IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 15, Issue 1 Ser. II (Jan – Feb 2019), PP 50-68
www.iosrjournals.org

Decisions Tree Building for Different Types Data

Ahmed Mohamed Mohamed Elsayed
Al-Obour High Institute For Management & Informatics
Department of Basic Science
Kilo 21 Cairo-Belbies Road, P.O. Box 27 Obour City, Egypt

Abstract: *Data mining is an important method for analysis the huge data that contained many types of variables. Data mining is widely used in many fields such as commerce, marketing, medicine, ...,etc. It is also an important field with contributions from many sciences. The important techniques of data mining include the classification tree, the regression tree and the clustering. In this paper, we will be interested with the two first techniques. The third technique is presented in another paper. These two techniques will be applied on the practical clinical data, containing different types variables, such as "nominal", "ordered", "numeric", "binary", "repeated measures". The different packages of R program will be used for classification and regression, we are concentrating on building trees. The packages "party", "rpart" and "randomForest" are used for this reason. The results of data analysis are demonstrated and compared, for different errors "R-square", "xstd", "xerror", indicated the path of each error during the process of building tree, taking in account the relationship between these errors. The impact variables and the optimal size of trees are constructed with complexity parameter with minimum "xerror". The pruned classification and regression tree, corresponding to a complexity parameter, are displayed. The comparisons of results that obtained from different techniques are investigated.*

Keywords: *Data mining; Classification; Regression; Trees; Cross-validation Errors.*

Date of Submission: 01-03-2019 Date of acceptance: 18-03-2019

I. Introduction

Data mining is the process to deduct an important knowledge from large data. Data mining is widely used in many fields, such as commerce, finance, communication and social media. It is constructed from many areas, such as statistics, machine learning, and information retrieval. The detailed information about data mining techniques can be found in the books [1, 2].

Recursive partitioning is a fundamental tool in data mining. It helps us to search the structure of a data set for predicting categorical or continuous outcomes. The classification and regression trees as methods of data mining can be built through many packages in R program such as: A laboratory for recursive partitioning "party" package [3] provides nonparametric regression trees for nominal, ordinal, numeric, censored, and multivariate responses. We can create a regression or classification tree using function "ctree" in "party" package.

The "rpart" package [4] is used also to build a decision tree. Function "rpart" is used for this reason, and the tree with the minimum error is selected.

The cross validation involves dividing a sample into complementary subsets: training set and test set. The goal of cross-validation is to test the model's ability to predict new data. In the classification trees, more levels in the tree mean that it has lower classification error. The complexity parameter (CP) is the amount of improved relative error when splitting the node. For example, splitting the original root node decreases the relative error from 1.0 to 0.5, so the CP of the root node is 0.5. The default limit of CP for deciding a split is 0.01, so the tree building stopped there.

There are many types of errors arise such as "xstd" in "rpart" package represents the variation in prediction across the validation samples. The "relative error" represents $(1-R^2)$ root mean squared error. It is estimated with the training data and thus it decreases as the tree increases. The "xerror" is related to the predicted residual error sum of squares.

A rule is to choose the lowest level of splits where (relative error + xstd < xerror).

The package "randomForest" [5] is used to build a predictive model. Random forests improve predictive accuracy by generating a huge number of bootstrapped trees, and deciding a final predicted outcome by combining the results. The margin of a data point is equal: Proportion of votes for the correct class - Maximum proportion of votes for other classes.

In this paper, we will present the decisions rules using the building of the classification and regression trees using different packages of R program such as "party", "rpart" and "randomForest" packages. These packages are applying on the clinical data that contained different types of variables. This reset of this paper will be organized as follow: Section II presents a building the classification tree with applications. Section III presents a building of the regression tree with applications. Section IV presents results and discussion. Finally, Section V presents some conclusions.

II. Building Classification Tree

In this section, we will present the classification tree, using "party", "rpart", "randomForest" packages in R program, applying on the respiratory clinical data [6]. These data include 444 observations (111 patients repeated with 4 visits) and 8 variables: id represents the repetitions of patient, sex(Male, Female), center(1,2), age of patient, treatment(Active, Placebo), outcome(good=1,poor=0), baseline reparatory status (good=1, poor=0) and visit (1,2,3,4). The data are containing different types variables: "id", "visit" and "center" represent the count variables, "sex" and "treatment" represent categorical variables, and "outcome" and "baseline" represent the binary variables. We start with building decisions tree for classification with package "party". Next we will use the packages "rpart" and "randomForest" respectively.

II.1 Using "party" Package

The type of tree created, will depend on the dependent categorical variable, Treatment (Active=A, Placebo=P). The explanatory variables are the other reset seven variables (id, sex, center, age, treatment, outcome, baseline and visit). Tree growth is based on statistical stopping rules, so pruning should not be required. The function "ctree" will be used to build the classification tree. The next two subsections; one of them presents the building of classification without training and set samples, and the other one presents the process with them.

II.1.1 Without (Training and Test) Samples

Applying the "party" package on the data, we have:
Conditional inference tree with 3 terminal nodes:

Number of observations: 444.
Node (1) outcome <= 0.
Node (3) sex == {M}, weights = 150.
Node (4) sex == {F}, weights = 46.
Node (5) outcome > 0, weights = 248.

Misclassifications:

Treatment	A	P	Total
A	147	101	248
P	69	127	196

Figures (1-1, 1-2) show that: For node (1) we have 444 obs. at the left side of tree this is happened when the outcome ≤ 0, then we have 196 obs. belong to "sex" variable, this splits 150 directed to the node (3) and labeled as a male (0.433 of them belong A class and 0.567 of them belong to P class). The other 46 obs. direct to the node (4) as a female (0.087 of them belong to A and 0.913 of them belong to P). At the right side when outcome>0 we have 248 obs. directed to the node (5) (0.593 of them belong to A and 0.407 of them belong to P).

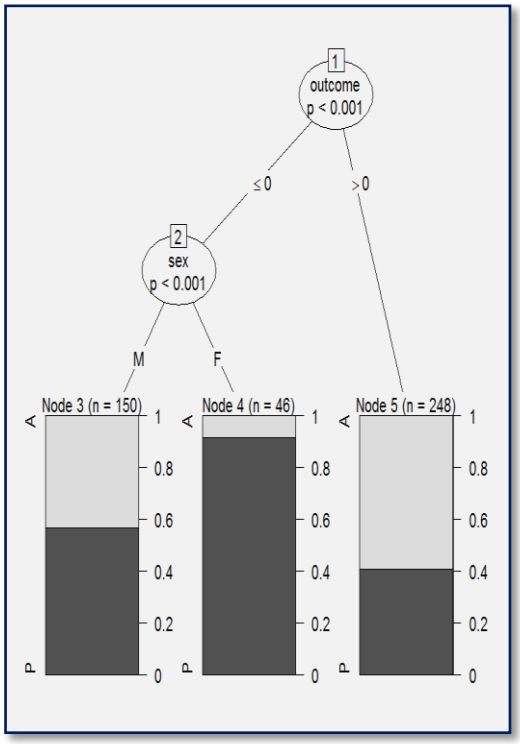

Figure 1—1

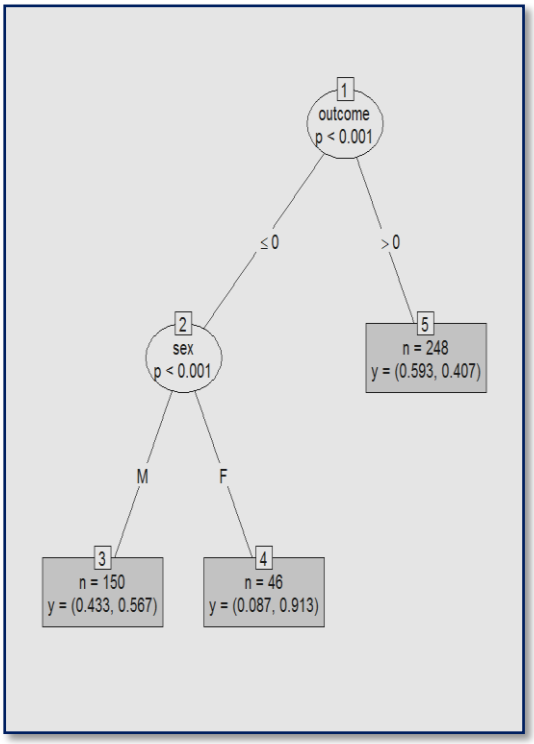

Figure 1—2

II.1.2 With (Training and Test) Samples

Before modeling the data we will split it into two subsets (samples); training (50%) and test (50%). The random seed is used to make the results unchangeable.

R-results are arisen from the training sample as:

Conditional inference tree with 2 terminal nodes:

Number of observations: 223.

Node (1) outcome <= 0.

Node (2) weights = 90.

Node (3) outcome > 0, weights = 133.

Misclassifications:

Treatment	A	P	Total
A	23	110	133
P	16	74	90

Figures (2-1, 2-2) show that: For node (1) we have 223 obs. at the left side of tree this is happened when the outcome ≤ 0, then we have 90 obs. directed to the node (2) (0.333 of them belong to A class and 0.667 of them belong to P class). At the right side when outcome > 0 the other 133 obs. directed to the node (3) (0.602 of them belong to A and 0.398 of them belong to P).

