Problem Chosen	2023 MCM/ICM Summary Sheet	Team Control Number
A		23080

A Human Behavior Classification Model Based on Feature Extraction and Multi-class Support Vector Machine

Summary

An important aspect of understanding human behavior is identifying and monitoring daily activities. Human behaviour can be classified using data from a range of different sensors.

In our paper, we successfully build a human behavior classification model based on feature extraction and multi-classification support vector machine.

In the first part of our paper, we analyze the given data, propose two methods to extract the feature data, and build a multi-classification support vector machine model to classify the data.

In the second part of our paper, we evaluate the generalization ability of the model by comparing three relevant indexes in machine learning with the performance of different kernel functions and different classifiers.

In the third part of our paper, we adjust the key parameters of the model through grid search, and finally obtain the optimal parameters to deal with the overfitting problem of the model and optimize the model.

Finally, we summarize the model, point out the shortcomings of the model, and propose feasible improvement methods, which can improve the classification accuracy, generalization ability and anti-overfitting ability of the model.

Keywords: support vector machine, kernel function, variance, Pearson correlation coefficient, grid search

Contents

1 Introduction

1.1 Background

Recognition and classification of human behavior is a hot topic in computer field. Activity recognition is to identify the actions performed by a person through a given series of data, thus we can understand people's behavior activities. Embedded intelligent devices and wearable sensor data can be used as information sources for activity recognition. Human activity recognition system based on sensor data is widely used in health care, dynamic monitoring, human-computer interaction and so on. Therefore, how to realize accurate recognition of human behavior through the data collected by sensors has become a top priority. In fact, people tend to identify movement more from its structural features. To put it simply, it is necessary to design a model according to the given human behavior data, extract and classify various behavior characteristics. Then machine learning is used to train the processed data to get the eigenvalues calculated by the model. Finally, it is compared with the data collected by the wearable activity recognition system, thus we can determine whether the data of the wearable device can be identified as a human-like behavior.

1.2 Promlems to be Solved

According to the question, we need to build a model to extract various human behavior characteristics, and compare the extracted characteristics with the collected data, so that the data can be classified as a certain human behavior. Since the wearable sensor collects real-time data every 5 seconds and the data set is very large, appropriate pretreatment methods should be adopted to reduce the dimension of the data in order to design a more efficient classification algorithm. The amount of data provided by the topic is limited, and the model used should be as suitable as possible for fresh data samples outside the given data set. Therefore, it is necessary to develop multidimensional evaluation indexes to evaluate the generalization ability of the model. Finally, considering the small amount of sample data and the high complexity of the model, the model may have inconsistent performance in the training set and the test set, so it is necessary to optimize the model to overcome the overfitting problem.

2 Problem Analysis

2.1 Restatement of the Problem

The core of the problem can be summarized as follows: given multiple samples belonging to 19 categories, each sample contains features collected by a total of 45 different sensors belonging to 5 sensor units. According to these characteristics, a model is built which can divide the sample data into 19 categories.

On this basis, we need to adopt appropriate and achievable strategies to evaluate and improve the generalization ability of the model for different data, and to solve the problem of over-fitting of the model.

2.2 Solution Analysis

(1) The given raw data has the characteristics of many features and large cardinality, so it cannot be directly used for the establishment of the model. Therefore, we need to perform data

preprocessing and feature extraction;

(2) The core of the problem involves the classification of a large amount of data, which we can implement by using relevant algorithms of machine learning;

(3) When evaluating the model, we can use the common evaluation indicators of machine learning, such as accuracy rate, precision rate, recall rate, F1 score and other indicators for evaluation. In addition, we can also switch to different classification models for classification and evaluate different results;

(4) For generalization and overfitting problems, we can use methods such as adjusting related parameters, parameter regularization, and feature dimensionality reduction.

To sum up, this problem can be solved by machine learning-based classification model. The solution flow is shown in Figure 1:

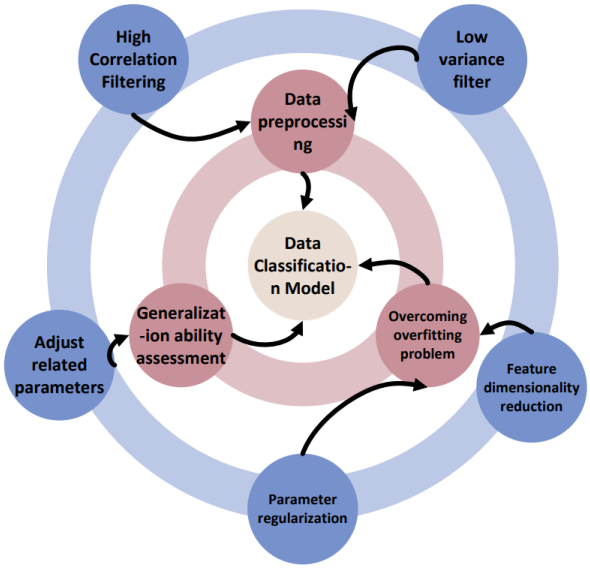

Figure 1: The solution flow

3 Models

3.1 Basic Model

To solve the above problems, we used low variance filtering and high correlation filtering to extract features from the original data, and used appropriate statistical variables to compress the data. After that, we adopted a machine learning classification model based on multi-classification support vector machine algorithm to classify the data, and conducted model evaluation and optimization.

3.1.1 Terms, Symbols and Definitions

Definition 1: Data characteristics – column definitions

For a given data matrix, define a feature of the data for each column of the matrix. In this problem, the characteristics of the sample population correspond to 45 columns of data collected by 45 sensors. These features are numbered by x_0,$x_1 \ldots x_{44}$.

