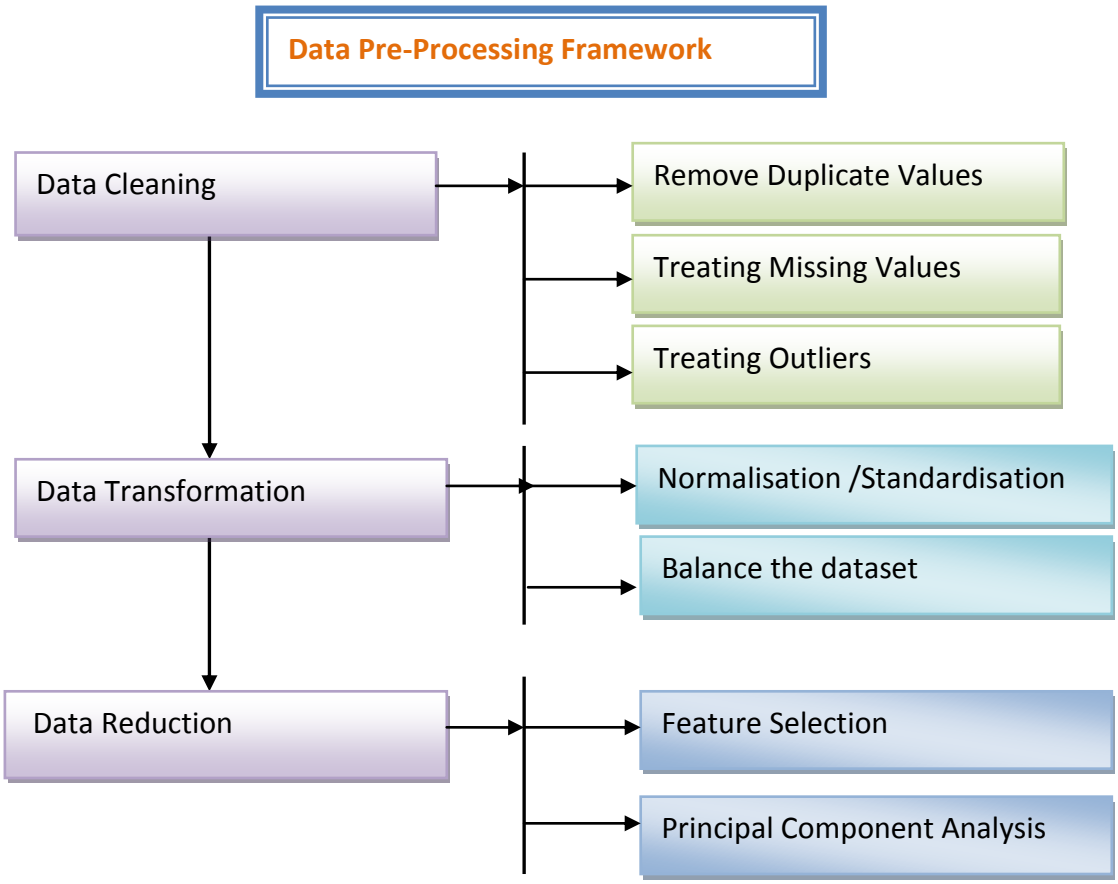

Figure 1. Data Pre-Processing Framework

III. RESULT AND DISCUSSION

The Australian weather dataset is chosen for study and the results are presented in this section. The dataset consists of 22 independent features and one dependant feature. The features, data type and number of missing values are shown in Figure 2.

The rows having missing values more than 50% are removed from the dataset. Remaining missing values are treated by replacing with standard methods. The effect of missing values on classification accuracy is studied in two stages. In the first stage the continuous data missing values are replaced with their mean values. The categorical missing values are replaced with their mode values. In the second stage the continuous missing values are imputed using "KNNImputer" from sklearn library and the categorical missing values are replaced with mode values. The results are compared.

The outliers are removed using stats models with features having z score higher than 3.