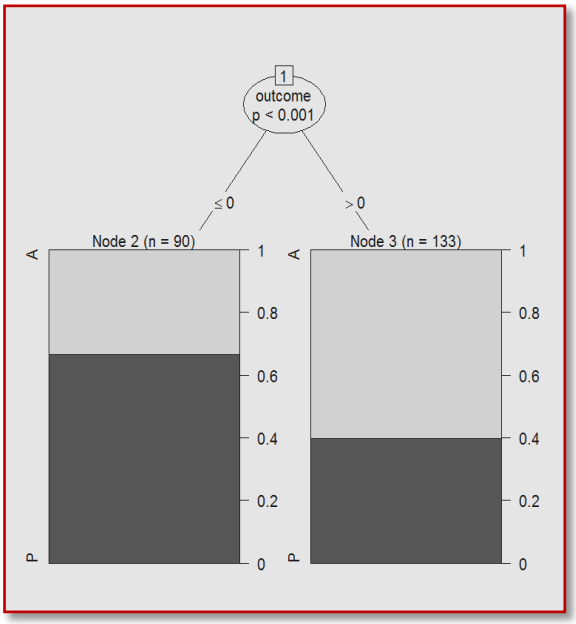

Figure 2-1

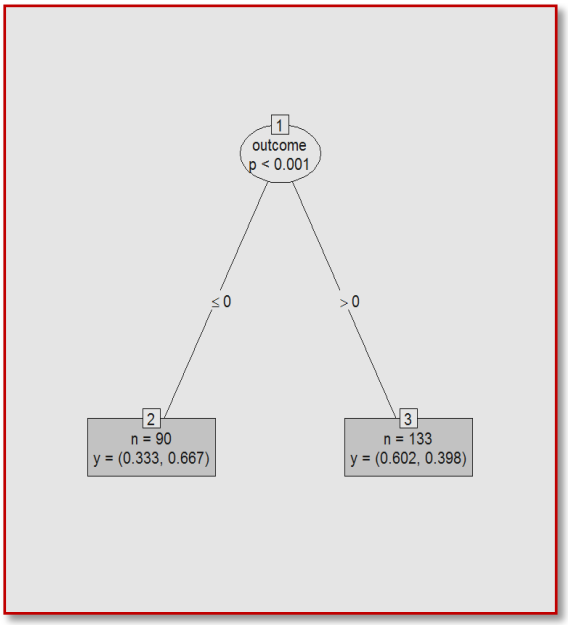

Figure 2-2

If the levels of a categorical variable (treatment) in the test sample are different of that in the training sample, we make a prediction from the test data. Then we have:

Misclassifications:

Treatment	A	P	Total
A	67	48	115
P	39	67	106

II.2 Using "rpart" Package

In this subsection, we will present the classification tree applying on the respiratory clinical data using "rpart" package. As in the "party" package we will use this package without/with training and test samples in the next two subsections.

II.2.1 Without (Training and Test) Samples

The next table presents the summary of classification tree results such as:
Terminal nodes (*), Index, Numbers of misclassifications (loss) for each node, Classes of outcome variable (A, P), Probability of misclassification, and Probability of correctness of classification.

Summary of classifications: (*) Terminal node

Node	Index	N	Loss	Class	P(loss)	1 - P(loss)
1	original root	444	216	P	0.48648649	0.51351351
2	outcome>=0.5	248	101	A	0.59274194	0.40725806
3	outcome<0.5	196	69	P	0.35204082	0.64795918*
4	id>=49.5	37	5	A	0.86486486	0.13513514
5	id<49.5	211	96	A	0.54502370	0.45497630
6	age<34.5	103	51	A	0.50485437	0.49514563
7	age>=34.5	93	17	P	0.18279570	0.81720430*
10	id<37.5	165	63	A	0.61818182	0.38181818
11	id>=37.5	46	13	P	0.28260870	0.71739130
12	id>=47	10	0	A	1.00000000	0.00000000*
13	id<47	93	42	P	0.45161290	0.54838710
20	sex=M	132	41	A	0.68939394	0.31060606
21	sex=F	33	11	P	0.33333333	0.66666667
22	id>=46.5	14	5	A	0.64285714	0.35714286*
23	id<46.5	32	4	P	0.12500000	0.87500000*
26	id<38	69	31	A	0.55072464	0.44927536
27	id>=38	24	4	P	0.16666667	0.83333333*
40	age>=20.5	97	22	A	0.77319588	0.22680412
41	age<20.5	35	16	P	0.45714286	0.54285714
42	id>=17.5	19	8	A	0.57894737	0.42105263*

43	id<17.5	14	0	P	0.00000000	1.00000000*
52	age>=19.5	59	23	A	0.61016949	0.38983051
53	age<19.5	10	2	P	0.20000000	0.80000000*
80	age<35.5	58	4	A	0.93103448	0.06896552*
81	age>=35.5	39	18	A	0.53846154	0.46153846
82	id<12.5	11	0	A	1.00000000	0.00000000*
83	id>=12.5	24	5	P	0.20833333	0.79166667*
104	id>=19.5	22	2	A	0.90909091	0.09090909*
105	id<19.5	37	16	P	0.43243243	0.56756757
162	age>=53.5	10	0	A	1.00000000	0.00000000*
163	age<53.5	29	11	P	0.37931034	0.62068966)
210	age<22.5	7	0	A	1.00000000	0.00000000*
211	age>=22.5	30	9	P	0.30000000	0.70000000
326	id>=12	20	9	A	0.55000000	0.45000000
327	id<12	9	0	P	0.00000000	1.00000000*
422	id<10.5	15	6	A	0.60000000	0.40000000*
423	id>=10.5	15	0	P	0.00000000	1.00000000*
652	baseline>=0.5	11	1	A	0.90909091	0.09090909*
653	baseline<0.5	9	1	P	0.11111111	0.88888889*

Root node error (misclassified) = 216/444 = 0.48648649.

This table indicates that the last nodes:
At the left side of tree "baseline" variable:
No. 652: 11 observation, predicted class = A,
P(node) = 0.02477477, class counts (10, 1), probabilities = (0.909, 0.091).
No. 653: 9 observations predicted class = P,
P(node) = 0.02027027, class counts (1, 8), probabilities = (0.111, 0.889).

At the right side of tree "id" variable:
No. 422: 15 observations, predicted class = A,
P(node) = 0.03378378, class counts (9, 6), probabilities= (0.6, 0.4).
No. 423: 15 observations, predicted class = P,
P(node) = 0.03378378, class counts (0, 15), probabilities = (0, 1).

The next table presents the importance of explanatory variables, which are used for building the classification tree. These values reflect the total sums of improvements for each variable through building the classification tree.

Id	Age	Baseline	Outcome	Sex	Center	Visit
84.142674	71.221255	14.818370	12.840235	11.493363	10.307844	1.221355

Figure 3, displays the importance of variables, and we conduct that the "id" variable is the most important variable follow the "age" and "baseline" variables. The reset variables are less importance.

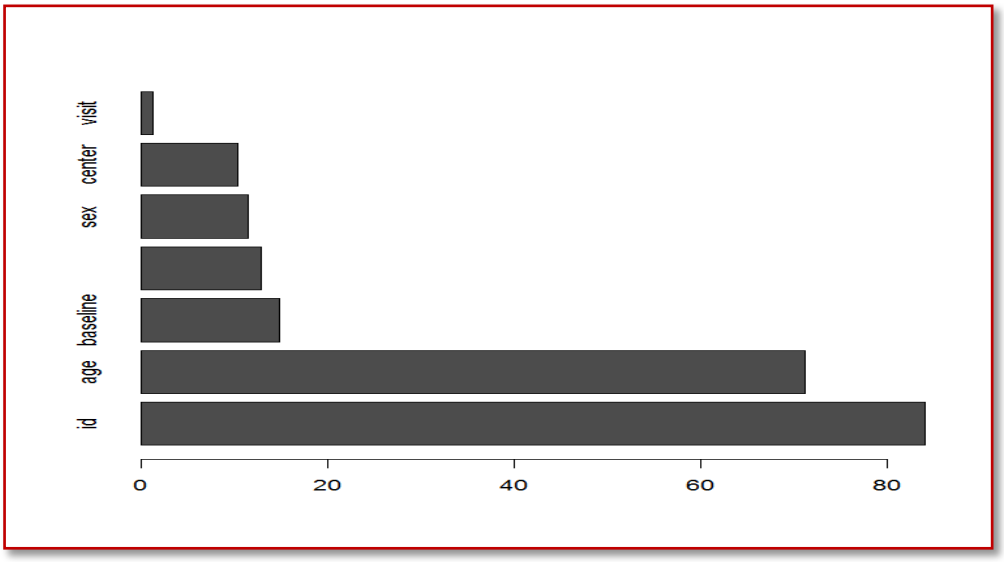

Figure 3: Variable importance

Figure 4 shows that the cross-validation errors: The dotted line is the minimum of the x-relative error curve plus 0.1 standard error. The smallest tree size is 20 nodes.

