DOI: 10.23977/acss.2023.070616
ISSN 2371-8838 Vol. 7 Num. 6

Research on flight technology evaluation based on machine learning algorithm

Kun Tang[1], Weijie Wang[1], Zhendong Guo[1,*], Junjie Liang[2,#], Kaipeng Yuan[3,#], Liuqing Huang[4,#]

[1]School of Mathematics and Computer Science, Guangdong Ocean University, Zhanjiang, China
[2]School of Mechanical Engineering, Guangdong Ocean University, Zhanjiang, China
[3]Binhai Agricultural College, Guangdong Ocean University, Zhanjiang, China
[4]College of Food Science and Technology, Guangdong Ocean University, Zhanjiang, China
**Corresponding author:1090904791@qq.com*
[#]These authors contributed equally to this work.

Keywords: Random forest classification, stacking model fusion, machine learning, aviation security

Abstract: In China's civil aviation transportation industry, flight safety has been the focus of attention. In this paper, a flight technology assessment model and an automated early warning model are established for aviation safety. First, data pre-processing is performed. Then the suitable indicators are continuously screened by multiple machine learning classifications, and then the screened data are fitted to continuously screen the suitable indicators, and the aircraft technology assessment is found to be more suitable for the integrated learning classification model. Subsequently, three unoptimized optimal models were derived as LightGBM, XGboost and Random Forest classification models. The results of these models are then fused by Stacking model to combine their advantages to build the final aircraft technology assessment prediction model. For the automated early warning mechanism, the aviation early warning mechanism needs to be established first by subclassing these data with the K-mean clustering model and visualizing the key data items such as avg (COG NORM ACCEL) based on the normal distribution, combined with the differentiated distribution for each category to set the implausible warning level to establish the aviation automated early warning model.

1. Introduction

The recent "March 21" air disaster has raised concerns about flight safety. Flight safety big data, such as Quick Access Recorder (QAR) data, which records aircraft flight parameters during the flight, is an important way for airlines to obtain aircraft flight parameters. At present, in the field of flight quality monitoring, it mainly involves the research and application of overrun events, but there is a deficiency in the analysis of overrun events and a lack of in-depth analysis of the causes of overruns. Therefore, there is a need to mine QAR full flight segment data to form flight quality records of specific personnel, and carry out targeted safety management, identify safety hazards and improve safety performance through data modeling, analysis, calculation and assessment of risk propensity.

Flight quality monitoring is an important task to ensure the stable and sustainable safety development of the civil aviation industry, and has an important role in preventing flight safety accidents in advance. Quick storage recorder (QAR) provides comprehensive and complete data for flight risk study by recording parameters such as position, flight attitude and flight operation during flight. The process of aligning an aircraft to the runway during the landing phase is called an approach, and an unstable approach can easily lead to serious events such as heavy landings, tail wipe, and runway deviation during the landing phase. Therefore, the risk analysis and accurate warning of unstable approach events can effectively guarantee the safety of aircraft landing. Therefore, this paper establishes a flight technology assessment model and an automated early warning model for aviation safety [1].

2. Flight Technology Assessment Model

2.1. Data pre-processing

(1) Disposal of redundant indicators

In this paper, we found that there are some indicators such as model, date, etc. These characteristic variables have no practical significance for model building, so they are directly deleted.

(2) Feature Code

For categorical data, such as raw data, the values span large and have multiple forms, as only numeric types can be computed. Therefore, for various special feature values, we need to encode them accordingly. Here we use Label encoding to encode the original feature values into custom numeric labels to complete the quantization encoding process. Such as V2_Method, we encode these R as 1 and C as 0.

(3) Missing value handling

Visualizing the missing values by python, there are only 2369 data in total. The null heat map of the aircraft parameter measurement data is shown in Figure 1.

