MLW-gcForest: A Multi-Weighted gcForest Model for Cancer Subtype Classification by Methylation Data

Yunyun Dong 1,2, Wenkai Yang 1, Jiawen Wang 1, Juanjuan Zhao 1 and Yan Qiang 1,*

1 College of Information and Computer, Taiyuan University of Technology, Taiyuan 030024, China
2 College of Information Technology and Engineering, Jinzhong University, Jinzhong 030619, China
* Correspondence: qiangyan@tyut.edu.cn; Tel.: +86-186-3516-8680

Received: 29 July 2019; Accepted: 27 August 2019; Published: 2 September 2019

Abstract: Effective cancer treatment requires a clear subtype. Due to the small sample size, high dimensionality, and class imbalances of cancer gene data, classifying cancer subtypes by traditional machine learning methods remains challenging. The gcForest algorithm is a combination of machine learning methods and a deep neural network and has been indicated to achieve better classification of small samples of data. However, the gcForest algorithm still faces many challenges when this method is applied to the classification of cancer subtypes. In this paper, we propose an improved gcForest algorithm (MLW-gcForest) to study the applicability of this method to the small sample sizes, high dimensionality, and class imbalances of genetic data. The main contributions of this algorithm are as follows: (1) Different weights are assigned to different random forests according to the classification ability of the forests. (2) We propose a sorting optimization algorithm that assigns different weights to the feature vectors generated under different sliding windows. The MLW-gcForest model is trained on the methylation data of five data sets from the cancer genome atlas (TCGA). The experimental results show that the MLW-gcForest algorithm achieves high accuracy and area under curve (AUC) values for the classification of cancer subtypes compared with those of traditional machine learning methods and state of the art methods. The results also show that methylation data can be effectively used to diagnose cancer.

Keywords: MLW-gcForest; TCGA; methylation data; cancer subtype

1. Introduction

Cancer is a heterogeneous disease and is the leading cause of death worldwide [1,2]. Most cancers have different subtypes that correspond to different prognoses [3–7]. The identification of cancer subtypes can provide valuable evidence for diagnosis and personalized treatment. With the rapid development of high-throughput technologies, a large amount of genomic data has been generated, making it possible to differentiate cancer subtypes. Over the past several years, various large-scale high-dimension genomic data have been used for the prediction and classification of cancer [8,9]. Different cancer subtype classification methods have been proposed [10–14]. However, due to the complexity of cancer pathogenesis, the classification methods of cancer subtypes still need further exploration.

In recent years, the development of machine learning and the cancer genome atlas (TCGA) project has provided new ideas for cancer research [15–18]. Cai et al. [19] applied machine learning methods (multi-Receiver Operating Characteristic (ROC) and random forest) to classify lung cancer subtypes. Guo et al. [20] proposed a hierarchical deep learning model to learn high-level representations in transcriptome data and gene expression data to classify cancer subtypes. Lu et al. [21] developed a three-level machine-learning model to identify glioma subtypes. Liao et al. [22] used a random forest method based on isomiR data to classify six cancers. Xiao et al. [23] developed a deep learning-based...
multi-model ensemble method based on Ribonucleic Acid (RNA)-seq data to identify three kinds of cancers.

Although the above methods have achieved certain success in the classification of cancer subtypes, due to the complexity of cancer subtypes, the following limitations exist in the use of machine learning methods for cancer subtype classification. (1) Due to the inherent small sample size and high dimension characteristics of genetic data, the training processes of these models are prone to overfitting, which leads to the model having poor generalization ability. (2) In the classification of cancer gene samples, category imbalance is prevalent, making it challenging to obtain high-performance classification models. Therefore, overcoming the problem of small sample sizes, high dimensionality, and category imbalances in cancer gene data and developing a stable and highly accurate cancer subtype classification model is an urgent problem to be solved.

Recently, deep learning has achieved great success in the fields of computer vision, image processing, and speech recognition [24,25]. A deep neural network can significantly improve classification ability because this method can combine multiple neurons and obtain the corresponding weight parameters, especially for speech and text. This method also provides an effective tool for predicting cancer subtypes [26,27].

However, due to the complexity of deep neural network models, the training of the network takes a long time and consumes a large number of resources. In the training process of the network, a large amount of data is needed to adjust the parameters of the network. Otherwise, the model easily falls into overfitting and local optimization. The initialization and adjustment of the hyper-parameters of a deep neural network significantly influence the classification performance of the model. It is still challenging to obtain stable and accurate classification of cancer subtypes using deep neural networks due to the small sample sizes, high dimensionality, and class imbalances of cancer gene data.

To take advantage of the multi-layer learning of deep learning and avoid the risk of overfitting due to small sample size, Zhou and Feng [28] proposed a novel decision tree integration method gcForest model. This model is a new strategy that combines machine learning algorithms and deep learning ideas. The model takes advantage of the multi-layer learning of deep learning and avoids overfitting due to small sample size.

The gcForest model consists of two modules: multi-grained scanning and cascade forest modules, as shown in Figure 1. Multi-grained scanning is an approximation of the convolution process of convolutional neural networks. Similar to a convolutional neural network, convolution kernels of different sizes are used to acquire the spatial structure of the pixels in the image and the receptive fields of different scales [29,30]. When inputting high-dimension sample data, multi-grained scanning can capture different levels of information by cutting the high-dimension sample data into different-scale sequences of features through sliding windows at different scales, enabling gcForest to be contextually or structurally aware. The second model is the cascade forest, which contains multiple-layer random forests. Each layer in the cascade forest receives information processed by the previous layer and transmits information to the next layer. This model can learn more distinct features and provide more accurate predictions. K-fold cross-validation is used to reduce the risk of overfitting when extending a new layer. In detail, the training data is divided into k folds; k-1 folds are selected as the training data in turn, and the remaining fold is used as the validation data. After extending the new layer, the performance of the entire cascade is estimated on the validation data, and if there is no significant gain in performance, the expanding process will be terminated [28]. Therefore, the number of gcForest model cascades is automatically determined. Compared with most deep neural networks, gcForest adaptively determines the model complexity by terminating the training when appropriate so that this model can be applied to data of different sizes, not just large-scale data.

The gcForest algorithm performs better than other machine learning algorithms for many applications [31]. However, the gcForest algorithm still has the following limitations regarding small-scale cancer gene data. (1) In the multi-grained scanning module of the standard gcForest algorithm, each random forest contributes equally to the final result, but actually, the classification
abilities of different forests differ. Similar to how the multi-scale feature maps generated by convolution kernels of different scales have different effects on the final performance [29,30], multi-grained scanning produces feature vectors of different scales. When using these differently scaled feature vectors to train random forests and completely random forests, the classification ability of each random forest trained is different. This effect is not considered in the standard gcForest, resulting in the features that are truly useful for classification not receiving the attention that these features deserve; such features are very valuable for classification. We should increase the weight of these features and try to increase their positive impact on the classification results. The weights of features that are less useful for classification should be reduced to avoid negative impacts on classification. (2) Furthermore, different sliding windows make different contributions to the final predictions, but the standard gcForest algorithm does not consider these differences. The class vectors derived from different scanning windows have different effects on the final classification results. Different sliding windows need to be given different weights to capture more complex and diverse features, further enhancing the characterization learning ability and improving the classification performance of the model on small samples.

