DigiOnco: A Pipeline to Unveil Digital Non-Invasive Biomarkers from Multi-parametric Radiomics Footprints

Santhi Natarajan, Anand Ravishankar, Bharathi Malakreddy A[a], G.Lohith, Kritika Sekar, Shivakumar Swamy, Kumar Kallur, Basavalinga Ajai Kumar, Mahesh Bandimegal, Krithika Murugan[b]

[a]*BMS Institute of Technology and Management, Visweswaraiah Technological Univesity, Bangalore, India*
[b]*Health Care Global Hospitals, Bangalore, India*

1. Appendix

1.1. Data Acquisition

1.2. Data Interpretation

Figure 1 breaks down the data interpretation process into a conglomerate of submodules which function serially to perform the classification task. The images obtained from the previous process are stored in a cataloging structure for easy access. Note that the images are tagged with

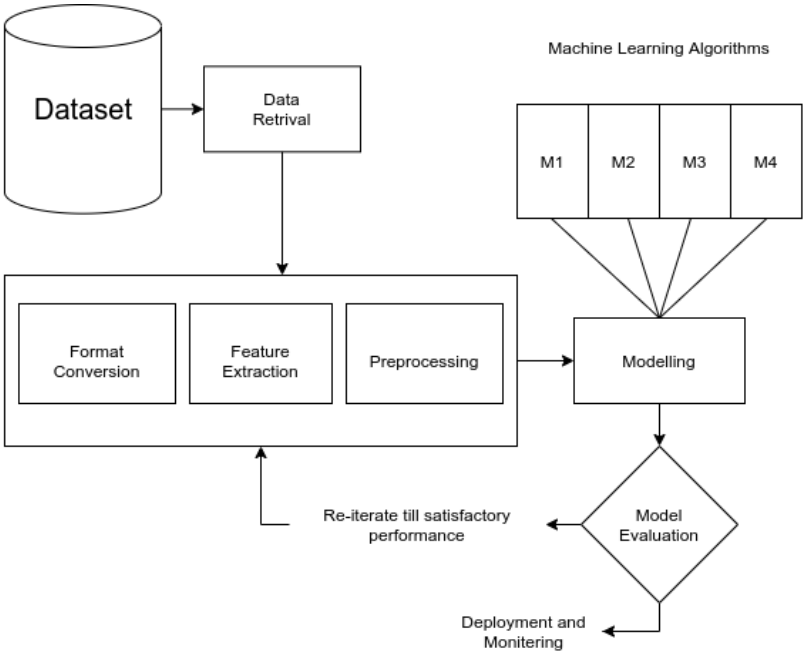

Figure 1: Flowchart showing the steps involved in obtianing the final results

their target class by a specialist beforehand. The images are converted to a suitable format for feature extraction. The end product is obtained in the form of a structured dataset consisting of the target class and an array of feature-filter mapping. Preprocessing techniques are applied to the normalize the values throughout the dataset and to remove any redundant entries. Principal Component Analysis (PCA) was applied to reduce the feature space whilst witholding components with the highest information content. The dataset is shuffled randomly and split into two categories (training and testing sets) in a ratio of 70:30. Multiple machine learning algorithms are queued onto the training set and the results are tabulated accordingly. The model evaluation process is repeated till either a set number of iterations are completed or till the performance metrics are acceptable. Once the parameters corresponding to the optimal performance are achieved, the test set is re-introduced into the model. Using the tuned parameters, the model makes a prediction on the test set and a score is generated with the test set's inital targets baseline. Provided a sufficient scored is obtained, statistical inferences can be drawn from the testing phase.

The preceding process is performed for 2 classification tasks: TN vs Non-TN, and HER vs Luminal-A vs TN. Tables 1 and 2 showcase the optimal feature-filter mapping for both the tasks respectively. Note that this mapping is a subset of the set of features and filters described in the main material.

