CourseWork2 Decision Trees

Team MCMC

(Junjie Lin, Soomin Myung, Wenhao Deng, Yinbin Li, Yunfei Wang)

1. Introduction

The purpose of this coursework is to implement a classification and regression tree learning algorithm using MATLAB (basic MATLAB functions). The two datasets trained and tested on the 'UCI Machine Learning Repository' are:

a. Airfoil self-noise data set for regression (Airfoil data),

b. Estimation of obesity levels based on eating habits and physical condition data set for classification (Obesity data).

Information Gain (entropy) as the splitting criterion was used in the classification tree and Sum-of-Square Residuals (variance) as the splitting criterion for the regression tree.

2. Data Pre-Handling

The data handling part of codes were written in **DataHandling.m**. It performs several functions as listed:

- Download the two dataset files from UCI to the current directory.
- For the classification dataset (Obesity data).
    a. Dataset has 2111 examples in total, 1500 examples were used for training, and 611 examples were used for testing.
    b. Transform the multi-class classification to the binary classification. There is a 7-classes label, which indicates the obesity levels of an individual. Based on the coursework requirement, these 7 classes were recategorised to 2 classes (no_obesity and obesity).
    c. One-hot encode the multi-class features.

Based on the requirement, a binary tree should be built after learning. That means all decision should be made binary (positive or negative) for internal nodes. However, in obesity datasets, there are many multi-class features. Therefore, as shown in figure 1, the discrete value of the original data was encoded to one-hot keys to simplify the training algorithm.

d. Randomise the dataset.

Randomise was done to the data to ensure the distribution of the segment data can reflect the original distribution of the dataset when cross-validation was conducted.

ObesityDataSetrawanddatasinthetic

Gender	Age	Height	Weight	family_his...	FAVC	FCVC	NCP	CAEC	SMOKE	CH2O	SCC	FAF	TUE	CALC	MTRANS	NObeyesd...
分类	数值	数值	数值	分类	分类	数值	数值	分类	分类	数值	分类	数值	数值	分类	分类	分类
Gender	Age	Height	Weight	family_hist...	FAVC	FCVC	NCP	CAEC	SMOKE	CH2O	SCC	FAF	TUE	CALC	MTRANS	NObeyes...
Female	21	1.62	64	yes	no	2	3	Sometimes	no	2	no	0	1	no	Public_Tra...	Normal_W...
Female	21	1.52	56	yes	no	3	3	Sometimes	yes	3	yes	3	0	Sometimes	Public_Tra...	Normal_W...
Male	23	1.8	77	yes	no	2	3	Sometimes	no	2	no	2	1	Frequently	Public_Tra...	Normal_W...
Male	27	1.8	87	no	no	3	3	Sometimes	no	2	no	2	0	Frequently	Walking	Overweig...
Male	22	1.78	89.8	no	no	2	1	Sometimes	no	2	no	0	0	Sometimes	Public_Tra...	Overweig...
Male	29	1.62	53	no	yes	2	3	Sometimes	no	2	no	0	0	Sometimes	Automobile	Normal_W...
Female	23	1.5	55	yes	yes	3	3	Sometimes	no	2	no	1	0	Sometimes	Motorbike	Normal_W...
Male	22	1.64	53	no	no	2	3	Sometimes	no	2	no	3	0	Sometimes	Public_Tra...	Normal_W...

ProcessedObesityDataset

FAF	TUE	Gender_Fe...	Gender_M...	family_his...	family_his...	FAVC_no	FAVC_yes	CAEC_Alw...	CAEC_Freq...	CAEC_So...	CAEC_no	SMOKE_no	SMOKE_yes	SCC_no	SCC_yes	CALC_Alw...	CALC_Freq...	CALC_Som...	CALC_no	MTRANS_...	MTRANS_...	MTRANS_...	MTRANS_...	MTRANS_...	NObeyesd...
数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值
FAF	TUE	Gender_F...	Gender_M...	family_hist...	family_hist...	FAVC_no	FAVC_yes	CAEC_Alw...	CAEC_Fre...	CAEC_So...	CAEC_no	SMOKE_no	SMOKE_yes	SCC_no	SCC_yes	CALC_Alw...	CALC_Fre...	CALC_So...	CALC_no	MTRANS_...	MTRANS_...	MTRANS_...	MTRANS_...	MTRANS_...	NObeyes...
2	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	1	0	0	0	0	1	0	0
1	1	1	0	0	1	0	1	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	0
1.887012	0	1	0	0	1	0	1	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	1
0.174475	1.261705	0	1	0	1	0	1	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	1
0	0	1	0	1	0	1	0	0	1	0	0	1	0	1	0	0	0	1	0	0	0	1	0	0
0	2	0	1	0	1	0	1	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	0
0.998391	0.85882	0	1	1	0	0	1	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	0
0.008127	1	1	0	0	1	0	1	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	1
1.862235	1	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	0
2	0.152985	1	0	0	1	0	1	0	1	0	0	1	0	1	0	0	0	1	0	0	0	1	0	0
0	1.119877	0	1	0	1	0	1	0	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	1

Figure 1. Comparison between original data and processed data

- For the regression dataset (Air-foil data).
    a. Dataset has 1503 examples in total, 1000 examples for training, and 502 examples for testing.
    b. Randomise the dataset.

Fortunately, there is no missing value in these two datasets, so how to deal with missing value data was not considered in the coursework. However, if the dataset did have missing value examples and if the dataset was big enough, the missing value examples could be deleted in the data pre-handling process.

Finally, after the handling of the data, the function would return classification features and its labels, also regression features and its labels.

3. Results

The main code file is **DecisionTreeLab.m**, which could be run to see the results and the code file **DecisionTreeLearning.m** is the interface implemented

following the guidance of the lab sheet.

- For the classification

  Splitting criterion: Information Gain (entropy).

  Leaf node determine condition: All examples have the same label value.

  1. The first learning with the training set and test set (Tree is shown in figure 2).

     [Training] Precision: 1 Recall: 1 F1-score: 1

     [Test] Precision: 0.99303 Recall: 0.98276 F1-score: 0.98787

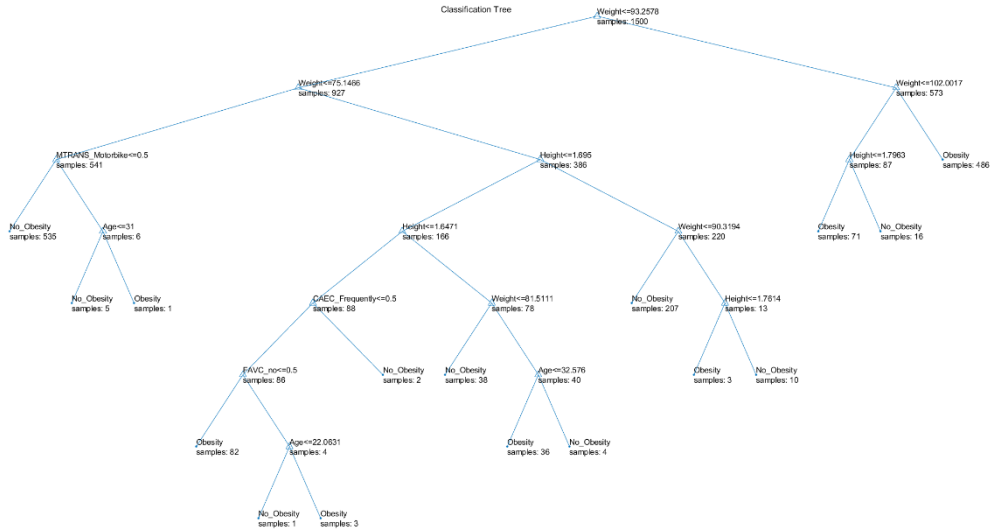


Figure 2. Classification tree learning from the training set.

  2. 10-fold cross-validation (Trees are shown in figure 3, also see appendix a).

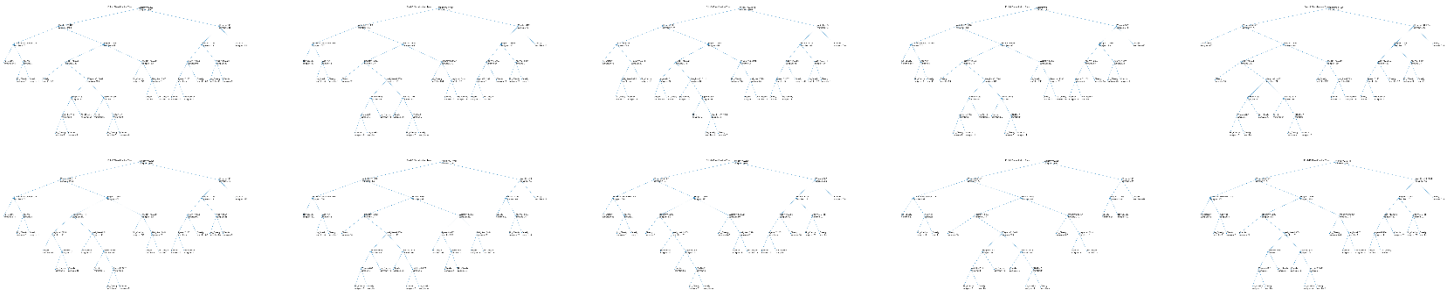


Figure 3. Trees learned from 10-fold cross-validation

[Training] Average Precision: 1 Average Recall: 1 Average F1-score: 1

[Test] Average Precision: 0.99362 Average Recall: 0.98891

Average F1-score: 0.99121

- For the regression

  Splitting criterion: Residuals Sum-of-Square (variance).

  Leaf nodes determine conditions:

A. All examples have the same label value.
B. The number of examples equals or less than 9.
C. The based residuals sum-of-square equals or less than 9.