In this paper, we propose an improved gcForest model, called the MLW-gcForest model, to solve the subtype classification problem of in cancer subtype classification. MLW-gcForest mainly includes two innovations. (1) Different weights are assigned to different random forests according to the classification abilities of the forests, fully exploiting the mutual synergies between different forests. (2) We propose a sorting optimization algorithm that assigns different weights to the feature vectors generated under different sliding windows and fully exploits the complementarity of the feature vectors under different scanning windows. In summary, the proposed multi-level weighting strategy can help deep forests extract more valuable and richer multi-level features, thus effectively improving the ability of the standard gcForest model to classify small samples of genetic data.

The proposed MLW-gcForest model is trained on the methylation data of five data sets: BRCA (breast invasive carcinoma), LUAD (lung adenocarcinoma), GBM (glioblastoma), LIHC (liver hepatocellular carcinoma), and STAD (stomach adenocarcinoma) from TCGA. The results suggest that the MLW-gcForest model is superior to the standard gcForest model in constructing the subtype classification model; the accuracy rate is higher than 0.87, and the area under curve (AUC) is higher than 0.88. The results demonstrate the superiority of our proposed algorithm regarding classification performance on the small sample sizes, high dimensionality, and class imbalances of gene data.

![Figure 1](image_url)  
**Figure 1.** The basic structure of the gcForest model [28].
2. Materials and Methods

2.1. Feature Selection

The small sample sizes and high dimensionality of cancer gene data can lead to a higher risk of overfitting and degradation of the classification performance. Feature selection is an excellent way to address these challenges. There are three main types of feature selection methods in supervised learning: the filter method, the encapsulation method, and the embedding method [32–34].

In this experiment, we selected the lasso regression method for feature selection [35]; this method is an embedded feature selection method. The lasso regression method has been successfully applied to microarray classification and gene selection [36].

2.2. gcForest Model

The gcForest model is an ensemble approach based on decision trees [37]. This model is composed of two parts, as shown in Figure 1, of multi-grained scanning and cascade forest. (1) The multi-grained scanning structure can improve the representation learning ability of the model. This structure adopts the sliding window strategy to cut high-dimension data into multi-instance feature vectors. These feature vectors are fed into different types of random forests to obtain class vectors. Then, these class vectors are concatenated as the output of the multi-grained scanning module. (2) The second module is the cascade forest, which learns the class distribution features by assembling the numbers of the decision trees. Each layer of the cascade forest structure receives information processed by the previous layer. The output of every layer is the class vectors after the different random forest classifications, and then, these vectors are concatenated with the original vector to be input in the next cascade layer (detailed in [28]). The confidence probability vector is output by passing through each layer of the cascade forest. More distinguishing features are learned from the cascade structure, and more accurate predictions are obtained. We use k-fold cross-validation to reduce the risk of overfitting when extending a new layer. When extending a new layer, the performance of the entire cascade is estimated for the validation data. If there is no significant gain in performance, the training process is terminated. Finally, the average of each class probability is calculated from the last output of the cascade layer. We use the maximum probability value as the classification result.

The multi-grained scanning module is shown on the left side of Figure 1. For the sequence data, we assume that the input has 400 dimensions (dim). The first sliding window has 100 dim, so a total of 301 scans are required, and 301 × 100 feature vectors are produced. Supposing the samples have three categories, each sample is trained using a random forest [38] and a completely-random forest [39], and the class vectors (1806 dim, 2 × 301 × 3 dim) are generated and concatenated. Similarly, when the sliding window sizes are 200 and 300, 1206-dim (2 × 201 × 3 dim) and 606-dim (2 × 101 × 3 dim) class vectors are generated, respectively.

The second module is the cascade forest. First, 1806-dimension class vectors are input in the cascade layer for training. After training in four forests (two random forests and two completely random forests), a 12 dim class vector (four forests, three classes) is generated. This vector is concatenated with the 1806-dim class vectors and 1818-dim vector as the input of the second layer (as shown in Figure 1). Similarly, the second-layer cascade forests output a 12-dim class vector, and then, this vector is concatenated with a 1206-dim class vector (generated by 200-dim sliding windows in the multi-grained scanning). Thus, a 1218-dim class vector is the input of the third layer. The third-layer cascade forests output a 12-dim class vector, and this vector is concatenated with a 606-dim class vector (generated by a 300-dim sliding window) as the input of the next layer. We repeat this process to generate a new layer. Whenever a new layer is generated, the overall performance of the algorithm is estimated in the validation set. If the performance does not improve, the expanding process will be terminated [28].
2.3. Multi-Weighted gcForest (MLW-gcForest)

However, two challenges of gcForest may limit the application of this method to small-scale biology data. (1) Each random forest makes different contributions to the final result, and the performance of each random forest is not considered in the feature learning process of the standard gcForest model. Thus, different weights are given to different forests to improve the performance of gcForest on small scale genetic data; we call these weights \( \alpha \). (2) The different granularity feature vectors generated under different sliding windows have different effects on the final classification results. The effects of different sliding windows are not considered in the original algorithm. Different weights are given to different sliding windows to capture more complex and diverse features and to enhance the characterization learning ability. We call these weights \( \beta \) and call the process of weighting the sorting optimization algorithm. The basic structure of MLW-gcForest is shown in Figure 2.

![Figure 2. The basic structure of the MLW-gcForest model.](image)

2.3.1. Calculation of Weight \( \alpha \)

To objectively assess the performance of each random forest, we use the AUC to evaluate the classification capability of each forest, given that this parameter has been widely used [37,38]. The most common definition of the AUC is the area under the receiver operating characteristics curve (ROC), as shown in Formula (1). To facilitate the calculation of the AUC, in this section, we calculated the AUC using the equivalent concept of the AUC, which is called the Wilcoxon-Mann-Whitney statistic [40], as shown in Formula (2).

\[
AUC = \int_0^1 ROC(u) du \quad u \in [0, 1] 
\]

(1)

\[
AUC = \frac{\sum_{j=1}^{n} \sum_{j=1}^{m} 1_{x_i > y_j}}{mn} 
\]

(2)

We assumed a classifier \( f \) and a dataset \( X \) that contain \( m \) positive class samples and \( n \) negative class samples, where \( x_i \) (1 ≤ \( i \) ≤ \( m \)) is the output of \( f \) for the positive samples and \( y_j \) (1 ≤ \( j \) ≤ \( n \)) is the output of \( f \) for the negative samples. For any of the samples in the positive class, if the probability that the classifier \( f \) divides the sample into positive samples is greater than the probability of the negative samples, then 1 is added. The same principle is used to accumulate negative samples. Then, we multiplied the two types of results by the product of the positive and negative samples, and the final result is the AUC.

