Journal of Informatics
Electrical and Electronics Engineering, 2021,
Vol. 02, Iss. 02, S. No. 021, pp. 1-8
ISSN (Online): 2582-7006

A Framework for Implementing Prediction Algorithm over Cloud Data as a Procedure for Cloud Data Mining

Safwan A. S. Al-Shaibani[1], Parag Bhalchandra[2]

[1,2]School of Computational Sciences S.R.T.M.University ,Nanded, MS, 431606, India
[1]safwan.srtmu@gmail.com, [2]srtmun.parag@gmail.com

How to cite this paper: Safwan A. S. Alshaibani, Parag Bhalchandra (2021) A Framework for Implementing Prediction Algorithm over Cloud Data as a Procedure for Cloud Data Mining. *Journal of Informatics Electrical and Electronics Engineering*, Vol. 02, Iss. 02, S. No. 021, pp. 1-8, 2021.

https://doi.org/10.54060/JIEEE/002.02.021

Received: 02/04/2021
Accepted: 23/05/2021
Published: 04/06/2021

Abstract

The cloud has become an important phrase in data storage for many reasons. Cloud services and applications are widespread in many industries including healthcare due to easy access. The limitless quantity of data available on the clouds has triggered the interest of many researchers in the recent past. It has forced us to deploy machine learning for analyzing the data to get insights as well as model building. In this paper, we have built a service on Heroku Cloud which is a cloud platform as a service (PaaS) and has 15 thousand records with 25 features. The data belongs to healthcare and is related to post-surgery complications. The boost prediction algorithm was applied for analysis and implementation was done in python. The results helped us to determine and tune some of the hyperparameters which have correlations with complications and the reported accuracy of training and testing was found to be 91% and 88% respectively.

Keywords

Heroku cloud ,CatBoost algorithm, prediction model, binary Classification

1. Introduction

In the last decade, people have migrated to the cloud for hassle free storage of data. The potential data available on the cloud have triggered attention of many researchers for its effective analysis and model building. Data mining such massive data is a challenging job. Data mining techniques implemented through cloud computing will allow us to retrieve useful information from virtually integrated data while lowering infrastructure and storage expenses. We take up a case of health care data to demonstrate how cloud data can be data mined? In the healthcare sector, data mining and machine learning have endless applications. These can help to streamline hospital administrative procedures, map and manage infectious diseases, customize medical treatments, etc[2]. They will play a key role in supporting clinical decision-making, allowing for earlier disease identification, and tailored treatment plans to ensure optimal outcomes. These may also be used to explain and advise patients with various care choices on possible disease pathways and outcomes [1,2]. The work undertaken belongs to surgical

complications for patients [1]. A surgical complication, for instance, is any adverse and unpredictable outcome of an operation affecting the patient [1]. It is not fixed, but depends on the level of surgical capacity and the facilities available [1]. It has two potential outcomes either a patient has a complication or has not. Hence this overall work belongs to binary classification. Binary classification determines the fate of the patient from expected results according to given dataset from health centers. Here we will use CatBoost algorithm as an example of prediction or supervised learning [3]. Due to their ability to generalize from data, predictive models have huge potential. Although, predictive models don't have the human expert's skills, they can deal with much greater amounts of data and can probably find subtle patterns in the data that a human can't. Predictive models depend a lot on training data, and are dependent on quality of data. This is promising when case is of clouds with voluminous data. Preferably, a model must extract the existing signal from the data and ignore any spurious patterns (noise). However, this is not an easy task, because data are usually far from perfect; small numbers of samples, irrelevant variables, missing values, and outliers are some of the imperfections [18].

In order to increase the predictive models' ability to extract important, we have followed standard data cleansing cycle including data preprocessing, smoothing for removing the superimposed noise, imputations for missing values, or excluding the outlier examples, common scaling and centering, etc. The later were part of advanced feature engineering techniques [18].