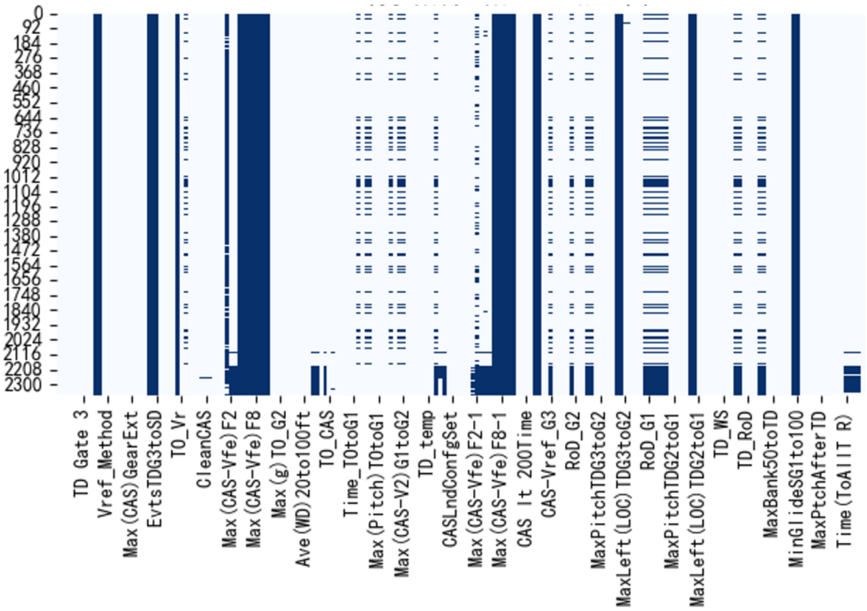

Figure 1: Aircraft parameter measurement data null heat map.

(4) KNN classification for outliers

For data with outliers in the metrics, we use KNN classification method to fill them.

2.2. Benchmark model building and solving

After performing missing value processing, outlier processing, and steps on the data, we obtained a new copy of the data. In calculating the model results, we combined the first question similar steps through multiple machine learning models important features to screen the main indicators, constantly screen the appropriate indicators, the data will be fitted to the sample data through integrated and linear models and other machine model algorithms to take 10-fold cross-validation method to get the relationship between each feature and the flight technology. Subsequently, three optimal models without optimization are derived as LightGBM, XGboost and random forest, respectively. The models are continuously optimized in terms of parameters, as well as changing indicators, and then the model fusion approach is adopted to fuse the results of LightGBM, XGboost and random forest by Stacking model, combining their advantages to build a final prediction of a flight based the optimal model is selected as the baseline model. The optimal model is selected as the baseline model, and the optimization is continued based on the baseline model. The model is built by Python 3.9+, scikit-learn, and the performance of the model is measured by Accuracy, Recall, Precision, and F1 values, which are calculated as follows.

Accuracy:

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \tag{1}$$

Recall:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{2}$$

Precision:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{3}$$

F1-Score:

$$F_1 = \left(1+\beta^2\right)\frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision}+\text{Recall}} \tag{4}$$

Where the symbolic meaning is as follows. TP (True Positive): Correct positive case, an instance is a positive class and is also determined to be a positive class. FN (False Negative): Wrong negative example, false positive, an instance is positive but judged to be false. FP (False Positive): False Positive, false positive, false class but determined to be positive. TN (True Negative): Correct counterexample, an instance is a false class and is also determined to be a false class.

Since the F1 value of the model metric is a composite metric, only the Accuracy and composite F1 values are shown below. Table 1 below shows the three model metrics for the flight technology assessment.

Table 1: Metrics for the three models of overall satisfaction with voice calls.

Models	Accuracy	F1
Random Forest	0.852	0.845
XGBoost	0.831	0.828
LightGBM	0.813	0.806

2.3. Random Forest Classification Model

Random forest belongs to a kind of integrated learning model, i.e., by building multiple learners to accomplish the learning task together, a set of base learners is generated first, and then some strategy is used to combine these learners, and decision trees are often used as base learners because they are weak learners themselves, but after integration, they often have strong prediction effect and become strong learners. Random forest is one of the decision tree integration models, which uses CART decision trees and is widely used because of its good prediction effect and stability [2].