We used the examples in the multi-grained scanning module of original standard gcForest (as shown in Figure 3) to explain the solution process of \( \alpha_1 \) and \( \alpha_2 \). For the sequence data, we assumed that the input characteristics have 400 dim. The first sliding window has 100 dim, so a total of 301 scans are required, and 301 × 100 feature vectors are produced. Supposing that the samples have three
categories, each sample is trained using the random forest and completely random forest. Finally, 301 3-dim class probability vectors are obtained. The corresponding category with the highest value for each 3-dim class probability vector is used as the prediction category, and then, the correct number of samples is statistically classified, according to Formula (2).

\[ A_u c_{\text{ly}} = \alpha A_u c_0 + \beta A_u c_1 \]

\[ A_u c_{\text{ly}} = \alpha A_u c_0 + \beta A_u c_1 \]

\[ A_u c_{\text{ly}} = \alpha A_u c_0 + \beta A_u c_1 \]

2.3.2. Sorting Optimization Algorithm (Calculation of Weight \( \beta \))

As the different feature vectors of the sliding window have different effects on the classification results, we considered assigning corresponding weights to different sliding windows. We call the weight setting process the sorting optimization algorithm, as shown in the right part of Figure 4.

Sorting Optimization Algorithm

1. The input is \( N_s \) and \( N_w \). \( N_s \) is the number of samples, and \( N_w \) is the number of sliding windows. \( M_o \) represents the dimension of the original features, and \( N_r \) represents the number of sample classes. For sample \( i \), the sliding window \( w \) (1 \( \leq \) \( w \) \( \leq \) \( N_w \)). The size of the sliding window \( w, S, (S = 100, 200 \ldots) \), is used to cut the original high-dimension data into multi-instance feature vectors. The step size of the scan is \( S_0 \) (default \( S_0 = 1 \)). The number of feature vectors after scanning is \( N_v \), as shown in Formula (5).

\[ N_v = \frac{M_o \cdot S}{S_0} + 1 \]

2. \( M_o \) original features are cut by the sliding window to generate \( S \)-dim feature vectors. The number of feature vectors generated is \( N_v \). Each \( S \)-dim feature vector is input into one random forest and one completely random forest. The random forest and completely random forest each output \( N_r \)-dim class vectors. The class vectors output from the random forest are concatenated into an \( N_v \cdot N_r \)-dim class vector (called \( RF_v \)); the class vectors output from the completely random forest are concatenated into an \( N_v \cdot N_r \)-dim class vector (called \( CRF_v \)).

3. \( RF_v \) and \( CRF_v \) are multiplied by the weights \( \alpha_1 \) and \( \alpha_2 \) and are concatenated as \( 2 \cdot N_v \cdot N_r \)-dim class vectors. The length of the vectors is \( L \) and \( L = 2 \cdot N_v \cdot N_r \).

4. The outputs of the random forest and completely random forest classification models are the confidence probabilities that the samples belong to the \( N_r \) class. The closer the maximum confidence probability is to 1, the stronger the ability of the forest is in distinguishing the sample categories.
We took the first $1/N_c$ class vectors to measure the prediction ability of the current sliding window. The $L$-dim class vectors obtained in the previous step are sorted in descending order. The top $1/N_c$ of the sorted class vectors are averaged. This calculation can approximate the strength of the prediction ability of the current window for the current sample $i$. We called this value the $\text{Pre}_i$, as shown in Formula (6).

$$\text{Pre}_i = \frac{\sum_{j=1}^{N_c} \text{Des}(\text{con}(\text{RF}_w \ast \alpha_1, \text{cRF}_w \ast \alpha_2))}{N_c}$$  \hspace{1em} (6)

where $\text{Des}$ represents the descending order and $\text{con}$ represents the concatenation operation.

(5) For each of the $N_s$ samples, we repeated steps (1)–(4) and obtained the prediction abilities of the $N_s$ samples, $\text{Pre}_1, \ldots, \text{Pre}_i, \ldots, \text{Pre}_N_s$.

(6) The prediction ability $W_{\text{ability}}$ of the sliding window, $w$ was obtained by averaging the prediction ability of the $N_s$ samples, as shown in Formula (7).

$$W_{\text{ability}} = \frac{\sum_{k=1}^{N_s} \text{Pre}_k}{N_s}$$  \hspace{1em} (7)

(7) For each window, we repeated steps (1)–(6) to obtain the prediction ability of each window, $W_{\text{ability}}_1 \ldots W_{\text{ability}}_w \ldots W_{\text{ability}}_{N_c}$.

(8) We normalized $W_{\text{ability}}$ to obtain the predictive weight $\beta_w$ for each sliding window, as shown in Equation (8). We obtained the weights for each window, $\beta_1, \beta_2 \ldots \beta_w \ldots \beta_{N_c}$.

$$\beta_w = \frac{W_{\text{ability}}_w}{\sum_{w=1}^{N_c} W_{\text{ability}}_w}$$  \hspace{1em} (8)

The detailed algorithm is shown in Algorithm 1. The class vectors obtained from each window were multiplied by the corresponding weights $\beta_w$. Then, we concatenated the vectors as the output of the first multi-grained scanning module, which is also the input of the second cascade forest module. We used the cascade forest component to predict the probability that an input sample will eventually belong to a certain class.

![Figure 4](image_url)  
**Figure 4.** Process of calculating weights $\alpha$ and $\beta$.  

---

**Algorithm 1**

1. For each of the $N_s$ samples, repeat steps (1)–(4) and obtain the prediction abilities of the $N_s$ samples, $\text{Pre}_1, \ldots, \text{Pre}_i, \ldots, \text{Pre}_N_s$.

2. For each window, repeat steps (1)–(6) to obtain the prediction ability of each window, $W_{\text{ability}}_1 \ldots W_{\text{ability}}_w \ldots W_{\text{ability}}_{N_c}$.