1. First learning with the training set and test set (Tree is shown in figure 4).
   RMSE of training set: 1.8941
   RMSE of test set: 3.0047

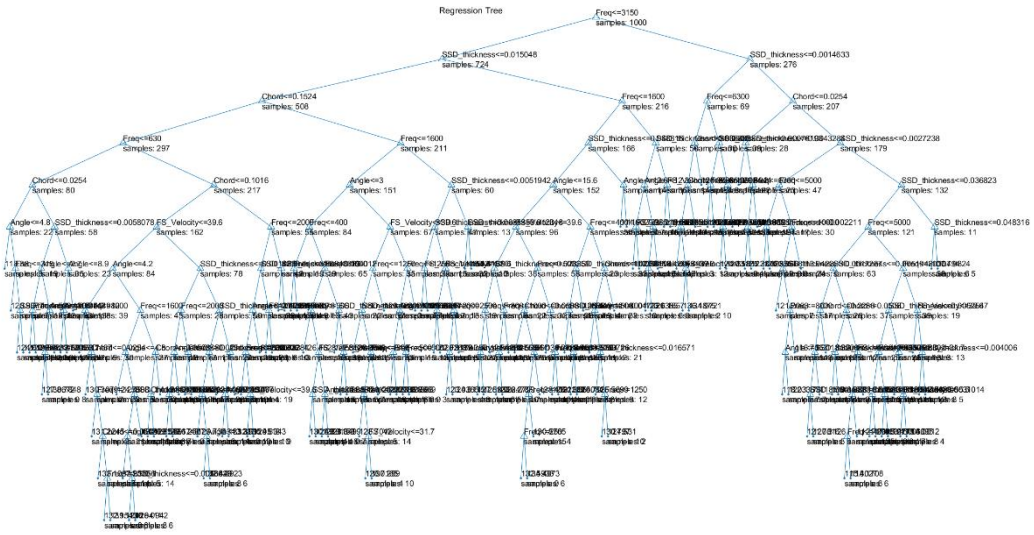


Figure 4. Regression tree learning from the training set.

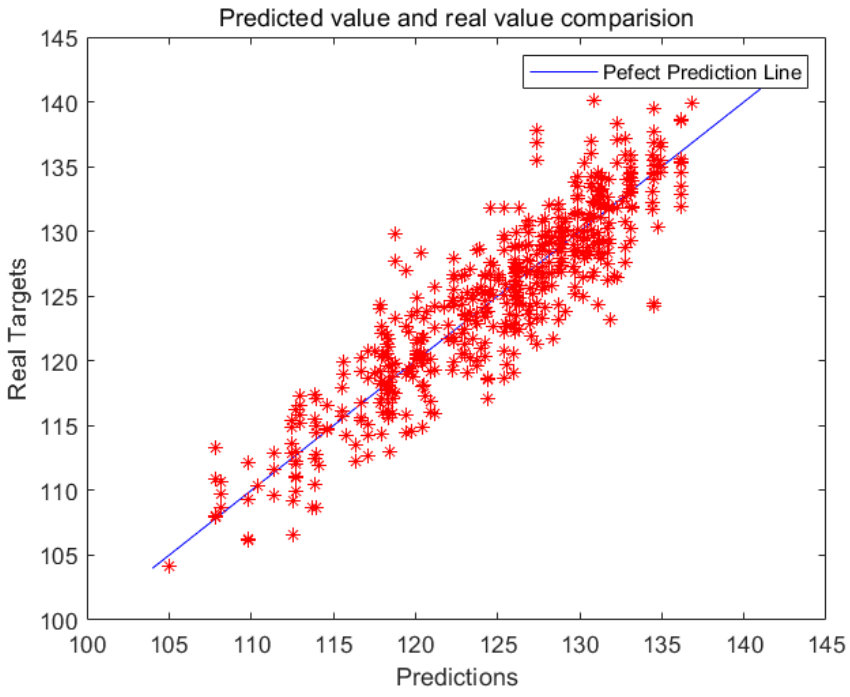


Figure 5. Predicted value vs real targets.

2. 10-fold cross-validation (10-fold Tree drawings see appendix b).
   Average RMSE of training sets: 1.8457
   Average RMSE of validation sets: 2.9665
3. Experiments with hyperparameters
   New leaf nodes determine condition was added after the 10-fold cross-validation: the max depth of the tree.
   As shown in figure 6, the RMSE of the testing set went flat following the increasing of tree depth after 11.

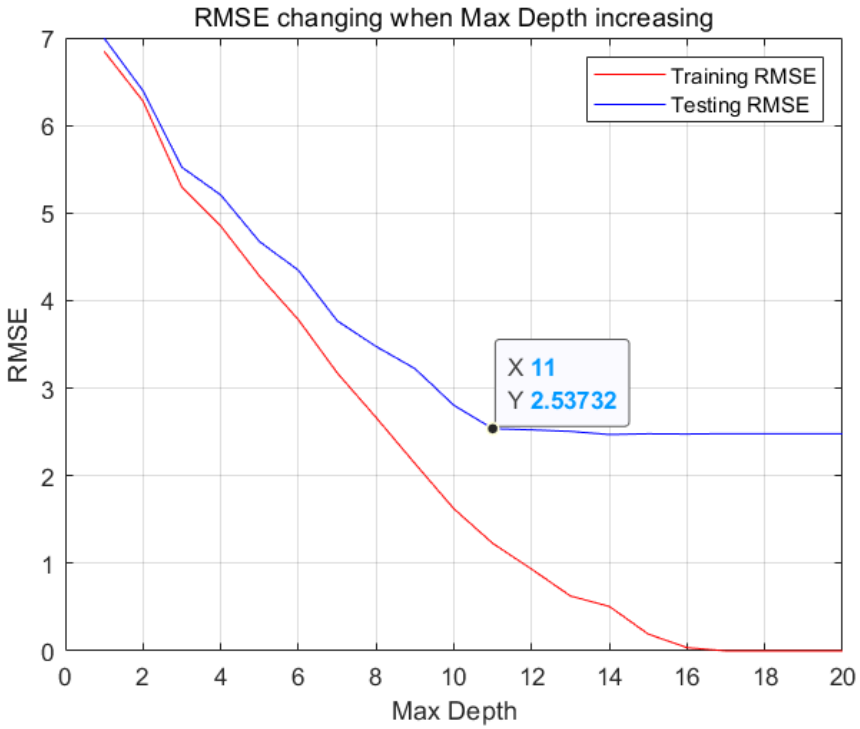


Figure 6. RMSE changing when Max Depth increasing

Therefore, we set the max depth as 11, minimal RSS and minimal numbers of examples were the same as the first training. As shown in figure 7, a tree which is smaller than the previous one was learned, and without losing accuracy which shown in figure 8 (RMSE of training set: 1.9335 RMSE of test set: 3.0084).

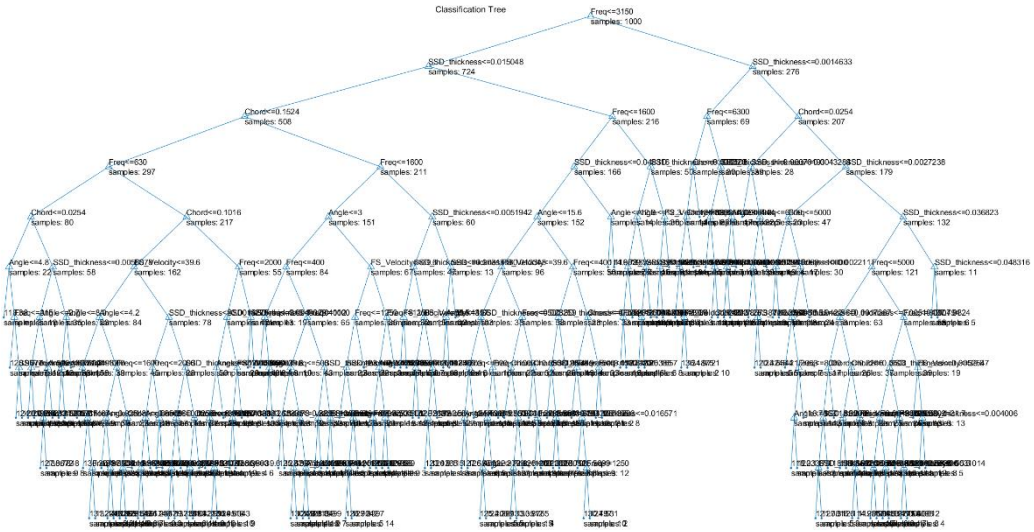


Figure 7. The regression tree with max depth as 11.

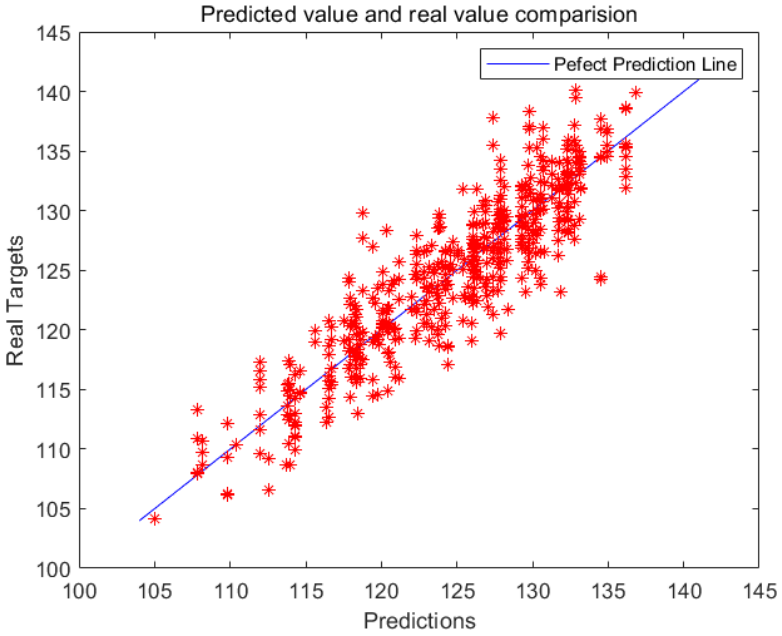


Figure 8. Predicted value vs real targets in max depth as 11.

