International Journal of Advanced Scientific Technologies ,Engineering and Management Sciences (IJASTEMS-ISSN: 2454-356X) Volume.3,Special Issue.1,March.2017

Optimized Machine Learning Model for categorizing Women's Health Risk Segments

Mrs. Jangam J S Mani#1 , Prof. K. Sandhya Rani *2
**** Computer Science Department, Sri Padmavathi Mahila University , Tirupati, Andhra Pradesh, India***
1 jsmani.jangam@gmail.com
2sandhyaranikasireddy@yahoo.co.in
1Keshav Memorial Institute of Technology affiliated to
JNTUH 1Research Scholar, Sri Padmavathi Mahila Viswa
Vidyalayam, Tirupati, AP, India

Abstract- Reproductive health status of a women is the benchmark of quality of life maintained by the people of that society. Women's reproductive health risks (such as HIV/AIDS infections) were categorized into different class of risk segments based on the set of demographics and her medical data collected in the form of questionnaire during their visit to the health care center. The classification of health risk segments are helpful both for the medical practitioner in giving personalized medication and education to help the subject in overcoming or minimizing their health risks especially during the reproduction. The problem is a multi-class classification problem and it was addressed using logistic regression, random forest techniques, etc. which are discussed in detail in the following sections

Keywords- demographics, health risk segment, multi-class classification, logistic regression, random forest.

I.INTRODUCTION

"When women progress, all the society get benefited, not only that the succeeding generations are given a better start in life"[1]. Over the past two decades there was a substantial progress in improving women health, thereby nearly 50% reduction in death during maternity, but much remains to be done.

The illness or death of a woman has serious impact on the health and career of her children, family and community as a whole. Every year, about 10 million women are facing life-threatening complications during pregnancy and childbirth, sometimes leading to either death or long term disability.

Early and unwanted childbearing, unsafe abortions, HIV/AIDS and other sexually transmitted infections(STIs), and pregnancy-related illnesses and deaths are becoming a significant proportion of the burden of illness experienced by women especially in low, middle income countries and continue to disproportionately affect marginalized groups in high income countries[2].

Millennium women health development goal focuses on reducing the maternal mortality ratio (MMR) by 75 percent between 1990 and 2015 and ensuring universal access to reproductive health by 2015[3]. Majority of the maternal deaths can be prevented through timely pre and postnatal care, skilled labor assistance during delivery, availability of emergency care to deal with complications and emergency, along with limiting the no. of births for mothers and children, as well as the gap between the pregnancy, etc[4].

The objective of this paper is to improve women's reproductive health outcomes in underdeveloped regions by suggesting an optimized machine learning algorithm that can accurately categorize the women patient (aged 15-30 years old) into different health risk segments and subgroups by considering the most important aspects for women's health include providing health information and contraceptive services, strengthening maternal health care, tackling non-communicable diseases, and preventing and responding to violence against women and girls.

RELATED WORK

There are many algorithms that could be applied on patient demographics to predict health risks such as cardiac problems, pulmonary problems, diabetes, cancer, etc. Decision Trees, Neural Networks, Genetic Algorithm methods are more popular ways of predicting health risks. The number of geographies, segments and subgroups, respectively will determine performance of the function. However, the most popular and common one is Multinomial Logistic Regression model is the most common one. This paper discusses three models that are implemented for making the predictions.

1. First, Multiclass logistic regression was to find the

combined class label for the segment, sub group for each geographical ID and also found the accuracy of it.
2. Then applied , Random forest model to predict the risk segment.
3. Finally, XGboost algorithm was applied to get a fast, scalable, efficient and accurate predictions.

DATA

Dataset used for building an optimized machine learning model for the categorizing the women reproductive health issues into health risk segment was acquired from Bill & Melinda Gates Foundation[5]. The data of this dataset was collected in the survey during 2015 exploring the wants, needs, and behaviors of women and girls age between 15-30 years old with regards to their sexual and reproductive health in nine geographies such as Bihar-India, Uttar Pradesh-India, Nothern, southern Nigeria, Kenya, Myanmar, congo, Ethiopia, and Burkina Faso. The dataset consists of 9000 subjects, dataset has been split into 75% training and 25% testing data to suffice the training and testing to make the classification model more accurate on unseen data. The dataset contains 9000 observations (1000 observations from each region) and 50 attributes related to the women reproductive health such as Religion, Income, literacy, tribe, HIVKNOW (knows HIV status), LaborDeliv (had both labour and delivery in HealthCare center), etc. The sexual and reproductive health risks were then evaluated by clinical practitioners and are assigned to different risk segments and subgroups.