The rest of this study is organized as, section 2 introduces research methodology where the authors have implemented CatBoost algorithm by following standard steps. Session 3 focuses on model building which is done using a python platform. Section 3 concludes with a discussion of the findings. Section 4 summarizes the overall findings, stating that overall accuracy is 91 percent.

2. Research Methodology

In this part, we will demonstrate the basic steps followed for the prediction and classification. The surgery complication database has numerical (quantity) data of both integer and float types. Our proposed model will work to analyzing the data and scaling that improve performance of the model by hyper parameters tuning to reach the level of reliability and safety to be used in the field of healthcare with flexibility and ease. **Figure** 1 shows patient surgical complication prediction process model from the beginning until its uploading to Heroku cloud for the end user to use.

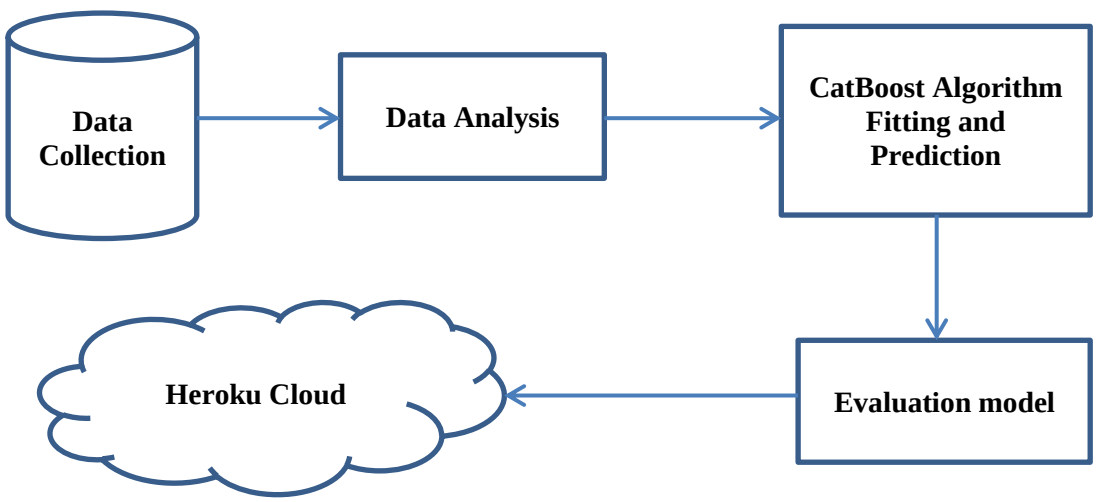

Figure 1. Prediction Process

2.1. Data Source

We have collected our dataset from Kaggle site which is considered a platform for all researchers interested in databases [3], that data set is called "Surgical-deepnet.csv". It has 25 features and 14635 records where all features are numerical (integer or float).

2.2. Features Engineering and Selection

It is the technique of extracting features from raw data using domain knowledge [4]. These features can be used to improve the performance of machine learning algorithms. So we checked each record to extract and handle noise, NAN, and imbalanced data then cleaning if possible based on size of dataset. During feature selection we pick out those features from the dataset that contribute and affect most to the target variable [5]. In other words, we choose the best predictors for the target variable. The classes in the “sklearn.feature_selection” module can be used for feature selection/dimensionality reduction on sample sets, either to improve estimators’ accuracy scores or to boost their performance on very high-dimensional datasets. While this dataset is a type of binary classification so there are independent variables and dependent variable which will impact the correlation between the two variables because whenever you change the values of independent variables will give us new expectations based on the target variable. Here, we select “SelectKBest” from the class "sklearn.feature_selection", this is one of the automatic feature determination techniques in Python that works to choose the features that have the strongest relationship with the target variable and this technique can determine the number of features that will be extracted from the data. In our case, we have determined the number of features to be ten features in the head of the dataset which is represented in **Table** 1 without showing the target variable that represents the tenth feature (label).