1.3. Statistical Analysis

Figures 2 to 9 depict the box and whisker plots obtained for the feature-filter mapping specified respectively. These plots summarize the variability and distribution of the select variable. Note that even though the feature-filter mapping is optimal, the data distribution might be too similar for a naked eye observation to make a clear distinction. Clear examples for this can be found in figures corresponding to entries 3, 11, and 13 in Table 1 and entries 1, 4, 5, and 15 in Table 2.

Table 1: **Final Feature-Filter Mapping**: TN vs Non-TN

Sr. No.	**Feature**	**Filter**
1	GLCM_Cluster_Shade	LoG
2	GLCM_Cluster_Shade	Nil
3	FirstOrder_Minimum	Square
4	NGTDM_Coarseness	Square
5	FirstOrder_Mean	Wavelet = HHH
6	FirstOrder_Skewness	Wavelet = HHH
7	GLCM_Correlation	Wavelet = HHH
8	FirstOrder_Energy	Wavelet = HHL
9	FirstOrder_Energy	Wavelet = HLH
10	FirstOrder_Mean	Wavelet = HLH
11	FirstOrder_Skewness	Wavelet = HLH
12	FirstOrder_Energy	Wavelet = HLL
13	GLCM_Cluster_Shade	Wavelet = LHH
14	FirstOrder_Energy	Wavelet = LLH
15	FirstOrder_Skewness	Wavelet = LLH
16	FirstOrder_Energy	Wavelet = LLL

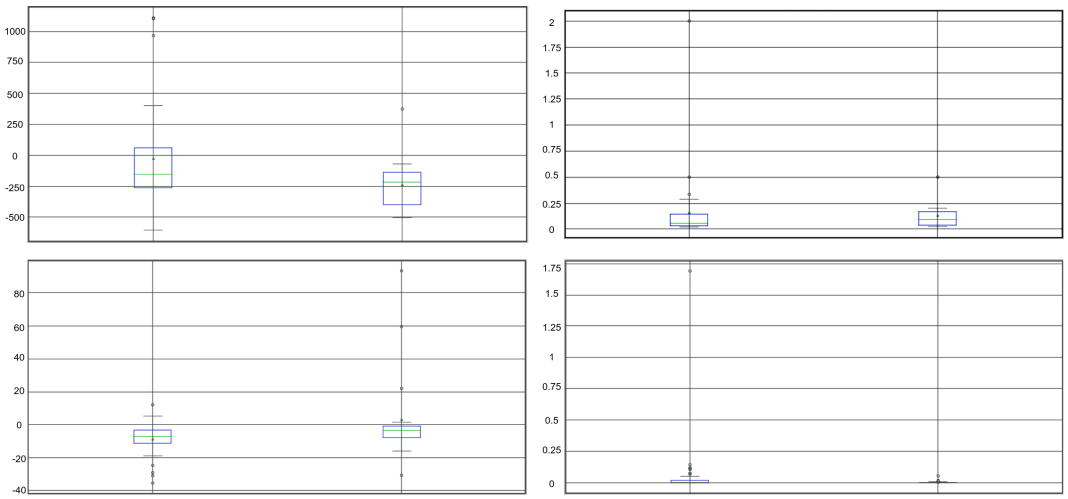

Figure 2: From top left, counter-clockwise, entries 1-4 of Table 1 depicting the distribution of select feature-filter mapping for TN (left) vs Non-TN (right)

2. Algorithms Employed

The data is classified based the features which contrast the classes with the highest information content. The process of data classification using Random Forest is shown in Algorithm 1. AdaBoost solves the problem of overfitting by presenting the system with the misclassified data and forcing it to improve the overall performance. The two flavours of AdaBoost i.e, SAMME

