European Journal of Scientific Research
ISSN 1450-216X / 1450-202X Vol. 133 No 1 June, 2015, pp.66-85
http://www.europeanjournalofscientificresearch.com

A Nifty and Accuracy Architecture of New Prediction Model to Improve Predictive Analytics in Healthcare

Samaher Al-Janabi
Corresponding Author, Department of Information Networks, Faculty of Information Technology
University of Babylon,40 Street, Babylon 00964, Iraq
E-mail: samaher@itnet.uobabylon.edu.iq

Hayder Fatlawi
Department Software, Faculty of Information Technology
University of Babylon,Najaf-Hilla Street, Babylon 00964, Iraq
Email: hayder_alnajfi2@uokufa.edu.iq

Abstract

Aim: Management of healthcare's resources contributes to improving the quality of medical services, thereby enhancing the level of health of society in general. This management requires providing prospective information about the need for patients admitting in a hospital, and the necessary medical resources.Prediction techniques represent an effective tool for knowledge discovery in huge and complex datasets in many fields including healthcare.

Methods: In this work, we design and implement a prediction model called a Modern Prediction Model for HealthCare Problem (MPM-HCP) which introduces two improvements for Gradient Boosting Machine (GBM) prediction technique. MPM-HCP developed (GBM) by inspiring positive sides of linear regression ***to replace splitting criterion with a correlation measure in regression tree building***. It also reduced the complexity of building boosted model by using a ***fast method for choosing best split point.***

Results: MPM-HCP has significant behavior in terms of prediction error and execution time. In comparison with tradition gradient boosting trees, the MPM-HCP has a testing error of 0.468,while original GBM based on sum of squared has error of0.491, and original GBM based on standard deviations has 0.481 error. Training time is also reduced more than 85%.

Conclusions: MPM-HCP implementation showed that there were three attributes frequent in binary regression trees building. Those attributes were gender of patient, number of claims to admit hospital, and the medical procedure group, which means those attributes is more correlated with the target of prediction (i.e. number of hospitalization days). MPM-HCP confirms the ability to produce precise prediction result, and the scalability to deal with huge dataset in suitable execution time.

Keywords: Correlation Measure, Gradient Boosting Machine, Healthcare Dataset, Predictive Analysis.

1. Introduction

Healthcare concerned applying all necessary medical procedures to restore health of people or prevent aggravation of health problems [Raym13]. The costs of those procedures grow rapidly to satisfy suitable quality of health system which makes the problems of this field have a significant effect on financial resources of modern world [Xian11];it also takes reasonable attention in research work. Many of the tools such as prediction techniquesare used to deal with challenges of healthcare problems.Prediction of future hospitalization could reduce unnecessary costs that are considered one of the difficulties of healthcare financial aid management. The two factors which control hospitalization costs are the number of patients who entered the hospital and how long each patient stays.

Many of problems could be mentioned here, such as inaccurate estimation of the stay period from medical staff, which may increase waste of resources, and high percentage of historical records have a zero value of days in the hospital,which means the data has an imbalance distribution, making predictive process more complicated. The main challenge is to build the algorithm of prediction models aiming to solve the problems above with precise prediction and less time. The goal of that prediction model is to predict precisely how long each patient is to be admitted to the hospital, depending on his medical records,so supporting medical staff can make the correct decision and reduce the costs, thereby enhancing the services of health and special government institutions.

In this work, MPM-HCP is presented to solve healthcare problems by replacing the split criteria of boosted regression tree with correlation measure,and then evaluate the resulted model by many error measures based on real and complex datasetto get the lowest prediction error and less time. MPM-HCP has three major stages: (i) preprocessing stage (i.e., dataset is made more suitable for the next stage; (ii) building prediction model using improved boosted regression; (iii) evaluatingthe results of the previous stages.

The related works in this field are varied between recommended systems and prediction models for unnecessary hospitalizations. Table (1) summarizes properties of every work and the differences from our work.

Duana et. al., 2010 created a recommended nursing clinical system to help make the right decision and improve clinical quality control by using association rules to find patterns in item sets of a community hospital in the Midwest dataset and use support, confidence , and lift as utility measurements [Duan11] . Our work is close to this work in dealing with the same field of healthcare, but we differ from it working in descriptive task, while our work is predictive task.

Xiang et. al., 2011 used machine learning algorithms to reduce unnecessary hospitalizations by predicting how long a patient will stay in the hospital for the next year according to his record in the prior year. Researchers used support vector machine (SVM), random forest, regression tree and boosting ensemble with HPN 2011 Dataset [Xian11]. Our work is near to Xiang's work by using same dataset, but with different types of prediction techniques.

Rashedur and Fazle, 2011 used and compared different decision tree classification techniques to classify admitted patients according to their critical condition and developedan application to diagnose and measure the criticality of the newly-arrived patient to mining hospital surveillance unit using C4.5 Decision Tree Classifier and evaluated the results using False positive rate (FP),Recall, Precision [Rash11]. We are similar to this work in the domain of healthcare, but vary in using different dataset, evaluation measures and regression rather than classification.

Hadi and Nima, 2012 proposed the correlation as a splitting criterion for multi-branch decision trees. They replaced traditional criterion for building decision trees, in which the feature having largest absolute correlation with the target is chosen as the best splitter in each node. For choosing best threshold on the selected feature, they used the ability of SVM to maximize the margin between classes [Hadi12] . Our work is close to this work in using the correlation as a splitting criteria, but we use it for predictive tasks with binary regression trees of GBM rather than classification tasks of multi-branch decision trees. We also use a simple method for finding the best threshold rather than using SVM.

Jufen et. al., 2013 constructed a Chi-Squared Automatic Interaction Detection (CHAID) classification tree with 10-fold cross-validation to predict probability of death or hospitalization for heart failure and compared the result with logistic regression (LR) models using ROC curve analysis based on TEN-HMS Dataset. They found the CHAID tree performed better than the LR-model for predicting the composite outcome [Jufe13] . We are similar to this work by utilizing the same task of data mining that is the prediction for healthcare;yet we vary in using a different dataset and a different technique of prediction.

Nannan, 2014 compared performance of three prediction techniques (i.e., linear regression, random forest and gradient boosting) on hospitalization dataset to explain which technique is the best. By experiments, Nannan found that the random forest technique provides the best prediction of patient hospitalization [Nann] . Our work is close to this work in using same dataset, but we differ from it by developing a new prediction model.

Table 1: Comparison among related works and our work

No	Author, Year	Data Mining Technique	Dataset	Our work difference Points
1	Duana et. al., 2010	Association rules	midwest dataset	Improving boosted regression tree instead of association rules
2	Xiang et. al., 2011	SVM, Random Forest, regression tree and boosting ensemble	HPN dataset first release	Improving boosted regression tree instead of using typical techniques
3	Rashedur and Fazle, 2011	C4.5 decision tree classifier	Hospital of ICDDR,B dataset	Utilize Gradient Boosting instead of single decision tree
4	Hadi and Nima, 2012	Multi branch decision tree, SVM, correlation	Iris, Pima, Glass, Zoo and others	Utilize correlation to improve binary regression trees rather than Multi branch decision tree
5	Jufen et. al., 2013	CHAID	TEN-HMS Dataset	Improving boosted regression tree instead of CHAID
6	Nannan, 2014	linear regression, random forest and gradient boosting	HPN dataset	Improving boosted regression tree instead of using typical techniques

The remainder of this paper layout is as follow: second section explains Concepts of Boosting Regression Trees. Third section shows the main stages of building the proposed model called a Modern Prediction Model for HealthCare Problem (MPM-HCP). We can show the main results of implementation of MPM-HCP with all details in fourth section. The limitations, assumptions and solutions of this work are explained in fifth section. Finally, Sixth section presents the conclusion and future works.

Concepts of Boosting Regression Trees

The term prediction includes both numeric prediction and class label prediction. While the binary and multi-classification techniques predict categorical (discrete, unordered) class labels, regression technique models continuous-valued functions. That is, regression is used for predicting unknown numerical data values instead of (discrete) class labels[Ali12b]. Regression analysis could be defined as a statistical methodology which is mostly used for numeric prediction [Jiaw13] . Healthcare data analysis in this work focused on prediction of continuous values, which is to try to answer the question, "How many days will be spent by a patient in the next year?"

