SMARTBRIDGE
Let's Bridge the Gap

INTERNSHIP REPORT APPROVAL FORM

July 1, 2019

With immense pleasure, this is to approve that the students of Sona College of Technology i.,e

Suhail Hafiz Khan J (1516102136),

Vikash M (1516102157),

Tamil Mani M (1516102144) and

Vijaiarivalagan K (1516102155)

successfully completed their Project and Project Report on **"Liver Patient Analysis"** under our guidance.

We are highly impressed with the work that they have done and commend them on their quick grasping skills. They have shown good intent to learn and have put the knowledge gained into application in the from of this project. We appreciate the hard work and commitment shown by them.

We, hereby approve that this document is completely checked and accepted by SmartBridge Technical Team. Its been an absolute pleasure to educate and mentor these students. We hope that this document will also serve as a Letter of Recommendation, to whomsover applied.

We wish them success in all future endeavors and a great career ahead.

GD Abhishek

AI Developer

TABLE OF CONTENTS

ABSTRACT

Problem Statement:

Patients with Liver disease have been continuously increasing because of excessive consumption of alcohol, inhale of harmful gases, intake of contaminated food, pickles and drugs. This dataset was used to evaluate prediction algorithms in an effort to reduce burden on doctors.

Requirements:

The Application Requirements for the Project is Anaconda platform for accessing Jupyter notebook, Ibm Watson, Node-Red for Web Form and Android Studio for developing an Application.

Outcome:

The outcome of the project gives us the prediction whether the person has the disease or not in either a Web Form or in the Liver Care App.

DATA PRE-PROCESSING

Pre-processing refers to the transformations applied to our data before feeding it to the algorithm. Data Pre-processing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis.

NEED FOR DATA PRE-PROCESSING

For achieving better results from the applied model in Machine Learning projects the format of the data has to be in a proper manner. Some specified Machine Learning model needs information in a specified format, for example, Random Forest algorithm does not support null values, therefore to execute random forest algorithm null values have to be managed from the original raw data set.

Another aspect is that data set should be formatted in such a way that more than one Machine Learning and Deep Learning algorithms are executed in one data set, and best out of them is chosen.

This article contains 3 different data pre-processing techniques for machine learning.

RESCALE DATA

- When our data is comprised of attributes with varying scales, many machine learning algorithms can benefit from rescaling the attributes to all have the same scale.

- This is useful for optimization algorithms in used in the core of

  machine learning algorithms like gradient descent.

- It is also useful for algorithms that weight inputs like regression and neural networks and algorithms that use distance measures like K-Nearest Neighbors.

- We can rescale your data using scikit-learn using the Min Max Scaler class

BINARIZE DATA (MAKE BINARY)

- We can transform our data using a binary threshold. All values above the threshold are marked 1 and all equal to or below are marked as 0.
- This is called binarizing your data or threshold your data. It can be useful when you have probabilities that you want to make crisp values. It is also useful when feature engineering and you want to add new features that indicate something meaningful.
- We can create new binary attributes in Python using scikit-learn with the Binarizer class.

STANDARDIZE DATA

- Standardization is a useful technique to transform attributes with a Gaussian distribution and differing means and standard deviations to a standard Gaussian distribution with a mean of 0 and a standard deviation of 1.
- We can standardize data using scikit-learn with the Standard-Scaler class.

IMPORTANT LIBRARIES

NumPy:

NumPy is a very popular python library for large multi-dimensional array and matrix processing, with the help of a large collection of high-level mathematical functions. It is very useful for fundamental scientific computations in Machine Learning. It is particularly useful for linear algebra, Fourier transform, and random number capabilities. High-end libraries like Tensor Flow uses NumPy internally for manipulation of Tensors.

Pandas:

Pandas is a popular Python library for data analysis. It is not directly related to Machine Learning. As we know that the dataset must be prepared before training. In this case, Pandas comes handy as it was developed specifically for data extraction and preparation. It provides high-level data structures and wide variety tools for data analysis. It provides many inbuilt methods for groping, combining and filtering data.

Matplotlib:

Matpoltlib is a very popular Python library for data visualization. Like Pandas, it is not directly related to Machine Learning. It particularly comes in handy when a programmer wants to visualize the patterns in the data. It is a 2D plotting library used for creating 2D graphs and plots. A module named pyplot makes it easy for programmers for plotting as it provides features to control line styles, font properties, formatting axes, etc. It provides various kinds of graphs and plots for data visualization, viz., histogram, error charts, bar charts, etc.

