1. Specificity Classifier Based on Random Forest
   1. Data preparation

We use data with vdjdb.score 0 as negative class data (0 labels) and data with vdjdb.score 1, 2, and 3 as positive class data (1 label). Due to the large number of negative class data compared to positive class data, in order to reduce model bias caused by class imbalance, we adopt a random sampling method to sample negative class data, randomly selecting the same amount of negative class data as positive class data as training data.

Next, we used N-gram models to encode the CDR3 sequence and antigen receptor sequence, respectively. The hyperparameters of the model are as follows:

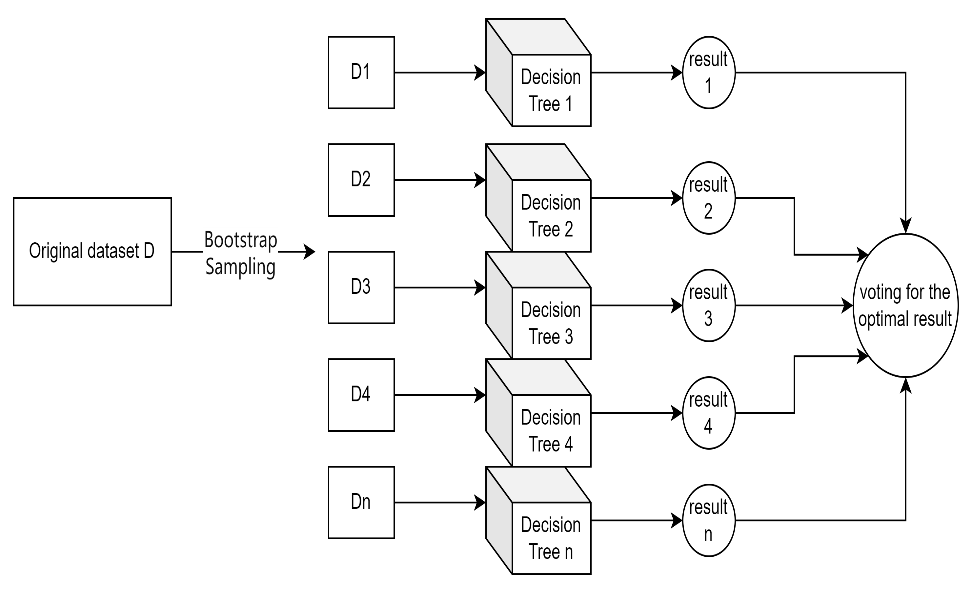
**Table 1.** Hyperparameter settings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| parameter | analyzer | max features | ngram\_range | lowercase |
| value | char | 2000 | (5,5) | False |

* 1. Construction of Random Forest

We will use the encoded CDR3 and antigen sequence as feature inputs and labels as model outputs to construct a random forest model(RF).

The model construction is shown in the following figure:



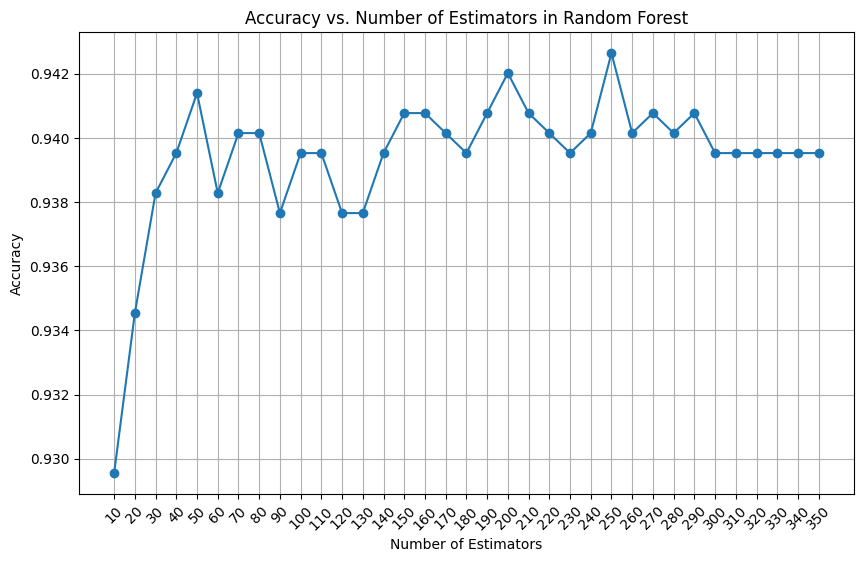
**Fig. 1.** RF classification principle

For the construction of decision trees, we use the GINI coefficient for evaluation, as shown by the following formula:

In formula (1.1), D represents the dataset, k represents the number of categories, and represents the proportion of samples belonging to category i in the dataset.