Binary Regression Tree

The regression tree is an extension for an ordinary decision tree for performing the numeric prediction tasks. In each leaf (terminal node), it stores the average value of target Y of data records that reach the leaf during the training process. For predicting any new data record, the regression tree is followed down to a leaf using the record's attributes values to make decisions at each node. When it reaches to a terminal node, the tree assigns previously stored value of target Y to that record.

Mostly, a decision tree chooses the splitting attribute to maximize the information gained; for numeric prediction, it is appropriate to minimize the variation in the target values in each node [Ali12a]. Less variation means more homogeneity in which the values close to each other and that will reduce error of prediction.

Aregression tree is growing as follow [Trev09] : data contain ***n*** records consists of ***k*** attributes and a target. Each record has a tuple (xi, yi) where i=1,2,…,N, and xi = (xi1, xi2, . . . , xik). A regression tree should decide which attribute will be used for splitting data atthe root of tree and what the splitting point will be(i.e. age < 20). A binary regression tree (**BRT**) starts with producing two child nodes from the root that will split the data into two regions, R1, and R2. For this level of tree, the target can be modeled as follows:

$$f(x) = c_{R1} I(x \in R1) + c_{R2} I(x \in R2) \quad (1)$$

Where c_{R1} represents the average of target's values in R1 as follows:

$$c_{R1} = avg(y_i | x_i \in R1) \quad (2)$$

and c_{R2} represents the average of target's values in R2 as follows:

$$c_{R2} = avg(y_i | x_i \in R2) \quad (3)$$

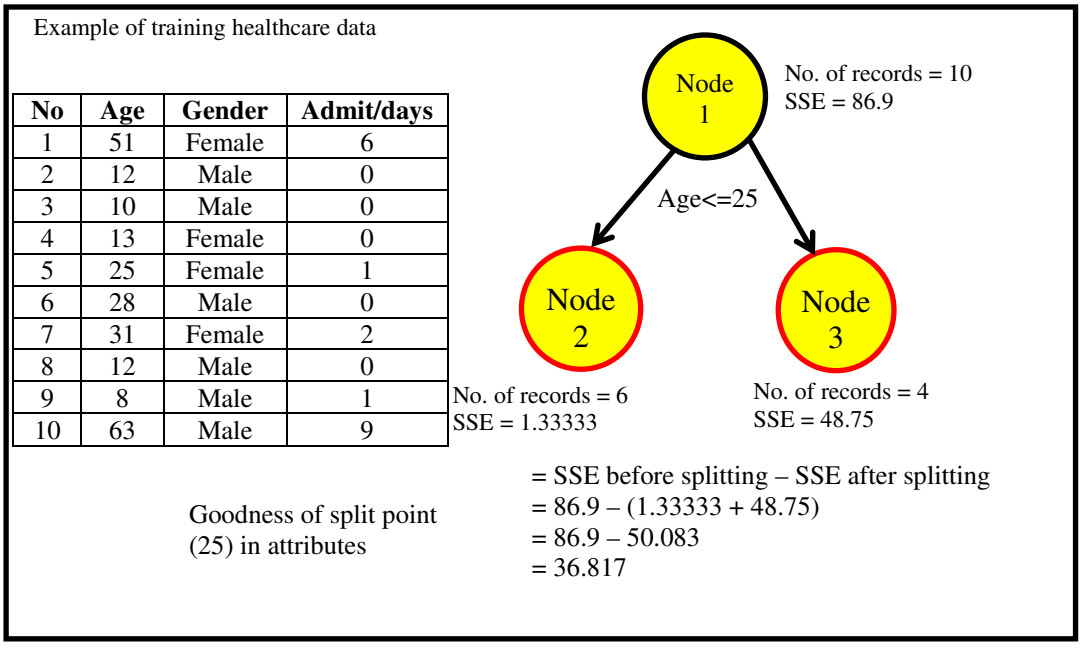
Also $I(x \in R1)$is a binary function that detects whether record *x* belongs to region1 (first child node) or not, and the same with $I(x \in R2)$, which detectswhether*x* belongs to region2 (second child node) or not. This procedure is repeated recursively for each child node to produce a new child node in which R1 will be split into R3, R4, and R2 will split into R5, R6. Each split operation is required choosing an attribute as a splitter and choosing the best split point to get more homogenous target's values.

The quality of an attribute and the quality of a specific split point for splitting the data are measured by sum of squared error (SSE). The goal is to minimize SSE between target values of a data record in a particular node and the average of target of that node in BRT[Max13] . Goodness of a split operation that split the data into two regions, R1, R2, can be measured by the following equation:

$$SSE = \sum_{yi \in R1} (yi - \overline{y1})^2 + \sum_{yi \in R2} (yi - \overline{y2})^2 \quad (4)$$

Where $\overline{y1}$ and $\overline{y2}$ are the averages of the target values in regions R1 and R2, respectively, BRT chooses the attribute and the split point which minimize SSE as little as possible. Figure (1) illustrates the evaluation of attribute's goodness of specific split point for one level of BRT.

Figure 1: Evaluation of attribute's goodness during BRT building



Standard deviation STD is another splitting criterion that is used to find the quality of an attribute (and split point). The value of STD represents an indicator for homogeneity of data before and after the splitting operation. If the overall sum of STD in child nodes is less than STD of parent node (closer to the zero), the splitting is preferred. Choosing splitting attributes (and split point) depends on which one has more reduction after splitting [12, 13]. Goodness of a split operation that splits the data into two regions, R1, R2, can be measured by the following equation:

$$\mathrm{STD}_{\mathrm{Reduction}} = \mathrm{STD}(\mathrm{R})_{before} - \left(\frac{n}{n_{R1}} \times STD_{R1} - \frac{n}{n_{R2}} \times STD_{R2}\right)\ldots \quad (5)$$

Boosted Ensemble Model

Models based on single tree have some weakness points such as model instability, which means that any slightchanges in data may change in the structure of the tree and thereby the interpretation of the tree model.Also,**BRT** produces rectangularregions that contain more homogeneous target values as a result of splitting the data. If the relationshipbetween selected attributes and the target cannot be sufficiently defined by rectangularregions, then **BRT** willhave a larger prediction error. To solve these problems, ensemble methods have been developed whichcombine many trees [Max13] .

Boosting is an ensemble forward, stage wise procedure for improving model accuracy [Elit08] . Boosting motivation is to combine the results of many "weak" prediction models to produce one powerful model [Trev09] . In boosting, **BRT** models are fitted to the training data iteratively, using suitable methods gradually to increase focusing on data records that incorrectly predicted. Boosting algorithms are different in method of quantification, lack of fit, and choosing settings to next iteration [Elit08] .Investigating in statistical framework of boosting leads, Jerome Friedman proposed a gradient boosting machine (GBM) algorithm. It was a simple, elegant, and highly adaptable technique for both class and numerical predictive tasks [Max13] .

Gradient Boosting Machine

Gradient boosting machine is a powerful and brilliant prediction technique that utilizesthe boosting concept for reducing error of prediction. There are two major parts in GBM- the first one is the ***loss function***, such us squared error; the second part is the ***weak learner***, like the binary regression tree, GBM algorithm builds an additive model for minimizing value of the loss function. It starts with the best guess of the target, typically the mean of values of the target. Residual of subtracting each target value and the mean is calculated - called ***the gradient***. After that, first **BRT** is built, considering the residual as a target [Sama15].

The second regression tree in GBM may contain very different attributes and split points comparing with the first regression tree. GBM process has stage-wise style, which means existing trees are not changed during the building of the model. ***Only the target value for each data record is re-estimated at each iteration, the aim being to give the new tree its contribution***[Elit08] . Algorithm (1) summarizes the major steps of gradient boosting algorithm.