A. Data Preprocessing :
After loading the dataset with 9,000 women patient's information:
step1: combine the geographical ID (geo) column and segment, and subgroup columns in the training data into a single label column so that, a single machine learning model can be build for all regions, segments, and subgroups.

```
combined_label <- 100*dataset1$geo +
10*dataset1$segment + dataset1$subgroup

data.set <- cbind(dataset1, combined_label)
data.set$combined_label <-
as.factor(data.set$combined_label)
```

step2: excluding the segment, subgroup columns from the dataset.
Step3: since patientID is unique for each row in both train and test datasets, so excluded it from the feature list.

Step 4: Cleaning the missing data by imputing missing values with 0 or mean. Religion column was dropped from the dataset, as it has 15 rows missing from the dataset and it is string type. As it might not play a major role in the classification even if it dropped.

B. Splitting and Cross Validation:

The dataset is actually split into train and test dataset by taking 75% of training data and remaining 25% of validation data to train and test machine learning models. As the training dataset has the balanced proportion of all the classes, stratification is not used while splitting it. Due to time constraint cross validation is not adopted here. If used the results would be more accurate and stronger.

```
nrows <- nrow(data.set)
sample_size <- floor(0.75 *
nrows) set.seed(98052)
train_ind <- sample(seq_len(nrows), size =
sample_size) train <- data.set[train_ind, ]
validation <- data.set[-train_ind, ]
```

METHODS IMPLEMENTATION

A. Multiclass Logistic Regression *model :*

It is an extension of logistic regression, which is conducted when the dependent variable is nominal with more than twolevels ie., it analyzes dichotomous (binary) dependents. It is used to describe data and to explain the relationship between one dependent nominal variable and one or more continuous-level(interval or ratio scale) independent variables [6].

To predict the risk segments, combine label is modeled against train dataset with all columns excluding patientID and subgroup, the multinom() function was imported from the library 'nnet'. It is repeated for about 500 iterations and tested upon validation dataset. The performance of the model evaluated or tested in terms of accuracy- it is nothing but the no. of subjects that are correctly classified under the right risk segment or category. The accuracy of this model on validation data is 76.230129%

```
> #formula for the model
> col_names <- colnames(validation)
> model_formula <- formula(paste('combined_label ~ ', paste(col_names[feature_index], collapse=' + '), sep=''))
>
> glmmodel <- multinom(model_formula, data = train, MaxNWts=3000, maxit = 500)
# weights:  2109 (2016 variable)
initial  value 14306.456770
iter  10 value 10854.290766
iter  20 value 8707.379297
iter  30 value 7794.934537
iter  40 value 6762.062936
iter  50 value 5089.525760
iter  60 value 4231.036315
iter  70 value 3836.678043
...
iter 400 value 816.006859
iter 410 value 811.757594
iter 420 value 808.170450
iter 430 value 805.752355
iter 440 value 803.627501
iter 450 value 801.697417
iter 460 value 799.863728
iter 470 value 797.711202
iter 480 value 796.073757
iter 490 value 794.323534
iter 500 value 793.049904
final  value 793.049904
stopped after 500 iterations
> predicted_labels <- predict(glmmodel, validation)
> #predicted_labels
> accuracy <- round(sum(predicted_labels==validation$combined_label)/nrow(validation) * 100,6)
> print(paste("The accuracy on validation data is ", accuracy, "%", sep=""))
[1] "The accuracy on validation data is 76.230129%"
```

B. *Random Forest Model:*
As discussed earlier our second implementation would be the Random Forest model. A Random Forest consists of a collection of simple tree predictors, each capable of producing a response when presented with a set of predictor values. For classification problems, this response takes the form of a class membership, which associates, or classifies, a set of independent predictor values with one of the categories present in the dependent variable. In the case of regression problems, the tree response is an estimate of the dependent variable given the predictors[7].

For classification problems, given a set of simple trees and a set of random predictor variables, the Random Forest method defines a margin function that measures the extent to which the average number of votes for the correct class exceeds the average vote for any other class present in the dependent variable. This measure provides us not only with a convenient way of making predictions, but also with a way of associating a confidence measure with those predictions[8].

The mean-square error for a Random Forest is given by:

$$\text{mean error} = (\text{observed} - \text{tree response})^2$$

The predictions of the Random Forest are taken to be the average of the predictions of the trees:

$$\text{Random Forest Prediction } s = \frac{1}{K}\sum_{K=1}^{K} K^{th} \text{ tree response}$$

wh ere the index k runs over the individual trees in the forest.
rfmodel = randomForest(model_formula, data = train,
ntree=180, nodesize=5)
To implement random forest algorithm, our first step is to construct the formula which is nothing but a data frame or a matrix of predictors or response variables in training set. In this case, formula is defined as a set of all the attributes other than the patientID, subgroup, because the model is supposed to predict the geographical region wise risk segment and its corresponding sub groups from the combined_columns. The idea to construct response by taking the average no. f votes for the correct class exceeds the average vote for any other class present in the dependent variable.