4. Additional question answers

1. Pruning

Pruning can reduce the size of the decision tree by deleting the redundant subtrees. By reducing the complexity of the tree, pruning can improve the generalisation ability of the tree. According to Brunk and Pazzani in 1991, a technique called REP (Reduce Error Pruning) could be applied to prune the decision tree.

This REP technique split the dataset to be one training set and one pruning set. After using the training set to generate a decision tree, the pruning set is used to test and evaluate which subtree could be removed from the decision tree from the bottom to the top.

Take one of our classification trees as an example:

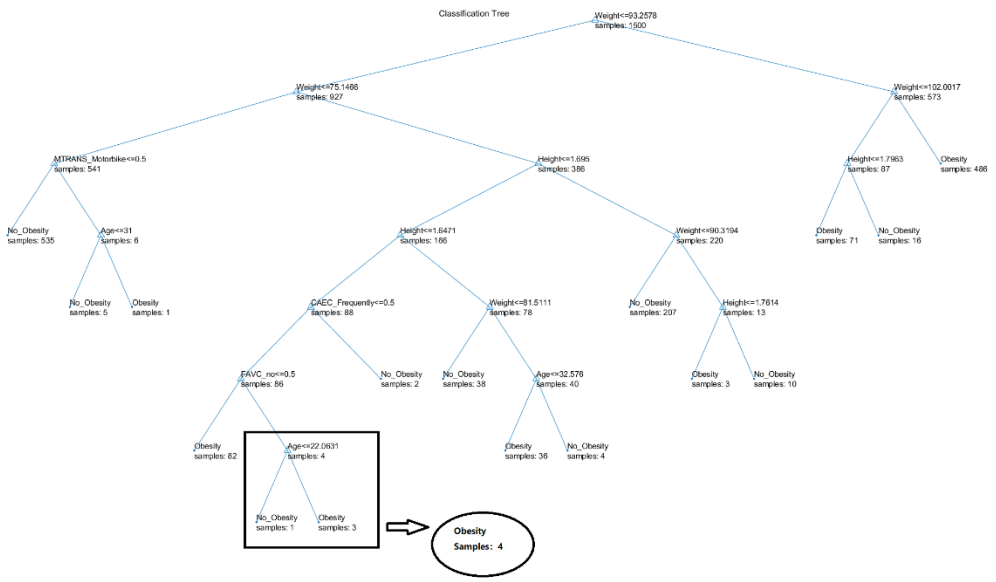


Figure 9. The first step to prune the tree

1. Firstly, the pruning set is used to the original tree to produce a based accuracy of the tree.
2. After this, the deepest leaf nodes are found and removed. All examples in these two nodes are put into their parent node. As a consequence, the parent node would become a new leaf node, and the prediction would become the majority value of targets corresponding to the examples within the node.

3. Then, this new tree is going to be tested using the pruning set and check whether the accuracy is improved. If the accuracy equals to the based accuracy or it is improved, then the tree stays in the current shape. Otherwise, those two removed leaf nodes should be recovered to the tree.
4. Repletely do step 3 and 4 to all nodes from bottom to the top, and finally, a pruned tree would be generated.

Suppose the original tree has a severe overfitting problem. In that case, the pruned tree should be in a smaller size and would have a higher predictive accuracy or at least has the same accuracy of the original tree.

2. Multiple independent binary labels prediction

A single binary tree could also implement multi-output classification. Although multi-label classification could be transformed to multi-class classification in data pre-handling process, it is not always a good manner to solve multi-label classification. For example, data instance pair (x1, [1,1,0]) could be transformed to pair (x1, 6) and instance pair (x2, [0,1,1]) could be transformed to pair (x2, 3), however, the number of transformed classes would grow exponentially when the number of labels grows. Therefore, an adapted algorithm for multi-labels output is worth to be considered.

According to Zhang and Zhou in 2013, an algorithm adaptation method for multi-label classification called Multi-Label Decision Tree was discussed. It also uses information gain as branch splitting criterion. However, to predict multi-binary-labels in a single tree, for all internal nodes in the tree, the entropy formula should be adapted. The entropy computation formula used in this coursework is:

$$Ent = -\frac{p}{p+n}\log_2\left(\frac{p}{p+n}\right) - \frac{n}{p+n}\log_2\left(\frac{n}{p+n}\right) \quad (1)$$

For predicting multiple binary labels, because each label is independent with each other, the entropy formula could be computed as the sum of the entropy of each label:

$$MLEnt = \sum_{i=1}^{k} -\frac{p_i}{p_i+n_i}\log_2\left(\frac{p_i}{p_i+n_i}\right) - \frac{n_i}{p_i+n_i}\log_2\left(\frac{n_i}{p_i+n_i}\right) \quad (2)$$

Here, the entropy of k different binary labels is computed one by one and added up together to be multi-labels entropy finally. The rest of the information gain computing is remaining the same as what is done in this coursework.

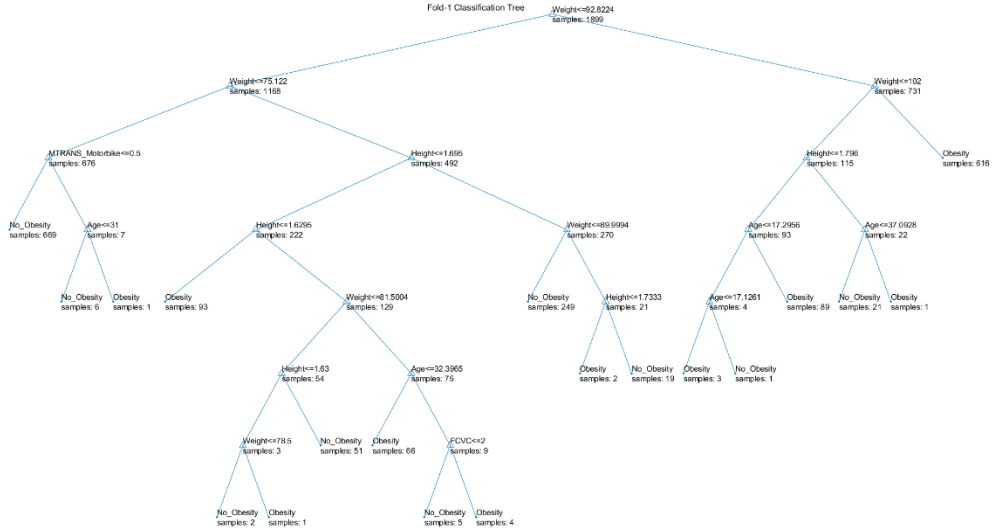
Moreover, what needs to predict would be a set of labels, so when the learning algorithm reaches the leaf node condition, the prediction method would be adapted to use the majority value of each node to compose as a set of labels (prediction). For example, for labels $[l_1, l_2, l_3 \ldots l_k]$, first, we find the majority value (positive or negative) of the 1st binary label using all examples within this node, and then find the majority value for the rest of labels one by one. Finally, the decision (prediction) can be made by composing all these majority values found as a set of label values.

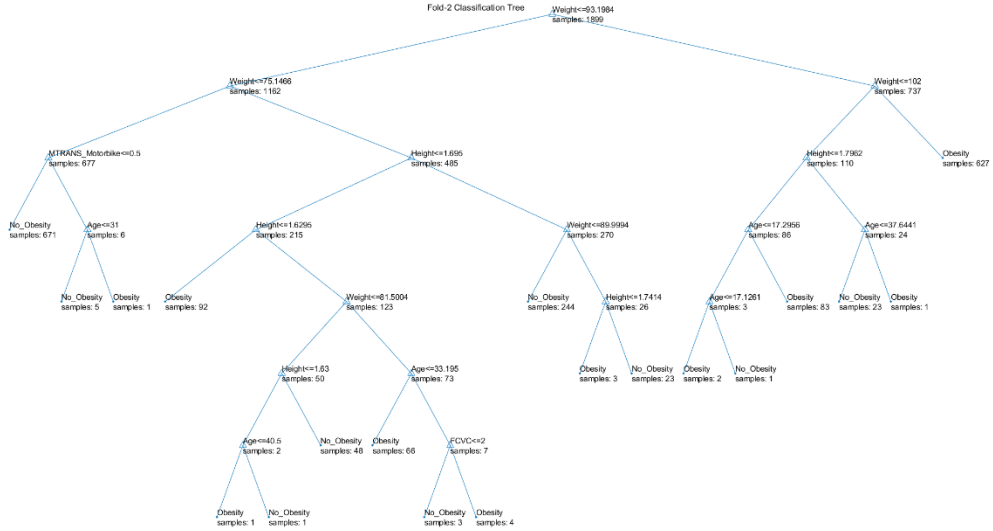
Reference

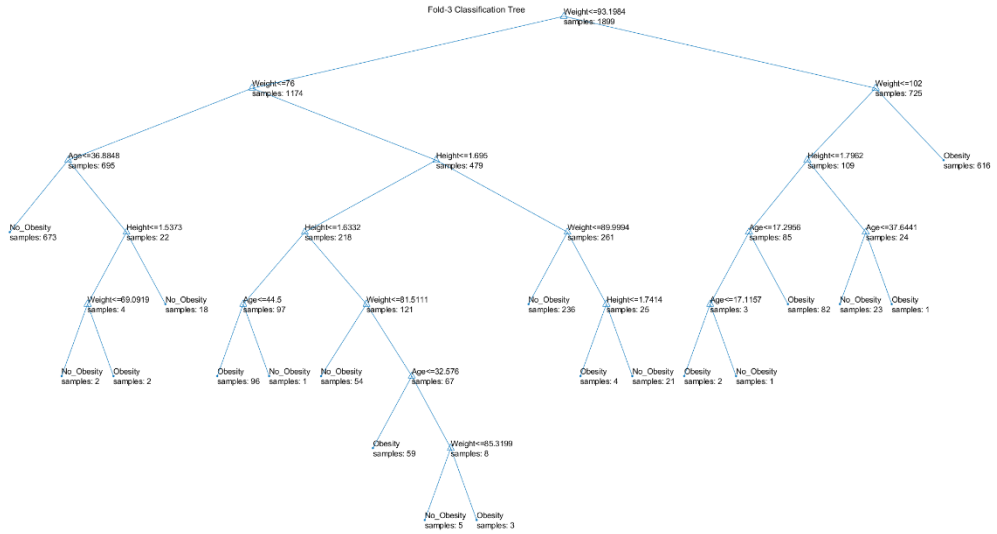
Brunk C. A. and Pazzani M J. (1991) An investigation of noise-tolerant relational concept learning algorithms. In **Proceedings of the 8th International Workshop on Machine Learning.** pp: 389-393. Evanston, Illinois. Available at: http://www.ics.uci.edu/~pazzani/Publications/An-Investigation-MLW-91.pdf [Accessed 25th November 2020].