The linear combination of many binary regression trees represents the final model of GBM. The performance of building process is best if it proceeds slowly; for this reason, the contribution of each **BRT** is reduced by a learning rate which has value of less than one. After the training process has been performed, predicted values are computed, the final model as summation of all regression trees multiplied by the learning rate. The results of Developed GBM usually are much more stable and precise than a single **BRT** model [Elit08]

Algorithm 1: Developed Gradient Boosting Machine

Input: *D: training data, Tm: maximum number of trees, Sk: learning rate, Mn: number of terminal nodes, Ms: number of data records in terminal node, N: number of data records in D, y: index of target.*

Output: *Prediction boosting model.*

Set: *Fx: array for predicted values of training data rows.*
Step1*: Find the initial guess for all data records in D as follow:*

1.1 For all data records in D, find mean of target values Mean(Y).

1.2 For i=0 to N-1 find residual between target value and mean:

$$Fx[i] = Mean(Y)$$

Step2*: For j=1 to Tm, build boosted model as following:*

2.1 Residual [0,i] = D [y,i] – Fx[i]

2.2 Build binary regression tree T based on D, Mn, and Ms with considering Residual of previous iteration as a target and ***based on correlation****.*

2.3 For each terminal node Tn in T, find mean of target values of data records in Tn: Mean (YTm)

2.4 For i=0 to N-1, update prediction value for record i :

$$Fx[i] = Fx[i] \times (Sk \times Mean(YTm(i)))$$

Step3*: Return boosted trees model with combination function Fx for training data records prediction.*
End

The Proposed Method

In this work, a Modern Prediction Model for HealthCare Problem (MPM-HCP) is presented to solve healthcare problems by replacing the split criteria of a boosted regression tree with a correlation measure, then evaluating the resulting model by many error measures based on real and complex dataset to get lowest prediction error and less time. MPM-HCP has three major stages: (i) preprocessing stage (i.e., dataset is made more suitable for the next stage), (ii) building prediction model using improved boosted regression, (iii) evaluation of the results of the previous stages .Those stages are illustrated in Figure (2)and summarized in Algorithm (2).

Preprocessing Stage

Real datasets like health care dataset usually have some inappropriate characteristics that need preprocessing before usingthem. Preprocessing may deal with missing values, noisy data, features selection, normalization, and binarization and data reduction [Jiaw06] . MPM-HCP performs preprocessing on healthcare dataset as follow:

Data Binarization:The main utility of prepressing is preparing data for mining in proper form. MPM-HCP uses correlation measure in the next stage to building prediction model and this require transforming categorical attributes of healthcare dataset to numerical attributes. Binarization includes expanding original attributes by convert each categorical value to new attribute. ***Features Construction:*** New sets of features are constructed to extend the original features set which improves error of prediction. There are two reasons for doing this step: (1) high prediction error when using the original features set; and (2) aggregating all claims of a specific year to be a single record requires separating the subcategories to be new features. Features construction performs by calculating statistical information such as the maximum value, the minimum value, mean and standard deviation, and retrieve sub features by converting each categorical value in the original feature to a newly separated feature. The values of new feature sets will be counter of the number of original categorical values in all claims in a year.

Data Rows Aggregation: All data rows, which means claims on healthcare for every patient at one year, are aggregated to be a single row. Structure of healthcare dataset consists of target table which contains the number of days in the hospital and related with other medical and personal information by patient identifier. This structure is concerning the main challenge of finding the number of hospitalization days for a next year, which enforced us to make this aggregation.

Features Construction: New sets of features are constructed to extend the original features set which improves error of prediction. There are two reasons for doing this step: (1) high prediction error when using the original features set; and (2) aggregating all claims of a specific year to be a single record requires separating the subcategories to be new features. Features construction performs by calculating statistical information such as the maximum value, the minimum value, mean and standard deviation, and retrieve sub features by converting each categorical value in the original feature to a newly separated feature. The values of new feature sets will be counter of the number of original categorical values in all claims in a year.

Data Rows Aggregation: All data rows, which means claims on healthcare for every patient at one year, are aggregated to be a single row. Structure of healthcare dataset consists of target table which contains the number of days in the hospital and related with other medical and personal information by patient identifier. This structure is concerning the main challenge of finding the number of hospitalization days for a next year, which enforced us to make this aggregation.

Preprocessing stage ends with combined all personal, hospitalization, drugs and labs information in tables of healthcare dataset together in one table to reduce the complexity of training processing which can be more efficient than separated tables.

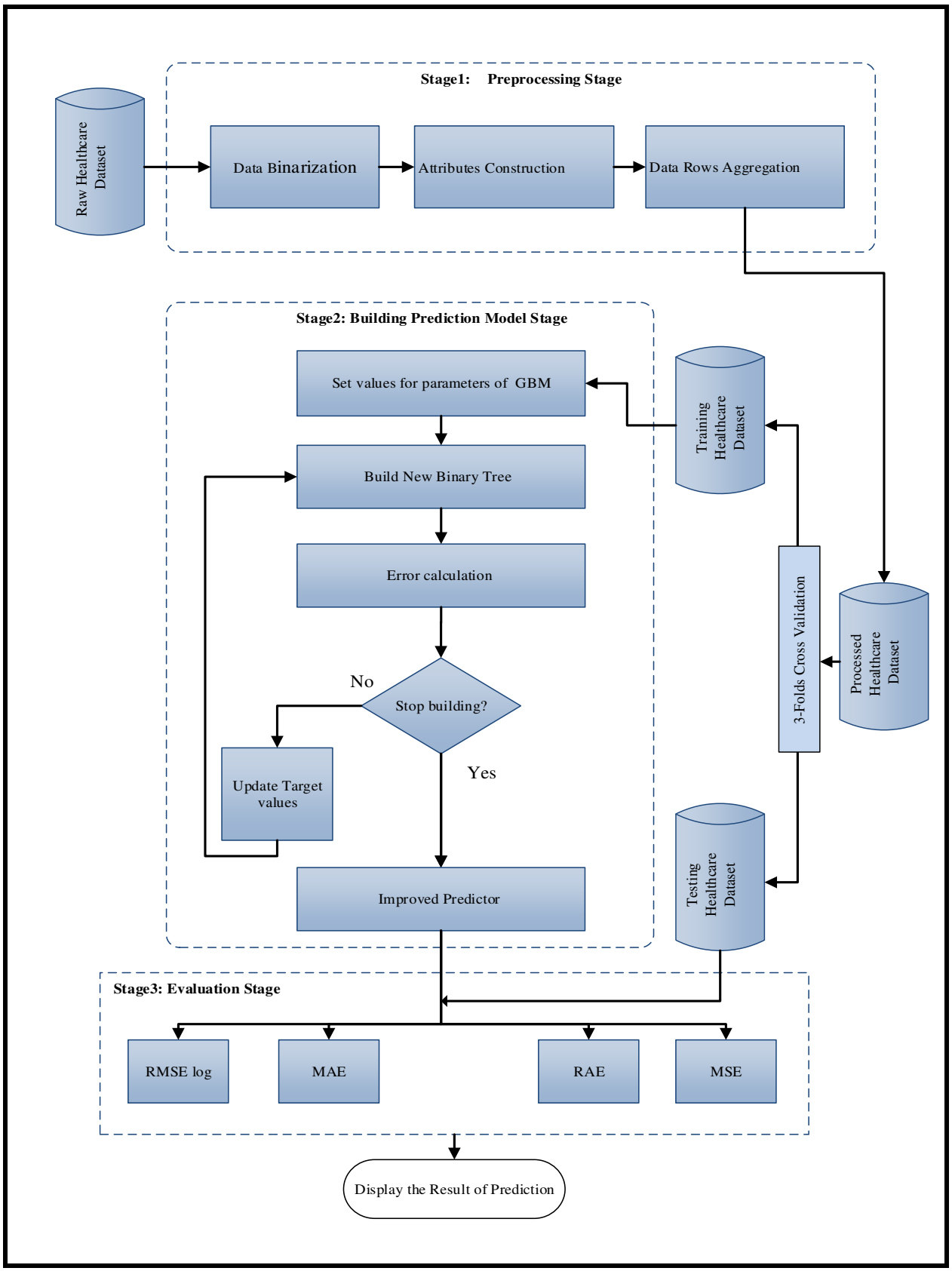
Building Prediction Model Stage

This stage could be considered the core of our MPM-HCP. It starts with detection of some important parameters which are needed for boosted regression. Then, in each iteration, a binary tree is built to reduce the error of a previous one until stop condition are satisfied. MPM-HCP model replaces the criteria for choosing best splitter feature inside binary tree procedure with more efficient criteria that is the correlation. The steps of this stage are clarified as follows:

Parameters detection: The first step in building reliable prediction models is to choose the parameters of the algorithm carefully by the user [Elit08] . Gradient Boosted Regression has four main parameters which should be selected:

- Maximum number of trees in the model that control the execution of the algorithm.
- Maximum number of terminal nodes in every single binary tree that controls the number of rules.
- Minimum number of samples in each terminal node that effect on the coverage of the rule of that node.
- Shrinkage that represents the learning rate in the training process.