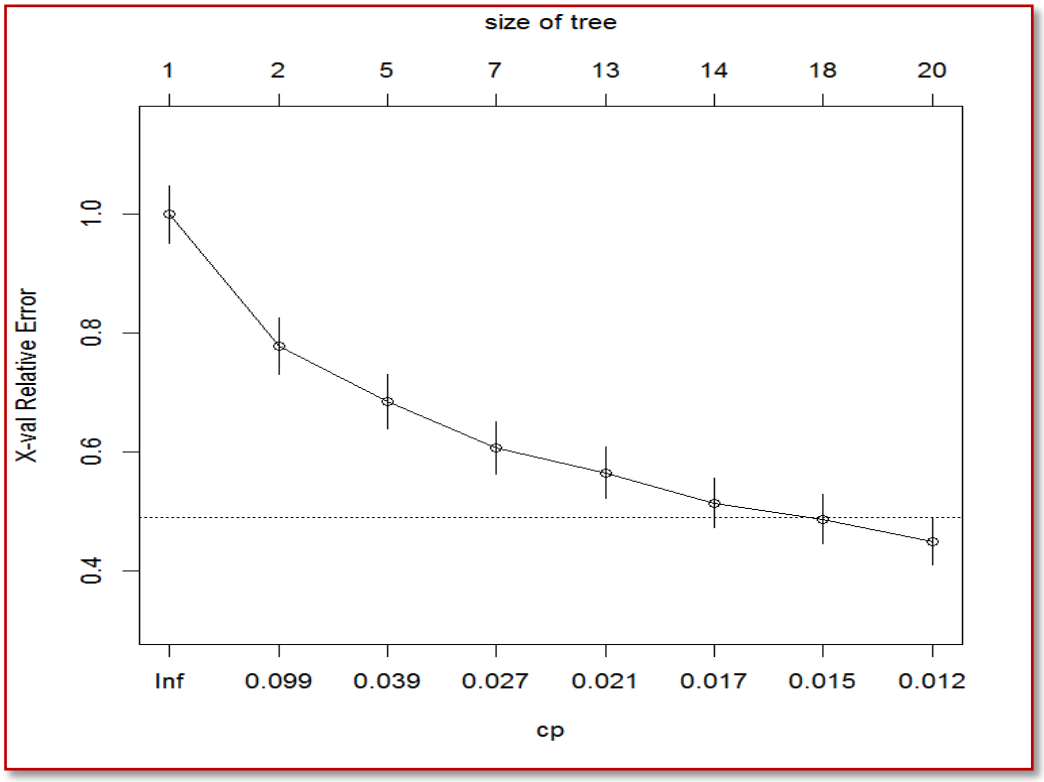

Figure 4: Cross-validation results

Figure 4, shows that a CP value of 0.01 corresponds to a tree with 19 splits (20 nodes) with xerror = 0.4490741 > (relative error = 0.2962963 + xstd error = 0.04030931). This value of CP = 0.01 will be used to prune the classification tree.

The next table is printing CP with errors to choose the optimal tree size:

CP	N-split	Relative error	Xstd	Xerror
0.21296296	0	1	0.04875836	1
0.04629630	1	0.7870370	0.04731120	0.7777778
0.03240741	4	0.6435185	0.04598662	0.6851852
0.02314815	6	0.5787037	0.04449002	0.6064815
0.01851852	12	0.4166667	0.04354744	0.5648148
0.01620370	13	0.3981481	0.04224141	0.5138889
0.01388889	17	0.3240741	0.04145236	0.4861111
0.01000000	19	0.2962963	0.04030931	0.4490741

We have saw that the CP = 0.01 at the 19 splits with 20 leaf nodes.

Figure 5 displays the classification tree after pruning it with CP = 0.01, using "rpart" package:

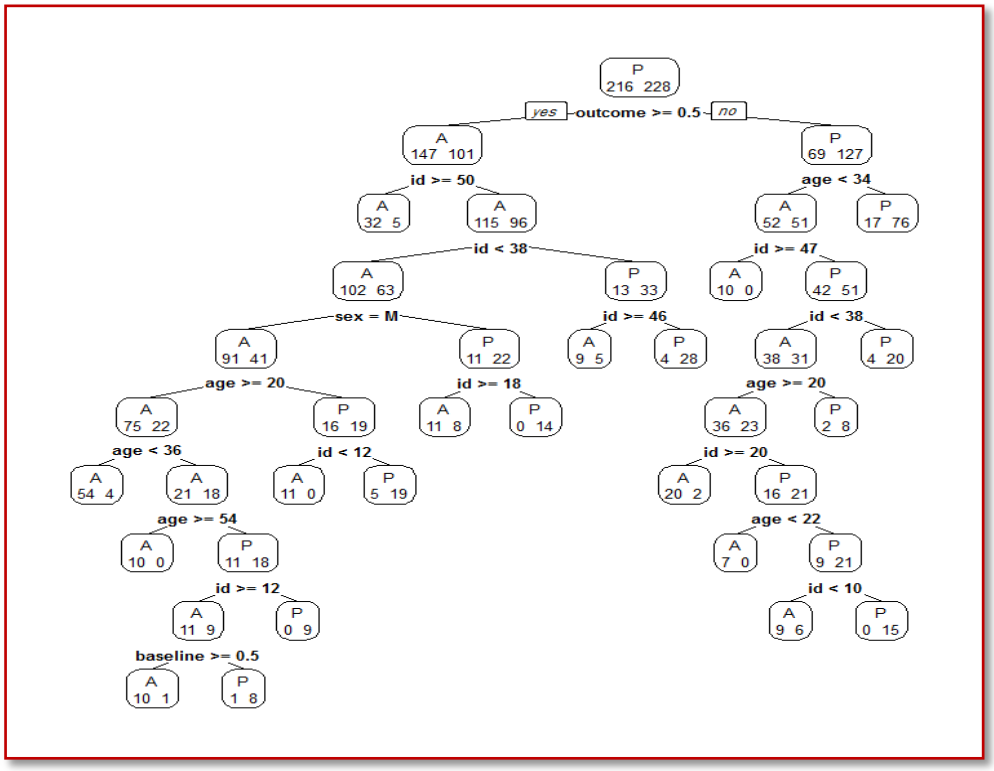

Figure 5: Classification tree after pruning

Figure 5 shows that: from 444 observations there P class (loss=216). If the outcome variable ≥ 0.5 is yes (at the left side of tree) the last nodes on the tree are represented as A=11 (loss=10) and P = 9 (loss=1) for baseline ≥ 0.5. If it is no (at the right side of tree) the last nodes is represented as A=15 (loss=9) and P=15(loss=0) for id < 10.

II.2.2 With (Training and Test) Samples

Before modeling the data we will split it into two subsets (samples); training (50%) and test (50%). Next table presents the summary of classification tree results with training sample as shown:

Node	Index	N	Loss	Class	P(loss)	1-P(loss)
1	original root	221	105	P	0.47511312	0.52488688)
2	id>=49.5	24	4	A	0.83333333	0.16666667)*
3	id<49.5	197	85	P	0.43147208	0.56852792)
6	id<38.5	148	73	A	0.50675676	0.49324324)
7	id>=38.5	49	10	P	0.20408163	0.79591837)
12	sex=M	111	43	A	0.61261261	0.38738739
13	sex=F	37	7	P	0.18918919	0.81081081
14	baseline<0.5	28	9	P	0.32142857	0.67857143
15	baseline>=0.5	21	1	P	0.04761905	0.95238095*
24	id>=35.5	7	0	A	1.00000000	0.00000000*
25	id<35.5	104	43	A	0.58653846	0.41346154
26	id>=17.5	18	7	P	0.38888889	0.61111111
27	id<17.5	19	0	P	0.00000000	1.00000000*
28	age<38.5	20	9	P	0.45000000	0.55000000
29	age>=38.5	8	0	P	0.00000000	1.00000000*
50	id<32.5	92	33	A	0.64130435	0.35869565
51	id>=32.5	12	2	P	0.16666667	0.83333333*
52	age>=36.5	12	5	A	0.58333333	0.41666667
53	age<36.5	6	0	P	0.00000000	1.00000000*
56	age>=27.5	7	0	A	1.00000000	0.00000000*
57	age<27.5	13	2	P	0.15384615	0.84615385*
100	age<35	66	17	A	0.74242424	0.25757576
101	age>=35	26	10	P	0.38461538	0.61538462
104	id<33	7	0	A	1.00000000	0.00000000*
105	id>=33	5	0	P	0.00000000	1.00000000*
200	age>=23.5	35	4	A	0.88571429	0.11428571*
201	age<23.5	31	13	A	0.58064516	0.41935484

202	age>=53.5	5	0	A	1.00000000	0.00000000*
203	age<53.5	21	5	P	0.23809524	0.76190476*
402	id<12.5	5	0	A	1.00000000	0.00000000*
403	id>=12.5	26	13	A	0.50000000	0.50000000
806	id>=15.5	13	6	A	0.68421053	0.31578947*
807	id<15.5	7	0	P	0.00000000	1.00000000*

Root node error = 105/221 = 0.47511.

This table indicates that the last nodes:
At the left side of tree (id):
No. 806: 19 observations, predicted class = A,
P(node) = 0.08597285, class counts (13, 6), probabilities (0.684, 0.316).
No. 807: 7 observations, predicted class = P,
P(node) = 0.03167421, class counts (0, 7), probabilities (0, 1).

At the right side of tree (age):
No 56: 7 observations, predicted class = A,
P(node) = 0.03167421, class counts (7, 0), probabilities (1,0).
No. 57: 13 observations, predicted class = P,
P(node) = 0.05882353, class counts (2 ,11), probabilities (0.154, 0.846).