3. Normalize $W_{\text{ability}}$ to obtain the predictive weight $\beta_w$ for each sliding window, as shown in Equation (8). The weights for each window, $\beta_1, \beta_2 \ldots \beta_w \ldots \beta_{N_c}$.
Algorithm 1: Sorting optimization algorithm

Input: $N_s$: Number of samples  
$N_w$: Number of sliding windows

For ($w = 1; w \leq N_w; w + +$) # current window
    For ($i = 1; i \leq N_s; i + +$) # current sample
        Number of feature vectors after scanning $N_v$
        $N_v = \frac{M_o - S_S}{S} + 1$
        ($M_o$: Number of original features  
$N_c$: Number of sample classes  
$S$: Size of the sliding window  
$S_0$: Scanning step size, default $S_0 = 1$)
        For ($j = 1; j \leq N_v; j + +$)
            Input S-dim feature vector into random forest, and output $N_c$-dim class vector
            Input S-dim feature vector into completely random forest, and output $N_c$-dim class vector
        End For
        Concatenate $N_v \times N_c$-dim class vector from random forest $RF_v$
        Concatenate $N_v \times N_c$-dim class vector from completely random forest $CRF_v$
        $RF_v$ and $CRF_v$ are multiplied by weights $\alpha_1$ and $\alpha_2$
        Concatenate these vectors (length $L = 2 \times N_v \times N_c$)
        Descend $L$-dim vector,
        Use the top $\frac{1}{N_c}$ of the sorted vectors to obtain the prediction ability of the current sample $i$ in the current window $Pre\_ability_i$, as detailed by the following formula
        \[
        Pre\_ability_i = \frac{\sum_{j=1}^{N_c} Des(con(RF_v \times \alpha_1, CRF_v \times \alpha_2))}{N_c}
        \]
    End For
W\_ability\_w: Prediction ability of sliding window $w$
    \[
    W\_ability\_w = \frac{\sum_{k=1}^{N_s} Pre\_ability_k}{N_s}
    \]
End for

Output: $\beta_1, \beta_2 \ldots \beta_w \ldots \beta_{N_w}$

3. Results

3.1. Dataset Preparation

We downloaded methylation datasets of BRCA, LUAD, GBM, LIHC, and STAD from TCGA. We selected these cancers because these cancer subtypes have been well verified in the past few years. The details of the five cancer datasets are shown in Table 1, where the clinical data sets for BRCA, GBM, and LUNG include the subtype information, while for LIHC and STAD, because there is no clear subtype category information in the clinical information, we labelled LIHC and STAD as different cancer subtypes based on the fields ‘viral_hepatitis_serology’ and ‘histological_type’, respectively.
Table 1. Information for five cancer datasets.

| Cancer                             | Total Samples | Available Samples | Feature Dimensions | Subtype Classes |
|------------------------------------|---------------|-------------------|--------------------|-----------------|
| Breast invasive carcinoma (BRCA)   | 1247          | 514               | 350                | 4               |
| Glioblastoma (GBM)                 | 629           | 546               | 240                | 4               |
| Lung adenocarcinoma (LUAD)         | 706           | 317               | 380                | 3               |
| Stomach adenocarcinoma (STAD)      | 580           | 508               | 250                | 4               |
| Liver hepatocellular carcinoma (LIHC) | 167          | 167               | 390                | 4               |

3.2. Experiments

In our research, for each cancer, we randomly divided the samples in a mutually exclusive manner, 80% for training, and 20% for independent testing. That is, our model is divided into two stages: training and independent testing. In the training stage, 10-fold cross-validation was performed. We divided the training datasets by ten-fold, with nine-fold as a training set and one-fold as a validation set. We repeated the process ten times until each set of data was used as training data and validated data once. The average accuracy of 10 validation sets was used as an estimate of the algorithmic accuracy.

To comprehensively evaluate the effectiveness of MLW-gcForest, we set up 500 decision trees for each random forest and completely random forest. In the discussion section, we compare in detail the influence of the number of decision trees in the forest on the final classification performance.

After the feature selection, the remaining dimensions of the methylation data are 350, 240, 380, 250, and 390 for the BRCA, GBM, LUAD, STAD, and LIHC, respectively, as shown in Table 1.

Different machine learning algorithms (support vector machine (SVM), K-nearest neighbor algorithm (KNN), logistic regression (LR), and random forest (RF)), gcForest and the proposed MLW-gcForest are used to establish cancer subtypes’ classification models. The evaluation index, area under the curve (AUC), accuracy (ACC), precision (Pre), Recall, and F1 score are used to evaluate the performance of the algorithms. Due to the unbalanced sample used, precise recall (PR) curves are required to process the highly skewed data.

3.3. Results

3.3.1. Classification Performance of Different Machine Learning Methods for Five Cancer Subtypes

We first compared the MLW-gcForest algorithm with the SVM, KNN, LR, and standard gcForest algorithms for five cancer subtypes to demonstrate the superiority of our proposed algorithm. Figure 5 shows the performances of the different machine learning methods on the five cancer subtypes. Figure 5a shows the classification results for breast cancer, which are divided into 4 subtypes: (luminal A (231), basal-like (98), luminal B (127), and HER2-enriched (58)). The experimental results suggest that MLW-gcForest obtains the highest AUC (0.99), which is 0.01 higher than that of standard gcForest and is always superior to the other traditional machine learning methods. Furthermore, the MLW-gcForest algorithm obtains an ACC of 90.5%, Pre of 90.8%, Recall of 89.6%, and F1 of 90.4%, which are superior to the results for the gcForest algorithm and traditional machine learning algorithms.

Figure 5b shows the classification results for LUAD, which are divided into three subtypes: (bronchioid (120), magnoid (83), and squamoid (114)). MLW-gcForest obtains the highest AUC (0.92), slightly higher than that of standard gcForest and consistently better than those of the other conventional methods. Furthermore, MLW-gcForest is slightly better than gcForest and significantly outperforms the conventional machine learning methods for the indicators ACC, Pre, Recall, and F1.

Figure 5c shows the classification results for LIHC, which are divided into four subtypes (Hepatitis B Virus (HBV) (73), Hepatitis C Virus (HCV) (15), hepatitis C antibody (56), and hepatitis B surface antigen (23)). MLW-gcForest obtains the highest AUC (0.91) and has better classification performance
than the gcForest algorithm; both algorithms perform significantly better than the traditional SVM, KNN, LR, and RF algorithms. The classification performance of MLW-gcForest for LIHC is slightly lower than those for BRCA and LUAD. The reason for this difference may be that the LIHC data set is relatively small, making it difficult to train a model with higher precision. However, in this case, our proposed algorithm still achieves better classification performance than that of the other algorithms, further proving the effectiveness of our algorithm on small data sets.

Figure 5d shows the classification results for GBM, which are divided into four subtypes (classical (150), mesenchymal (166), neural (90), and proneural (140)). MLW-gcForest obtains the same AUC as standard gcForest, which is better than those of the other conventional machine learning methods. Besides, the MLW-gcForest algorithm obtains an ACC of 0.885, a Pre of 0.863, a Recall of 0.878, and an F1 of 0.870, outperforming the gcForest algorithm (ACC of 0.836, Pre of 0.857, Recall of 0.850, and F1 of 0.853). Both algorithms perform significantly better than the traditional machine learning methods regarding ACC, Pre, Recall, F1, and AUC.