Figure 2: Block Diagram of MPM-HCP



Algorithm 2: MPM-HCP

Input*: T Healthcare Dataset, Tmax: maximum number of trees Tnmax: maximum number of terminal nodes, Smin: minimum number of samples in terminal node, RC: training row count, Shr: shrinkage.*

Output: *Tmodel: regression trees model, Number of days in a hospital for each patient in next year.*

Step 1*: Pre-Processing Raw Healthcare Dataset.*

Step 2*: Split Processed Healthcare Dataset based on Cross Validation into Training dataset Tr and Testing Dataset Ts.*

Step 3: *Building Developed Boosting regression model using Algorithm (1) and retrieve Tmodel.*

Step 4: *Testing Tmodel on Test dataset: Testing_Trees_model (Tmodel ,Ts , Shr)*

End

Build Developed Binary Regression Tree: While the current number of trees in the model is still less than the maximum number of trees, a new binary tree is created. Each tree works on the same set of features, except the values of the target of prediction would be updated by prediction of previous trees, and it must be minimized [Ali12a]. The steps of building new regression tree are explained in Algorithm (3) as follows:

Best Splitter Feature Choosing: The most important step in binary tree building is to choose a splitter feature that has more relation to the target according to specific criteria. Binary tree for a classification problem depends on information gain, gain ratio and Gini index for choosing best feature, while regression problem uses sum of squared error SSE and standard deviation. MPM-HCP developed this important step by using the correlation between each feature and the target as an indicator of feature quality for splitting operation, while the correlation is used usually for feature selection in pre-processing. Calculation of the correlation value is clarified in step (1) of Algorithm (3).

Best Split Point Choosing: After the best feature is chosen, the best split points from all possible points should be selected. The typical method is to try all possible split points and evaluate the quality of each point. This operation is a costly computational task because the effort increases as the values of feature increases. For example, with acontinuous feature that has k values, there are k-1 possible split points. MPM-HCP usesa simple and efficient method to choose the best split point, which depends on the fraction of squared summation of target values dividing by number of data row for both right and left child node [Luis99] . Equation 1 explainsthe calculation of split error.

$$Err(sp) = \frac{Sum_L{}^2}{N_{tL}} + \frac{Sum_R{}^2}{N_{tR}} \quad (6)$$

Where, SumL: summation of target values for left child node; SumR: summation of target values for right child node, NtL: number of data rows in left child, NtR: number of target values on right child node.

Algorithm 3 : Develop Regression Tree Building

Input :*Tr Healthcare Training Dataset, Tnmax: maximum number of terminal nodes, Smin: minimum number of samples in terminal node, Rc: Training rows count, L: level_index, Lmax: maximum level index, Rcc: Current rows count .*

Output: *RegressionBinary Tree.*

Step 1: *If the number of the terminal nodes less than Tnmax, do the following:*

- *If L>= LmaxorRcc<= Smin then, do the following :*
  - *Create Terminal Leaf Node Lf*
  - *Lf.predected_value = mean (Trt)*
  - *Return Lf*

Else

- *Create internal node INode*
- *For each feature xj in features set, find correlation wj between xj and target according to the equation:*

$$wj = \frac{\sum xjY - \frac{\sum xj \sum Y}{N}}{\sqrt{\left(\sum xj^2 - \frac{(\sum xj)^2}{N}\right)\left(\sum Y^2 - \frac{(\sum Y)^2}{N}\right)}}$$

- *Choose best feature bf with highest correlation value as splitter feature for node INode .*
- *Choose splitting point sp on bf categories which minimize the error according to the equation :*

$$Err(sp) = \frac{Sum_L{}^2}{N_{tL}} + \frac{Sum_R{}^2}{N_{tR}}$$

Where $Sum_L = \sum_{DtL} yi, Sum_R = \sum_{DtR} yi$

- *Split Data Tr based on sp into two parts : TrL , TrR.*
- *INode.feaure_splitter =bf .*
- *INode.splite_point = sp .*
- *INode.Left_child= Develop Regression Tree Building (TrL , Tnmax, Smin, Rc, L+1)*
- *INode.Right_child= Develop Regression Tree Building (TrR , Tnmax, Smin, Rc,L+1)*

End if

End if

Step2: *return regression tree T.*

End

Data Splitting: In this stage, current data is divided into two parts according to the condition of the best split point detected in section B.2. The data rows that have value of splitter feature less than the split point become the data of new left child node, and those that have value equal or larger than the split point become the data of new right child node.

Prediction Value in Terminal Nodes: Each path from the root of a tree to a terminal node is a rule that consists of many conditions. The rule should lead to predicate a specific value for the target of prediction. In the terminal node, mostly there is more than one data row, and each one has a target value. The prediction value of a particular node simply could be the mean of all values of data rows of such node.

Training Error Calculation:

Gradient Boosted Regression depends on the concept of minimizing gradient of error. The error of every data row is calculated by subtracting the original target value of that data row from the predicted value of it. Average training error for a specific tree could be found by dividing the summation of the error of all data rows by the number of them.

Target Values Updating:

The value of the target in each data row is updating by adding the prediction of the new treemultiplying with shrinkage rate. This step is considered the most important in Gradient Boosted Regression, and it is differentiated in this technique from other prediction techniques. It could be explained as looking for the best point which is as close to all target values as possible.

Evaluation Model Stage

In this stage, all tree models built by Gradient Boosted Regression are evaluated based on test dataset that have never seen before. RMSE, RMSE Log, MSE, RAE and MAE measures are used in this step to evaluate the prediction error of every tree,and then find the total error for combination of all tree models.

Experimental Results

MPM-HCP described in the previous section has been tested with different parameters values, and results will be reviewed in this section. A real and a huge dataset have been used as an implementation example to discover the behavior of that model. MPM-HCP has many stages that deal with the dataset to get required prediction values; the experimental result of each stage is shown and explained. The deficiency of real and reliable healthcare data in Iraqi health government institutions led us to utilize a foreign dataset in this implementation, which has extensive and trusted healthcare data that are considered resistant-proof for the ability of MPM-HCP to handle difficult and complicated dataset.It is taken from URL[1].

Description of Healthcare dataset

Heritage Provider Network (HPN) provided healthcare dataset for researchers and data miners aiming to reduce the costs by predicting hospitalization of patients. It contains data of more than (113,000) patients with nine tables linked by patient identifier and described in Table (2). Each patient has one or more claims during a year. The claim data included information about the condition causing hospitalization and the medical procedure required for patient treatment. Features of claims table are explained in Table (3).

The values of days in a hospital are between 0 and 15; duration of more than 15 days is rounded to 15 in this dataset. Those integer and continuous values for target made the prediction process considered a regression problem while grouping those values in multiple ranges could make the process a multi-classification problem by grouping target values in specific ranges. MPM-HCP treats this dataset as a regression model to get more precise and reliable results that can reduce the total error of prediction and prevent wasting unnecessary costs.

[1]https://www.heritagehealthprize.com

Table 2: Description of Healthcare dataset

Table Name	Description
Claims	The main table of healthcare dataset and contains information of medical case and patient.
Members	Contain personal information such as age.
DaysInHospital_Y2	Contain the number of days that spend by patient in the second year, it used in the training stage.
DaysInHospital_Y3	Contain the number of days that spend by patient in the third year, it used in the training stage.
DrugCount	Number of drugs that consumed by patient.
LabCount	Number of Labs tests that consumed by patient.
PrimaryConditionGroup	Describe of Primary Condition Group coding.
LookupProcedureGroup	Describe of medical Procedure Group coding.