Table 1. Dataset Head

bmi	Age	asa_status	baseline _cancer	baseline_ charison	Baseline_ osteoart	ccsComplication Rate	ccMort03Rate	complication_rsi	mortality_rsi
39.56	44.0	0.0	0.0	0.0	0.0	0.142512	0.004026	0.00	0.00
39.19	43.0	1.0	0.0	1.0	1.0	0.081977	0.002959	-0.32	-0.16
31.03	48.0	0.0	0.0	0.0	0.0	0.466129	0.012903	0.58	0.09
27.20	74.0	1.0	0.0	0.0	1.0	0.081977	0.002959	-2.86	-1.96
25.88	73.0	1.0	0.0	0.0	0.0	0.105720	0.000789	-1.45	0.08

Of the benefits that we get reduces over fitting that has less redundant data. This means less possibility of making decisions based on redundant data/noise. Hence, reduces Training Time where algorithms train faster. Below **Figure** 2 shows the features that were selected from the dataset according to the strength of their correlation with the dependent variable or label.

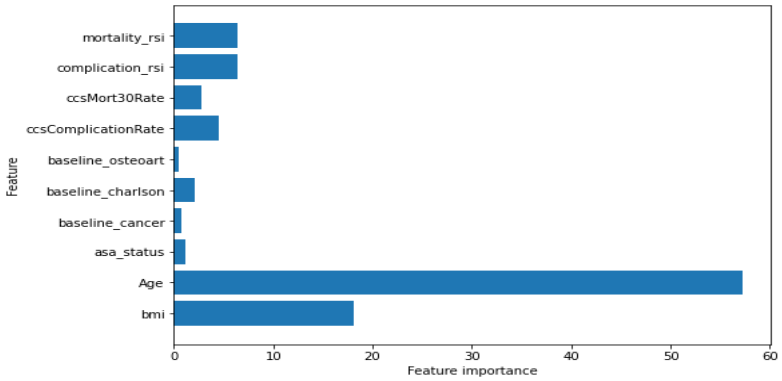

Figure 2. Features Selection

2.3. Data Scaling

Data scaling and data normalization terminologies are used inter changeably and their purpose is to standardize or transfer data into ranges and forms, suitable for modeling and mining[19]. Compared to models trained on un-scaled data, models trained on scaled data typically have significantly higher performance. Data scaling is regarded as an important phase in data preprocessing[20]. Data preparation involves using techniques such as normalization and standardization to rescale input and output variables prior to training machine learning model[6]. It is noticeable in our dataset that each feature has its own range. This will generate a slow process of processing this data. We used one of the methods provided in data scaling techniques namely "Standardization" where the values are centered on the mean with a unit standard deviation where is the mean of the feature values and is the standard deviation of the feature values. Here's the formula for standardization:

$$X' = \frac{X - \mu}{\sigma} \tag{2.1}$$

2.4. CatBoost Algorithm

The CatBoost is an algorithm for machine learning using gradient boosting on decision trees[15]. It exists as a library of open source platforms like python. It's an open- source algorithm that based on the decision trees. CatBoost was developed by Yandex. It is the successor of the MatrixNet algorithm that is widely used within the company for ranking tasks, forecasting and making recommendations. It is universal and can be applied across a wide range of areas and problems[16]. The goal of the training is to select a model that correctly solves the given problem classification or multi-classification for any input object based on certain features[9]. The validation dataset (test dataset), which has data in the same format as the training dataset, is checked for accuracy, but is only used for the evaluation of the standard of training[10]. A set of decision trees are constructed consecutively during training. In contrast to the previous trees, each successive tree is constructed with reduced loss. The starting parameters control the number of trees. We used the over fitting detector to prevent over fitting[11]. When it is triggered, trees stop being built. Gradient boosting is the best method for problems with noisy data, complex dependencies as well as heterogeneous features. It has a lot of implementations in search engines, weather forecasting, and recommendation systems. Gradient boosting combines iteratively weaker models (base predictors). In a function space, such a procedure corresponds to gradient descent[17]. As mentioned above, decision trees are the most common base predictor for gradient boosting algorithm. XGBoost, LightGBM and H2O are the most used algorithms. But the state-of-the-art algorithm is CatBoost. The CatBoost uses the following algorithm for updating the hyperparameters and the weight value corresponding to each model (**Algorithm** 1)[17].