The modeling computation process of random forest classification algorithm is:

(1) Draw the training set from the original sample set. In each round, n training samples (with put-back sampling) are drawn from the original sample set using the Bootstraping method. A total of k rounds are performed to obtain k training sets. (The k training sets are independent of each other)

(2) One model is obtained using one training set at a time, and a total of k models are obtained from k training sets.

(3) For the classification problem: the k models obtained in the previous step are used to obtain the classification results by voting.

The random forest optimized metrics are shown in Table 2.

Table 2: Random Forest Optimized Metrics.

Random Forest Classification	**Accuracy**	**F1**
Flight technique score	0.883	0.859

2.4. XGBoost classification model

2.4.1. Model Building

When training the model, the XGBoost algorithm [4] generates each tree by determining whether the node has "gained" before and after the node split, and then specifying whether the node is split or not, while controlling the depth of the tree through parameters. After a decision tree is generated, it needs to be pruned to prevent overfitting. The tree generated in the mth round learns the residuals between the true value and the predicted value in the m-1th round, so that the model prediction gradually approximates the true value.

XGBoost has several benefits:

(1) A regularization term is added to the objective function to reduce the possibility of overfitting, and not only the first-order derivative is used, but also the second-order derivative is used.

It also uses second-order derivatives, which makes the loss function more accurate and allows customizing the loss.

(2) Parallel optimization is possible, and XGBoost is parallel in terms of feature granularity.

(3) Considering the handling of sparse values, the ability to set the default direction of branching for missing values or specified values greatly improves the efficiency of the algorithm.

(4) Column sampling is allowed, which can suppress overfitting and reduce computational effort at the same time.

2.4.2. Parameter tuning and solving of the model

We first call Python to wrap XGBoost to fit customer satisfaction, and then perform parametric tuning optimization on XGBoost. In the parameter tuning of XGboost, we divide the main steps into three steps:

Step 1: When building the XGboost model, we first construct the model with the default values of each parameter and calculate the evaluation criteria of the initial model.

Step 2: The parameters learning_rate, n_estimators, and max_depth are tuned. The learning_rate is the learning rate, default is 0.3, which is used to control the iteration rate and suppress overfitting in the classification task; n_estimators is the number of boosting iterations (the number of weak classifiers); max_depth is the maximum depth of the tree, which is usually used to avoid overfitting. max_depth depth is the number of iterations of boosting (number of weak classifiers); max_depth is the maximum depth of the tree. The tuning of the regularization parameters lambda, alpha, which reduce the complexity of the model and thus improve the performance of the model.

Step 3: The optimal model is constructed using the optimal combination of parameters selected in step 2, and the evaluation criterion values of the optimized model are calculated, and the fit of the optimized model is significantly improved compared with the pre-optimized model.

The calculated optimized metrics are shown in Table 3 below.

Table 3: XGBoost optimization table.

XGBoost Classification	Accuracy	F1
Flight technique score	0.892	0.861

2.5. LightGBM Classification Model

2.5.1. Model Building

LightGBM is an efficient implementation of XGBoost. The idea is to discretize the continuous floating-point features into k discrete values and construct a histogram of width k. The histogram is then traversed through the training data to calculate the cumulative statistics of each discrete value in the histogram. The training data is then traversed and the cumulative statistics of each discrete value in the histogram is calculated. For feature selection, only the discrete values of the histogram need to be traversed to find the optimal segmentation points; and the use of a leaf-wise strategy with a depth limit saves a lot of time and space expenses.

LightGBM is a high-speed, distributed, and high-performing gradient boosting framework based on decision tree algorithms for sorting, classification, regression, and many other machine learning tasks. The core ideas of LighGBM algorithm are: histogram algorithm, Leaf-wise splitting strategy, direct support for class features, histogram-based feature optimization histogram Histogram-based feature optimization [3].