MPM-HCP implementation on HPN Dataset

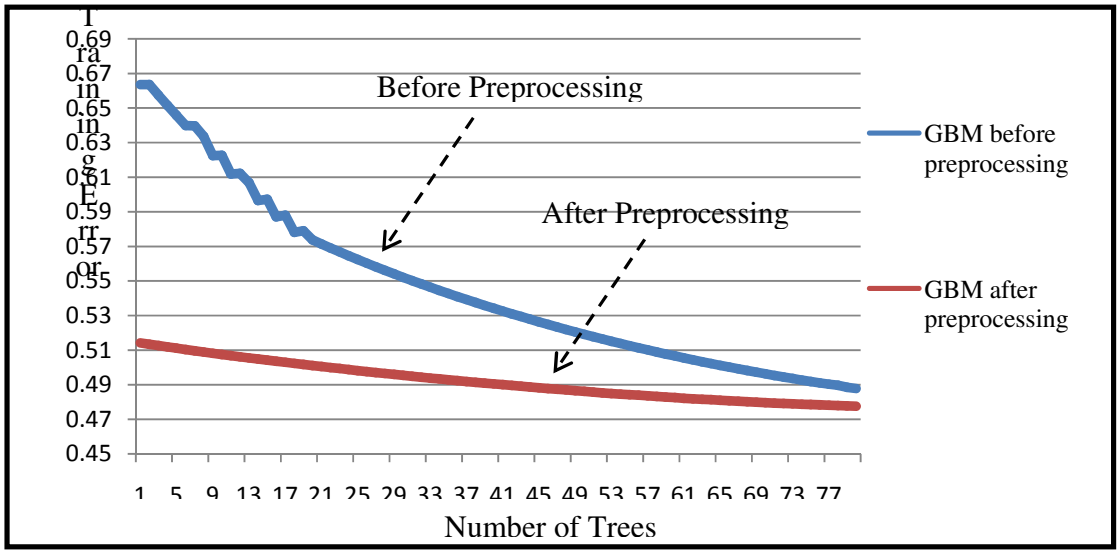
All the steps of the preprocessing stage of MPM-HCP are applied on HPN dataset because it has many tables with different data types, which make the training process more complicated. For any other dataset, it is not necessary to always apply all steps of preprocessing because it depends on the nature of data and the structure of tables in such dataset.

The benefit of the preprocessing stage is apparent in Figure (3); the difference between training error before and after applying steps of preprocessing is distinguishable. GBM had 0.668 error measure when applying on HPN dataset without preprocessing; the error reduced after 80 iterations to reach 0.49 while preprocessed dataset produced error starting from 0.515 and ending with 0.48 according to RMSE log measure.

Table 3: Description of Claims table

Feature Name	Description
MemberID	Patient Identifier that link all tables.
ProviderID	Provider Identifier that provide healthcare financial aid.
Vendor	Vendor Identifier.
PCP	Primary care physician Identifier.
Year	The year of claim Y1; Y2; Y3.
Specialty	General specialty of patient's condition.
PlaceSvc	General place of healthcare service.
PayDelay	Number of days delays between the date of service and the date of payment.
LengthOfStay	Number of days of hospitalization that mean the delay between discharge date and admission date.
DSFS	Days since the first claim.
PrimaryConditionGroup	Code of medical diagnostic which describe in PrimaryConditionGroup table.
CharlsonIndex	A value of affect diseases.
ProcedureGroup	Code of procedure diagnostic which describe in ProcedureGroup table.
SupLOS	Binary value of suppression of claim.

Figure 3: Comparison Error of GBM before and after preprocessing



All claims for a year of a specific patient are aggregated, which make the process of choosing the best feature as a splitter in the building tree step more efficient. This aggregation reduces the overall complexity of the prediction problem with respect to the knowledge in HPN dataset. The number of data rows of claims table would reduce from more than one million to 147,473 records after aggregating. Before building the prediction model, we need to combine all tables of HPN dataset, which is explained in Table (2), into one table that contain 136 features, including the target of prediction and 147,473 data rows. These datasets are considered as an input to the training process to build prediction model. HPN dataset is used to build a boosted set of the regression tree in which the training error should be optimized. Dataset has been split according to the Cross Validation concept. It included three folds, in each fold, HPN dataset split into two parts; 2/3 for training stage (103231 records) and 1/3 for testing stage (44242 records). Building prediction model had two steps as follows:

Applying Parameters Values Selection:The choosing of parameters controls the behavior of the training process and affects the result of this process. Multiple values of Gradient Boosted Machine parameters have been used for HPN dataset aiming to find optimal values for them.

Another parameter that should be selected for GBM is the maximum number of terminal nodes that relate with the complexity of trees and number of rules. The popular depth of a regression tree in GBM is between 2-4, which means the maximum number of terminal nodes is in the range of 4-16 nodes.

In MPM-HCP, depth of threefour and five levels is tested and the stable results were using five levels, which means the maximum number of terminal nodes with HPN dataset was 32 nodes, considered medium complexity for dataset with 128 features.

The third parameter in GBM is the minimum number of samples in terminal nodes, which means if the number of data rows in the node is equal or less than this parameter, the node would be a terminal node and the split operation should stop. This parameter was set to 20 samples with HPN dataset that represent 0.000017 ration from the total number of data rows.

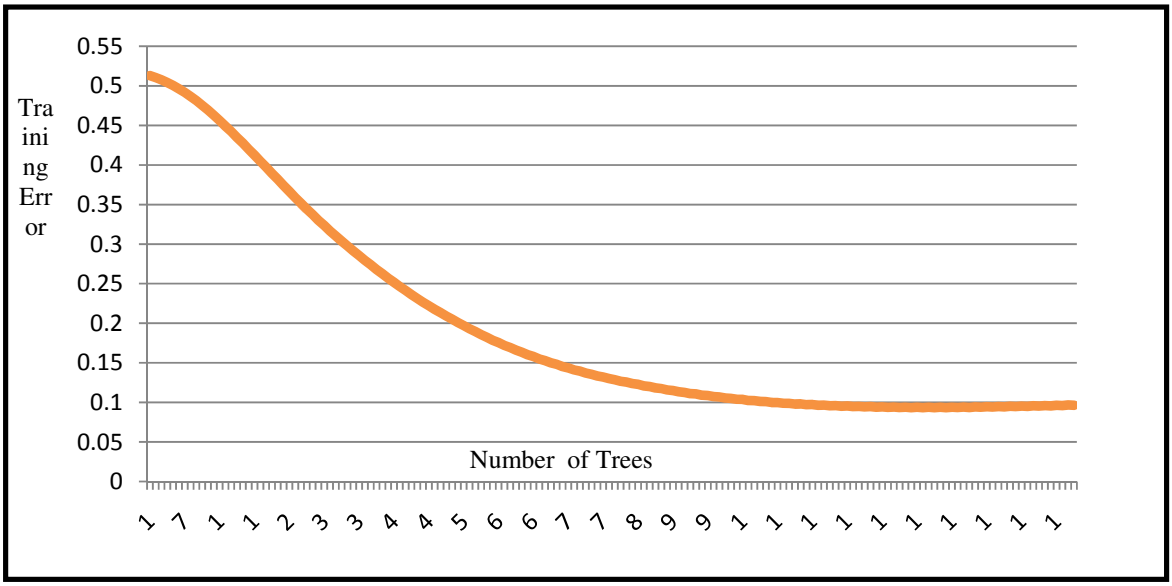
The number of trees is the most important parameter in this process that needs to be chosen carefully; with HPN dataset, the range of 80 – 160 trees is used. The effect of increasing the number of trees could be seen in Table (4) and Figure (4).