Input: $\{(X_k,Y_k)\}_{k=1}^{n}$ ordered according to σ ,
the number of trees I
$M_i \leftarrow 0$ *for* $i = 1..n$
for *iter* $\leftarrow$ *1 to I* **do**
  for $i \leftarrow$ *1 to* n **do**
    for $j \leftarrow$ 1 to i-1 **do**
      $g_i \leftarrow \frac{d}{da}\text{Loss}(y_j,a)|_{a=Mi\ (xj)}$
    end
    $M \leftarrow$ *LearnOneTree*$((X_j,g_j)$ for j = 1..i-1)
    $M_i \leftarrow M_i + M$
  end
end
return $M_1..\ M_n;\ M_1(X_1)..M_n(X_n)$

Algorithm 1 CatBoost algorithm [17]

2.5. Evaluation

For any classification problem, the Classification Report can be used to display the precision, F1, recall, and support scores for the model, to support problem detection and easier interpretation. Where Precision (also called positive predictive value) that seeking about "what proportion of identifications was correct?". And therefore defined as the ratio of true positives to the sum of true and false positives. Where ***tp*** is true positive and ***fp*** is false positive, It is defined as follows:

$$Precision = \frac{tp}{tp+fp} \tag{2.2}$$

Recall (also known as sensitivity) trying to answer this query "what attribution of actual positives was identified correctly?". Thus, It's the ratio of true positives to the sum of true positives and false negatives**.** Where ***fn*** is false negative and other factors same existing in precision. Mathematically, It is defined as follows:

$$Recall = \frac{tp}{tp+fn} \tag{2.3}$$

When you seek a balance between Precision and Recall, F1 value is needed, noting that the best score is 1.0 and the worst is 0.0, thus F1 Score might be a better measure to use if we need to seek a balance between Precision and Recall. Generally speaking, F1 scores are lower than accuracy measures as they integrate precision and recall into their computation. The formula is as follows:

$$F_1 = \left(\frac{2}{recall^{-1}+precision^{-1}}\right) = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{2.4}$$

Support is the number of actual class occurrences in the dataset defined. Support does not change between models, but diagnoses the process of assessment instead.

2.6. Heroku Cloud

For end-user access to data anytime and anywhere, we must publish the dataset and model in a type of cloud. So we have used Heroku cloud which is a platform for deploying and running modern apps as a service built on a controlled container structure, with integrated data services and a powerful ecosystem as well as supporting several programming languages[13]. The Heroku

developer experience is an app-centered software delivery approach, integrated with the most common developer tools and workflows at the moment[14]. This model was deployed in the Heroku cloud after being ready for end-user use.

3. Model Building and Results

Using Python language, the experimental models were built. CatBoost is a standalone library. It is the robust algorithm and does not need extensive hyper- parameter tuning. After gathering data from healthcare centers and websites interested in the healthcare area, we have done cleaning the dataset from null values (NAN), imbalanced data, and delete duplicated rows then we chose essential features according to correlation with the target(label). After that, we have done scaling of this dataset and finally splitting the dataset into training and test data to apply CatBoost algorithm on the dataset. We have got accuracy in test 88% and accuracy in training 91%. **Figure** 3 shows plotting for some features from both complication and non-complication.