Figure 5e shows the classification results for STAD, which are divided into four subtypes (stomach, adenocarcinoma, not otherwise specified (NOS) (208), stomach, intestinal adenocarcinoma (207), stomach, adenocarcinoma, diffuse type (80), stomach adenocarcinoma, signet ring type (13)). MLW-gcForest obtains the same AUC as standard gcForest and obtains higher ACC, Pre, Recall, and F1 than gcForest. Both algorithms performed significantly better than traditional machine learning methods. Additionally, the STAD dataset has a strong class imbalance, and our MLW-gcForest algorithm shows better classification performance than the other algorithms.

The five corresponding Precision-Recall (PR) curves are shown in Figure 5(a3,b3,c3,d3,e3). In the PR curves, we observed that the areas obtained by the proposed MLW-gcForest are larger than the areas of the standard gcForest algorithms and traditional machine learning. The result shows that the MLW-gcForest method handles the imbalance of clinical samples well.
Figure 5. Comparisons of overall performances of MLW-gcForest, gcForest, SVM, KNN, LR, and RF on five data sets: (a) BRCA, (b) LUAD, (c) LIHC, (d) GBM, and (e) STAD.

Table 2 shows the results of the independent test datasets for the different algorithms for different types of cancer. Through comparison and analysis of the experimental results, MLW-gcForest achieves better performance than the other algorithms in the classification of the five cancer subtypes and significantly outperforms the conventional machine learning methods. In summary, our proposed MLW-gcForest algorithm improves the classification ability of standard gcForest in small sample size, high dimensionality, and class imbalances of genetic data. The main reasons are as follows: (1) Our MLW-gcForest algorithm fully exploits the mutual synergy between different forests, considers the classification ability of diverse forests, and gives the forests the corresponding weights. (2) The sorting optimization algorithm determines the feature vectors generated under different sliding windows that are most valuable to the final prediction results and gives these vectors higher weights, fully exploiting the complementarity of the feature vectors under different sliding windows. Therefore, the prediction performance of the model is greatly improved.
Table 2. Comparison of different methods on the independent test datasets.

|       | BRCA | LUAD | LIHC | GBM | STAD |
|-------|------|------|------|-----|------|
| SVM   |      |      |      |     |      |
| ACC   | 0.752| 0.762| 0.714| 0.694| 0.674|
| Pre   | 0.755| 0.754| 0.726| 0.723| 0.619|
| Recall| 0.709| 0.742| 0.693| 0.732| 0.574|
| F1    | 0.731| 0.748| 0.709| 0.727| 0.596|
| KNN   |      |      |      |     |      |
| ACC   | 0.745| 0.750| 0.688| 0.631| 0.706|
| Pre   | 0.774| 0.746| 0.708| 0.683| 0.697|
| Recall| 0.743| 0.739| 0.686| 0.736| 0.736|
| F1    | 0.758| 0.742| 0.697| 0.709| 0.716|
| LR    |      |      |      |     |      |
| ACC   | 0.730| 0.746| 0.718| 0.669| 0.658|
| Pre   | 0.728| 0.756| 0.708| 0.683| 0.609|
| Recall| 0.663| 0.726| 0.701| 0.726| 0.559|
| F1    | 0.694| 0.741| 0.704| 0.704| 0.583|
| RF    |      |      |      |     |      |
| ACC   | 0.691| 0.676| 0.716| 0.730| 0.674|
| Pre   | 0.527| 0.532| 0.693| 0.751| 0.546|
| Recall| 0.475| 0.508| 0.699| 0.753| 0.563|
| F1    | 0.500| 0.520| 0.696| 0.752| 0.554|
| gcForest |    |      |      |     |      |
| ACC   | 0.852| 0.820| 0.804| 0.836| 0.757|
| Pre   | 0.859| 0.821| 0.798| 0.857| 0.733|
| Recall| 0.826| 0.819| 0.778| 0.850| 0.788|
| F1    | 0.842| 0.820| 0.788| 0.853| 0.760|
| MLW-gcForest | |      |      |     |      |
| ACC   | 0.915| 0.866| 0.873| 0.885| 0.876|
| Pre   | 0.923| 0.863| 0.845| 0.863| 0.872|
| Recall| 0.916| 0.852| 0.829| 0.878| 0.821|
| F1    | 0.919| 0.857| 0.837| 0.870| 0.846|

To verify the proposed MLW-gcForest algorithm on small sample-sized datasets, we set up experiments to compare the AUC values of different methods for different cancer subtypes with samples of different size scales, as shown in Figure 6. The results in Figure 6 show that with increasing data sample size, the AUC value of the traditional machine learning algorithm and the standard gcForest algorithm increase linearly, while MLW-gcForest is always superior to the standard gcForest algorithm for all proportions of samples. Further, when the sample size is quite small (30% and 50%), the traditional machine learning algorithms and standard gcForest algorithm obtain AUC values lower, while the proposed MLW-gcForest algorithm can reach a higher AUC (0.7–0.79). From the above comparison, the proposed MLW-gcForest algorithm shows better classification performance for five cancer subtypes with samples of different size scales.
Appl. Sci. 2019, 9, 3589

3.3.2. Comparison with the State of the Art

We compared the performance of our proposed algorithm with other studies and used the results reported in these papers as a comparison, as shown in Table 3.

Liao et al. [22] used a random forest classification algorithm to classify six cancers by extracting five features of the isomiRs and achieved an accuracy of greater than 0.84. The classification accuracy in Liao’s study for BRCA and STAD is lower than that in our study, while the accuracy for LUAD is higher than that of our method. Telonis et al. [41] evaluated the ability of isomiRs and used the top 20% abundant isomiRs to construct a binary classifier. The classifier could label tumor samples with 93% average sensitivity. These researchers compare the SVM classification using the miRNA arm (B) expression profile and obtain results as shown in Table 2; the accuracy for BRCA in Telonis’s study is similar to that of our method, the accuracy for LIHC is higher than that of our method, and the accuracies for LUAD and STAD are lower than those of our method. Both studies showed that isomiR is very helpful in the classification of cancer subtypes.

For BRCA subtype classification: Li et al. [42] obtained an AUC of 0.89, and Sherafatian et al. [43] obtained an ACC of 0.89 and a Pre of 0.90. Our algorithm has a clear advantage in classification performance. The results show that our improved strategy for the gcForest algorithm can learn more discriminative features and achieve better classification performance.