Table 4: Comparison of original GBM and MPM-HCP according to the number of trees in the training process

No. of Trees	Original GBM-STD		Original GBM-SSE		MPM-HCP	
	Best Tree	Training Error	Best Tree	Training Error	Best Tree	Training Error
80	80	0.477421	80	0.477421	80	0.12996
120	97	0.476155	97	0.476155	120	0.095122
160	97	0.476155	97	0.476155	136	0.093330

In the original GBM that used standard deviation, the training error is reduced and reached the lowest error (0.466) after 104 iterations of training; then it was increasing again. In the SSE-based GBM, the lowest training error (0.475) has been reached after 99 iterations.

Figure 4: Training Error of MPM-HCP in model of 160 trees

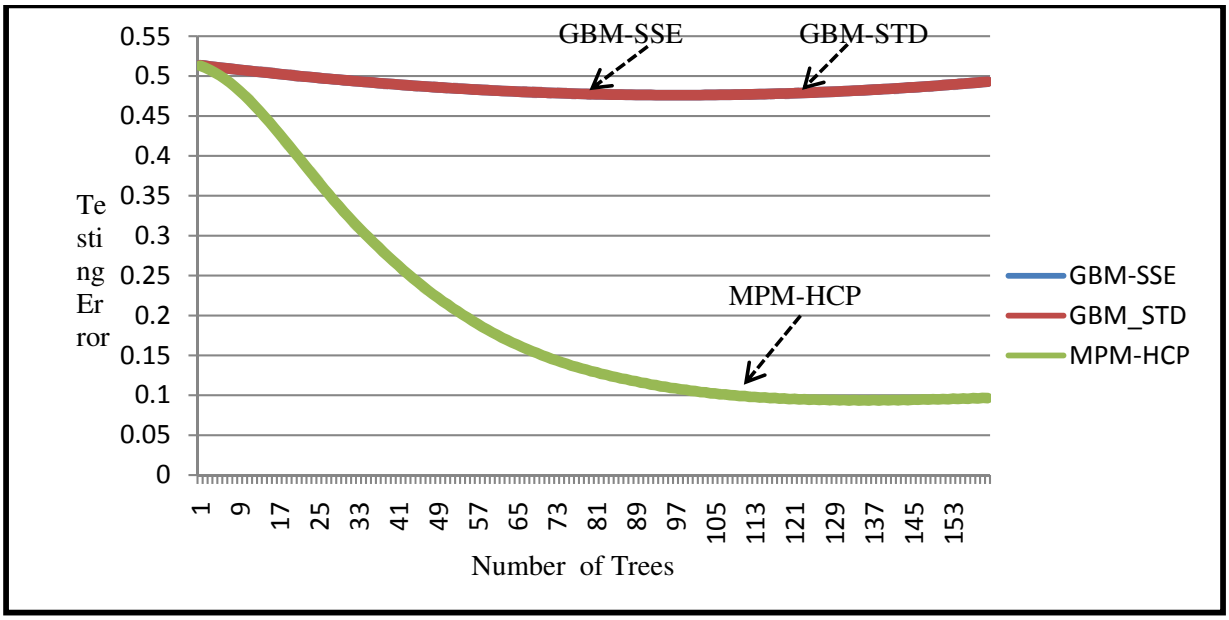


Training error of MPM-HCP was reduced gradually until it reached the lowest value (0.094) after 142 iterations, which represents an apparent improvement comparedto original GBM.

Building Binary Regression Tree Implementation:The parameters that have been set in the previous section would be used to build a binary regression tree. In the first iteration, an initial guess of target feature is used rather than building a tree; it could be simply the average of all target values. HPN dataset has more than 89% of its target values as a zero value; for this reason, the initial guess in MPM-HCP prefers to be zero. The main improvement of MPM-HCP is replacing the traditional splitting criteria of the regression tree by using a correlation that leads to improving the training error. Two observations could be seen after building 160 regression trees in both original and improved GBM:

- **First:** training error of standard deviation based GBM and SSE based GBM starts to increase after (100) tress in them both, also there was matching in the result of two measures.
- **Second:** MPM-HCP has more stable behavior and less training error from both original GBM methods, also the gradient of error doesn't increase even (160) trees were build. Both observations are shown in Figure (5).

Figure 5: Comparison Training Errors of GBM and MPM-HCP



MPM-HCP Evaluation

Reliable evaluation for data mining techniques should be make based on test data which never seen before during training stage. According to Cross validation [Alja14], In each fold HPN dataset is split into two parts in which the test data is 1/3 from original data and the rest for training, that means the number of test data rows is (44242) records. Training stage is finished with building (160) regression trees that need to be evaluate, and three folds cross validation is applied.

Evaluating based on Error

There are many possible methods to test the final model; (i) taking the final tree for testing only (ii) random choosing of trees to contribute the evaluation process (iii) use all trees which build in training process for evaluating the model and control contribution of each tree by learning rate parameter.

The first method depended on the final tree which has accumulative model from all previous trees. In this method, original GBM based on SSE and STD had (0.586) average testing error in three folds and RMSE log measure. MPM-HCP training error was (0.48) which represent the best result in the first method. Table (5) shows comparison of this method using four error measures.

Table 4.4 shows that MPM-HCP has best result in most measures with three folds. Also it shows that both GBM-SSE and GBM-STS have same result in the first method of testing.

In the second method, multiple random number is used (10, 20, 50, 100) trees without replacement. The result shows there is difference between GBM based on SSE and GBM based on STD in this method, also all of random ranges have test error worse than using all trees combination. Tables (6) show comparison among GBM and MPM-HCP in three folds of cross validation.

Third method includes testing every record on (160) tree sequentially and multiply the result of each tree by shrinkage learning rate. In this method original GBM based on SSE and STD had (0.495) training error in RMSE log measure. MPC-HCP also had lowest error (0.468) in this method as shown in Table (7).

Table 5: Comparison Testing Error of GBM and MPM-HCP in Last Tree

		MAE	RAE	MSE	RMSE Log
Fold 1	GBM-SSE	0.903645	1.167224	2.460890	0.586971
	GBM-STD	0.903645	1.167224	2.460890	0.586971
	MPM-HCP	0.519348	0.675684	2.591641	0.480892
Fold 2	GBM-SSE	0.904521	1.192644	2.481206	0.588235
	GBM-STD	0.904521	1.192644	2.481206	0.588235
	MPM-HCP	0.520267	0.677308	2.554392	0.481644
Fold 3	GBM-SSE	0.892517	1.207927	2.388095	0.584159
	GBM-STD	0.892517	1.207927	2.388095	0.584159
	MPM-HCP	0.520409	0.677494	2.554360	0.481694

Table 6: Comparison Testing Error of GBM and MPM-HCP with random choosing trees in Fold 1

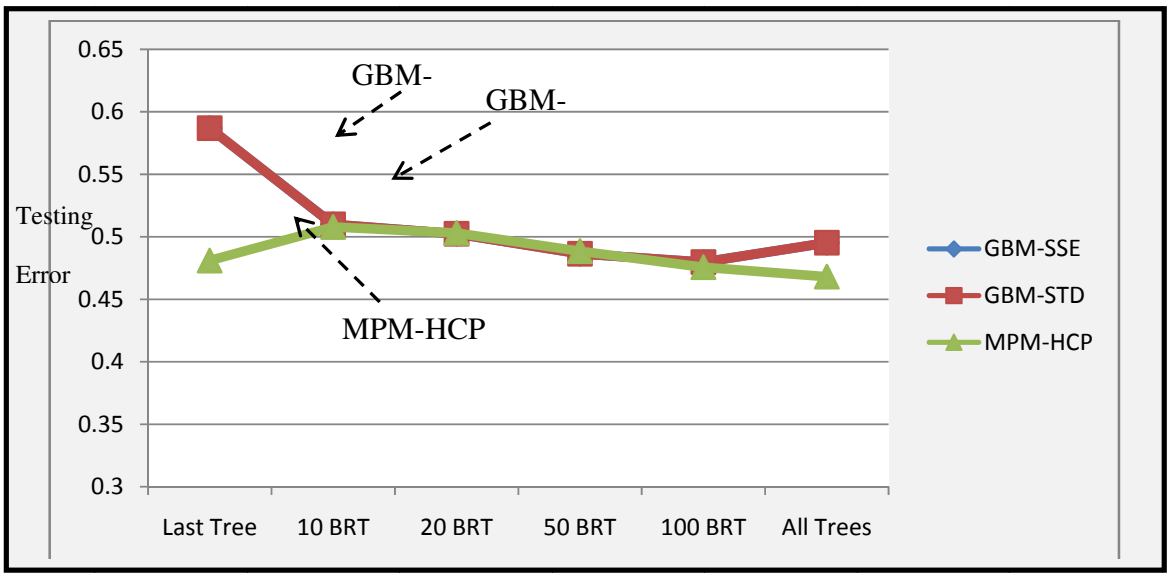
		MAE	RAE	MSE	RMSE Log
10 random trees	GBM-SSE	0.472279	0.610036	2.632628	0.509846
	GBM-STD	0.472084	0.609784	2.632940	0.509954
	MPM-HCP	0.463047	0.602436	2.684503	0.507777
20 random trees	GBM-SSE	0.486762	0.628742	2.609880	0.502378
	GBM- STD	0.486897	0.628917	2.609685	0.502317
	MPM-HCP	0.470304	0.611877	2.670683	0.503036
50 random trees	GBM-SSE	0.531399	0.686400	2.547301	0.486363
	GBM- STD	0.531270	0.686233	2.547453	0.486393
	MPM-HCP	0.496538	0.646009	2.622814	0.488493
100 random trees	GBM-SSE	0.604714	0.781099	2.468867	0.479705
	GBM- STD	0.604742	0.781136	2.468844	0.479707
	MPM-HCP	0.533072	0.693540	2.566358	0.475638