In the data transformation stage the categorical labels are encoded with numerical values. The standard scaler is applied to all columns. The "RainTomorrow" classes are imbalanced with an imbalance ratio of No Rain to Rain equal to 3.53. The class details are shown in Figure 3. The data set is balanced using SMOTE oversampling method. The results are compared without balancing the dataset. The study is carried out using three types of ML models namely Logistic Regression, XGBOOST and Random Forest Classifier. The models performance results are shown in Table 1.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 145460 entries, 0 to 145459
Data columns (total 23 columns):
 #   Column         Non-Null Count   Dtype         Missing Values
---  ------         --------------   -----         --------------
 0   Date           145460 non-null  object            0
 1   Location       145460 non-null  object            0
 2   MinTemp        143975 non-null  float64        1485
 3   MaxTemp        144199 non-null  float64        1261
 4   Rainfall       142199 non-null  float64        3261
 5   Evaporation    82670  non-null  float64       62790
 6   Sunshine       75625  non-null  float64       69835
 7   WindGustDir    135134 non-null  object        10326
 8   WindGustSpeed  135197 non-null  float64       10263
 9   WindDir9am     134894 non-null  object        10566
 10  WindDir3pm     141232 non-null  object         4228
 11  WindSpeed9am   143693 non-null  float64        1757
 12  WindSpeed3pm   142398 non-null  float64        3062
 13  Humidity9am    142806 non-null  float64        2654
 14  Humidity3pm    140953 non-null  float64        4507
 15  Pressure9am    130395 non-null  float64       15065
 16  Pressure3pm    130432 non-null  float64       15028
 17  Cloud9am       89572  non-null  float64       55888
 18  Cloud3pm       86102  non-null  float64       59358
 19  Temp9am        143693 non-null  float64        1767
 20  Temp3pm        141851 non-null  float64        3609
 21  RainToday      142199 non-null  object         3261
 22  RainTomorrow   142193 non-null  object         3267

 dtypes: float64(16), object(7)
```

Figure 2. Feature space variables and missing values

```
df_test["RainTomorrow"].value_counts()
0.0    112391
1.0     31780
Name: RainTomorrow, dtype: int64
```

Figure 3. Imbalanced classes details

Table 1. Models Performance Summary

Model	Missing Value Treatment	Balancing of Dataset	k-fold cross validation	Model Accuracy
Logistic Regression	Mean and Mode Values	Yes	No	77.19%
		No	No	84.10%
		No	Yes	84.42%

	KNN Imputer	Yes	No	78.70%
		No	No	84.40%
		No	Yes	84.52%
XGBOOST	Mean and Mode Values	Yes	No	85.75%
		No	No	85.42%
		No	Yes	87.53%
	KNN Imputer	Yes	No	84.79%
		No	No	85.10%
		No	Yes	88.17%
Random Forest	Mean and Mode Values	Yes	No	85.32%
		No	No	85.30%
		No	Yes	99.40%
	KNN Imputer	Yes	No	84.36%
		No	No	85.25%
		No	Yes	99.87%

The model performance is studied for 6 different cases on each model as shown in Table 1. It is observed that missing value treatment by replacing with mean and mode values yielded similar performance when replaced with KNNImputer. The k-fold cross validation with Random Forest Model gave the best accuracy than any other model. In the present study balancing of dataset did not influence the model accuracy. Using k-fold cross validation alone with Random Forest model gave the highest accuracy in predicting RainTomorrow class.

IV. CONCLUSIONS

- Data-preprocessing is an important step in ML model building. A framework is discussed and implemented on weather Australia dataset.
- Replacing the continuous missing values with mean and categorical variables with mode yielded similar result when replaced with KNNImputer.
- The k-fold cross validation with Random Forest is seen to do well in predicting the model accuracy when dataset is imbalanced.

REFERENCES

[1]. S. Banumathi, Dr. A. Aloysius, "An Enhanced Preprocessing Algorithms And Accuracy Prediction Of Machine Learning Algortihms", International Journal of Scientific & Technology Research Volume 8, Issue 08, August 2019
[2]. Carlos Vladimiro Gonzalez Zelaya, "Towards Explaining the Effects of Data Preprocessing on Machine Learning", IEEE 35th International Conference on Data Engineering (ICDE), 2019
[3]. Stamatios-Aggelos N. Alexandropoulos, Sotiris B. Kotsiantis And Michael N. Vrahatis, "Data preprocessing in predictive data mining", The Knowledge Engineering Review, Vol. 34, e1, 1–33, 2019
[4]. Changming Zhu, Daqi Gao, "Influence of Data Preprocessing", Journal of Computing Science and Engineering, Vol. 10, No. 2, June 2016, pp. 51-57, 2016
[5]. Theodoros Iliou, Christos-Nikolaos Anagnostopoulos, Marina Nerantzaki, "A Novel Machine Learning Data Preprocessing Method for Enhancing Classification Algorithms Performance", 16th EANN workshops, September 25-28, 2015, Rhodes Island, Greece, 2015
[6]. Nazri Mohd Nawi* , Walid Hasen Atomi, M. Z. Rehman, "The Effect of Data Pre-Processing on Optimized Training of Artificial Neural Network", Science Direct, Elsevier Procedia Technology 11 (2013) 32 – 39
[7]. Sotiris Kosiantis, Dimitris Kanellopoulos, P.E.Pintelas, "Data Preprocessing for Supervised Learning", International Journal Of Computer Science Volume 1 Number 1 2006 Issn 1306-4428
[8]. A. Famili a, Wei-Min Shen b, Richard Weber , Evangelos Simoudis, "Data Preprocessing and Intelligent Data Analysis", Intelligent Data Analysis 1, Elsevier, 3-23, 1997
[9]. https://www.kaggle.com/datasets