The next table presents the importance of explanatory variables:

Id	Age	Sex	Baseline	Outcome	Center	Visit
57.0858191	42.1803863	10.9818393	7.3903268	5.2370108	2.7326729	0.9953081

Variable importance is displayed by Figure 6:

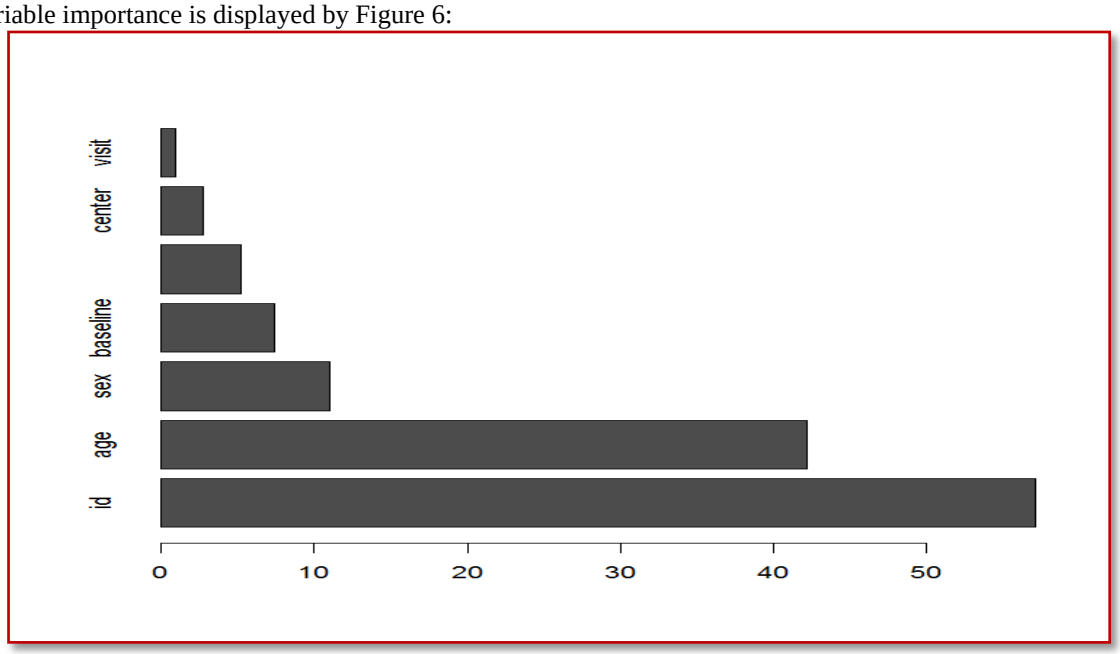

Figure 6: Variable importance

From figure 6 we see that the most important variable is "id" follows variable "age" and variable "sex" and the reset variables are less importance.
Figure 7 displays the cross-validation errors with training sample:

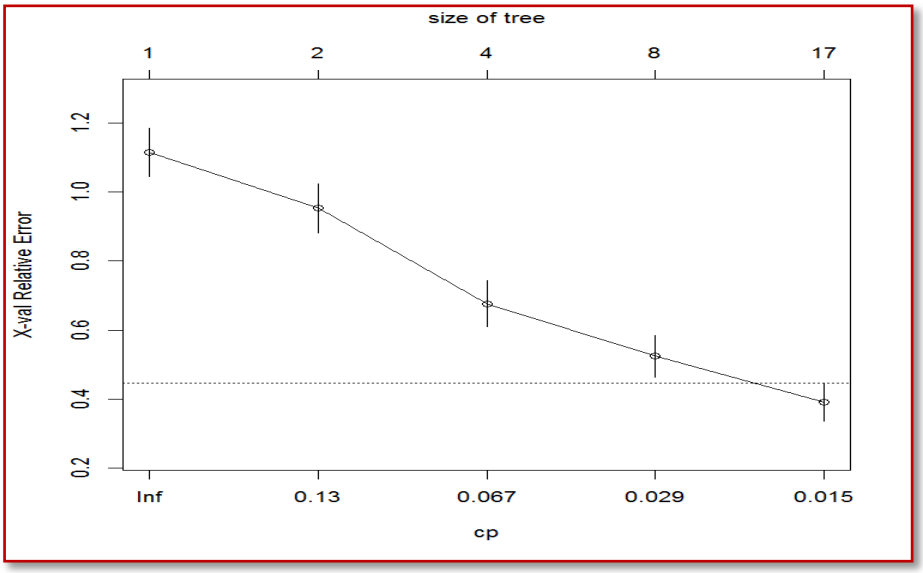

Figure 7: Cross-validation errors

Figure 7 shows that CP = 0.01 corresponds to a tree with 16 splits (17 nodes) with xerror = 0.39048 > (relative error = 0.22857 + xstd error = 0.055035). This value of CP = 0.01 will be used to prune the classification tree.

The next table presents CP with these errors:

CP	N-split	Xstd	Relative error	Xerror
0.152381	0	0.070668	1	1.11429
0.119048	1	0.070470	0.84762	0.95238
0.038095	3	0.066113	0.60952	0.67619
0.022222	7	0.061214	0.42857	0.52381
0.010000	16	0.055035	0.22857	0.39048

The optimal size of tree is 17 nodes (16 splits) with CP = 0.01
Figure 8 displays the pruning of classification tree with training sample using the "rpart" package.

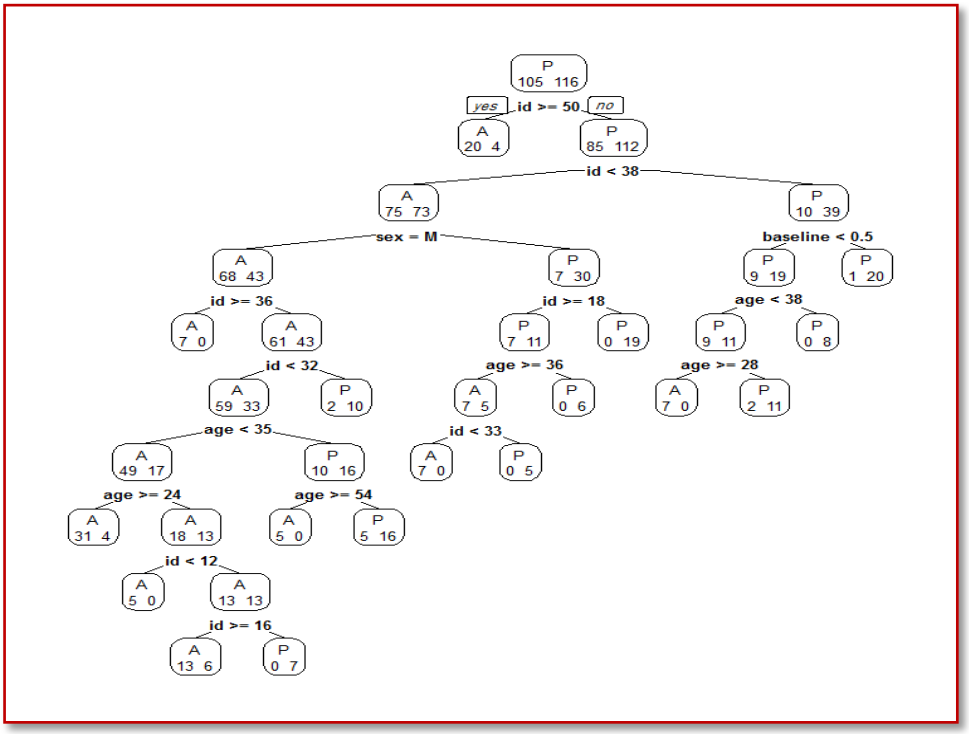

Figure 8: Classification tree after pruning

Figure 8 shows that: from 221 observations P class (loss = 105), if the id variable < 0.5 is yes (at the left side of tree) the last nodes is represented as A=24 (loss=20).

If it is no (at the left side of tree) the last nodes is represented as A=19 (loss=13) and P=7 (loss=0) for id ≥ 16 at the left side, and the last nodes is represented as A=7 (loss=0) and P=13 (loss=2) for age ≥ 28 at the right side.

Since the train sample = 221 obs. (A = 105, P = 116), we can summarize the attributes of predicted values for two classes (A, P) for the treatment with the observed values from test samples = 223 obs. (A=111, P=112).

A	P
Min. :0.0000	Min. :0.0000
1st Qu. :0.1538	1st Qu. :0.1143
Median :0.6842	Median :0.3158
Mean :0.5087	Mean :0.4913
3rd Qu. :0.8857	3rd Qu. :0.8462
Max. :1.0000	Max. :1.0000

Figure 9 displayed relation between the predicted values using the test sample and compared with the observed value for A class (Large Mean=0.5087).

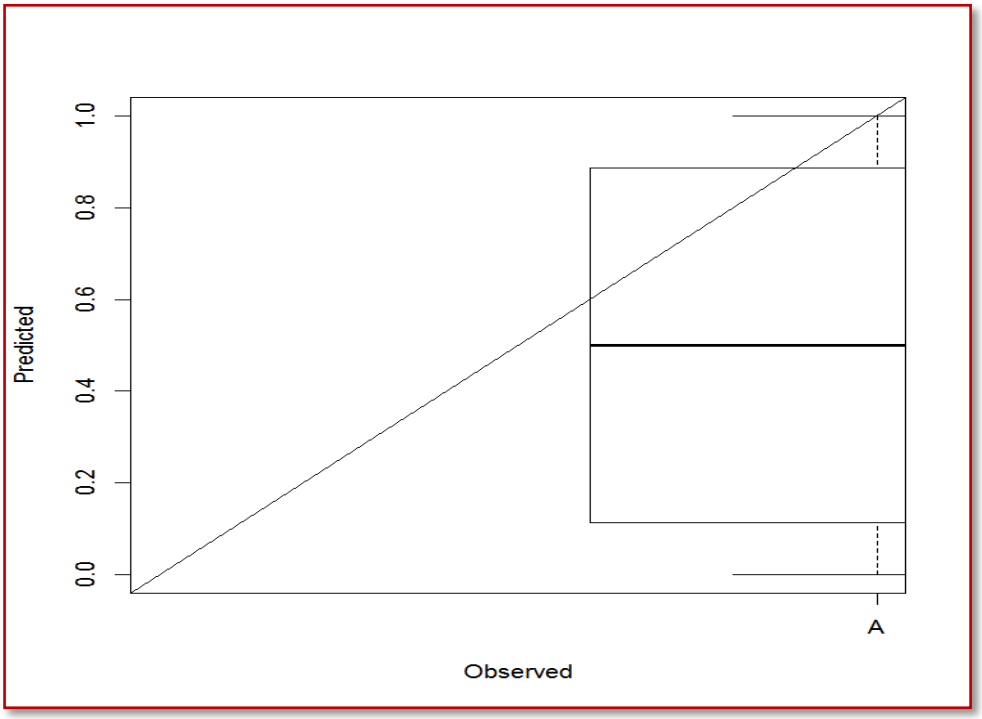

Figure 9: Predicted and observed values

This means that the predicted and observed values are too closed using the test sample, this reflect the accuracy of classification process.