Table 7: Comparison of testing error between GBM and MPM-HCP with all trees combination

		MAE	RAE	MSE	RMSE Log
Fold 1	GBM-SSE	0.690653	0.892105	2.415575	0.495156
	GBM-STD	0.690653	0.892105	2.415575	0.495156
	MPM-HCP	0.583212	0.758773	2.498579	0.468068
Fold 2	GBM-SSE	0.685725	0.904154	2.421763	0.491485
	GBM-STD	0.685725	0.904154	2.421763	0.491485
	MPM-HCP	0.584536	0.760977	2.460593	0.468859
Fold 3	GBM-SSE	0.673076	0.910937	2.323039	0.486992
	GBM-STD	0.673076	0.910937	2.323039	0.486992
	MPM-HCP	0.584536	0.760978	2.454479	**0.468077**

The behavior of original GBM based on STD and MPM-HCP in three testing methods is illustrated in Figure 6. They show that MPM-HCP has best result (i.e. less prediction error) in the first and third testing methods in all folds of cross validation. In the second method of testing (i.e. random choosing of BTRs), MPM-HCP has same result of approximately in first fold.

Figure 6: Comparison Testing Error of GBM and MPM-HCP in first fold of cross validation



Evaluation based on Time

Time of execution considered important factor to evaluate scalability of the prediction model. Complexity of computation operations of building regression tree represents difficult challenge to have best prediction error in suitable time. The major parameters that effect training time is the number of trees in the model, number of trees 's nodes, number of Attributes and number of data rows.

MPM-HCP spend (270) minutes to build (160) trees while original GBM need more than (1500) minutes to build same trees. It means MPM-HCP enhanced training time more than 85%. Figure (7) shows comparison of training time between original GBM and MPM-HCP.

Testing time for both original GBM and MPM-HCP is less than (32) second for combination of all trees. It minimize with random choosing method depending to the number of trees. Method of testing last tree spend (6) seconds to predicate unseen testing data. Figures (8) show comparison of testing time between original GBM and MPM-HCP.

Figure 7: Comparison Training Time of GBM and MPM-HCP (in minutes)

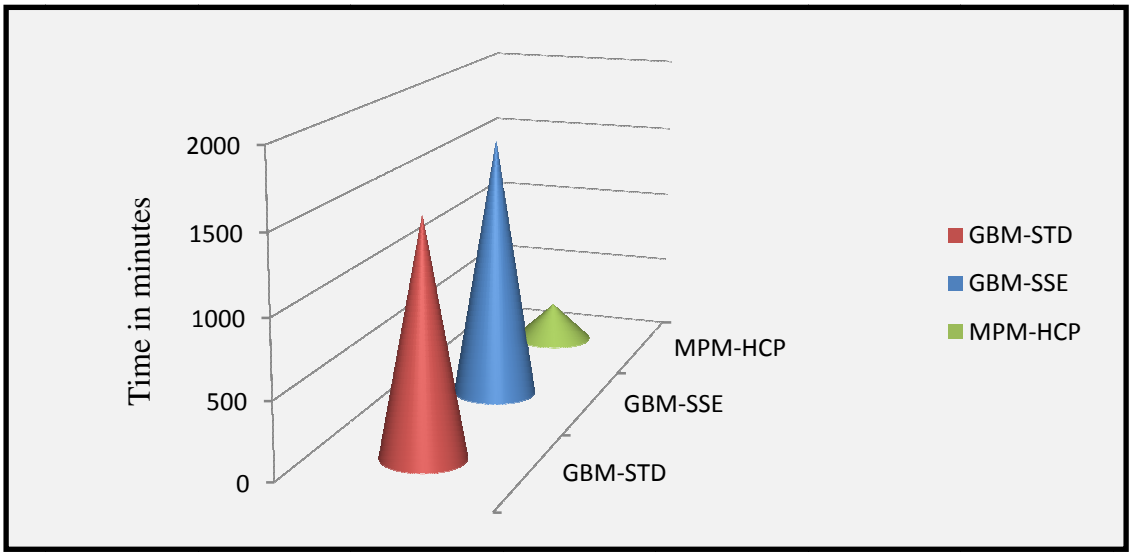
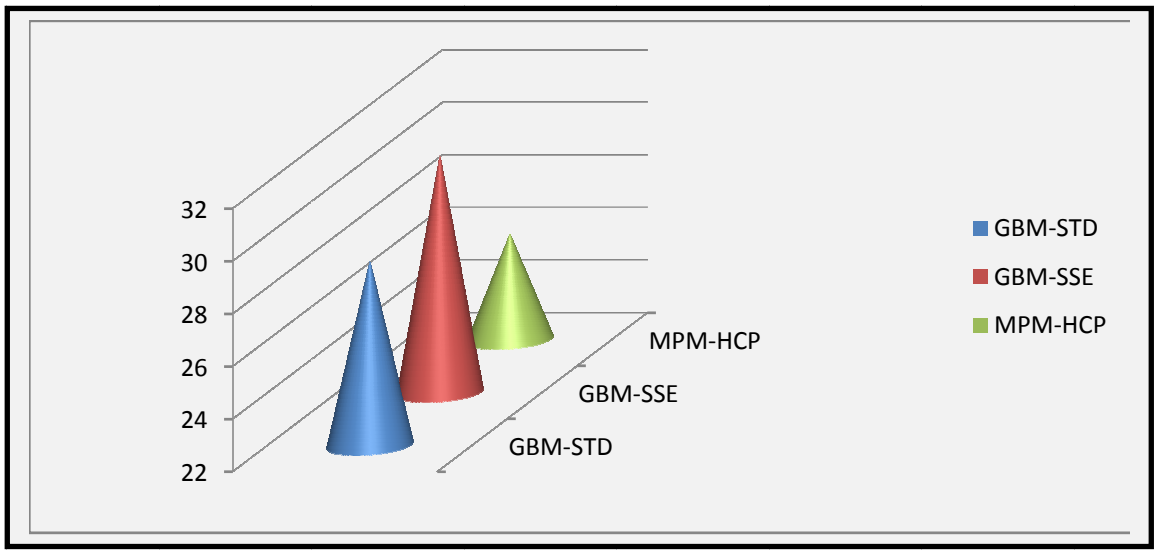


Figure 8: Comparison Testing Time of GBM based and MPM-HCP (in seconds)



Discussion

The work in the field of prediction needs knowledge in aspects of mathematics and statistics and techniques for machine learning to find suitable and efficient solutions to the requirements of this field. In this section, we discuss the most important points that have been subjected in this work.During the review of previous work in the field of prediction in health care and the review of the techniques used in the algorithm, it show that GBM has the best results, often in spite of it depending on the decision tree in the construction of the model, and that leads to three essential difficulties[Sama15]., namely:

Storage: The main problem is that the GBM uses recursion in building decision trees, and since it requires the presence of all the data in memory during training time, it consumes a large storage resource in casesof huge data. In our proposed model, the problem has been solved by using a number of methods of optimization of memory (such as using pointers, object-oriented programming, and the use of local variables and not public is deleted from memory after the end of the task).