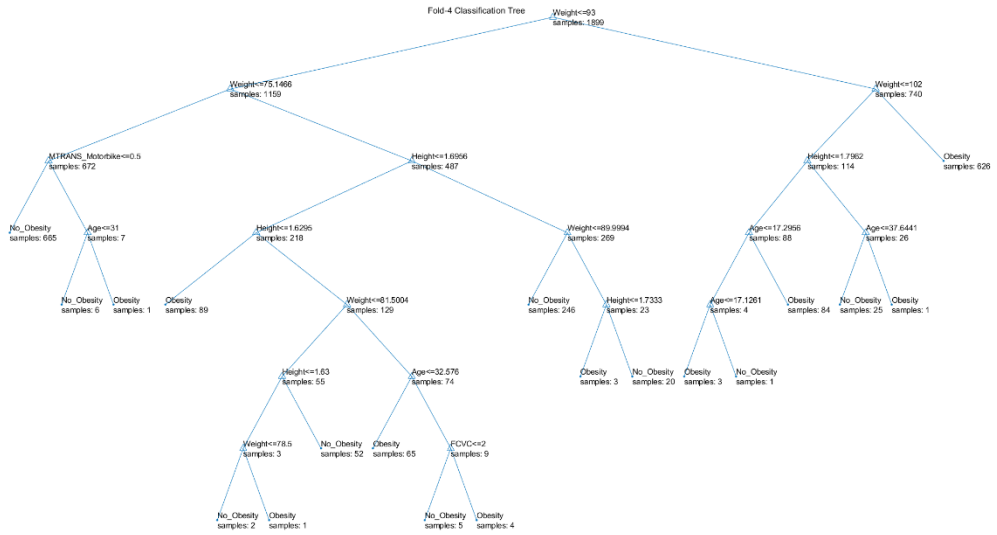
Zhang M L, Zhou Z H. (2013) A review on multi-label learning algorithms[J]. **IEEE transactions on knowledge and data engineering**. 26(8). pp: 1819-1837. Available at: https://romisatriawahono.net/lecture/rm/survey/machine%20learning/Zhang%20-%20Multi-Label%20Learning%20Algorithms%20-%202013.pdf [Accessed 24th November 2020].

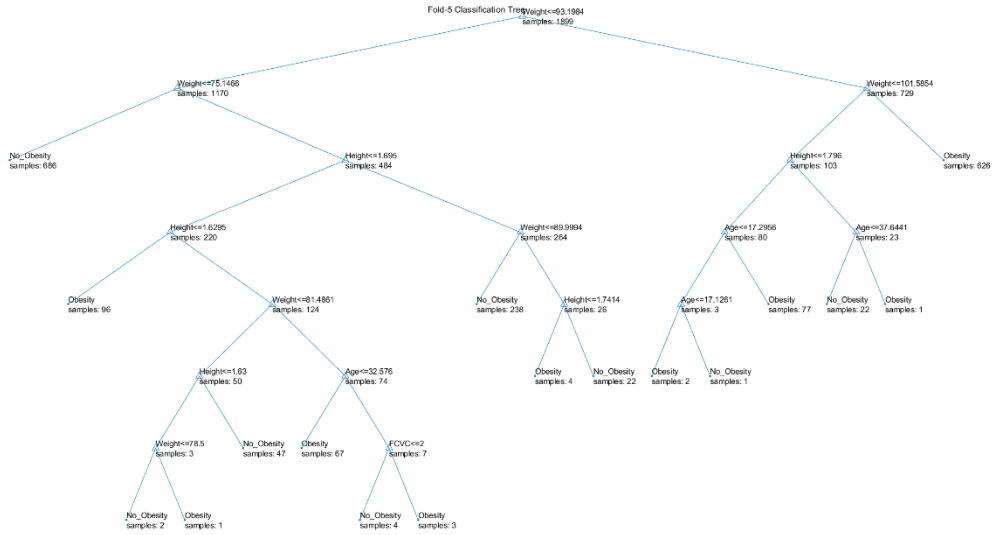
Appendix a: classification 10-fold tree drawings