* 1. Selection of hyperparameters

For a random forest, selecting how many evaluators (number of decision trees) is an important hyperparameter. We use k-fold validation to calculate the average accuracy of samples for different numbers of random forest models, as shown in the following figure:



**Fig. 2.** Accuracy vs. Number of Estimators in Random Forest

From the graph, we can intuitively see that when the number of decision trees is 250, the model achieves the best accuracy. So we have chosen 250 evaluators, and our choices for other hyperparameters and k-fold validation settings are shown in the table below:

**Table 2.** hyperparameters settings

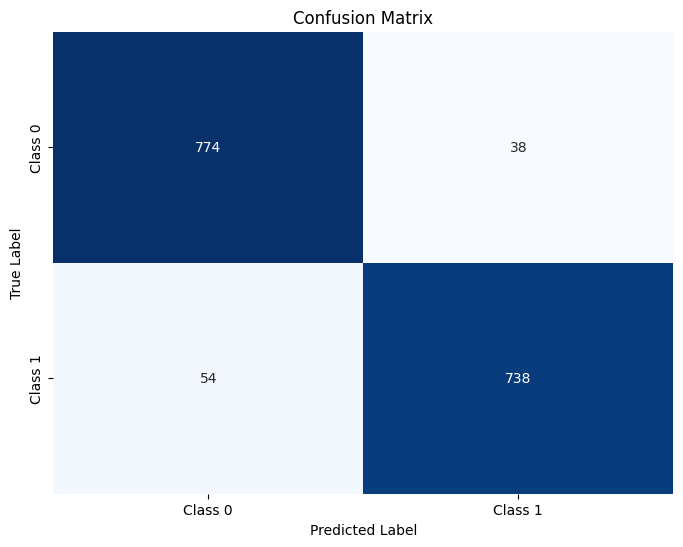
|  |  |
| --- | --- |
| parameter | setting |
| max\_depth | [None, 5, 10, 15] |
| max\_features | ['auto', 'sqrt', 'log2'] |
| min\_samples\_split | [2, 5, 10] |
| cv | 5 |
| scoring | accuracy |

The final hyperparameter design determined by grid search is shown in the table below:

**Table 3.** final hyperparameter design

|  |  |
| --- | --- |
| parameter | setting |
| max\_depth | None |
| max\_features | log2 |
| min\_samples\_split | 2 |
| n\_estimators | 250 |

* 1. Prediction of Random Forest

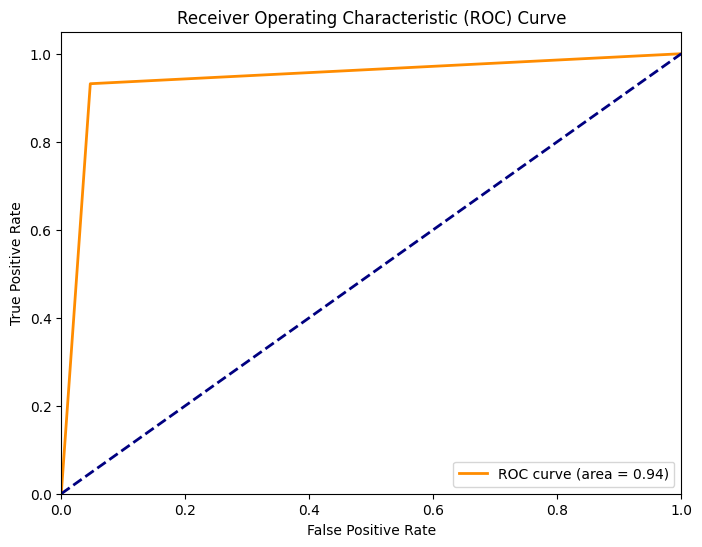


Recall: 0.9318181818181818

Precision: 0.9510309278350515

F1 Score: 0.9413265306122448

Accuracy: 0.942643391521197



1. Conclusion

In this section, we will summarize our conclusions and insights

1. Further Work and Improvement

In this section, we will focus on better data augmentation methods, such as using GANs to enhance data.