Execution Time: A large number of data rows and the number of features in healthcare dataset need a long time to be processed using typical prediction techniques. Training process includes many splitting and comparing steps for tree building, and those steps are repeated many times to create sequences of decision trees. Required time to execute all those steps is considered a problem, and we solved it in our proposed model by using the correlation splitting criteria, which reduces comparing steps.

Parameters Detection: There are four basic parameters that must be carefully selected in GBM, which are the minimum number of samples that can be contained to terminal node, the maximum number of terminal nodes, the maximum number of trees allowed in the model, and the learning coefficient., Both the first and second parameter affect the building of the tree and decision-making, while the third and fourth parameters are related to the process of reducing the error rate in the construction of the integrated model. The process of choosing values for such parameters is critical and effective in the results of training and therefore must be selected very carefully. In our proposed model, we used the principle of the experiment and the error in the choice of parameters values, and we did not use any particular strategy.

Most techniques of sampling are used as a part of preprocessing huge data, and that leads to creating unrealistic results, especially with healthcare data. In our proposed model MPM-HCP, we didn't use any sampling technique to get precise and realistic results that can be relied upon by the users of this model.

Conclusion

MPM-HCP is a domain dependent prediction model which attempts to solve the problem of healthcare. MPM-HCP uses the correlation between features and the target as new splitting criteria to choose the best feature during the building of a regression tree. It has significant behavior in terms of prediction error and execution time. In comparison with tradition gradient boosting trees, the proposed system gave clarity improvement in the training process in which the training error is closer to zero and the testing error is better than SSE and standard deviation-based GBM. Training time is also reduced more than 85%, which gives MPM-HCP more scalability to deal with large and complex healthcare datasets.The experimental result of MPM-HCP implementation showed that there were three attributes frequent in binary regression trees building. Those attributes were **gender of patient, number of claims to admit hospital, and the medical procedure group**, which means those attributes is more correlated with the target of prediction (i.e. number of hospitalization days). We suggestions for future works using intelligent algorithm for choosing suitable correlation type of splitting criteria by analysis the data. Using optimization techniques in MPM-HCP to select suitable values for Developed GBM parameters thereby reduce required time of training process.Build a mathematical regression model by using most frequent attributes of final boosted model.Utilizing MPM-HCP to predicate other target like specialty of future medical statues or required medical procedure, the send early notification to avoid aggravation of health suddenly.Develop other prediction and classification techniques which depend on binary regression tree concept such as random forest to use correlation splitting criteria.

References

[1] [Alja14] Al-Janabi, S.; Patel, A.; Fatlawi, H.; Kalajdzic, K.; Al Shourbaji, I., (2014), "Empirical rapid and accurate prediction model for data mining tasks in cloud computing environments," *Technology, Communication and Knowledge (ICTCK), 2014International Congress on* , vol., no., pp.1,8, 26-27 Nov. 2014 doi:10.1109/ICTCK.2014.7033495
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7033495&isnumber=7033487

[2] [Ali12a] Ali, S.H., (2012), "Miner for OACCR: Case of medical data analysis in knowledge discovery," IEEE, *Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012 6th International Conference on* , vol., no., pp.962,975, 21-24 March 2012
doi: 10.1109/SETIT.2012.6482043,

[3] URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6482043&isnumber=6481878

[4] [Ali12b] Ali, S.H., (2012), "A novel tool (FP-KC) for handle the three main dimensions reduction and association rule mining," IEEE, *Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012 6th International Conference on* , vol., no., pp.951,961, 21-24 March 2012
doi: 10.1109/SETIT.2012.6482042,
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6482042&isnumber=6481878

[5] [Duan11] L. Duana, W.N. Streeta and E. Xu (2011) Healthcare information systems: data mining methods in the creation of a clinical recommender system. Enterprise Information Systems, Vol. 5, No. 2

[6] [Elit08] J. Elith, J. R. Leathwick, and T. Hastie (2008) A working guide to boosted regression trees. Journal of Animal Ecology, vol (77), p.p 802–813

[7] [Hadi12] Hadi Yazdi and Nima Moghaddami (2012) Multi Branch Decision Tree: A New Splitting Criterion. International Journal of Advanced Science and Technology, Vol. 45, August, 2012

[8] [Hast09] Hastie, T.; Tibshirani, R.; Friedman, J. H, (2009) The Elements of Statistical Learning, 2nd Edition. New York: Springer, pp. 337–384, ISBN 0-387-84857-6

[9] [Ian 05] Ian H. Witten andEibe Frank (2005) Data Mining: Practical Machine Learning Tools and Techniques, 2nd Edition. ISBN-13: 978-0120884070, Morgan Kaufmann

[10] [Ian11] Ian H. Witten, Eibe Frank, and Mark A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques, 3nd Edition, ISBN-13: 978-0123748560, Morgan Kaufmann

[11] [Jiaw06] Jiawei Han and Micheline Kamber (2006) Data Mining: Concepts and Techniques, 2nd edition, ISBN 10: 1-55860-901-6, Elsevier

[12] [Jiaw13] Jiawei Han and Micheline Kamber, (2013) Data Mining: Concepts and Techniques, 3th edition. ISBN 978-0-12-381479-1, Elsevier

[13] [Jufe13] Jufen Zhang, Kevin M. Goode, Alan Rigby, Aggie H.M.M. Balk and John G. Cleland (2013) Identifying patients at risk of death or hospitalization due to worsening heart failure using decision tree analysis: Evidence from the Trans-European Network-Home-Care Management System (TEN-HMS) Study. International Journal of Cardiology, vol (163), p.p 149-156, Elsevier

[14] [Krzy07] Krzysztof Cios, Witold Pedrycz, Roman Swiniarski, and Lukasz Kurgan (2007) Data Mining A Knowledge Discovery Approach. Springer, ISBN-13: 978-0-387-33333-5

[15] [Luis99] Luis Torgo (1999) Inductive Learning of Tree-based Regression Models. Ph.D. Thesis. University of Porto

[16] [Max13] Max Kuhn and Kjell Johnson (2013) Applied Predictive Modeling. ISBN 978-1-4614-6849-3. Springer

[17] [Nann] Nannan He (2014) Data Mining for Improving Health-Care Resource Deployment. Master thesis. University of California Santa Cruz

[18] [Rash11] Rashedur M., Fazle Rabbi (2011) Using and comparing different decision tree classification techniques for mining ICDDR, B Hospital Surveillance data.Expert Systems with Applications vol (38), p.p 11421–11436.

[19] [Raym13] Raymond L. Goldensteel, Karen Goldensteel (2013) U.S. Healthcare system ,7th Edition. ISBN: 978–0-8261–0930-9, Springer Publishing

[20] [Robe09] Robert Nisbet, Gary Miner, and John Elder (2009) Handbook of Statistical Analysis and Data Mining Applications. ISBN-13: 978-0123747655, Academic Press,

[21] [Step11] Stéphane Tufféry (2011) Data Mining and Statistics for Decision Making, First Edition. ISBN: 978-0-470-68829-8, John Wiley & Sons, Ltd

[22] [Sama15]Samaher Al_Janabi (2015) A Novel Agent-DKGBM Predictor for Business Intelligence and Analytics toward Enterprise Data Discovery, Journal of Babylon University/Pure and Applied Sciences/ No.(2)/ Vol.(23): PP 482-507.

[23] [Trev09] Trevor Hastie, Robert Tibshirani, and Jerome Friedman (2009) The Elements of Statistical Learning, 2nd Edition. ISBN 0-387-84857-6, New York: Springer, pp. 337–384

[24] [Xian11] Xiang Peng, Wentao and Wu Jia Xu (2011) Leveraging Machine Learning in Improving Healthcare.Association for the Advancement of Artificial Intelligence