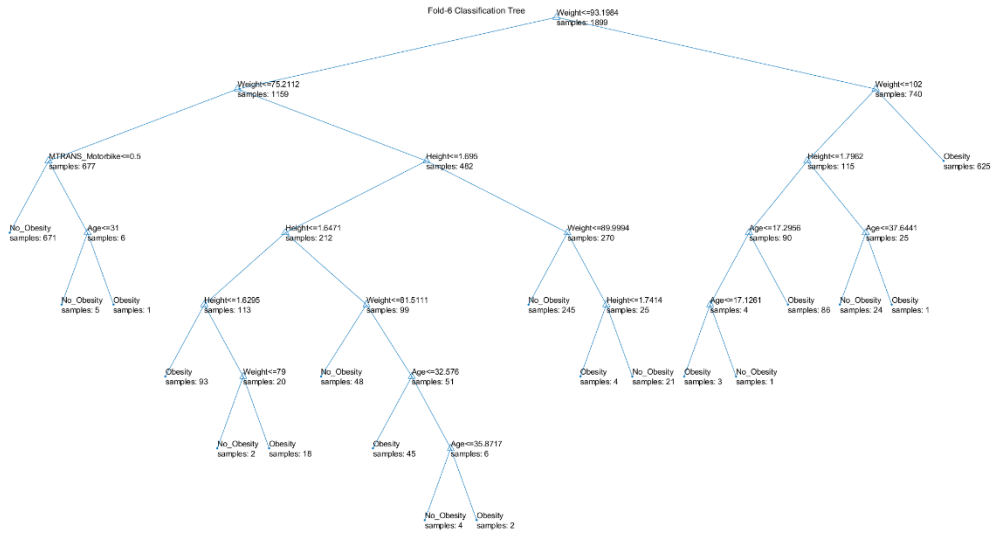


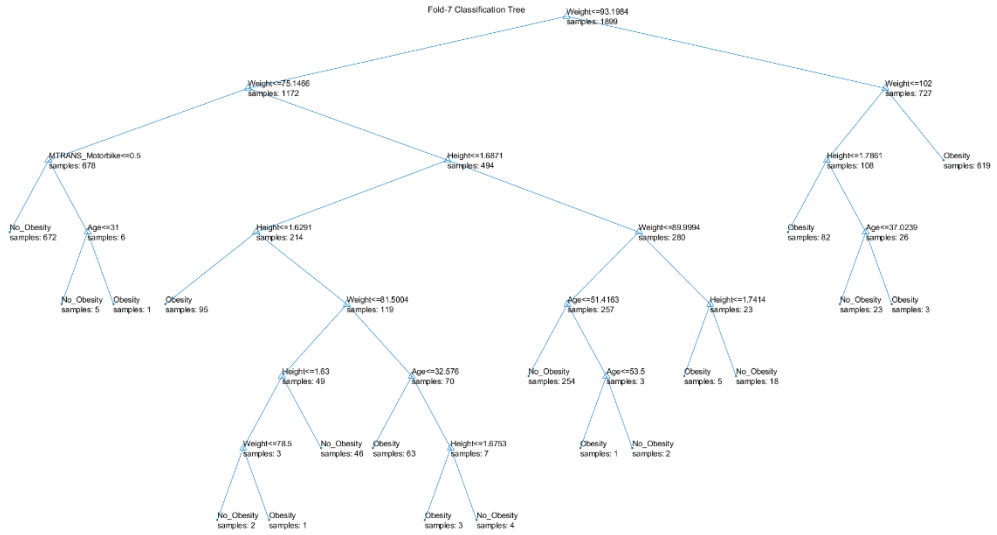


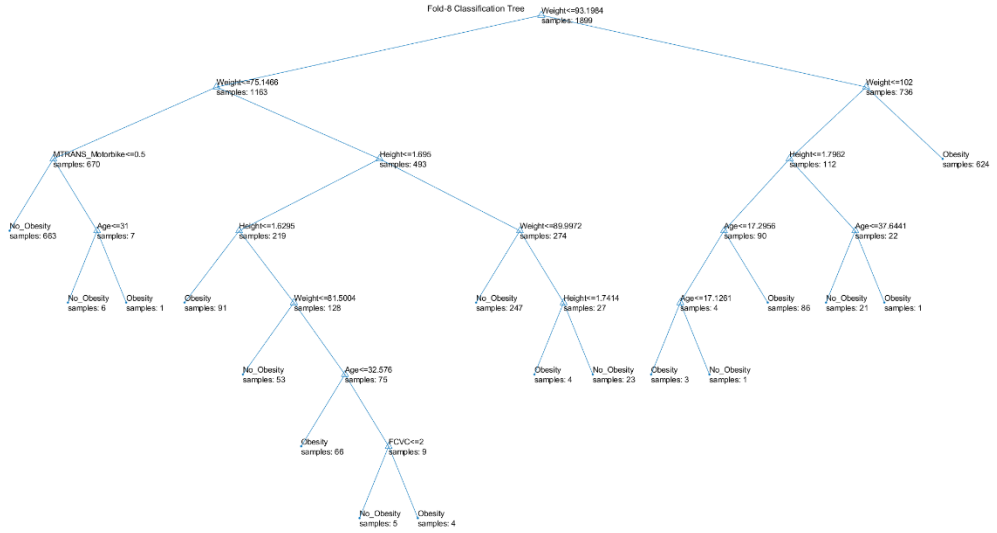

Fold-3 Classification Tree
Weight<=93.1984
samples: 1899
Weight<=76
samples: 1174
Weight<=102
samples: 725
Age<=36.8848
samples: 695
Height<=1.695
samples: 479
Height<=1.7952
samples: 109
Obesity
samples: 616
No_Obesity
samples: 673
Height<=1.5373
samples: 22
Height<=1.6332
samples: 218
Weight<=89.9994
samples: 261
Age<=17.2956
samples: 85
Age<=37.6441
samples: 24
Weight<=69.0919
samples: 4
No_Obesity
samples: 18
Age<=44.5
samples: 97
Weight<=81.5111
samples: 121
No_Obesity
samples: 236
Height<=1.7414
samples: 25
Age<=17.1157
samples: 3
Obesity
samples: 82
No_Obesity
samples: 23
Obesity
samples: 1
No_Obesity
samples: 2
Obesity
samples: 2
Obesity
samples: 96
No_Obesity
samples: 1
No_Obesity
samples: 54
Age<=32.576
samples: 67
Obesity
samples: 4
No_Obesity
samples: 21
Obesity
samples: 2
No_Obesity
samples: 1
Obesity
samples: 59
Weight<=85.3199
samples: 8
No_Obesity
samples: 5
Obesity
samples: 3

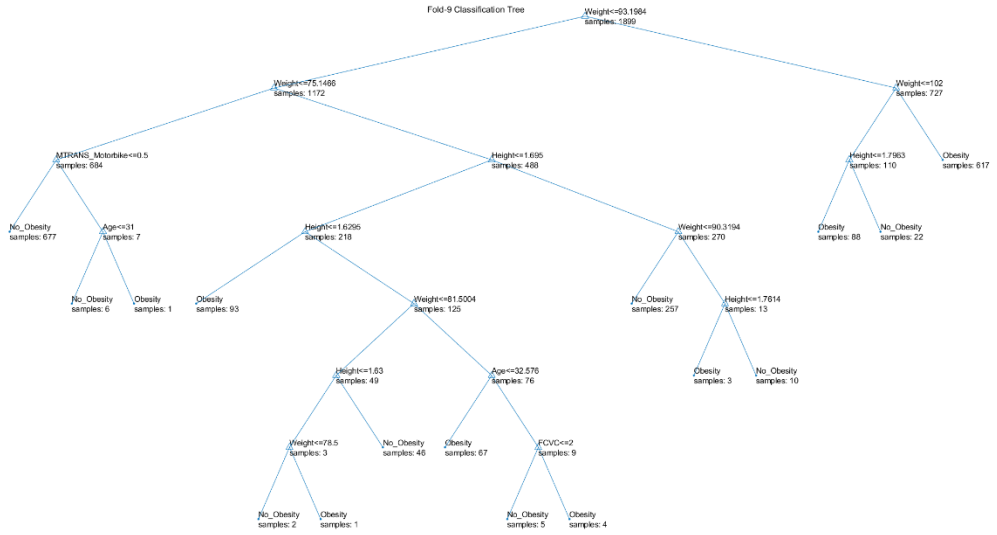

Fold-4 Classification Tree
Weight<=93
samples: 1899
Weight<=75.1466
samples: 1159
Weight<=102
samples: 740
MTRANS_Motorbike<=0.5
samples: 672
Height<=1.6956
samples: 487
Height<=1.7952
samples: 114
Obesity
samples: 626
No_Obesity
samples: 665
Age<=31
samples: 7
Height<=1.6295
samples: 218
Weight<=89.9994
samples: 269
Age<=17.2956
samples: 88
Age<=37.6441
samples: 26
No_Obesity
samples: 6
Obesity
samples: 1
Obesity
samples: 89
Weight<=81.5004
samples: 129
No_Obesity
samples: 246
Height<=1.7333
samples: 23
Age<=17.1261
samples: 4
Obesity
samples: 84
No_Obesity
samples: 25
Obesity
samples: 1
Height<=1.63
samples: 55
Age<=32.576
samples: 74
Obesity
samples: 3
No_Obesity
samples: 20
Obesity
samples: 3
No_Obesity
samples: 1
Weight<=78.5
samples: 3
No_Obesity
samples: 52
Obesity
samples: 65
FCVC<=2
samples: 9
No_Obesity
samples: 2
Obesity
samples: 1
No_Obesity
samples: 5
Obesity
samples: 4

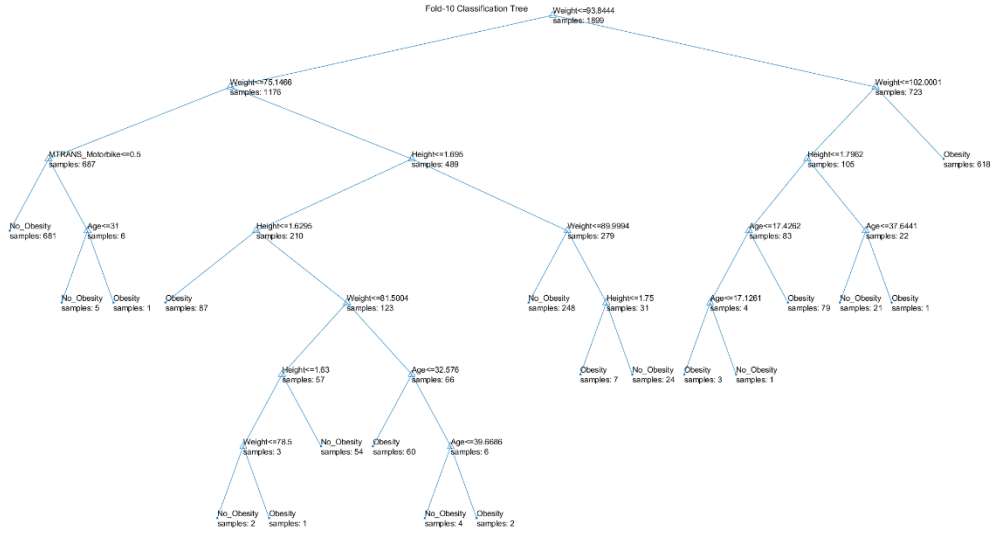

Fold-5 Classification Tree
Weight<=93.1984
samples: 1899
Weight<=75.1466
samples: 1170
Weight<=101.5854
samples: 729
No_Obesity
samples: 686
Height<=1.695
samples: 484
Height<=1.796
samples: 103
Obesity
samples: 626
Height<=1.6295
samples: 220
Weight<=89.9994
samples: 264
Age<=17.2956
samples: 80
Age<=37.6441
samples: 23
Obesity
samples: 96
Weight<=81.4861
samples: 124
No_Obesity
samples: 238
Height<=1.7414
samples: 26
Age<=17.1261
samples: 3
Obesity
samples: 77
No_Obesity
samples: 22
Obesity
samples: 1
Height<=1.63
samples: 50
Age<=32.576
samples: 74
Obesity
samples: 4
No_Obesity
samples: 22
Obesity
samples: 2
No_Obesity
samples: 1
Weight<=78.5
samples: 3
No_Obesity
samples: 47
Obesity
samples: 67
FCVC<=2
samples: 7
No_Obesity
samples: 2
Obesity
samples: 1
No_Obesity
samples: 4
Obesity
samples: 3


Fold-6 Classification Tree
Weight<=93.1984
samples: 1899
Weight<=75.2112
samples: 1159
Weight<=102
samples: 740
MTRANS_Motorbike<=0.5
samples: 677
Height<=1.695
samples: 482
Height<=1.7962
samples: 115
Obesity
samples: 625
No_Obesity
samples: 671
Age<=31
samples: 6
Height<=1.6471
samples: 212
Weight<=89.9994
samples: 270
Age<=17.2956
samples: 90
Age<=37.6441
samples: 25
No_Obesity
samples: 5
Obesity
samples: 1
Height<=1.6295
samples: 113
Weight<=81.5111
samples: 99
No_Obesity
samples: 245
Height<=1.7414
samples: 25
Age<=17.1261
samples: 4
Obesity
samples: 86
No_Obesity
samples: 24
Obesity
samples: 1
Obesity
samples: 93
Weight<=79
samples: 20
No_Obesity
samples: 48
Age<=32.576
samples: 51
Obesity
samples: 4
No_Obesity
samples: 21
Obesity
samples: 3
No_Obesity
samples: 1
No_Obesity
samples: 2
Obesity
samples: 18
Obesity
samples: 45
Age<=35.8717
samples: 6
No_Obesity
samples: 4
Obesity
samples: 2