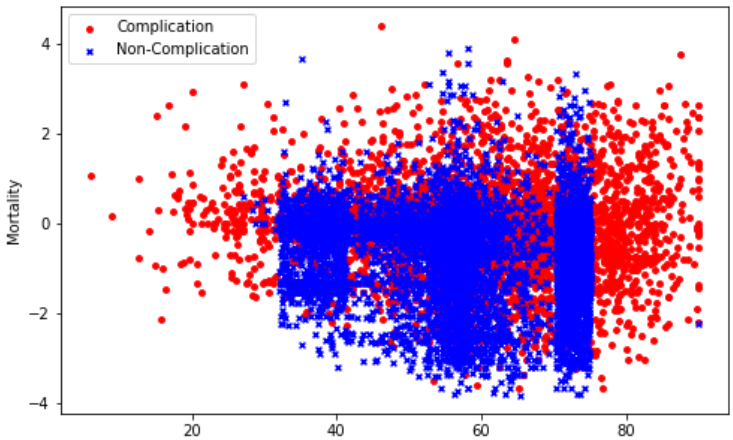

Figure 3 Classification between two parameters

In **Figure** 4 we can see the number of records which contain patients who have complications and non-complications.

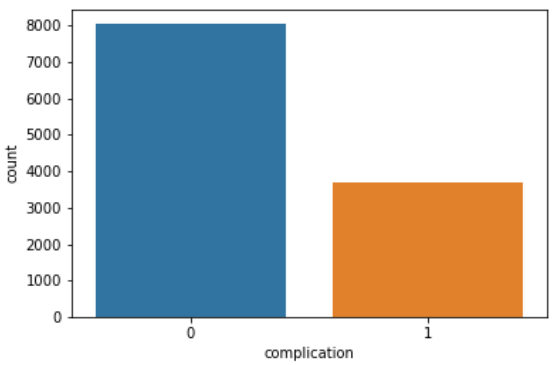

Figure 4 Rate of complications

A confusion matrix, also known as an error matrix, is a particular table layout that allows the performance of an algorithm to be visualized, with each matrix row representing instances in a predicted class, while each column represents instances in

an actual class [7]. **Figure** 5 indicates the rate between the values that are real and predicted.

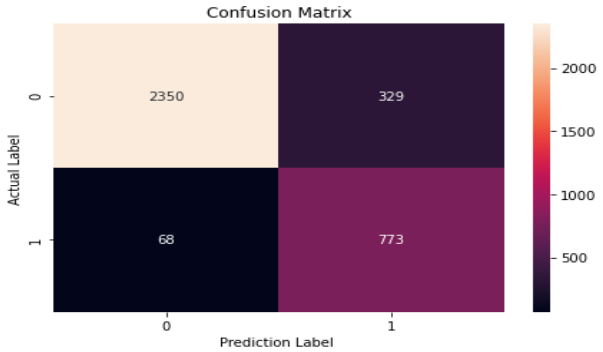

Figure 5. Confusion matrix

A ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds [8]. Fig. 6 shows curve plots between the two parameters which are the actual and predicted values.

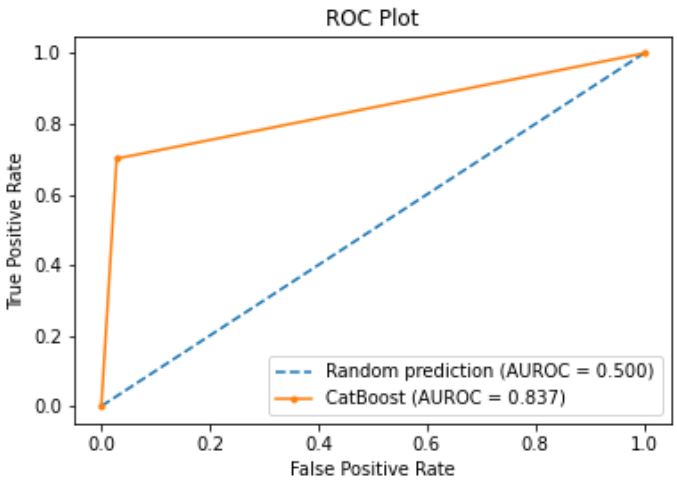

Figure 6. ROC curve

4. Conclusions

The underlined research work is an evidence of data mining in clouds where our suggested model works as an add on and mines massive data in the cloud with significant accuracy. Here we have built a model that predicts the presence of complications of surgery of the patient. We have uploaded our devised out a model to Heroku cloud where receive the related parameters for providing expected results. The overall implementations were done in Python. The reported accuracy rate in the test model was 88% and on the training model, it was observed to be 91%. The overall work can be useful as a role model for the healthcare industry for analysis of their data on the clouds.