2.5.2. Parameter tuning and solving of the model

The LightGBM model has more parameters, but the tuning rules are similar to those of every decision tree based model, first determine 0.1 as the initial learning rate, which can make the model converge faster. Then, we determine the number of decision trees and the maximum depth of decision trees by grid search, then determine the maximum number of leaf nodes, and finally adjust the minimum number of samples of leaf nodes to prevent overfitting of the model.

The optimized metrics are obtained in the following table 4.

Table 4: LightGBM Optimized Metrics.

LightGBM Classification	Accuracy	F1
Flight technique score	0.885	0.860

2.6. Model fusion solving

Stacking [5] is to use the initial training data to learn several base learners and then use the prediction results of these learners as a new training set to learn a new learner. In this paper, we have trained three base learners, Random Forest, LightGBM and XGBoost, and the output of these three learners is used as a subset training set for the secondary learners. For the secondary learners, we take the classification model because our base learner is a strong learner, and the secondary learners choose a simple model to avoid the overfitting phenomenon because the learning effect of different learners is combined, and the stacking method can make the final fused model more stable and perform better.

The summary of the metric scores of each model is shown first, and the fused model is found to be better than the strong learner alone from the training effect, and the fused fits of Random Forest, LightGBM and XGBoost and all three are shown in the table 5:

Table 5: LightGBM Optimized Metrics.

Models	Accuracy	F1
Random Forest	0.883	0.859
XGBoost	0.892	0.861
LightGBM	0.885	0.860
Stacking Fusion Model	0.913	0.898

3. Automated Early Warning Model

3.1. K-means clustering modeling

The K-means algorithm proposed by B. MacOueen in 1967 is by far one of the most influential techniques among the many clustering algorithms used for scientific and industrial applications.

① For the set of data objects, K objects are arbitrarily selected as the initial class centers.

② We reassign each object to the most similar class based on the mean of the objects in the class.

③ We update the class average, i.e., calculate the average value of objects in each class.

④ We repeat step (②) and (③) until no more changes occur.

The combination of Python and SPSSpro yielded the following results, for the above feature variables were divided into a total of five categories, and the clustering visualization was as shown in figure 2:

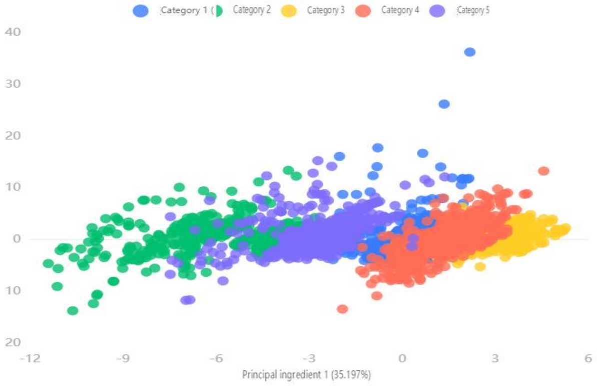

Figure 2: Aircraft parameter measurement data null heat map.

It is obvious that avg (COG NORM ACCEL), PITCH ATT RATE, avgROLL ATT, PITCH ATT RATE, Inertial Vertical Speed are divided into 5 categories, and category 1 has the largest weight.

For each of these five eigenvalues we have selected for normal distribution visualization, only avg COG NORM ACCEL and avgROLL ATT normal distribution visualizations are shown in figure 3:

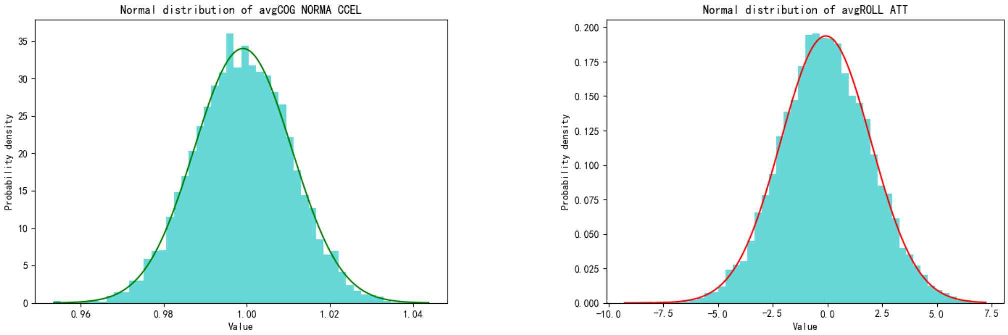

Figure 3: Visualization of normal distribution.

3.2. K-means clustering modeling

3.2.1. Decision tree modeling

① Calculate the information gain of all chemical content for the current sample set

The information gain of feature B on the training data set D is g(D, B) with the value of the chemical content as feature B and the value of the missing value processed in Annex II data as the training data set D.

$$g(D,B)=H(D)-H(D\,|\,B)$$

$$H\left(\mathrm{x}\right)=-\sum_{i=1}^{n}p(X_i)\log(p(X_i))$$

H(x) represents the information entropy, where p(xi) represents the probability of random event X_i.

② Select the attribute with the greatest information gain as the test attribute, and classify the samples with the same value of the test attribute into the same subsample set: if the class attribute of the subsample set contains only a single attribute, branch to the leaf node: otherwise call the algorithm recursively for the subsample set.

Program in Python to bring the data into fast discriminative classification.

3.2.2. Automation realization

Combined with the use of python, while loop, real-time data acceptance and combined with decision trees for fast classification.

The specific implementation steps are as follows:

(1) Develop the categories by k-means clustering in advance by dividing the 5 classes into value ranges;

(2) Using a while loop;

(3) Loading real-time flight data and adding it to the buffer;

(4) Performing data updates;

(5) Perform decision tree classification quickly in real time and determine whether to issue a warning and exactly how many levels.

4. Conclusions

For the flight technology assessment model, the results of LightGBM, XGboost and Random Forest were fused by Stacking model in this paper, and their advantages were combined to build the final aircraft technology assessment prediction model. This flight technology assessment model has an accuracy of 0.913 and an F1 value of 0.898. In building the automated warning model, the landing G-value data were subclassified into a total of five classes using a K-means clustering model and yielded significant p-values of 0.000*** for all, indicating a reasonable classification.

The sample of this question has multiple features, and the use of random forest can handle high-dimensional data well, and random forest is usually more accurate than a single decision tree, and can handle multiple types of data. The random forest is able to handle both numerical and categorical data involved in the question. Using XGBoost integrated learning method, XGBoost can greatly improve the training speed due to its parallel processing and caching mechanism, and XGBoost usually has a good performance with high accuracy. With stacking fusion model, the prediction information of multiple basic classifiers can be used and combined to produce the final prediction result, so the accuracy of stacking fusion model will be higher than that of a single basic classifier. The rationality and sensitivity of using k-means clustering method to establish early warning mechanism level indicators are good and have strong practical application value.

References

[1] Xie Jiayi. Research on Unstable Approach Risk Analysis and Early Warning Technology Based on QAR Flight Big Data [D]. Wuhan University, 2021.

[2] Ma Li. Optimization and improvement of random forest algorithm [D]. Jinan University, 2016.

[3] Shu Shiwen. LightGBM model and its application [J]. Information Record Materials, 2022, 23(07): 219-222.

[4] Sun Ruishan; Xiao Yabing. Research on the structure of civil aviation pilot operation characteristics index based on QAR record data [J]. China Safety Production Science and Technology, 2012(11)

[5] Shi Jiaqi, Zhang Jianhua. Load forecasting method based on multi-model fusion Stacking integrated learning approach [J]. Chinese Journal of Electrical Engineering, 2019, 39(14):4032-4042.