For LUAD subtype classification: Podolsky et al. [44] obtained an AUC of 0.92, and Cai et al. [19] obtained an ACC of 0.85 and a Pre of 0.86. Podolsky et al. [44] obtained the same AUC as that of our proposed algorithm, but these researchers used the highest AUC as the final result, and our result is an average. Therefore, the AUC obtained by our algorithm is more reliable.

For LIHC subtype classification: Tan et al. [45] obtained an AUC of 0.77 and an ACC of 0.83 and Friemel et al. [46] obtained an ACC of 0.87.

For GBM subtype classification: Ryu et al. [47] proposed a three-level machine-learning model and obtained 0.83 AUC and 0.8 ACC. Lu et al. [21] obtained an AUC and ACC of 0.92 and 0.88, respectively.

Figure 6. The AUC values of different methods for different cancer subtypes with samples of different size scales. (a) BRCA, (b) LUAD, (c) LIHC, (d) GBM, (e) STAD.
Though these researchers obtained an AUC higher than that of our method, the ACC value is lower than that of our method.

We did not find literature for specialized classification of STAD subtypes. The subtype classification of STAD may have not yet found prognostic significance.

Compared with the methods proposed in the literature, MLW-gcForest achieves better performance for the five cancer subtypes. These results demonstrate that the deep forest structure can capture more complex and diverse features, making this method able to achieve better cancer subtype classification abilities compared with standard gcForest and traditional machine learning algorithms. Furthermore, the proposed multi-level weighting strategy can help deep forests extract more valuable multi-level features, thus effectively improving the classification ability of standard gcForest on small samples of genetic data. Additionally, most of the algorithms in the literature make subtype classifications and prognosis predictions for one particular cancer. Our proposed method can achieve excellent performance in the classification of multiple cancer subtypes, further demonstrating the superiority of our proposed multi-level weighted gcForest algorithm.

### Table 3. Comparison with the state of the art from the three indicators of AUC, ACC, and Pre on five data sets: (a) BRCA, (b) LUAD, (c) LIHC, (d) GBM, and (e) STAD.

| Cancer | Methods     | AUC  | ACC  | Pre  |
|--------|-------------|------|------|------|
| BRCA   | Liao [22]   | N/A  | 0.87 | N/A  |
|        | Guo [20]    | N/A  | N/A  | 0.88 |
|        | Telonis [41]| N/A  | 0.91 | N/A  |
|        | Li [42]     | 0.89 | N/A  | N/A  |
|        | Sherafatian [43] | N/A  | 0.89 | 0.90 |
|        | MLW-gcForest| 0.98 | 0.91 | 0.92 |
| LUAD   | Liao [22]   | N/A  | 0.91 | N/A  |
|        | Guo [20]    | N/A  | N/A  | 0.88 |
|        | Telonis [41]| N/A  | 0.86 | N/A  |
|        | Podolsky [44] | 0.92 | N/A  | N/A  |
|        | Cai [19]    | 0.85 | 0.86 |
|        | MLW-gcForest| 0.92 | 0.87 | 0.86 |
| LIHC   | Guo [20]    | N/A  | N/A  | 0.82 |
|        | Telonis [41]| N/A  | 0.90 | N/A  |
|        | Tan [45]    | 0.77 | 0.83 | N/A  |
|        | Friemel [46]| N/A  | 0.87 | N/A  |
|        | MLW-gcForest| 0.91 | 0.87 | 0.85 |
| GBM    | Guo [20]    | N/A  | N/A  | 0.78 |
|        | Lu [21]     | 0.92 | 0.88 | N/A  |
|        | Ryu [47]    | 0.83 | 0.80 | N/A  |
|        | MLW-gcForest| 0.87 | 0.89 | 0.86 |
| STAD   | Liao [22]   | N/A  | 0.84 | N/A  |
|        | Telonis [41]| N/A  | 0.85 | N/A  |
|        | MLW-gcForest| 0.88 | 0.87 | 0.87 |

### 4. Discussion

The gcForest algorithm is a fusion of traditional machine learning algorithms and deep learning thought, but the standard gcForest algorithm may face over-fitting problems due to the small sample sizes and high dimensionality of genetic data. The MLW-gcForest algorithm is an improvement of the gcForest algorithm. Dynamically setting multi-level weights according to the classification performance of each random forest and different sliding windows can alleviate the problem of overfitting and improve the ability to classify small samples of gene data.
To demonstrate that the proposed MLW-gcForest can alleviate overfitting, we plotted the accuracy curves of MLW-gcForest and gcForest on the training and validation sets as the number of samples increased for five cancer subtypes. The results in Figure 7 show that under the five different cancer types, although the standard gcForest achieved good classification accuracy, the accuracy curves of standard gcForest on training and validation sets are further apart in position. It demonstrates that the accuracy of the standard gcForest on the validation set is much lower than that on the training set, which indicates that the standard gcForest still has some overfitting. The accuracy curve of MLW-gcForest on the training and validation sets is closer in position, compared to the standard gcForest. It demonstrates that the MLW-gcForest has a smaller difference in the accuracy of the validation set and the training set, which indicates that MLW-gcForest can effectively alleviate the over-fitting problem of the standard gcForest. Therefore, our dynamic multi-level weighting strategy can more effectively alleviate the over-fitting problem.

![Figure 7. Cont.](image-url)
Figure 7. Accuracy curves of MLW-gcForest and gcForest on the training and validation sets for five cancer subtypes.

We compared the MLW-gcForest model with the standard gcForest model and several commonly used machine learning methods. We found that in the process of classifying cancer subtypes, the MLW-gcForest model and the gcForest model are superior to traditional classification methods. The most likely reason for the difference in performance is that deep forests can learn more meaningful advanced features through supervised learning. Among the subtypes of most cancers, the MLW-gcForest model is superior to the gcForest model, proving that our proposed multi-level weighting idea improves the classification ability of the standard gcForest algorithm for small sample high-dimension data and provides a good model for the classification of cancer subtypes.

To determine the number of decision trees in the random forest that achieves the best performance, we designed an experiment to change the number of decision trees to see the effect on the final classification performance. The accuracy results for different numbers of trees on five cancers are shown in Figure 8. Figure 8 suggests that when the number of decision trees is set to 30, the algorithm performs the worst. When the number of trees is set to 500 or 600, the algorithm performs the best. When the number of trees continues to increase beyond 600, the accuracy slowly decreases. Based on the above results, 500 trees are used as the final experimental parameter because 500 and 600 achieved similar results, but the time cost and calculation cost of 500 trees are lower.