References

[1]. D.K. Sokol, J. Wilson, “What is a surgical complication?” World journal of surgery, vol.32,no.6,pp.942-944, 2008.

[2]. R. Bhardwaj, A.R. Nambiar, & D. Dutta, “A study of machine learning in healthcare.,” In IEEE 41st Annual Computer Software and

Applications Conference (COMPSAC), vol. 2, pp. 236-241,2017.

[3]. M. Ramaswami and R. Bhaskaran, “A CHAID based performance prediction model in educational data mining,” 2010.

[4]. A. Zheng, and C. Amanda,” Feature engineering for machine learning: principles and techniques for data scientists. " O'Reilly Media, Inc.", 2018.

[5]. R. Abdulhammed, M. Faezipour, A. Abuzneid et al., "Effective Features Selection and Machine Learning Classifiers for Improved Wireless Intrusion Detection," *International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1-6, 2018.

[6]. X. H. Cao, I. Stojkovic, and Z. Obradovic, “A robust data scaling algorithm to improve classification accuracies in biomedical data,” BMC Bioinformatics, vol. 17, no. 1, p. 359, 2016.

[7]. J. Xu, Y. Zhang, and D. Miao, “Three-way confusion matrix for classification: A measure driven view,” Inf. Sci. (Ny), vol. 507, pp. 772–794, 2020.

[8]. M. C. Sachs, “PlotROC: A tool for plotting ROC curves,” J. Stat. Softw., vol. 79, no. Code Snippet 2, 2017.

[9]. G. Huang, L. Wu, X. Ma, et al., “Evaluation of CatBoost method for prediction of reference evapotranspiration in humid regions,” J. Hydrol. (Amst.), vol. 574, pp. 1029–1041, 2019.

[10]. L. Prokhorenkova, G. Gusev, A. Vorobev, et al., CatBoost: unbiased boosting with categorical features. In Advances in neural information processing sys-tems pp. 6638-6648,2018.

[11]. H. Li, H. Huang, and Z. Zheng., "Research on Credit Risk of P2P Lending Based on Cat-Boost Algorithm." vol.9, no.3, pp.137-141,2019.

[12]. J. P. Craig, K. K. Nichols, E. K. Akpek et al., “TFOS DEWS II definition and classification report,” Ocul. Surf., vol. 15, no. 3, pp. 276–283, 2017.

[13]. P.K. Das, N. Sinha, & B. Annappa,” Data privacy preservation using aes-gcm encryption in Heroku cloud (No. 2615),” EasyChair, pp.1-8, 2020.

[14]. B. H. Lee, E. K. Dewi and M. F. Wajdi, "Data security in cloud computing using AES under HEROKU cloud," *27th Wireless and Optical Communication Conference (WOCC)*, pp. 1-5,2018.

[15]. L. Breiman, “Arcing the edge,” Technical Report 486, Statistics Department, University of California at Berkeley, pp.1-14,1997.

[16]. A. V. Dorogush, V. Ershov, and A. Gulin, “CatBoost: gradient boosting with categorical features support,” arXiv:1810.11363,2018.

[17]. A. Malakhov, F. Goncharov and E. Gryazina, "Testing machine learning approaches for wind plants power output,"*International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE)*, pp. 1-6,2019.

[18]. X. H. Cao, I. Stojkovic, and Z. Obradovic, “A robust data scaling algorithm to improve classification accuracies in biomedical data,” BMC Bioinformatics, vol. 17, no. 1, p. 359, 2016.

[19]. J. Han, M. Kamber, & J. Pei,” Data mining: concepts and techniques 3rd edn,” Morgan Kaufmann,2011.

[20]. S. SHaykin, “Neural networks and learning machines 3rd edn,” Simon Haykin. Prentice hall, pp.1-917,2009.