In the fitted randomforest model, the second parameter called data is an optional data frame containing the variables in the model.

Third parameter, ntree = 180 is nothing but the no. of trees to grow and this should not be set too small.

The Fourth parameter in the above formula, nodesize. It is the minimum size of the terminal nodes. The default values for classification and regression are different and they are 1 and 5 respectively.

Like every model fitted, random forest also has components like, type (classification or regression), votes, predicted, confusion, etc.
To get their respective values, use the modelname$component.
Eg: rfmodel$type - gives us the classification as output.

Finally the model should be applied upon the validation data set to test the accuracy in predicting the class labels for different risk segments and subgroup region-wise. The accuracy of randomforest model =84.102952%.

```
> library(randomForest)
> set.seed(155)
> rfmodel = randomForest(model_formula, data = train, ntree=180, nodesize=5)
> predicted_labels <- predict(rfmodel, validation)
> accuracy <- round(sum(predicted_labels==validation$combined_label)/nrow(validation) * 100,6)
> print(paste("The accuracy on validation data is ", accuracy, "%", sep=""))
[1] "The accuracy on validation data is 84.102952%"
```

C. XGBoost Algorithm :

The considered problem is a multiclass classification problem, most of the decision tree classifiers are binary and Since the majority of the given dataset's attributes/features are also binary, hence it is good to apply xgboost algorithm to the dataset inorder to categorize the subjects into different risk segments.

XGBoost, a scalable machine learning system for tree boosting. It has both linear model solver and tree learning algorithms. xgboost is at least 10 times faster than existing gradient boosting implementations[9]. It supports various functionality like classification, regression and also ranking. XGBoost only works with numeric vectors, it demands us to convert all other forms of data into numeric vectors. A simple method to convert categorical variable into numeric vector in the form of 0's and 1's[10].XGboost () supporting many parameters grouped into three categories such as i) general parameters - talk about the general functionality of the algorithm ii) Booster parameters - talk about the tree boosting features like child weight, depth of the tree, etc, iii) learning task parameters - meant for optimizing the functionality of the algorithm [11].

To apply xgboost algorithm to do the multiclass classification:
step1: load the library(xgboost)
step2: load the dataset
step 3: data pre-pocessing and feature engineering

In this step, we will impute all the missing data with mean or

zero, remove the class label feature and the features with the string type such as religion, etc.
step 4: partition the dataset into train and test data
step 5: tune and run the model

```
##Train an XGBoost model with the training data
xgb11<-xgboost(xgb.DMatrix(
        data.matrix(train[-c(1,19,49,50,51,52)]),
        label=t(train[52]), missing = NaN),
        max.depth = 30, eta = 0.025, nround = 500,
        objective = "multi:softmax",
        num_class = 38, gamma = 1)
```

where data.matrix - is taking the sparse form of the predictors from the train data, along with class label to be found, it considers all missing values as NaN.

Other parameters for this xgboost() are: maximum depth of the tree, in this case it is taken as 30, maximum no of iterations as 500, no.of classes as 38, 100, and 1000, etc (it should be used only with multiclass classification problem), etc.

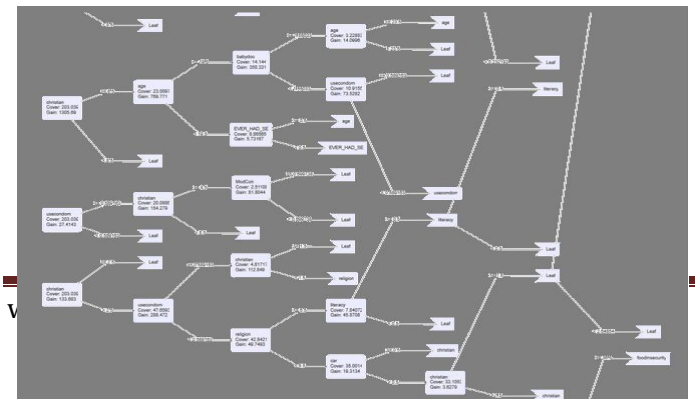
Once the model is tuned, evaluate its performance on the

3. XGBoost Decision Tree Algorithm: Performs best with really high accuracy, but when test set is large, this method is slower than the above two. The accuracy is 86.53% on the validation set.