Table 2: **Final Feature-Filter Mapping**:HER vs Luminal-A vs TN

Sr. No.	**Feature**	**Filter**
1	FirstOrder_Kurtosis	Exponential
2	GLCM_Cluster_Shade	LoG
3	GLCM_Cluster_Shade	Nil
4	GLCM_Cluster_Prominence	Square
5	GLCM_Cluster_Shade	Square
6	GLCM_Cluster_Tendency	Square
7	GLCM_MCC	Square
8	FirstOrder_Energy	Wavelet = HHL
9	FirstOrder_Mean	Wavelet = HHL
10	FirstOrder_Energy	Wavelet = HLL
11	FirstOrder_Skewness	Wavelet = HLL
12	FirstOrder_Energy	Wavelet = LHH
13	FirstOrder_Energy	Wavelet = LHL
14	FirstOrder_Skewness	Wavelet = LLH
15	GLCM_Cluster_Prominence	Wavelet = LLH

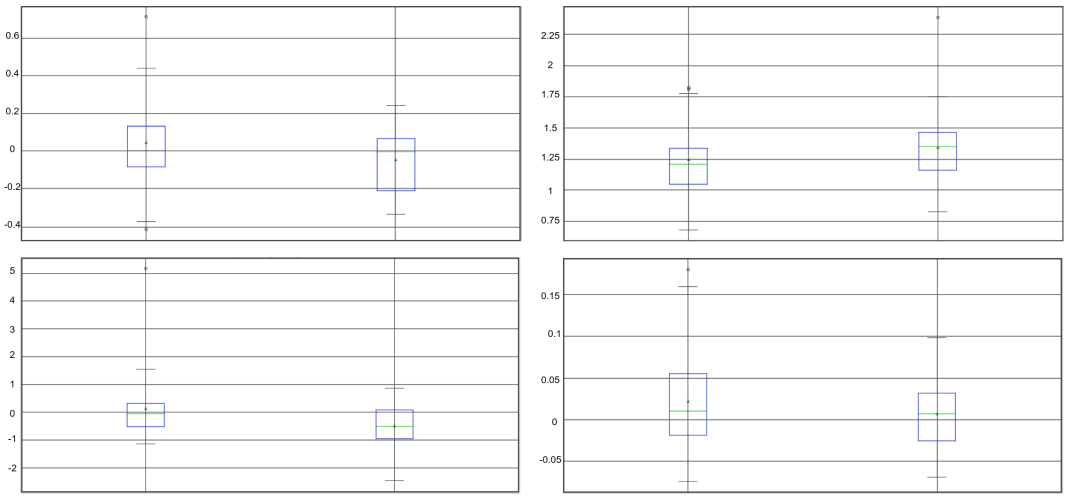

Figure 3: From top left, counter-clockwise, entries 5-8 of Table 1 depicting the distribution of select feature-filter mapping for TN (left) vs Non-TN (right)

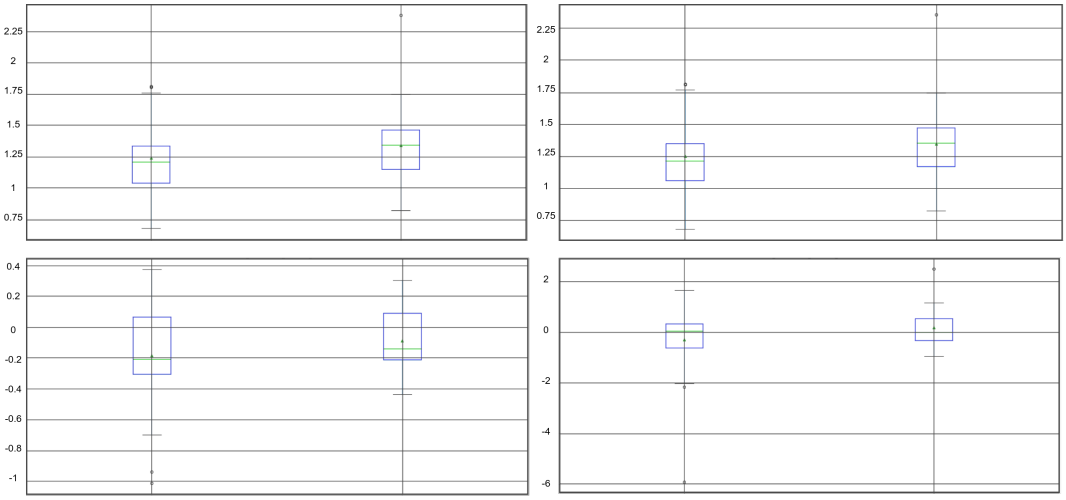

Figure 4: From top left, counter-clockwise, entries 9-12 of Table 1 depicting the distribution of select feature-filter mapping for TN (left) vs Non-TN (right)

