Homewor1 Answer

Course: CSE6250 BH4H

Name: Yichao Zhang

GT User ID: yzhang3414

**2 Descriptive Statistics**

|  |  |  |  |
| --- | --- | --- | --- |
| Metric | Deceased patients | Alive patients | Function to complete |
| Event Count |  |  | event count metrics |
| 1. Average Event Count | 982.014 | 498.118 |  |
| 2. Max Event Count | 8635 | 12627 |  |
| 3. Min Event Count | 1 | 1 |  |
| Encounter Count |  |  | encounter count metrics |
| 1. Average Encounter Count | 23.038 | 15.452 |  |
| 2. Max Encounter Count | 203 | 391 |  |
| 3. Min Encounter Count | 1 | 1 |  |
| Record Length |  |  | record length metrics |
| 1. Average Record Length | 127.532 | 159.2 |  |
| 1. Max Record Length | 1972 | 2914 |  |
| 1. Min Record Length | 0 | 0 |  |

Table 2: Descriptive statistics for alive and dead patients

**4 Predictive Modeling**

**4.1 Model Creation**

**4.1 b.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Accuracy | AUC | Precision | Recall | F1-Score |
| Logistic Regression | 0.9545 | 0.9454 | 0.9869 | 0.8988 | 0.9408 |
| SVM | 0.9940 | 0.9945 | 0.9882 | 0.9970 | 0.9926 |
| Decision Tree | 0.7763 | 0.7475 | 0.7922 | 0.6012 | 0.6836 |

Table 3: Model performance on training data

**4.1 c.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Accuracy | AUC | Precision | Recall | F1-Score |
| Logistic Regression | 0.7381 | 0.7375 | 0.6804 | 0.7333 | 0.7059 |
| SVM | 0.7381 | 0.7389 | 0.6768 | 0.7444 | 0.7090 |
| Decision Tree | 0.6714 | 0.6569 | 0.6329 | 0.5556 | 0.5917 |

Table 4: Model performance on test data

**4.1 d.**  **Based on the performance metrics on training and test data, please propose some strategies to improve the test performance and also provide the justification for your recommendation. For example, the strategies can be “gather more training data” or “do parameter tuning more on the algorithms”.**

The result shows that the training scores are very high, while the validation score are much lower than training score. That means the models are over-fitted on training data. So the performance and generalizability are bad.

To avoid overfitting and raise the generalizability:

1. Use k-fold cross validation;
2. Use ensemble learning method such as Random Forest;
3. Simplify the model complexity, due to Occam's Razor principle;

To improve the performance:

1. Do a better feature engineer, because the upper bound of performance is determined by feature selection;
2. Use grid search to fine tune the parameters;
3. Gather more training data, which improve both performance and generalizability.

**4.2 Model Validation**

**4.2 b.**

|  |  |  |
| --- | --- | --- |
| CV strategy | Accuracy | AUC |
| K-Fold | 0.7213 | 0.7076 |
| Randomized | 0.7357 | 0.7188 |

Table 5: Cross Validation

**4.3 Self Model Creation**

**4.3 b. Write a short paragraph on your best predictive model (based on cross validation and AUC) and the other models that you tried. What was the rationale behind your approach? Did your model perform better than in the previous section?**

My best model is a Random Forest Classifier.

1. Random Forest is an ensemble learning method, which is less likely overfitting than other models.
2. The performance can be raised by grid search the parameters, my search ranges are:

'n\_estimators': [300, 400, 500, 600, 700],

'max\_depth': [50,60,70 ,80 ,90, 100]

‘max \_features’: [‘log2’, ‘sqrt’(default)]

‘criterion’: [‘gini’(default), ‘entropy’]

The best parameters are:

max\_features='log2'

criterion = ‘gini’(default)

n\_estimators = 500

max\_depth = 80

With the best AUC score about 0.82 on validation set, which is much better than in the previous section. The other models I tried include: SVM, Naïve Bayes, AdaBoost and etc, the performance are lower than Random Forest here.