II. 3 Using "randomForest" Package

In this subsection, we will demonstrate the classification process using "randomForest" package in two subsections: the first one the classification tree without training and test samples; the second one with training and test samples.

II.3.1 Without (Training and Test) Samples

Applying the "randomForest" package on the respiratory data, we have the next results:
Number of trees: 500
Number of variables tried at each split: 2
Estimate of error rate: 12.16%

Confusion matrix for all data 444 observations:

Treatment	A	P	Total	Classification Error
A	201	15	216	0.06944444
P	39	189	228	0.17105263

Figure 10 displays the estimate of classification error rate with numbers of trees:

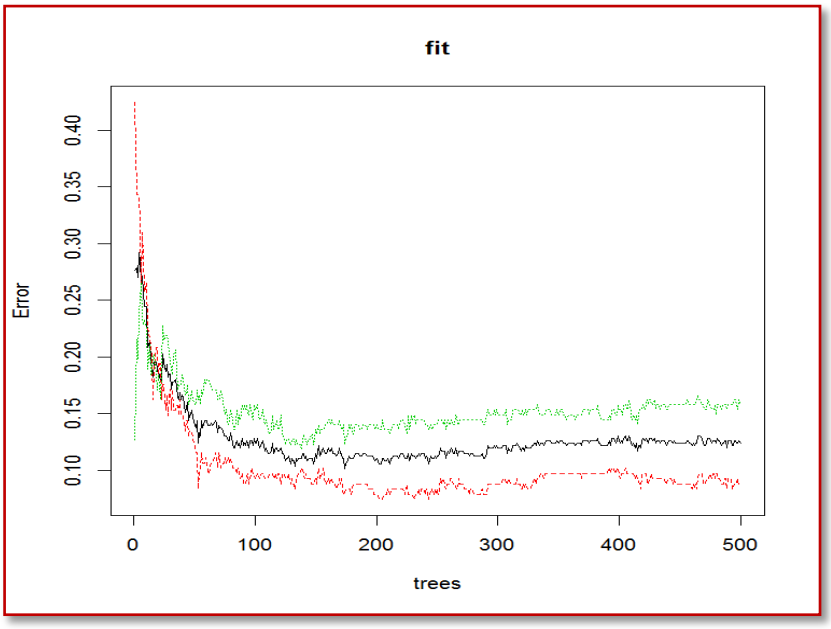

Figure 10: Estimate of classification error rate

The black curve of figure 10 indicates that: the estimate of error rate is decreases from 30% to 12.16% as the size of tree increased from 0 to 500 trees.

The next table presents importance of each predictor:

Id	Age	Sex	Outcome	Baseline	Center	Visit
59.886319	51.767213	13.296771	12.383189	7.355138	6.469766	6.270895

Figure 11 displays the importance of variables using Mean Decrease Gini:

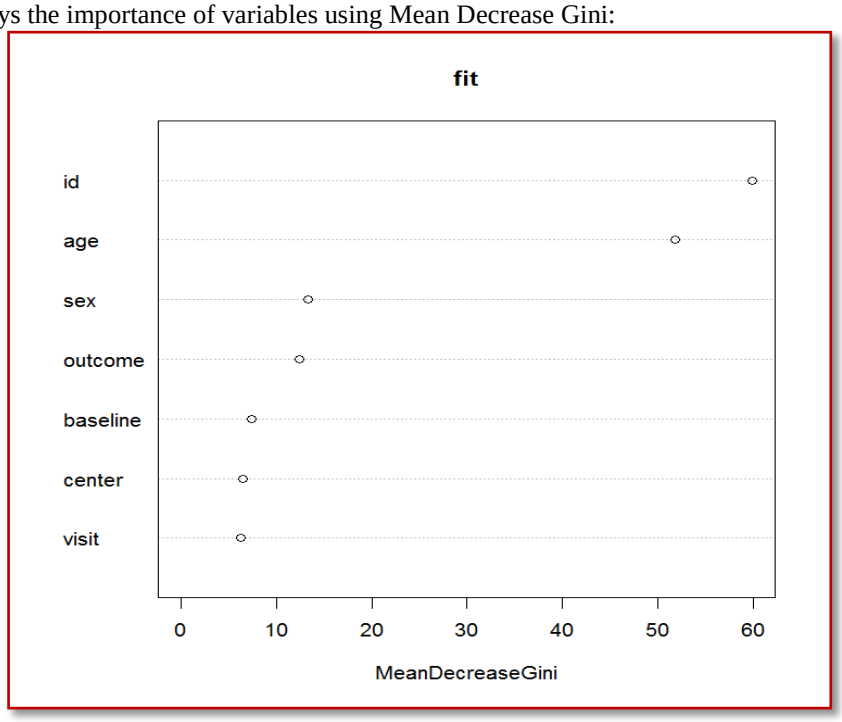

Figure 11: Importance of variables

From figure 11, we see that the variable "id" is the most important variable, followed the variable "age", then the variable "sex", the reset variables are less importance.

2.3.2 With (Training and Test) Samples

Before modeling the data we will split it into two subsets (samples); training (50%) and test (50%). Next table presents the summary of classification tree results with training sample using "randomForest" package as shown:

Number of trees: 100
No. of variables tried at each split: 2
Estimate of error rate: 19.13%

Confusion matrix for train set 130 obs.:

Treatment	A	P	Total	Classification Error
A	87	22	109	0.2018349
P	22	99	121	0.1818182

Figure 12 displays the estimate of classification error rate with numbers of trees:

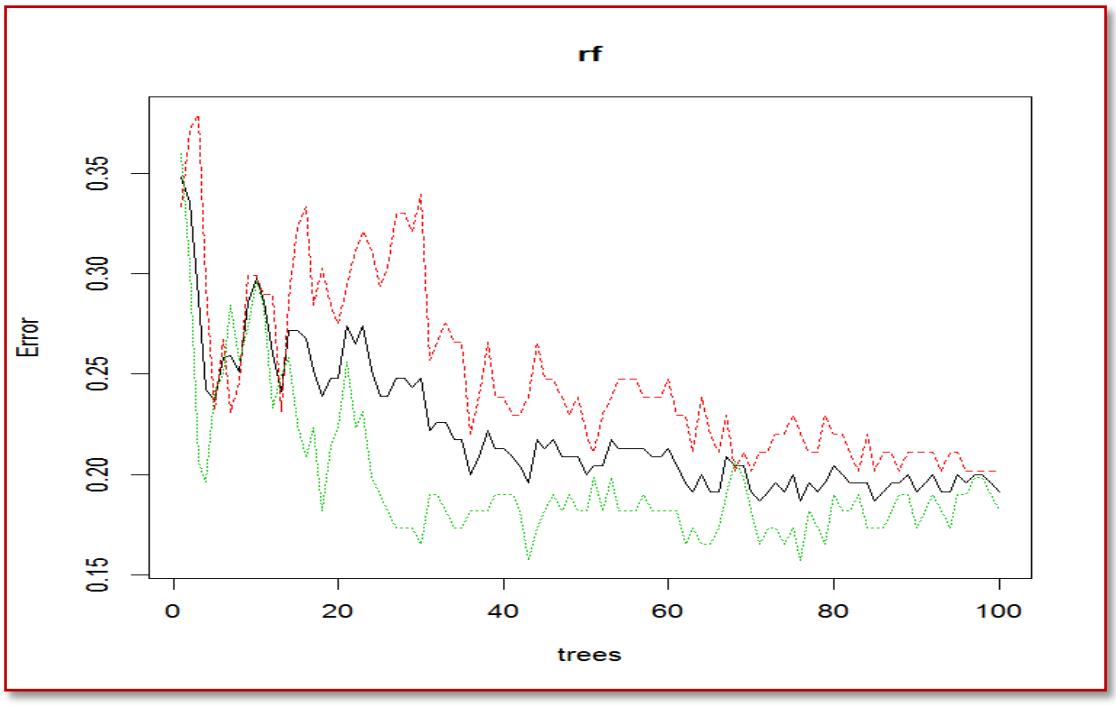

Figure 12: Estimate of error rate with numbers of trees

The black curve of figure 12 indicates that: the estimate of error rate is decreased from 35% to 19.13% as the size of tree increased from 0 to 100 trees.