We explain why methylation is selected for subtype classification in the following. The data we can obtain includes methylation data, RNA data, and CNV data from TCGA, which are subclassified using our improved MLW-gcForest method. Table 4 shows the classification performance of MLW-gcForest on different types of cancer using different types of data. The table shows that the methylation data provides the strongest discriminating ability for MLW-gcForest and achieves better results for five types of cancer. The classification ability using RNA data is the second best, but in particular, the ability to classify STAD is relatively weak. The CNV data provide the worst subtype classifications for cancer, especially for LIHC and STAD. Therefore, after comprehensive consideration, we chose methylation data to classify cancer subtypes.
Cancer is a highly heterogeneous disease, and different subtypes of cancer require different treatments. The subtype classification of cancer plays an important role in the diagnosis and treatment of cancer. In this paper, we propose a learning model called MLW-gcForest, which is an improved version of the standard gcForest algorithm, to improve the subtype classification ability of small sample and high dimensionality cancer genetic data. We fully consider the mutual synergy between different forests, assigning different weights to different random forests according to the classification ability of different subtypes of cancer.

Furthermore, we propose a sorting optimization algorithm that assigns different weights to the feature vectors generated under different sliding windows, fully considering the complementarity of the feature vectors under different scanning windows. Specifically, the methylation data of five types of cancers, BRCA, LUAD, LIHC, GBM, and STAD, were used for classification. The experimental results show that the MLW-gcForest algorithm is superior to common machine learning algorithms in various evaluation metrics and also has better classification performance than the standard gcForest algorithm. The effectiveness of the proposed multi-level weighting scheme is shown, fully considering the diversity and complementarity of different random forests and different sliding windows; thus, more abundant and differentiated feature information is obtained, greatly improving the classification accuracy. Our study shows that methylation data are beneficial in the classification of cancer subtypes.

Table 4. Classification performance for different types of cancer and different types of data.

|        | Methylation | RNA | CNV |
|--------|-------------|-----|-----|
|        | ACC  | Pre  | Recall | F1  | ACC | Pre  | Recall | F1  | ACC  | Pre  | Recall | F1  |
| BRCA   | 0.915| 0.923| 0.916  | 0.919| 0.844| 0.851| 0.846  | 0.848| 0.757| 0.722| 0.745  | 0.733|
| LUAD   | 0.866| 0.863| 0.852  | 0.857| 0.807| 0.826| 0.824  | 0.825| 0.739| 0.746| 0.726  | 0.736|
| LIHC   | 0.873| 0.845| 0.829  | 0.837| 0.796| 0.816| 0.810  | 0.813| 0.726| 0.731| 0.744  | 0.737|
| GBM    | 0.885| 0.863| 0.878  | 0.870| 0.843| 0.832| 0.846  | 0.839| 0.750| 0.733| 0.742  | 0.737|
| STAD   | 0.876| 0.872| 0.821  | 0.846| 0.739| 0.725| 0.714  | 0.719| 0.668| 0.673| 0.676  | 0.674|

Although our model achieves certain results in the classification of cancer subtypes, there are certain limitations: (1) In our study, we only subtyped five common cancers, and whether our method can be extended to other cancers requires further exploration. (2) We did not consider whether the fusion of multi-modal data is feasible for the classification of cancer subtypes; this consideration will be explored in future research.

5. Conclusions

Cancer is a highly heterogeneous disease, and different subtypes of cancer require different treatments. The subtype classification of cancer plays an important role in the diagnosis and treatment of cancer. In this paper, we propose a learning model called MLW-gcForest, which is an improved version of the standard gcForest algorithm, to improve the subtype classification ability of small sample and high dimensionality cancer genetic data. We fully consider the mutual synergy between different forests, assigning different weights to different random forests according to the classification ability of the forest.
Author Contributions: Methodology, Y.D. and W.Y.; Software, J.W.; Validation, J.Z.; Writing—review and editing, Y.D., W.Y., and Y.Q.

Funding: This research was funded in part by National Natural Science Foundation of China, grant number 61872261, in part by the open funding project of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University (Grant No. 2018-VRLAB2018B07) and in part by Research Project Supported by Shanxi Applied Basic Research Project (201801D121139).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Noone, A.M.; Cronin, K.A.; Altekruse, S.F.; Howlader, N.; Lewis, D.R.; Petkow, V.I.; Penberthy, L. Cancer incidence and survival trends by subtype using data from the Surveillance Epidemiology and End Results Program, 1992–2013. *Cancer Epidemiol. Biomark. Prev.* 2016, 26, 632. [CrossRef] [PubMed]

2. Choi, W.; Ochoa, A.; McConkey, D.J.; Aine, M.; Högblom, M.; Kim, W.Y.; Real, F.X.; Kiltie, A.E.; Millsom, I.; Dyrskjøt, L.; et al. Genetic alterations in the molecular subtypes of bladder cancer: Illustration in the cancer genome atlas dataset. *Eur. Urol.* 2017, 72, 354–365. [CrossRef] [PubMed]

3. Dai, X.; Li, T.; Bai, Z.; Yang, Y.; Liu, X.; Zhan, J.; Shi, B. Breast cancer intrinsic subtype classification, clinical use and future trends. *Am. J. Cancer Res.* 2015, 5, 2929. [PubMed]

4. Feng, P.; Chen, T.; Lin, Y.; Chi, S.; Zhang, Y.; Yu, K.; et al. Comparison of glioblastoma (GBM) molecular classification methods. In *Seminars in Cancer Biology*; Academic Press: Cambridge, MA, USA, 2018; Volume 53, pp. 201–211.

5. Cristescu, R.; Lee, J.; Nebozhyn, M.; Kim, K.M.; Ting, J.C.; Song, S.S.; Liu, J.; Yue, Y.G.; Wang, J.; Yu, K.; et al. Molecular analysis of gastric cancer identifies subtypes associated with distinct clinical outcomes. *Nat. Med.* 2018, 21, 449. [CrossRef] [PubMed]

6. Muhamed Ali, A.; Zhuang, H.; Ibrahim, A.; Rehman, O.; Huang, M.; Wu, A. A Machine Learning Approach for the Classification of Kidney Cancer Subtypes Using miRNA Genome Data. *Appl. Sci.* 2018, 8, 2422. [CrossRef] [PubMed]
17. Sun, D.; Wang, M.; Li, A. A multimodal deep neural network for human breast cancer prognosis prediction by integrating multi-dimensional data. IEEE/ACM Trans. Comput. Biol. Bioinform. 2018, 16, 841–850. [CrossRef] [PubMed]

18. Becker, A.S.; Marcon, M.; Ghafoor, S.; Wurnig, M.C.; Frauenfelder, T.; Boss, A. Deep learning in mammography: Diagnostic accuracy of a multipurpose image analysis software in the detection of breast cancer. Investig. Radiol. 2017, 52, 434–440. [CrossRef]

19. Cai, Z.; Xu, D.; Zhang, Q.; Zhang, J.; Ngai, S.M.; Shao, J. Classification of lung cancer using ensemble-based feature selection and machine learning methods. Mol. BioSyst. 2015, 11, 791–800. [CrossRef]