and SAMME.R have been descibed in Algorithms 2 and 3. SFORCE combines the strength of Random Forests and takes care of the drawbacks by using a Boosting algorithm to make the search process more concentrated as shown in Algorithm 4.

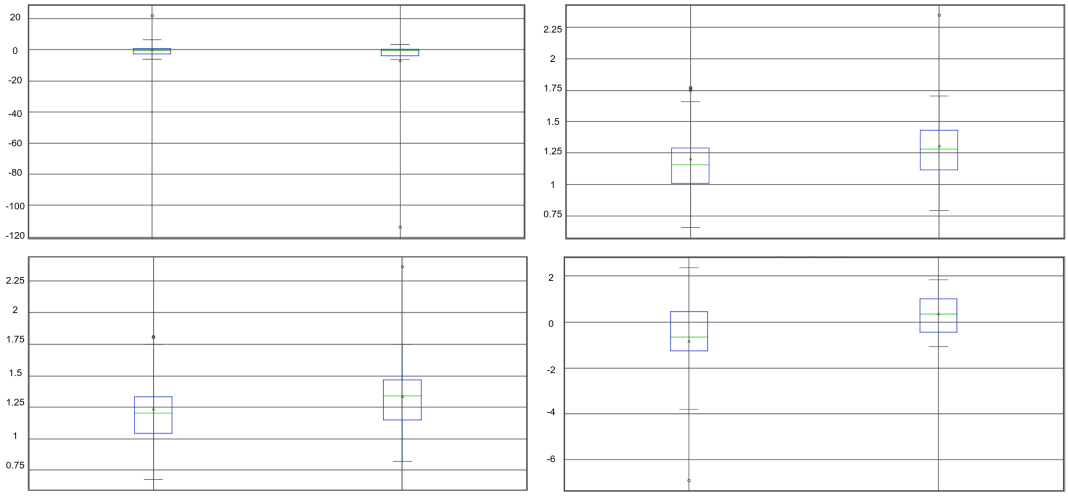

Figure 5: From top left, counter-clockwise, entries 13-16 of Table 1 depicting the distribution of select feature-filter mapping for TN (left) vs Non-TN (right)

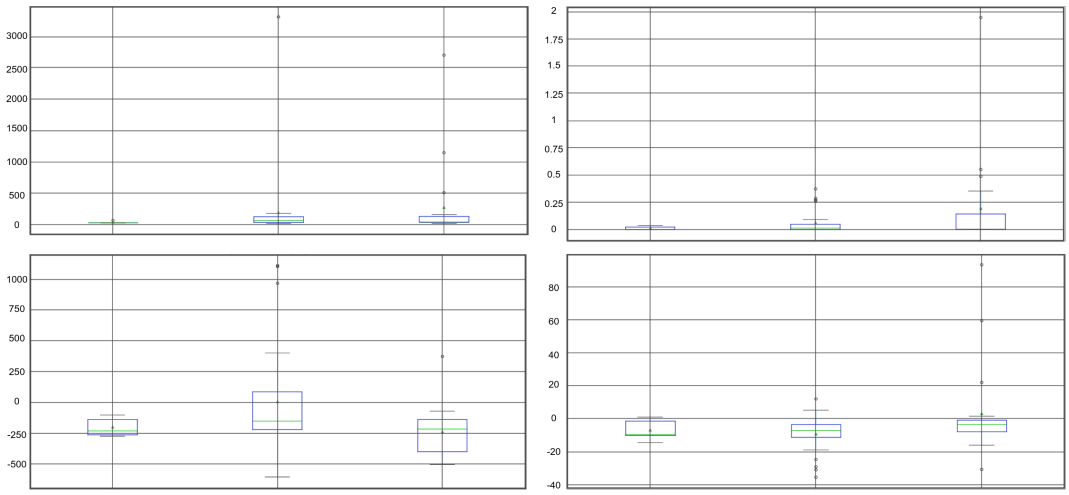

Figure 6: From top left, counter-clockwise, entries 1-4 of Table 2 depicting the distribution of select feature-filter mapping for HER (left) vs Luminal-A (center) vs TN (right)