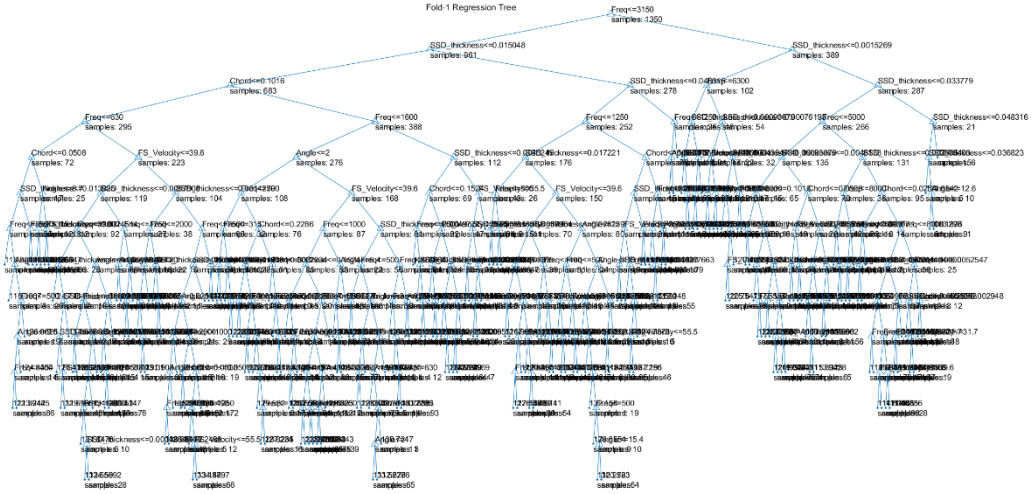

Fold-7 Classification Tree
Weight<=93.1984
samples: 1899
Weight<=75.1466
samples: 1172
Weight<=102
samples: 727
MTRANS_Motorbike<=0.5
samples: 678
Height<=1.6871
samples: 494
Height<=1.7861
samples: 108
Obesity
samples: 619
No_Obesity
samples: 672
Age<=31
samples: 6
Height<=1.6291
samples: 214
Weight<=89.9994
samples: 280
Obesity
samples: 82
Age<=37.0239
samples: 26
No_Obesity
samples: 5
Obesity
samples: 1
Obesity
samples: 95
Weight<=81.5004
samples: 119
Age<=51.4163
samples: 257
Height<=1.7414
samples: 23
No_Obesity
samples: 23
Obesity
samples: 3
Height<=1.63
samples: 49
Age<=32.576
samples: 70
No_Obesity
samples: 254
Age<=53.5
samples: 3
Obesity
samples: 5
No_Obesity
samples: 18
Weight<=78.5
samples: 3
No_Obesity
samples: 46
Obesity
samples: 63
Height<=1.6753
samples: 7
Obesity
samples: 1
No_Obesity
samples: 2
No_Obesity
samples: 2
Obesity
samples: 1
Obesity
samples: 3
No_Obesity
samples: 4

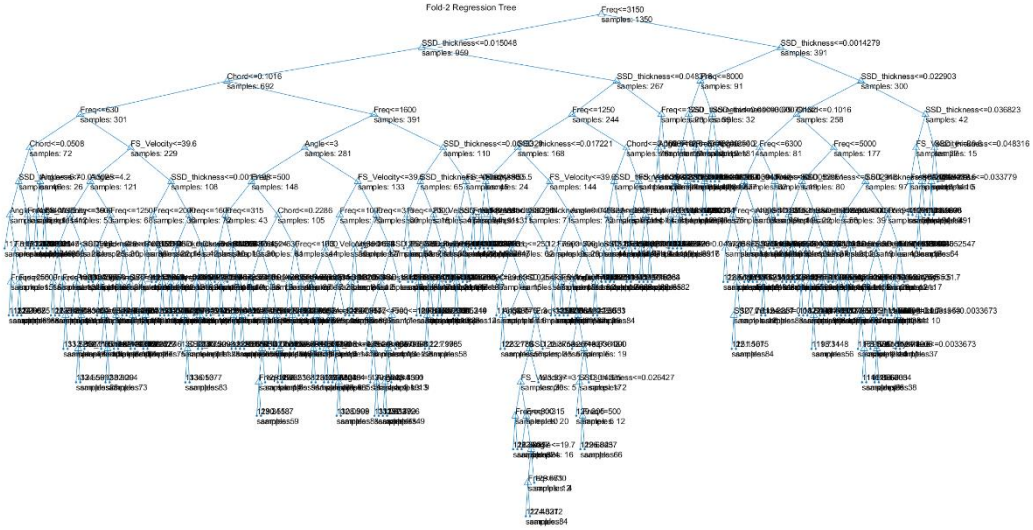

Fold-8 Classification Tree
Weight<=93.1984
samples: 1899
Weight<=75.1466
samples: 1163
Weight<=102
samples: 736
MTRANS_Motorbike<=0.5
samples: 670
Height<=1.695
samples: 493
Height<=1.7962
samples: 112
Obesity
samples: 624
No_Obesity
samples: 663
Age<=31
samples: 7
Height<=1.6295
samples: 219
Weight<=89.9972
samples: 274
Age<=17.2956
samples: 90
Age<=37.6441
samples: 22
No_Obesity
samples: 6
Obesity
samples: 1
Obesity
samples: 91
Weight<=81.5004
samples: 128
No_Obesity
samples: 247
Height<=1.7414
samples: 27
Age<=17.1261
samples: 4
Obesity
samples: 86
No_Obesity
samples: 21
Obesity
samples: 1
No_Obesity
samples: 53
Age<=32.676
samples: 75
Obesity
samples: 4
No_Obesity
samples: 23
Obesity
samples: 3
No_Obesity
samples: 1
Obesity
samples: 66
FCVC<=2
samples: 9
No_Obesity
samples: 5
Obesity
samples: 4

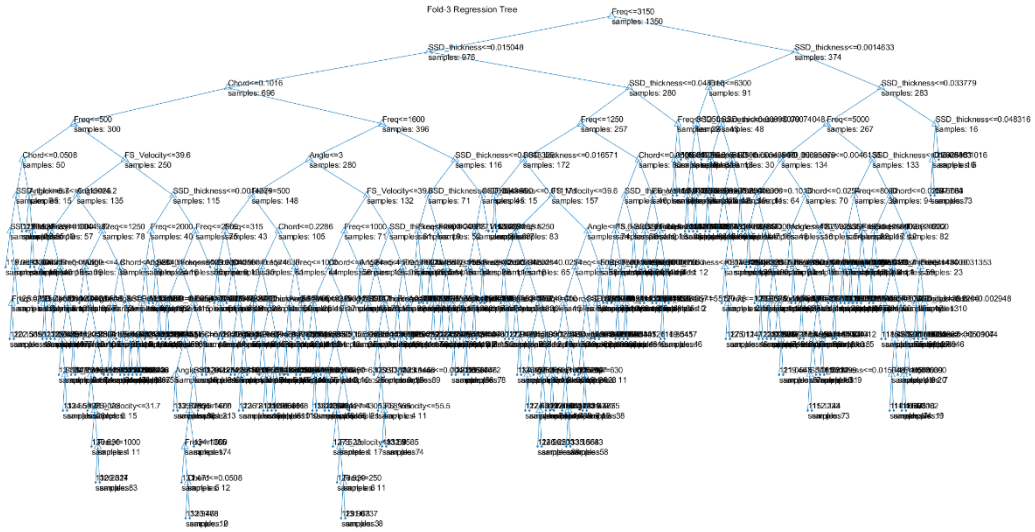

Fold-9 Classification Tree
Weight<=93.1984
samples: 1899
Weight<=75.1466
samples: 1172
Weight<=102
samples: 727
MTRANS_Motorbike<=0.5
samples: 684
Height<=1.695
samples: 488
Height<=1.7983
samples: 110
Obesity
samples: 617
No_Obesity
samples: 677
Age<=31
samples: 7
Height<=1.6295
samples: 218
Weight<=90.3194
samples: 270
Obesity
samples: 88
No_Obesity
samples: 22
No_Obesity
samples: 6
Obesity
samples: 1
Obesity
samples: 93
Weight<=81.5004
samples: 125
No_Obesity
samples: 257
Height<=1.7614
samples: 13
Height<=1.63
samples: 49
Age<=32.576
samples: 76
Obesity
samples: 3
No_Obesity
samples: 10
Weight<=78.5
samples: 3
No_Obesity
samples: 46
Obesity
samples: 67
FCVC<=2
samples: 9
No_Obesity
samples: 2
Obesity
samples: 1
No_Obesity
samples: 5
Obesity
samples: 4

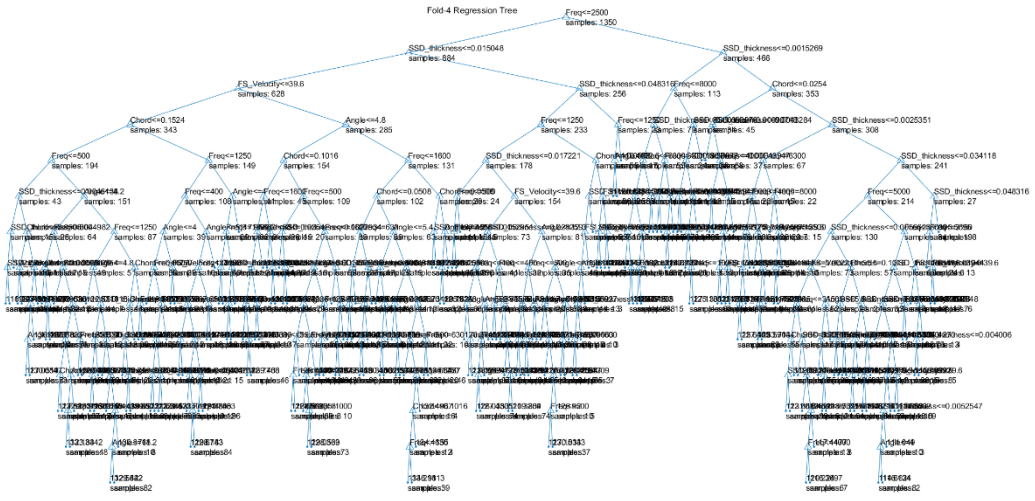

Fold-10 Classification Tree
Weight<=93.8444
samples: 1899
Weight<=75.1466
samples: 1176
Weight<=102.0001
samples: 723
MTRANS_Motorbike<=0.5
samples: 687
Height<=1.695
samples: 489
Height<=1.7982
samples: 105
Obesity
samples: 618
No_Obesity
samples: 681
Age<=31
samples: 6
Height<=1.6295
samples: 210
Weight<=89.9994
samples: 279
Age<=17.4262
samples: 83
Age<=37.6441
samples: 22
No_Obesity
samples: 5
Obesity
samples: 1
Obesity
samples: 87
Weight<=81.5004
samples: 123
No_Obesity
samples: 248
Height<=1.75
samples: 31
Age<=17.1261
samples: 4
Obesity
samples: 79
No_Obesity
samples: 21
Obesity
samples: 1
Height<=1.63
samples: 57
Age<=32.576
samples: 66
Obesity
samples: 7
No_Obesity
samples: 24
Obesity
samples: 3
No_Obesity
samples: 1
Weight<=78.5
samples: 3
No_Obesity
samples: 54
Obesity
samples: 60
Age<=39.6686
samples: 6
No_Obesity
samples: 2
Obesity
samples: 1
No_Obesity
samples: 4
Obesity
samples: 2