XGBoost Model		
Train Dataset size	No. of Iterations	Best Performance
75%	1000	86.525360%
75%	500	86.501506%
60%	1000	85.666982%
60%	500	85.619678%

Finally XCBoost algorithm is considered to do prediction in the case multiclass classification with more accuracy, which is higher than the expected accuracy. But the accuracy was got by just testing on 3800 observations in the validation set, which is not large enough, so the reliability of the accuracy should be doubted. But still this paper have reason to justify that the xgboost algorithm works better than the multinomial logistic regression and random forest, this is classifying the dataset with more accuracy than the other two models.

The classification of risk segments and their respective subgroups obtained using xgboost algorithm:



validation dataset by calling a predict() by passing xgboost model fitted and the validation dataset as parameters.

The accuracy of the xgboost model upon the validation dataset is : 86.52536%.

```
> # Predict - evaluate the performance of the trained model on the remaining portion of
 the split data.
> pred <- predict(xgb11, xgb.DMatrix(data.matrix(validation[-c(1,19,49,50,51,52)]),labe
l=t(rep(0,nrow(validation))), missing = NaN))
> pred <- lab.lev[pred]
> accuracy <- round(sum(pred==validation$combined_label)/nrow(validation) * 100,6)
> # Print the accuracy
> print(paste("The accuracy on validation data is ", accuracy, "%", sep=""))
[1] "The accuracy on validation data is 86.52536%"
```

RESULTS AND JUSTIFICATION

i Multinomial Logistic Regression Model: The most efficient model, very fast, but the performance is not good as my expectation.

Multinomial Logistice Regression Model	
No. of Iterations	Best Performance
100	73.883422%
500	76.230129%
600	76.760030%

i Random Forest: It is faster than the multinomial logistic regression model, but when the no. of trees are changed the performance is similar fluctuating.

Randon Forest Model	
No. of Trees	Best Performance
50	84.178653%
100	84.935655%
180	84.102952%

CONCLUSION

The main objective of this paper is to categorize reproductive health risks of women based on segment and as per their respective subgroups. To categorize them, a three digit code comprising of segment, single, and subgroup columns are taken and tried to properly map them to geographic ID. To address this problem, multinomial logistic regression model was applied and accuracy was evaluated, then due to its poor performance, random forest with bagging was applied on the same dataset and achieved good accuracy and further tried with xgboost algorithm, got more and consistent accuracy than the earlier two models. Hence xgboost algorithm was considered to be better choice for decision tree based multi class classification problems[9] as it can yield pretty satisfactory accuracy than other model.

As XGBoost algorithm can be applied in a parallel, distributed environment, In future it can be applied upon the huge datasetswith hadoop and spark environment to provide in-memory, scalable machine learning algorithm to address the women health risks especially to find and suggest the root causes for still births, maternal deaths, prolonged disability

due to maternal conditions.

FUTURE WORK

1.This paper given a good try in categorizing the women reproductive health risks segment-wise and its sub group wise. As the dataset size was not very huge, classification is done by using R language.

i In reality, health care data is enormous, it is better to segment the health risks using big data technologies like hadoop, spark, etc.

ii The entire women health risks were not addressed completely yet. Hence, there is a lot of future research scope especially when you take each kind of a health risk faced by women as research topic and given a solution, definitely the society would get benefited.

REFERENCES

[1] http://www.truth-out.org/news/item/32012-how-women contribute-3-trillion-to-global-healthcare.

[2] World Health Organization. Global health estimates 2000-2012. WHO, 2014.

[3] World Health Organization (WHO), United Nations Children's Fund (UNICEF), United Nations Population Fund (UNFPA), World Bank. Trends in maternal mortality: 1990-2013. WHO, UNICEF, UNFPA and The World Bank estimates. WHO, 2014.

[4] World Health Organization. Unsafe abortion: global and regional estimates of the incidence of unsafe abortion and associated mortality in 2008. 6th ed.WHO, 2011.

[5] http: //az754797.vo.msecnd.net /competition /whra /data /WomenHealth_Training.csv

[6] http://www.statisticssolutions.com/mlr/

[7] Trover Hastie, Robert Tibshirani. An Introduction to Statistical Learning with Applications in R, http://www.springer.com/series/417

[8] https://www.statsoft.com/Textbook/Random-Forest

[9] Tianqi Chen, Carlos Guestrin. XGBoost: A Scalable Tree Boosting System, ACM, 2016

[10] https://www.analyticsvidhya.com/blog/2016/01/xgboost-algorithm-easy-steps/ [11]http://xgboost.readthedocs.io/en/latest/R-package /xgboostPresentation.html