The next table presents the importance of each predictor:

Id	Age	Sex	Outcome	Visit	Center	Baseline
31.069341	30.588146	8.122679	7.880642	6.151862	3.966221	3.887506

Figure 13 displays the importance of explanatory variables:

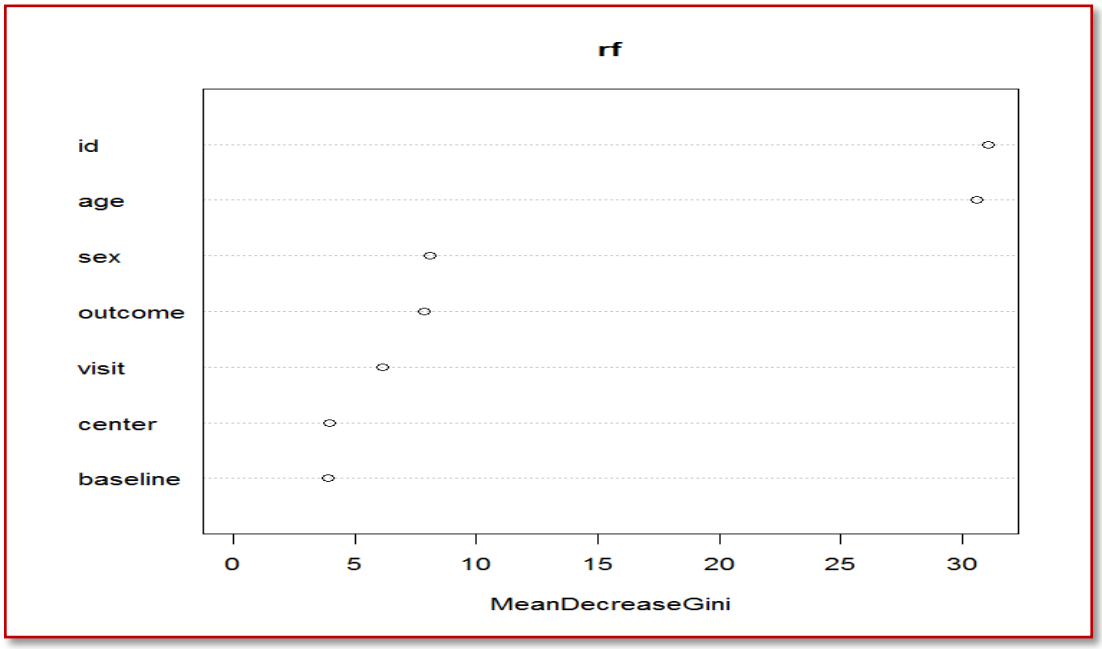

Figure 13: Importance of variables using Mean Decrease Gini- Train sample

From figure 13, we see that the variable "id" is the most important variable, followed the variable "age", then the variable "sex", the reset variables are less importance.

Using the test sample (114 obs.), we have:

Treatment	A	P	Total
A	87	24	111
P	20	83	103

Figure 14 displays the true and false positive rates.

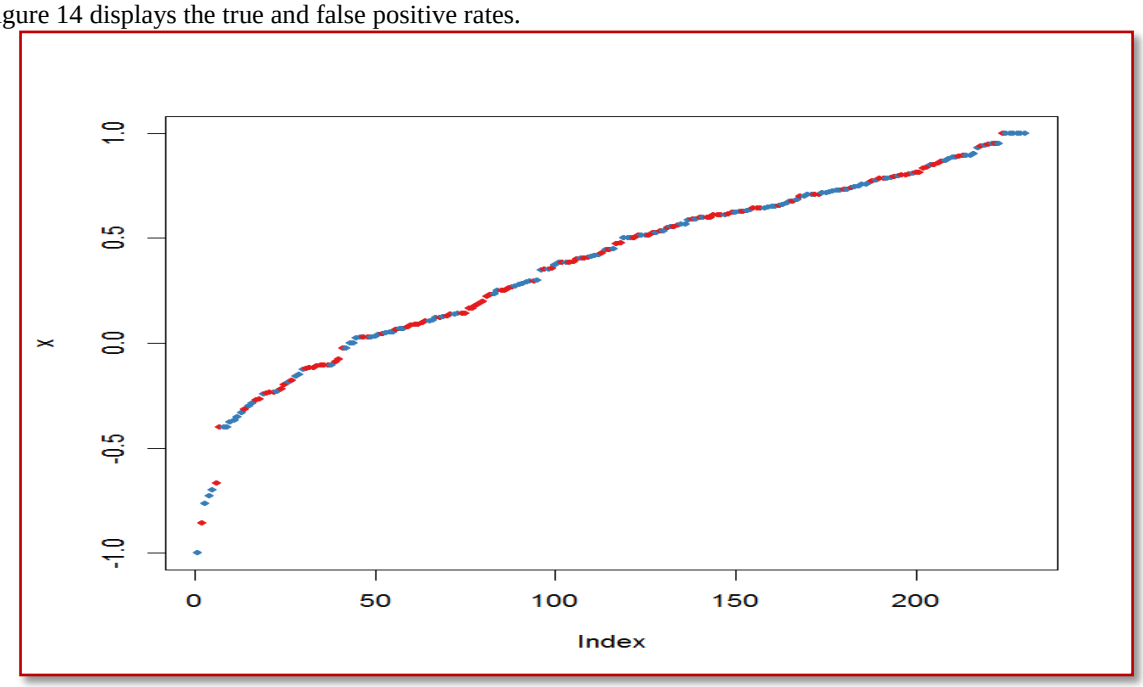

Figure 14: True-False positive rates

The true positive rates is achieved after nearly tree size = 50 nodes, and continue increasing as tree size increase up to be stable and equals 1 after 200 nodes. The true positive rates indicate a good classification process with test sample.

III. IBuilding of Regression Tree

In this section we will use "rpart" package to build the regression tree for the respiratory data, since the "outcome" variable represents the dependent binary variable; the other variables "visit", "sex", "baseline", "treat", "id" and "center" represent the predictor variables.

The next table presents the importance of the used variables:

Age	Baseline	Id	Center	Treat	Sex
19.805696	18.661002	12.576842	8.600325	5.254234	3.246308

The variables which actually used in the regression tree construction are "age", "baseline", "center", "id" and "treat". This means the "visit" variable was not use.

Figure 15 displays the important variables before pruning:

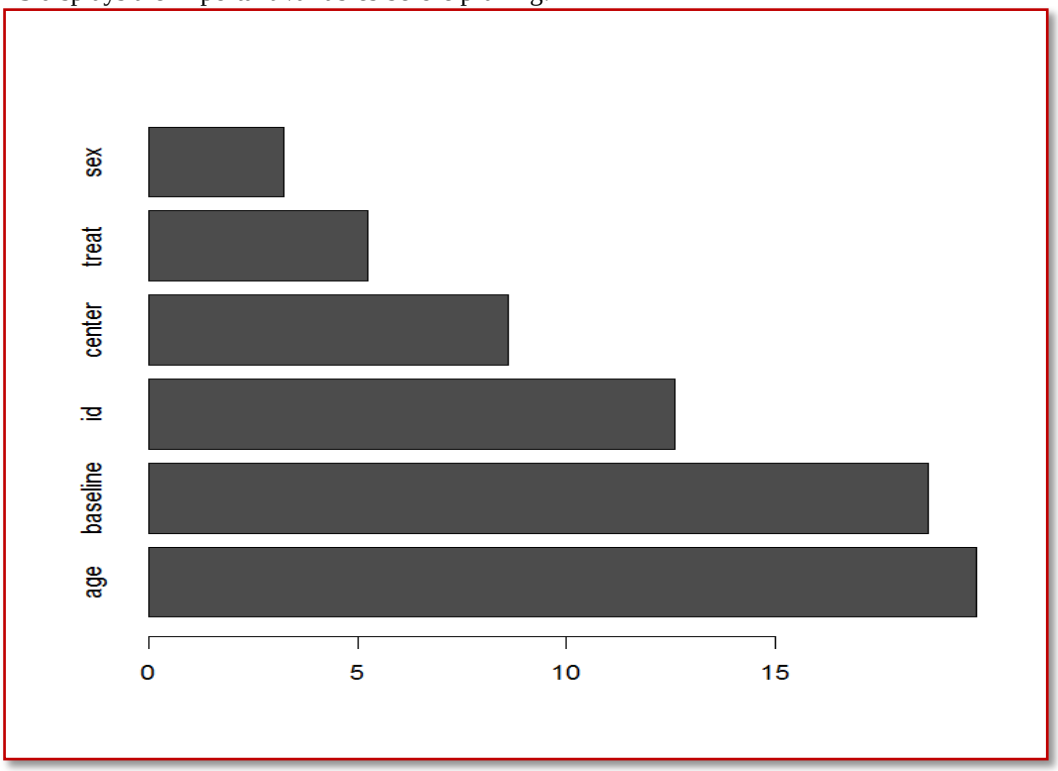

Figure 15: Variable importance before pruning

From Figure 15 we conclude that the most important variables are the "age", "baseline" and "id", the rest variables are less importance.

The next table presents the results of regression tree that contains: Node number, Index splits, Numbers of obs., Deviance and Mean of y-value:

Node	index	*N*	Deviance	y-value Mean
1	original root	444	109.4775	0.5585586
2	baseline<0.5	244	57.06148	0.3729508
3	baseline>=0.5	200	33.75500	0.7850000
4	center<1.5	152	29.94079	0.2697368
5	center>=1.5	92	22.82609	0.5434783
6	treat=P	104	22.88462	0.6730769
7	treat=A	96	8.15625	0.9062500*
8	age>=25.5	112	15.77679	0.1696429*
9	age<25.5	40	9.900000	0.5500000
10	age>=37.5	48	11.25000	0.3750000
11	age<37.5	44	8.727273	0.7272727
12	age>=31.5	52	12.98077	0.5192308
13	age<31.5	52	7.442308	0.8269231
18	treat=P	16	3.4375	0.3125*
19	treat=A	24	4.958333	0.7083333*
20	id<51	40	7.5	0.25

21	id>=51	8	0	1*
22	id>=14.5	28	6.678571	0.6071429
23	id<14.5	16	0.9375	0.9375
24	age<49.5	44	10.90909	0.4545455
25	age>=49.5	8	0.875	0.875*
40	id<14	12	0	0
41	id>=14	28	6.428571	0.3571429
44	age<31	16	3.75	0.375
45	age>=31	12	0.9166667	0.9166667*
82	id>=41	8	0	0*
83	id<41	20	5	0.5

Root node error: 109.48/444 = 0.24657

No.1: 444 obs., CP = 0.1704552, mean = 0.5585586, MSE = 0.2465709 , left son (244 obs.), right son (200 obs.).