Seaborn:

Seaborn is built on top of Python's core visualization library Matplotlib. It is meant to serve as a complement, and not a replacement. However, Seaborn comes with some very important features. Let us see a few of them here. The features help in −

- Built in themes for styling matplotlib graphics
- Visualizing univariate and bivariate data
- Fitting in and visualizing linear regression models
- Plotting statistical time series data
- Seaborn works well with NumPy and Pandas data structures
- It comes with built in themes for styling Matplotlib graphics

In most cases, you will still use Matplotlib for simple plotting. The knowledge of Matplotlib is Recommended to tweak Seaborn's default.

WORKING WITH MISSING DATA IN PANDAS:

Missing Data can also refer to as NA (Not Available) values in pandas. In Data Frame sometimes many datasets simply arrive with missing data, either because it exists and was not collected or it never existed.

In Pandas missing data is represented by two value

None: None is a Python singleton object that is often used for missing data in Python code.

NaN: NaN (an acronym for Not a Number), is a special floating-point value recognized by all systems that use the standard IEEE floating-point representation

Pandas treat None and NaN as essentially interchangeable for indicating missing or null values. To facilitate this convention, there are several useful functions for detecting, removing, and replacing null values in Pandas Data Frame:

isnull(), isnan(), isna().

ONE HOT ENCODING

Sometimes we need to convert string values in a pandas data frame to a unique integer so that the algorithms can perform better. So we assign unique numeric value to a string value in Pandas Data Frame.

Say we have a table containing names and gender column. In gender column, there are two categories male and female and suppose we want to assign 1 to male and 2 to female.

STANDARDIZATION

It is a step of Data Pre Processing which is applied to independent variables or features of data. It basically helps to normalize the data within a particular range. Sometimes, it also helps in speeding up the calculations in an algorithm.

Why and Where to Apply Standardization?

Real world dataset contains features that highly vary in magnitudes, units, and range. Normalization should be performed when the scale of a feature is irrelevant or misleading and not should Normalize when the scale is meaningful.

The algorithms which use Euclidean Distance measure are sensitive to Magnitudes. Here feature scaling helps to weigh all the features equally.

Formally, if a feature in the dataset is big in scale compared to others then in algorithms where Euclidean distance is measured this big scaled feature becomes dominating and needs to be normalized.

Examples of Algorithms where Feature Scaling matters

1. K-Means uses the Euclidean distance measure here feature scaling matters.

2. K-Nearest-Neighbors also require feature scaling.

3.Gradient Descent: Calculation speed increase as Theta calculation becomes faster after feature scaling.

MODEL FITTING

Let us consider that the system is designing a machine learning model. A model is said to be a good machine learning model, if it generalizes any new input data from the problem domain in a proper way. This helps us to make predictions in the future data, that data model has never seen.

UNDERFITTING

A statistical model or a machine learning algorithm is said to have under fitting when it cannot capture the underlying trend of the data. (It's just like trying to fit undersized pants!) Under fitting destroys the accuracy of our machine learning model. Its occurrence simply means that our model or the algorithm does not fit the data well enough. It usually happens when we have less data to build an accurate model and also when we try to build a linear model with a non-linear data. In such cases the rules of the machine learning model are too easy and flexible to be applied on such a minimal data and therefore the model will probably make a lot of wrong predictions. Under fitting can be avoided by using more data and also reducing the features by feature selection.

OVERFITTING

A statistical model is said to be over fitted, when we train it with a lot of data (just like fitting ourselves in an oversized pants!). When a model gets trained with so much of data, it starts learning from the noise and inaccurate data entries in our data set. Then the model does not categorize the data correctly, because of too much of details and noise. The causes of overfitting are the non-parametric and non-linear methods because these types of machine learning algorithms have more freedom in building the model based on the dataset and therefore they can really build unrealistic models. A solution to avoid overfitting is using a linear algorithm if we have linear data or using the parameters like the maximal depth if we are using decision trees.

HOW TO AVOID OVERFITTING

The commonly used methodologies are:

CROSS –VALIDATION

A standard way to find out-of-sample prediction error is to use 5-fold cross validation.

EARLY STOPPING

Its rules provide us the guidance as to how many iterations can be run before learner begins to over-fit.

PRUNING

Pruning is extensively used while building related models. It simply removes the nodes which add little predictive power for the problem in hand.

REGULARIZATION

It introduces a cost term for bringing in more features with the objective function. Hence it tries to push the coefficients for many variables to zero and hence reduce cost term.