20. Guo, Y.; Shang, X.; Li, Z. Identification of cancer subtypes by integrating multiple types of transcriptomics data with deep learning in breast cancer. Neurocomputing 2019, 324, 20–30. [CrossRef]

21. Lu, C.F.; Hsu, F.T.; Hsieh, K.L.C.; Kao, Y.C.J.; Cheng, S.J.; Hsu, J.B.K.; Tsai, P.H.; Chen, R.J.; Huang, C.C.; Yen, Y.; et al. Machine learning–based radiomics for molecular subtyping of gliomas. Clin. Cancer Res. 2018, 24, 4429–4436. [CrossRef]

22. Liao, Z.; Li, D.; Wang, X.; Li, L.; Zou, Q. Cancer diagnosis through IsomiR expression with machine learning method. Curr. Bioinform. 2018, 13, 57–63. [CrossRef]

23. Xiao, Y.; Wu, J.; Lin, Z.; Zhao, X. A deep learning-based multi-model ensemble method for cancer prediction. Comput. Methods Programs Biomed. 2018, 153, 1–9. [CrossRef] [PubMed]

24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

25. Ciresan, D.; Meier, U.; Schmidhuber, J. Multi-column deep neural networks for image classification. arXiv 2012, arXiv:1202.2745.

26. Ha, R.; Mutasa, S.; Karcich, J.; Gupta, N.; Van Sant, E.P.; Nemer, J.; Sun, M.; Chang, P.; Liu, M.Z.; Jambawalikar, S. Predicting Breast Cancer Molecular Subtype with MRI Dataset Utilizing Convolutional Neural Network Algorithm. J. Digit. Imaging 2019, 32, 276–282. [CrossRef] [PubMed]

27. Coudray, N.; Ocampo, P.S.; Sakellaropoulos, T.; Narula, N.; Snuderl, M.; Fenyö, D.; Moreira, A.L.; Razavian, N.; Tsirigos, A. Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. Nat. Med. 2018, 24, 1559. [CrossRef] [PubMed]

28. Zhou, Z.H.; Feng, J. Deep Forest: Towards an Alternative to Deep Neural Networks. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017.

29. LeCun, Y.; Bottou, L.; Bengio, Y.; Haﬀner, P. Gradient-based learning applied to document recognition. Proc. IEEE 1998, 86, 2278–2324. [CrossRef]

30. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classiﬁcation with deep convolutional neural networks. In Proceedings of the Neural Information Processing Systems, Harrahs and Harveys, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.

31. Ray, S. Disease Classiﬁcation within Dermoscopic Images Using features extracted by ResNet50 and classiﬁcation through Deep Forest. arXiv 2018, arXiv:1807.05711.

32. Meinshausen, N.; Bühlmann, P. Stability selection. J. R. Stat. Soc. 2010, 72, 417–473. [CrossRef]

33. Huang, X.; Zhang, L.; Wang, B.; Li, F.; Zhang, Z. Feature clustering based support vector machine recursive feature elimination for gene selection. Appl. Intell. 2017, 48, 1–14. [CrossRef]

34. Vinh, L.T.; Lee, S.; Park, Y.T.; d’Auriol, B.J. A novel feature selection method based on normalized mutual information. Appl. Intell. 2012, 37, 100–120. [CrossRef]

35. Tibshirani, R. The lasso method for variable selection in the cox model. Stat. Med. 1997, 16, 385–395. [CrossRef] [PubMed]

36. Lin, Y.; Liu, X.; Hao, M. Model-free feature screening for high-dimensional survival data. Sci. China Math. 2018, 61, 1617–1636. [CrossRef]

37. Quinlan J, R. Induction of decision trees. Mach. Learn. 1986, 1, 81–106. [CrossRef]

38. Breiman, L. Random forests. Mach. Learn. 2001, 45, 5–32. [CrossRef]

39. Fan, W.; Wang, H.; Philip, S.Y.; Ma, S. Is random model better? On its accuracy and efficiency. In Proceedings of the Third IEEE International Conference on Data Mining, Melbourne, FL, USA, 22 November 2003; p. 51.

40. Cortes, C.; Mohri, M. AUC optimization vs. error rate minimization. In Advances in Neural Information Processing Systems; Neural Information Processing Systems Foundation: Vancouver, BC, Canada, 2004; pp. 313–320.
41. Telonis, A.; Magee, R.; Loher, P.; Chervoneva, I.; Londin, E.; Rigoutsos, I. The presence or absence alone of miRNA isoforms (isomiRs) successfully discriminate amongst the 32 TCGA cancer types. *bioRxiv* 2016, 082685.

42. Li, H.; Zhu, Y.; Burnside, E.S.; Huang, E.; Drukker, K.; Hoadley, K.A.; Fan, C.; Conzen, S.D.; Zuley, M.; Net, J.M.; et al. Quantitative MRI radiomics in the prediction of molecular classifications of breast cancer subtypes in the TCGA/TCIA data set. *NPJ Breast Cancer* 2016, 2, 16012. [CrossRef]

43. Sherafatian, M. Tree-based machine learning algorithms identified minimal set of miRNA biomarkers for breast cancer diagnosis and molecular subtyping. *Gene* 2018, 677, 111–118. [CrossRef]

44. Podolsky, M.D.; Barchuk, A.A.; Kuznetcov, V.I.; Gusarova, N.F.; Gaidukov, V.S.; Tarakanov, S.A. Evaluation of machine learning algorithm utilization for lung cancer classification based on gene expression levels. *Asian Pac. J. Cancer Prev.* 2016, 17, 835–838. [CrossRef]

45. Tan, P.S.; Nakagawa, S.; Goossens, N.; Venkatesh, A.; Huang, T.; Ward, S.C.; Sun, X.; Song, W.M.; Koh, A.; Canasto-Chibuque, C.; et al. Clinicopathological indices to predict hepatocellular carcinoma molecular classification. *Liver Int.* 2016, 36, 108–118. [CrossRef]

46. Friemel, J.; Rechsteiner, M.; Frick, L.; Böhm, F.; Struckmann, K.; Egger, M.; Moch, H.; Heikenwalder, M.; Weber, A. Intratumor heterogeneity in hepatocellular carcinoma. *Clin. Cancer Res.* 2015, 21, 1951–1961. [CrossRef]

47. Ryu, Y.J.; Choi, S.H.; Park, S.J.; Yun, T.J.; Kim, J.H.; Sohn, C.H. Glioma: Application of whole-tumor texture analysis of diffusion-weighted imaging for the evaluation of tumor heterogeneity. *PLoS ONE* 2014, 9, e108335. [CrossRef] [PubMed]