From the previous variable we have:
At the left side "id" variable:
No. 82: 8 obs., mean=0, MSE=0.
No. 83: 20 obs., mean=0.5, MSE=0.25.

At the right side "age" variable:
No. 24: 44 observations, mean=0.4545455, MSE=0.2479339.
No. 25: 8 observations, mean=0.875, MSE=0.109375.

Figure 16 displays the cross-validation errors with optimal tree size:

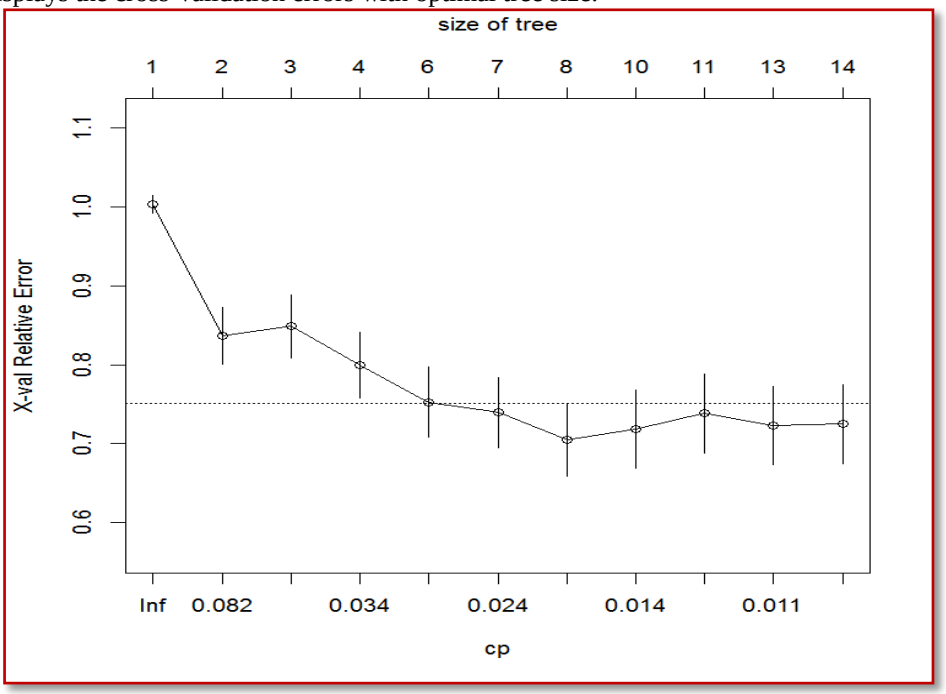

Figure 16: Cross-validation errors

Figure 16, shows that CP value of 0.01 corresponds to a tree with 13 splits (14 nodes) with "xerror" = 0.725072 > (relative error = 0.5677829 + "xstd" error = 0.04985353). The value of CP = 0.01 will be used to prune the regression tree.
The next table represents all errors with n-splits and cp value.

CP	N-split	xstd	relative error	xerror
0.17045517	0	0.01131539	1	1.0036320
0.03922815	1	0.03585762	0.8295448	0.8363913
0.03894868	2	0.03994132	0.7903167	0.8489486
0.03013777	3	0.04139475	0.7513680	0.7993496
0.02479172	5	0.04398607	0.6910925	0.7527769
0.02248443	6	0.04445974	0.6663007	0.7398937
0.01426369	7	0.04659961	0.6438163	0.7050106
0.01373951	9	0.04913372	0.6152889	0.7185560

0.01141787	10	0.04986646	0.6015494	0.7385410
0.01093082	12	0.04971359	0.5787137	0.7229020
0.01000000	13	0.04985353	0.5677829	0.7250720

We see from the previous table the smallest "xerror" is achieved at *n*-splits = 7, with CP = 0.01426369, which the regression tree must be pruned at 7 splits and 8 nodes. The relative errors are decreased as *n*-splits increase, but "xstd" is increased as *n*-splits increase.

Figure 17 displays the differences between apparent and x-relative errors.

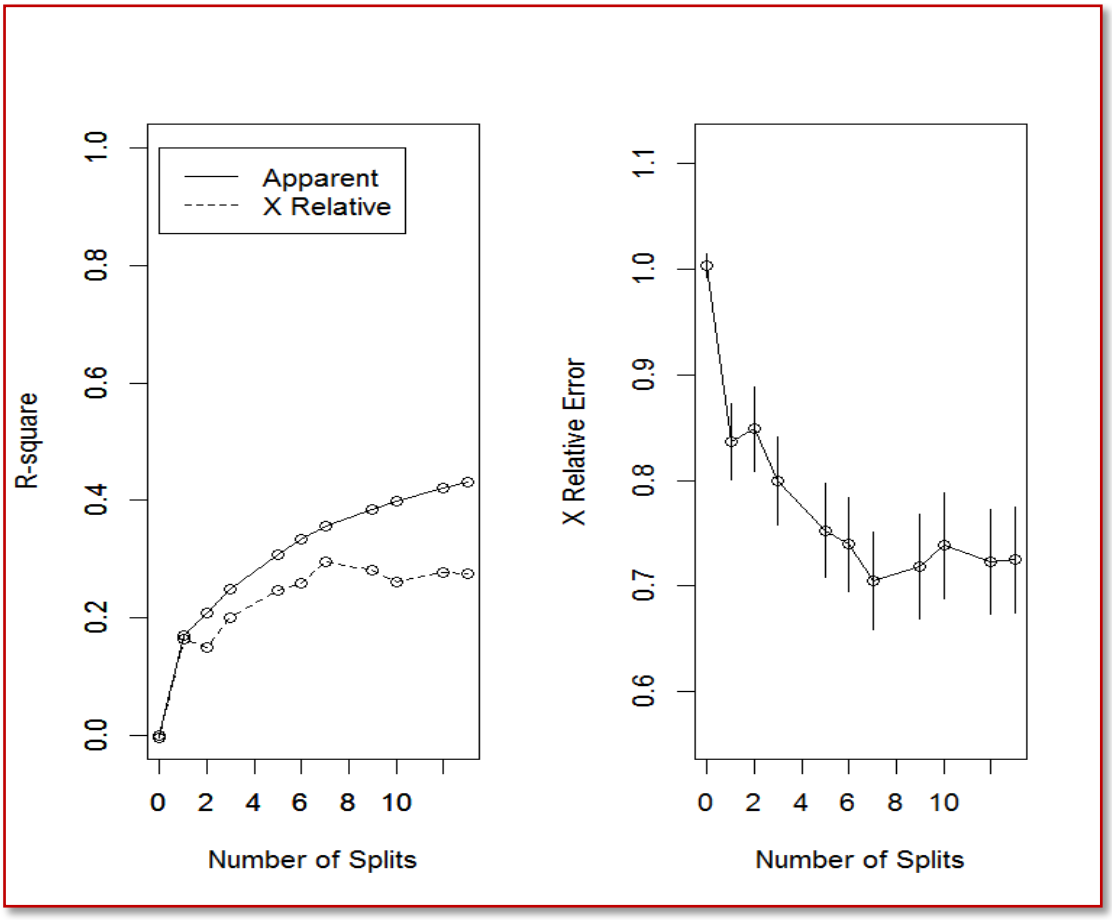

Figure 17: Apparent and X-relative errors

We see from figure 17 while the relative errors are decreased as *n*-splits, the R-square increased as *n*-splits increase, since R-square error > x-relative error.

To prune the regression tree we must stop at CP=0.01 with tree size = 14 nodes (13 splits).

Variable importance after pruning:

Age	Baseline	Id	Center	Treat	Sex
29	27	18	13	8	5

Figure 18 displays the variable importance after pruning.

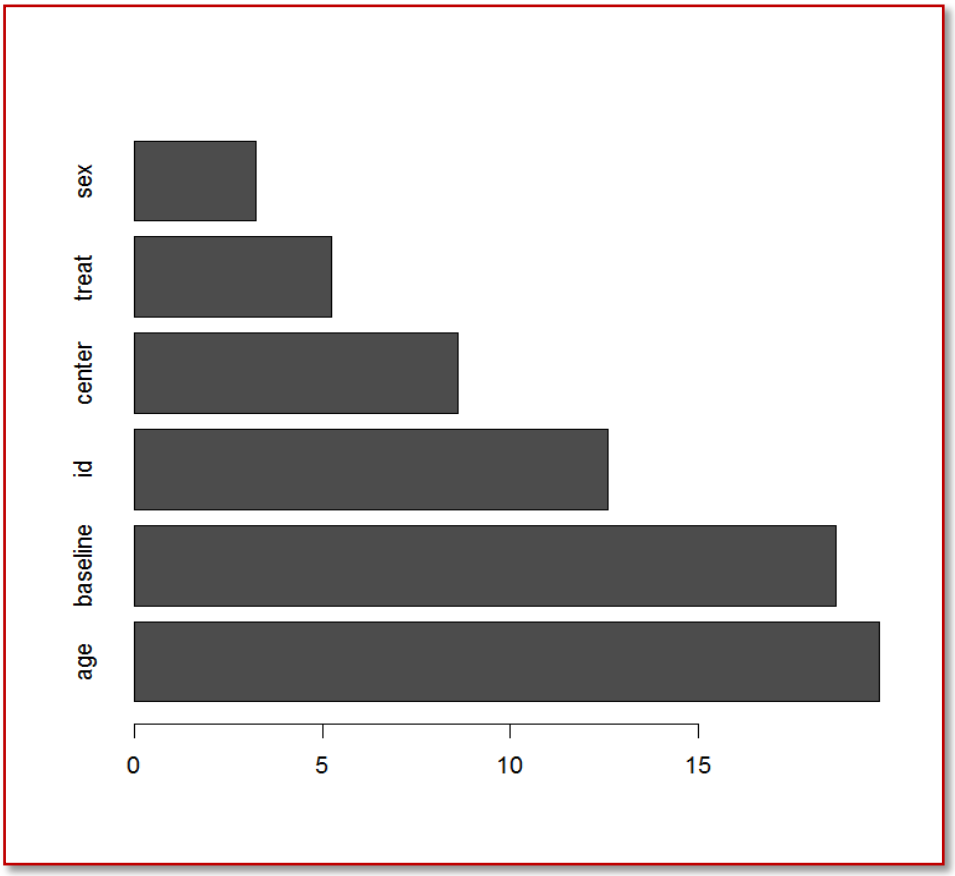

Figure 18: Variable importance after pruning

Figure 19 displays the pruning of regression tree using the rpart package.