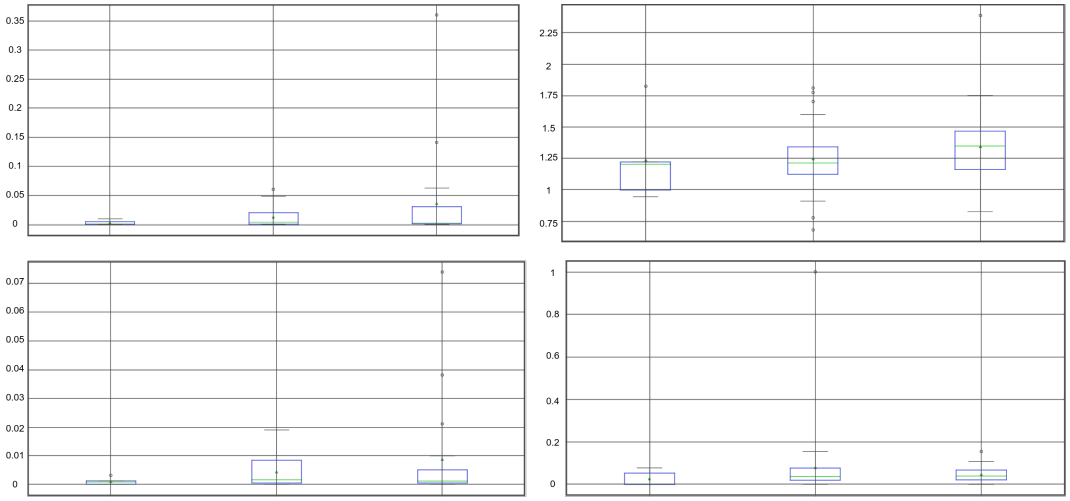

Figure 7: From top left, counter-clockwise, entries 5-8 of Table 1 depicting the distribution of select feature-filter mapping for HER (left) vs Luminal-A (center) vs TN (right)

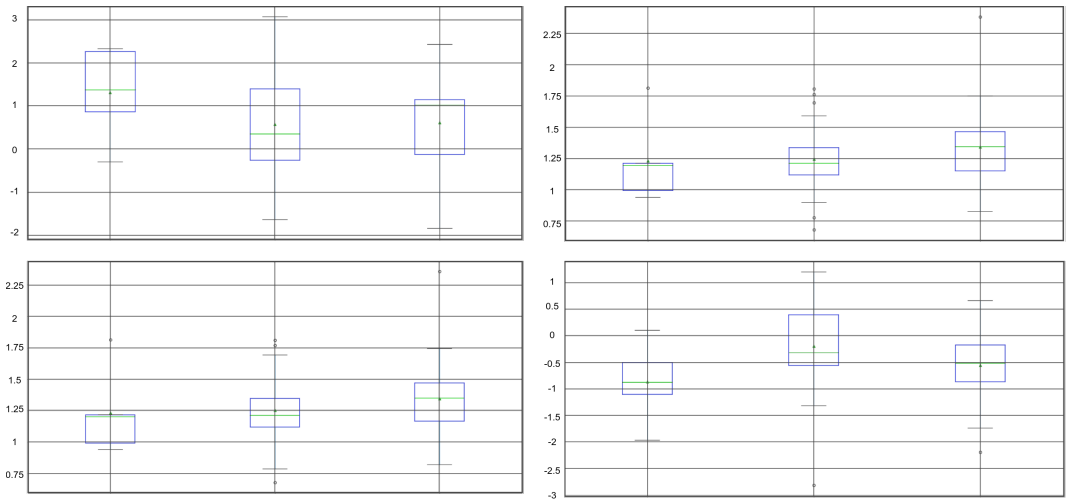

Figure 8: From top left, counter-clockwise, entries 9-12 of Table 1 depicting the distribution of select feature-filter mapping for HER (left) vs Luminal-A (center) vs TN (right)