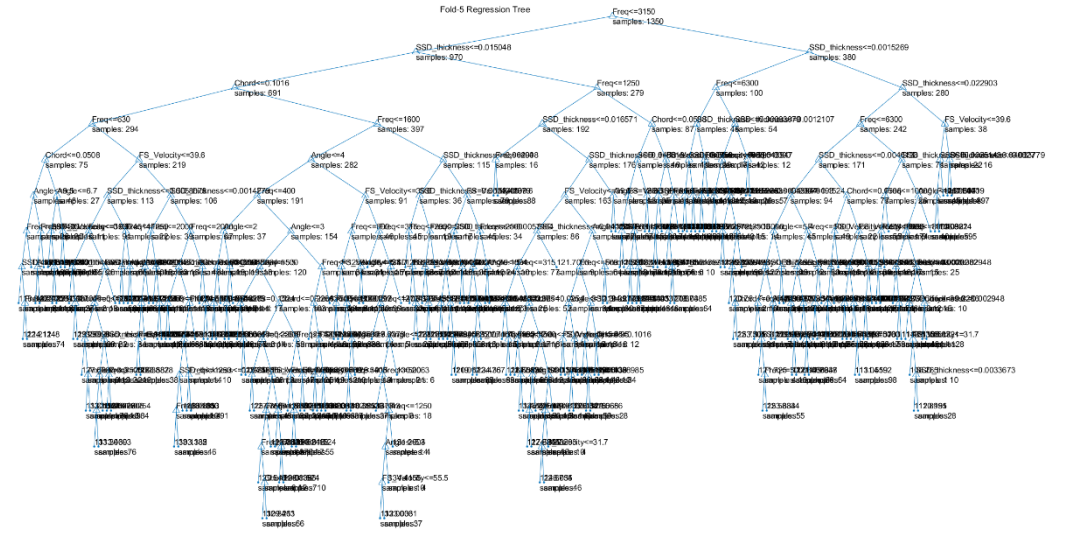
Appendix b: regression 10-fold tree drawings

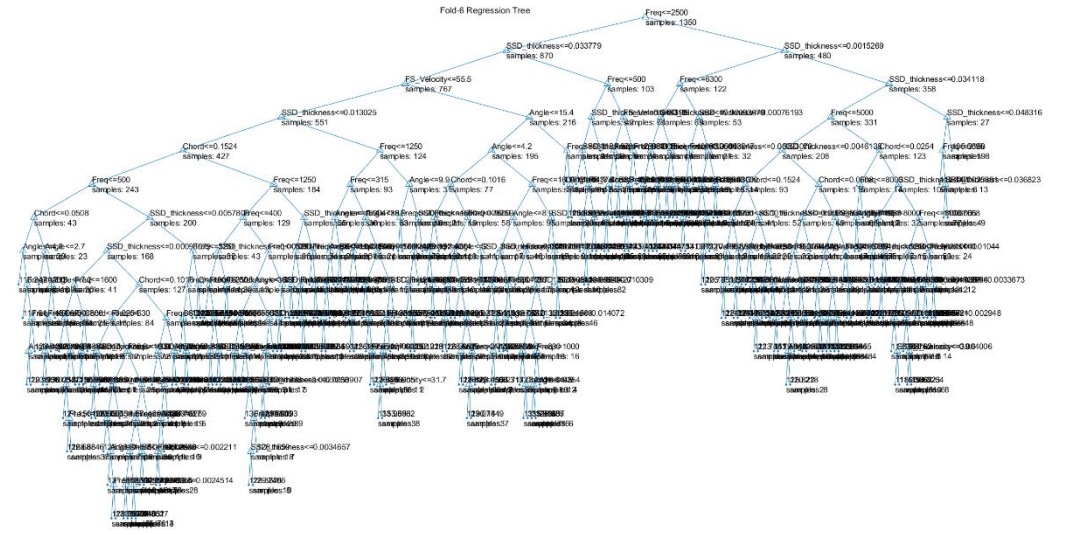


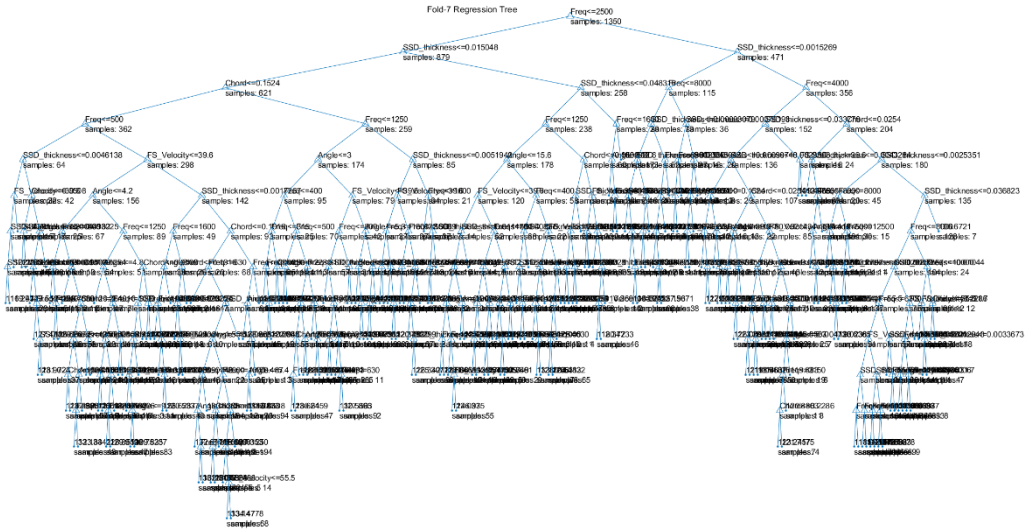


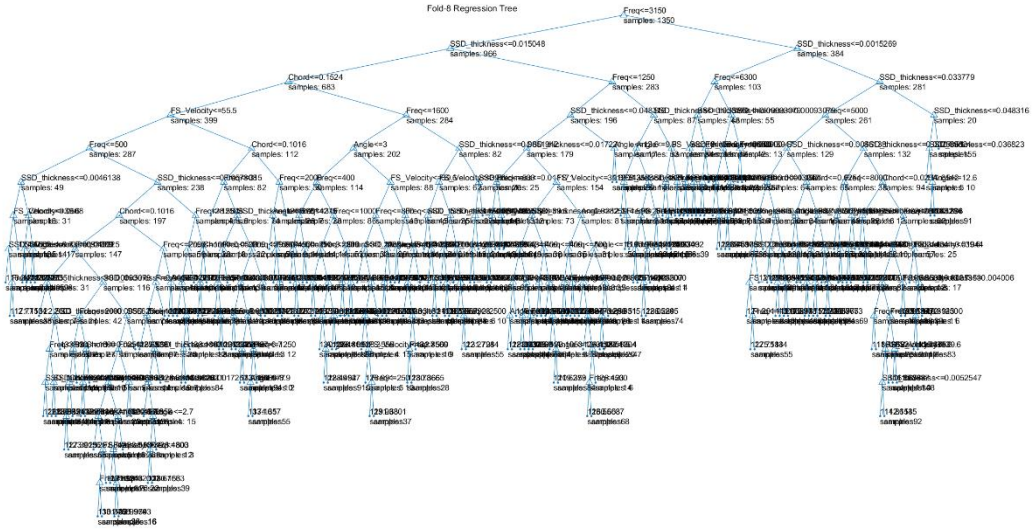

Fold-3 Regression Tree
Freq<=3150
samples: 1350
SSD_thickness<=0.015048
samples: 976
SSD_thickness<=0.0014633
samples: 374
Chord<=0.1016
samples: 696
SSD_thickness<=0.048316
samples: 280
SSD_thickness<=0.033779
samples: 283
Freq<=500
samples: 300
Freq<=1600
samples: 396
Freq<=1250
samples: 257
Freq<=5000
samples: 267
SSD_thickness<=0.048316
samples: 16
Chord<=0.0508
samples: 50
FS_Velocity<=39.6
samples: 250
Angle<=3
samples: 280
SSD_thickness<=0.016571
samples: 172
SSD_thickness<=0.0046153
samples: 133

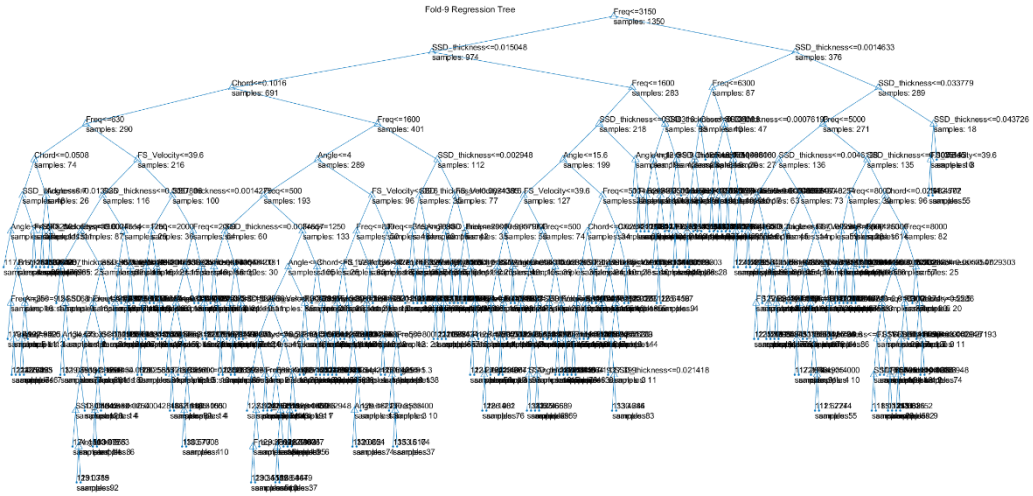

Fold-4 Regression Tree
Freq<=2500
samples: 1350
SSD_thickness<=0.015048
samples: 884
SSD_thickness<=0.0015269
samples: 466
FS_Velocity<=39.6
samples: 628
SSD_thickness<=0.048316
samples: 256
Freq<=8000
samples: 113
Chord<=0.0254
samples: 353
Chord<=0.1524
samples: 343
Angle<=4.8
samples: 285
Freq<=1250
samples: 233
SSD_thickness<=0.0025351
samples: 308
Freq<=500
samples: 194
Freq<=1250
samples: 149
Chord<=0.1016
samples: 154
Freq<=1600
samples: 131
SSD_thickness<=0.017221
samples: 178
SSD_thickness<=0.034118
samples: 241
Freq<=5000
samples: 214
SSD_thickness<=0.048316
samples: 27