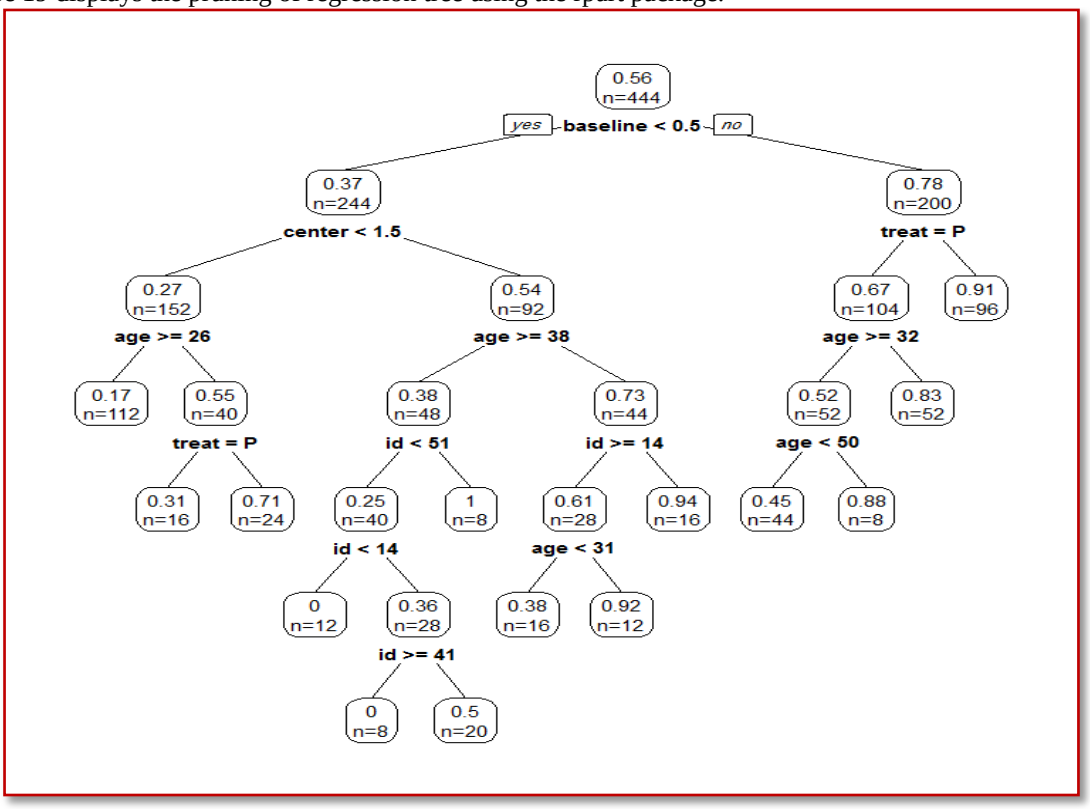

Figure 19: Regression tree after pruning using rpart package

Figure 19 shows that: we start from 444 obs. with mean = 0.56. If the "baseline" variable < 0.5 is yes, then the left side of tree is 244 obs. with mean = 0.37. If the "baseline" < 0.5 is no, then the right side of tree is 200 obs. with mean = 0.78.
The regression tree is pruned from the left when the variable "id" ≥ 41, then the last nodes are 8 obs. with mean = 0, and 20 obs. with mean = 0.5. Also, from the right when the variable "age" < 50, then the last nodes are 44 obs. with mean = 0.45, and 8 obs. with mean = 0.88.

IV. Results and Discussion

From the previous results, we have the next summary:

For the "party" package: we have devoted the classification tree in two cases. The first one is without training and test samples. The tree is starting from the "outcome" variable for 444 obs. and 5 nodes (4 splits). The most important variables are "sex" and "outcome". The second one is with training and test samples with 223 obs. for training sample and 3 nodes (2 splits). The important variable is "outcome".

For "rpart" package: we have also devoted the classification tree in two cases. The first one is without training and test samples. In this case the tree is starting from the "outcome" variable for 444 obs. and loss = 216/444 = 0.486 for P class . The last nodes for "id" variable with 20 obs. (11 obs. for A class with loss = 0.9091) and (9 obs. for P class with loss = 0.1111). The important variables according to sum of improvements respectively are : (id, age, baseline, outcome, sex, center and visit). The tree is pruning at CP = 0.01 with tree size = 20 nodes (19 splits) and minimum xerror = 0.44907.
The second one is with training and test samples. In this case the tree is starting with the "id" variable for 221 obs. and loss = 105/221 = 0.475 for P class . The last nodes for "id" variable with 26 obs. (19 obs. for A class with loss = 0.6842) and (7 obs. for P class with loss = 0.000). The important variables according to sum of improvements respectively are: (id, age, sex, baseline, outcome, center and visit). The tree is pruning at CP = 0.01 with tree size = 17 nodes (16 splits) and minimum xerror = 0.39048.

For "randomForest" package: we have also devoted the classification tree in two cases. The first one is without training and test samples with 444 obs. The misclassification error for A class is 0.6944, and for P class is 0.1711. The important variables according to sum of improvements respectively are: (id, age, sex, outcome, baseline, center and visit). The estimate of error rate is decreased from 30% to 12.16% with 500 trees.
The second one is with training and test samples, with train sample 230 obs. The misclassification error for A class is 0.2018, and for P class is 0.1818. The important variables according to Mean Decrease Gini are: (id, age, sex, outcome, visit, center and baseline). The estimate of error rate is decreased from 35% to 19.13% with 100 trees.

For "rpart" package: we have devoted the regression tree. With 444 obs. The regression tree is starting from "baseline" variable, deviance = 109.4775 and mean = 0.56. The last nodes with "id" variable was 28 obs. (8 obs. for A class, deviance = 0, mean = 0) and (20 obs. for P class, deviance = 5, mean 0.5). The important variables according to sum of improvements respectively are: (age, baseline, id, center, treatment, sex). The "visit" variable is not used in the regression tree. An optimal value of CP to prune the regression tree is 0.014 with minimum xerror = 0.705 with 7 splits. The regression tree is pruned at CP = 0.01 and xerror = 0.725 with tree size = 14 nodes (13 splits).

V. Conclusions

In this paper, the classification trees are built from the respiratory clinical data using different packages of R program such as "party", "rpart" and "randomForest" packages. These data are contained different types variables such as categorical, ordered, binary and continuous variables. The complexity parameter CP is used to prune the classification tree, and the cross-validation errors are constructed with the suitable tree size with minimum "xerror". The classification trees are displayed after pruning that explained the last nodes. The importance variables are detected according to the sum of improvements along the process of building the classification tree. For "rpart" package the training and test samples reduces the tree size and "xerror". The "randomForest" package indicated that an estimate of error rate decreased along the numbers of trees. But the training and test samples increase the estimate of error rate and decrease the numbers of trees.

Also, the process of building the regression tree is explained using "rpart" package with different types of errors that reflects the important variables of the regression tree. The used predictor variables are different variables and dependent variable is binary variable. The deviances and mean values of the dependent variables at each node presented. As in classification tree the important variables are "age", "baseline" and "id", the reset

variables "center", "treat" and "sex" are less important variables. But the "visit" variable is not used in the regression tree.

References

[1]. J. Han and M. Kamber, *Data mining: Concepts and techniques* (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000).
[2]. D. J. Hand, H. Mannila and P. Smyth, *Principles of data mining: Adaptive computation and Machine learning* (The MIT Press, 2001).
[3]. T. Hothorn, K. Hornik, C. Strobl and A. Zeile, Party: A laboratory for recursive partitioning, *R package version 1.0*, 23, 2015.
[4]. T. Therneau, B. Atkinson and B. Ripley, rpart: Recursive partitioning and regression trees, *R package version 4*, 2015,1-9.
[5]. A. Liaw and M. Wiener, Classification and regression by randomForest. *R News, 2(3),* 2002, 18-22.
[6]. M. E. Stokes, C. S. Davis and G. G. Koch, *Categorical data analysis using the SAS system* (SAS Program, 1995).

Ahmed Mohamed Mohamed Elsayed. " Decisions Tree Building for Different Types Data ." IOSR Journal of Mathematics (IOSR-JM) 15.1 (2019): 50-68.