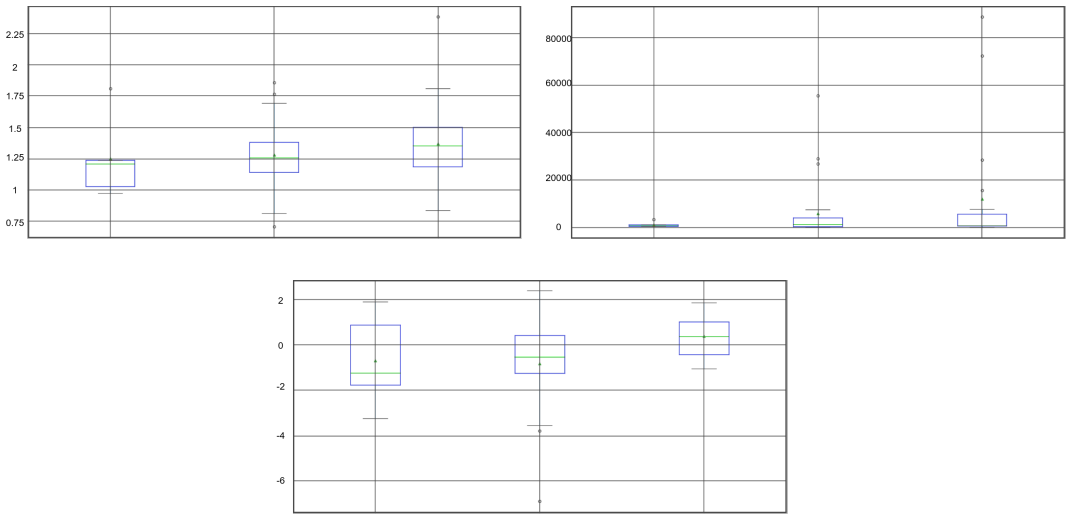

Figure 9: From top left, counter-clockwise, entries 13-15 of Table 1 depicting the distribution of select feature-filter mapping for HER (left) vs Luminal-A (center) vs TN (right)

Algorithm 1 : Ensemble Learning: Random Forest

// Input: Data Set D = $\{(x_1, y_1),(x_2, y_2), \ldots \ ((x_m, y_m)\}$, Feature Set F, Randomization Factor R, Number of trees T
// Output: Root node of i^{th} tree
- -
for $\forall i \in \{1, 2, \ldots T\}$ **do**
  $N_i \leftarrow$ Root node of i^{th} tree
  if All targets belong to same class i.e y_i or $F \in \emptyset$ **then**
    Return N_i
  end if
  $D_i \leftarrow$ bootstraped sample from D
  for Each node **do**
    $f \leftarrow$ Randomly selected R features from F
    $N_f \leftarrow$ Best Feature from f features
    $N_p \leftarrow$ Best Split based on N_f
  end for
end for
return N_i

Algorithm 2 : Stagewise Additive Modeling: SAMME

// Input: Data Set D = $\{(x_1, y_1),(x_2, y_2), \ldots \ ((x_m, y_m)\}$, Number of Learning Rounds T, Learning Algorithm ϵ
// Output: sign$(\sum_{t=1}^{T} \alpha_t.C_t)$
- -
$D_1(x) = 1/m$ {Initialize the weight distribution}
for $t = \{1, 2, \ldots T\}$ **do**
  $C_t = \epsilon$(D,D_t) {Create classifier C_t}
  $e_t = P_{x\sim D}(h_t(x) \neq f(x))$ {Calculate error e_t}
  $\alpha_t = log\dfrac{1 - e_t}{e_t} + \log(K\text{-}1)$ {Calculate the weight h_t}
  $D_i(x) \leftarrow D_i(x).\exp(\alpha_t.\mathrm{P}(C_i \neq f(x)))$ {Update the distribution D_t}, $i = \{1, 2, \ldots m\}$
  Renormalize $D_t(x)$
end for