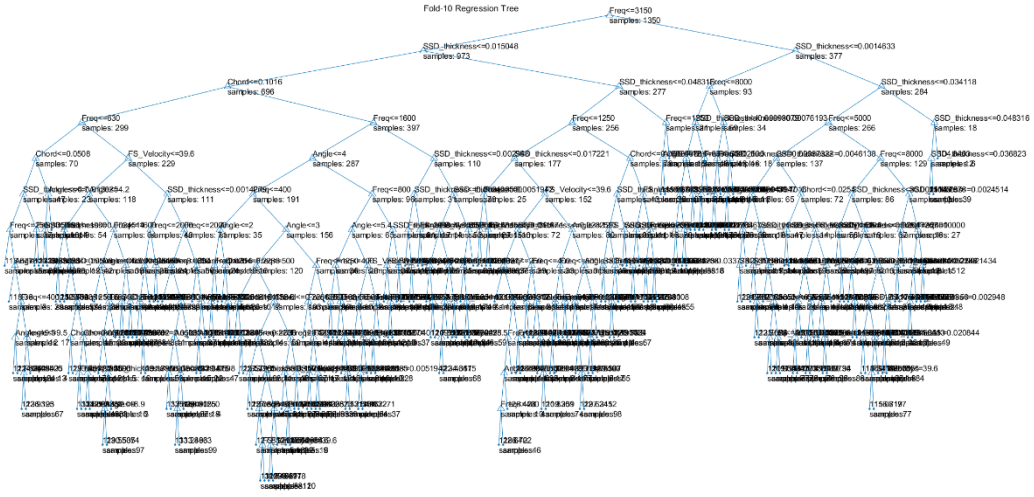

Fold-5 Regression Tree
Freq<=3150
samples: 1350
SSD_thickness<=0.015048
samples: 970
SSD_thickness<=0.0015269
samples: 380
Chord<=0.1016
samples: 691
Freq<=1250
samples: 279
Freq<=6300
samples: 100
SSD_thickness<=0.022903
samples: 280
Freq<=630
samples: 294
Freq<=1600
samples: 397
SSD_thickness<=0.016571
samples: 192
Freq<=6300
samples: 242
FS_Velocity<=39.6
samples: 38
Chord<=0.0508
samples: 75
FS_Velocity<=39.6
samples: 219
Angle<=4
samples: 282
Freq<=6300
samples: 171


Fold-6 Regression Tree
Freq<=2500
samples: 1350
SSD_thickness<=0.033779
samples: 870
SSD_thickness<=0.0015269
samples: 480
FS_Velocity<=55.5
samples: 767
Freq<=600
samples: 103
Freq<=6300
samples: 122
SSD_thickness<=0.034118
samples: 358
SSD_thickness<=0.013025
samples: 551
Angle<=15.4
samples: 216
Freq<=5000
samples: 331
SSD_thickness<=0.048316
samples: 27
Chord<=0.1524
samples: 427
Freq<=1250
samples: 124
Angle<=4.2
samples: 196
Freq<=500
samples: 243
Freq<=1250
samples: 184
Freq<=315
samples: 93
Angle<=9.9
samples: 31
Chord<=0.1016
samples: 77
Chord<=0.0508
samples: 43
SSD_thickness<=0.00578
samples: 200
Freq<=400
samples: 129


Fold-7 Regression Tree
Freq<=2500
samples: 1350
SSD_thickness<=0.015048
samples: 879
SSD_thickness<=0.0015269
samples: 471
Chord<=0.1524
samples: 621
SSD_thickness<=0.0048
samples: 258
Freq<=8000
samples: 115
Freq<=4000
samples: 356
Freq<=500
samples: 362
Freq<=1250
samples: 259
Freq<=1250
samples: 238
SSD_thickness<=0.033276
samples: 152
Chord<=0.0254
samples: 204
SSD_thickness<=0.0046138
samples: 64
FS_Velocity<=39.6
samples: 298
Angle<=3
samples: 174
SSD_thickness<=0.0051941
samples: 85
Angle<=15.8
samples: 178
SSD_thickness<=0.0025351
samples: 180
Angle<=4.2
samples: 156
SSD_thickness<=0.0017
samples: 95
SSD_thickness<=0.036823
samples: 135
Freq<=1250
samples: 89
Freq<=1600
samples: 49
Chord<=0.1524
samples: 93
Freq<=12500
samples: 7
SSD_thickness<=0.0033673
samples: 12


Fold-8 Regression Tree
Freq<=3150
samples: 1350
SSD_thickness<=0.015048
samples: 966
SSD_thickness<=0.0015269
samples: 384
Chord<=0.1524
samples: 683
Freq<=1250
samples: 283
Freq<=6300
samples: 103
SSD_thickness<=0.033779
samples: 281
FS_Velocity<=55.5
samples: 399
Freq<=1600
samples: 284
SSD_thickness<=0.048316
samples: 20
Freq<=500
samples: 287
Chord<=0.1016
samples: 112
Angle<=3
samples: 202
SSD_thickness<=0.036823
samples: 155
SSD_thickness<=0.0046138
samples: 49
Chord<=0.1016
samples: 197
Freq<=8000
samples: 58
Chord<=0.0254
samples: 94
Chord<=0.1524
samples: 147
Freq<=2000
samples: 42
Freq<=1250
samples: 55
Freq<=1250
samples: 74
Freq<=1250
samples: 55
Freq<=1250
samples: 68
SSD_thickness<=0.0052547
samples: 92


Fold-9 Regression Tree
Freq<=3150
samples: 1350
SSD_thickness<=0.015048
samples: 974
SSD_thickness<=0.0014633
samples: 376
Chord<=0.1016
samples: 591
Freq<=1600
samples: 283
Freq<=6300
samples: 87
SSD_thickness<=0.033779
samples: 289
Freq<=830
samples: 290
Freq<=1600
samples: 401
SSD_thickness<=0.043726
samples: 18
Chord<=0.0508
samples: 74
FS_Velocity<=39.6
samples: 216
Angle<=4
samples: 289
SSD_thickness<=0.002948
samples: 112
Angle<=15.6
samples: 199
SSD_thickness<=0.0046138
samples: 135
FS_Velocity<=39.6
samples: 127
Freq<=8000
samples: 55
Freq<=500
samples: 193
FS_Velocity<=39.6
samples: 96
Chord<=0.0254
samples: 63
SSD_thickness<=0.0024514
samples: 96
SSD_thickness<=0.0076193
samples: 47
Freq<=5000
samples: 271
SSD_thickness<=0.0046138
samples: 136
Freq<=8000
samples: 73
Chord<=0.0254
samples: 82
Freq<=1250
samples: 256
SSD_thickness<=0.0046138
samples: 137
SSD_thickness<=0.0024514
samples: 86
SSD_thickness<=0.0076193
samples: 34
Freq<=5000
samples: 266
SSD_thickness<=0.048316
samples: 18
SSD_thickness<=0.0046138
samples: 129
SSD_thickness<=0.0024514
samples: 65
Freq<=8000
samples: 72
SSD_thickness<=0.0024514
samples: 64
SSD_thickness<=0.036823
samples: 25
SSD_thickness<=0.0046138
samples: 18
SSD_thickness<=0.0024514
samples: 39
SSD_thickness<=0.0046138
samples: 27


Fold-10 Regression Tree
Freq<=3150
samples: 1350
SSD_thickness<=0.015048
samples: 973
SSD_thickness<=0.0014633
samples: 377
Chord<=0.1016
samples: 896
SSD_thickness<=0.048316
samples: 277
Freq<=8000
samples: 93
SSD_thickness<=0.034118
samples: 284
Freq<=830
samples: 299
Freq<=1600
samples: 397
Freq<=1250
samples: 256
Freq<=5000
samples: 266
SSD_thickness<=0.048316
samples: 18
Chord<=0.0508
samples: 70
FS_Velocity<=39.6
samples: 229
Angle<=4
samples: 287
SSD_thickness<=0.002948
samples: 110
SSD_thickness<=0.017221
samples: 177
SSD_thickness<=0.0046138
samples: 129
SSD_thickness<=0.036823
samples: 25
SSD_thickness<=0.0014654
samples: 111
SSD_thickness<=0.0014654
samples: 400
samples: 191
Freq<=800
samples: 96
SSD_thickness<=0.0051743
samples: 25
FS_Velocity<=39.6
samples: 152
SSD_thickness<=0.0046138
samples: 137
SSD_thickness<=0.0024514
samples: 86
Chord<=0.0254
samples: 72
SSD_thickness<=0.0024514
samples: 65
Angle<=3
samples: 156
Angle<=5.4
samples: 72
SSD_thickness<=0.0024514
samples: 27
SSD_thickness<=0.0046138
samples: 18
SSD_thickness<=0.0024514
samples: 39
Angle<=9.5
SSD_thickness<=0.0046138
samples: 59
SSD_thickness<=0.020844
samples: 49
SSD_thickness<=0.0046138
samples: 64