VISUALIZATION

Python is a great language for doing data analysis, primarily because of the fantastic ecosystem of data-centric Python packages. Data Visualization is a critical step for building a powerful and efficient machine learning model. It helps us to better understand the data, generate better insights for feature engineering, and, finally, make better decisions during modeling and training of the model. we will use the seaborn and matplotlib libraries to generate the visualizations. Matplotlib is a MATLAB-like plotting framework in python, while seaborn is a python visualization library based on matplotlib. It provides a high-level interface for producing statistical graphics. In this blog, we will explore different statistical graphical techniques that can help us in effectively interpreting and understanding the data. Although all the plots using the seaborn library can be built using the matplotlib library, we usually prefer the seaborn library because of its ability to handle DataFrames.

MODEL EVALUATION

ACCURACY

In accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. One may think that, if we have high accuracy then our model is best. Yes, accuracy is a great measure but only when you have symmetric datasets where values of false positive and false negatives are almost same. Therefore, you have to look at other parameters to evaluate the performance of your model. For our model, we have got 0.803 which means our model is approx. 80% accurate.

Accuracy = TP+TN/TP+FP+FN+TN

ROC

ROC curves are frequently used to show in a graphical way the connection/trade-off between clinical sensitivity and specificity for every possible cut-off for a test or a combination of tests. In addition, the area under the ROC curve gives an idea about the benefit of using the test(s) in question.

ROC curves are used in clinical biochemistry to choose the most appropriate cut-off for a test. The best cut-off has the highest true positive rate together with the lowest false positive rate

NODE RED

Node-RED is a programming tool for wiring together hardware devices, APIs and online services. Primarily, it is a visual tool designed for the Internet of Things, but it can also be used for other applications to very quickly assemble flows of various services.

Node-RED is open source and was originally created by the IBM Emerging Technology organization. It is included in IBM’s Bluemix (a Platform-as-a-Service or PaaS) IoT starter application package. Node-RED can also be deployed separately using the Node.js application. At present, Node-RED is a JS Foundation project.

Node-RED enables users to stitch together Web services and hardware by replacing common low-level coding tasks (like a simple service talking to a serial port), and this can be done with a visual drag-drop interface. Various components in Node-RED are connected together to create a flow. Most of the code needed is created automatically.

ANDROID APPLICATION

In android studio the user-interface part of the machine learning model is developed as a mobile application. Android Studio is the official open source integrated development environment (IDE) for Google's Android operating system, built on Jet Brains IntelliJ IDEA software and designed specifically for Android development. It is available for download on windows, MacOS and Linux based operating systems. XML (Extensible Markup Language) is used for layout and Java programming language is used for the application.

The android application has doctor login and patient login separately. The patients can login using their patient-id and patient should provide the test report values for prediction. Then the values are sent to the node-red and the prediction is done in the ibm-cloud and the predicted output is displayed to the patients. In the doctor login the doctor’s id is authenticated and the doctor can get results of all the patients with the predicted values.

REFERENCES:

1. T. M. Lakshmi, A. Martin, R. M. Begum, and V. P.Venkatesan, “An Analysis on Performance of Decision Tree Algorithms using Student’s Qualitative Data,” Int. J. Mod. Educ. Comput. Sci., vol. 5, no. 5, pp. 18–27, 2013.

2.P. Sharma and A. P. R. Bhartiya, “Implementation of Decision Tree Algorithm to Analysis the Performance,” Int. J. Adv. Res. Comput.Commun.Eng., vol. 1, no. 10, pp. 861– 864, 2012.

3. Ho, Tin Kam (1995). Random Decision Forests *(PDF)*. Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282. Archived from the original (PDF) on 17 April 2016. Retrieved 5 June 2016.

4.Whitaker, Jeffrey. "The Matplotlib Basemap Toolkit User's Guide (v. 1.0.5)". Matplotlib Basemap Toolkit documentation. Retrieved 24 April 2013.

5.Galkin, Alexander (November 28, 2011). "What is the difference between test set and validation set?". Retrieved 10 October 2018

6. Grossman, Robert; Seni, Giovanni; Elder, John; Agarwal, Nitin; Liu, Huan (2010). "Ensemble Methods in Data Mining: Improving Accuracy Through Combining Predictions". Synthesis Lectures on Data Mining and Knowledge Discovery. Morgan & Claypool. 2: 1–126. doi:10.2200/S00240ED1V01Y200912DMK002.