Algorithm 3 : Stagewise Additive Modeling for Real Value Predictions: SAMME.R

// Input: Data Set D = $\{(x_1, y_1),(x_2, y_2), \ldots \ ((x_m, y_m)\}$, Number of Learning Rounds T, Learning Algorithm ϵ
// Output: sign$(\sum_{t=1}^{T} \alpha_t.C_t)$
- -
$D_1(x)$ = 1/m {Initialize the weight distribution}
for $t = \{1, 2, \ldots T\}$ **do**
    $C_t = \epsilon$(D,D_t) {Create classifier C_t}
    $p_{kt}(x)$ = Prob($y = k|x$), $k = \{1, 2, \ldots K\}$
    $h_{kt}(x) \leftarrow (K$ - 1)$(log p_{kt}(x)$ - $\frac{1}{K}.\sum_{k'} log p_{k't}$(x))
    $D_i(x) \leftarrow D_i(x)$.exp$(\frac{1-K}{K}.y_i^{\mathbf{T}}.log(p_t(x_i)))$ {Update the distribution D_t, $i = \{1, 2, \ldots m\}$ }
    Renormalize $D_t(x)$
end for

Algorithm 4 : Ensemble of Ensemble: SFORCE

// Input: Data Set D = $\{(x_1, y_1),(x_2, y_2), \ldots \ ((x_m, y_m)\}$, Feature Set F, Randomization Factor R, Number of trees T,Number of Learning Rounds T', Learning Algorithm ϵ
// Output: Root node of i$^{\text{th}}$ Boosted Tree
- -
Random Forest
for $\forall i \in \{1, 2, \ldots T\}$ **do**
    $N_i \leftarrow$ Root node of i$^{\text{th}}$ tree
    if All targets belong to same class i.e y_i or $F \in \emptyset$ **then**
        Call SAMME.R with N_i
    end if
    $D^i \leftarrow$ bootstraped sample from D
    for Each node **do**
        $f \leftarrow$ Randomly selected R features from F
        $N_f \leftarrow$ Best Feature from f features
        $N_p \leftarrow$ Best Split based on N_f
        Call SAMME.R with N_i
    end for
end for
return N_i
- -
SAMME/SAMME.R
$D_1(x)$ = 1/m {Initialize the weight distribution}
for $t = \{1, 2, \ldots T\}$ **do**
    $C_t = \epsilon$(D,D_t) {Create classifier C_t}
    $p_{kt}(x)$ = Prob($y = k|x$), $k = \{1, 2, \ldots K\}$
    $h_{kt}(x) \leftarrow (K$ - 1)$(log p_{kt}(x)$ - $\frac{1}{K}.\sum_{k'} log p_{k't}$(x))
    $D_i(x) \leftarrow D_i(x)$.exp$(\frac{1-K}{K}.y_i^{\mathbf{T}}.log(p_t(x_i)))$ {$i = \{1, 2, \ldots m\}$}
    Renormalize $D_t(x)$
    Call Random Forest with $(\sum_{t=1}^{T'} \alpha_t.C_t)$
end for

Algorithm 5 DigiOnco: Algorithmic Flow

```
//Input Image dataset D_n and masks D_m
//Output Predicted Class
for Each image i in D_n do
    Convert image to a suitable format using conversion software
    Call the pre-processsing techniques on the formatted images
    Using mask j for corresponding i, extract radiomics features
    Create a grid of p-values
    for EACH value in grid do
        Call Algorithm 4 with related feature set
    end for
    Obtain accuracy levels and digital bio-markers
end for
```