Definition 2: Data categories, testers, time segments, and individual samples – row definitions

According to the name and number of the original data file, the following definitions are made: This problem intends to classify 19 kinds of human activities, and use 1-19 for numbering; The data of each activity came from 8 testers, represented by p_0,$p_1 \ldots p_8$; Each tester collected a total of 60 time segments (represented by s_{01},$s_{02} \ldots s_{60}$). Each segment contained data collected by the relevant sensor at a total of 125 specific moments within 5 seconds, represented by l_0,$l_1 \ldots l_{124}$.

Definition 3: total data set, training set and test set

A given set of all data is called the total data set. Among them, the data set that actually participates in model training is called training set, and the data set that actually participates in model testing is called test set. In this problem, for each category, p_1-p_6 is taken as the training set and p_7 and p_8 as the test set.

Definition 4: Pearson correlation coefficient

Pearson correlation coefficient is used to measure the degree of linear correlation between two variables, and its value is between -1 and 1. The calculation formula of Pearson correlation coefficient is shown in equation1 .

$$\rho_{X,Y} = \frac{\sum(X - \overline{X})(Y - \overline{Y})}{\sqrt{\sum(X - \overline{X})^2}\sqrt{\sum(Y - \overline{Y})^2}} \tag{1}$$

3.1.2 Assumptions

It is assumed that the given 19 types of data all have obvious characteristics, which can be effectively classified, and the model can solve the problem in a relatively short time, and the data in the training set and data set are in line with the format specification.

3.1.3 The Foundation of Model

(1) Feature filtering method

- Low variance filtering:

  Since the data set is very large and not every feature has high differentiation, it is necessary to select the feature of the data. The principle of low variance filtering is: Firstly, each feature is measured according to divergence. That is, to calculate the variance corresponding to each characteristic value in the sample, select an expected threshold, and compare the variance with the threshold. The variance represents whether the sample shows significant differences in characteristics. If the variance of a certain characteristic value is too small, lower than the threshold value, it means that the feature cannot distinguish all data significantly. Therefore, by retaining all features above the threshold value, the feature dimension of the dataset can be reduced and the accuracy of the model can be improved.

- High correlation filtering:
  In order to reduce the complexity of data set, it is necessary to further reduce the dimension of data set features. Correlation between data sets is a measure of how related they are. If two features are highly correlated, it means they have similar trends and may carry similar information. Therefore, when the correlation coefficient exceeds a certain threshold, one of the features can be discarded, thus reducing the dimension of the data set and improving the model performance.

(2) Normalization and standardization

- Max-min normalization:
  In the process of data preprocessing, it is necessary to standardize or normalize the data in order to eliminate the dimensional influence. The maximum and minimum normalization processes the maximum and minimum values in the data, and the processed values are between $[0, 1]$. We normalized the data in feature engineering, and its calculation method is shown in equation 2.

$$x^{'} = \frac{x - min}{max - min} \tag{2}$$

- Z-Score standardization
  Z-Score standardization is a centralized method based on the mean and standard deviation of the original data. After standardization, the mean value of the data is 0 and the variance is 1. We do a Z-Score normalization of the data in machine learning.

$$\sigma = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(x_i - \mu)^2} \tag{3}$$

$$z = \frac{x - \mu}{\sigma} \tag{4}$$

(3) Support Vector Machine algorithm (SVM)

Support vector machine (SVM) is a kind of generalized linear classifier for binary classification of data according to supervised learning. Its decision boundary is the maximum margin hyperplane for solving learning samples. SVM is a sparse and robust classifier that uses hinge loss function to calculate empirical risk and adds regularization term to the solving system to optimize structural risk. SVM is one of the common kernel learning methods, which can carry out nonlinear classification by kernel method.

Standard SVM is an algorithm designed based on binary classification problems, which cannot directly deal with multiple classification problems. The calculation process of standard SVM is used to construct multiple decision boundaries in an orderly manner to realize multi-classification of samples, which are usually implemented as ”one-to-many” and ”one-to-one”.

A one-to-many SVM establishes m decision boundaries for m classifications, and each decision boundary determines the attribution of one classification to all other classifications. One-to-one SVM is a voting method. Its calculation process is to establish decision boundaries for any 2 of m classifications, that is,

$$\frac{m(m-1)}{2} \tag{5}$$

decision boundaries in total. Sample categories are selected according to the category with the highest score in the discriminant results of all decision boundaries. One - to - many SVM can compute all decision boundaries iteratively by modifying the optimization problem of standard SVM. Here, we adopted a "one-to-one" SVM.

Considering that the mathematical principle of the SVM model is relatively complex, this paper will not explain its detailed principle, but only introduce three important parameters that affect the establishment of the model.

- Kernel function

  Common kernel functions are shown in Table 1

Table 1: Common kernel functions

Name	Expression
Linear Kernel Function: kernel= 'linear'	kernel=$< x, x^{'} >$
Polynomial Kernel Function: kernel= 'poly'	kenel=$(\gamma < x, x^{'} > +r)^{d}$
Radial Basis Kernel Function: kernel= 'rbf'	kernel=$exp(-\gamma \left\|x - x^{'}\right\|^{2})$
Sigmod Kernel Function kernel= 'sigmod'	kernel=$tanh(\gamma < x, x^{'} > +r)$

- Regularization parameter C

  The intensity of regularization is inversely proportional to C and must be strictly positive. The regularization coefficient C is introduced, which can be understood as allowing the weight of errors to be divided (the larger the value, the less errors are allowed). When C is small, a small number of samples are allowed to be divided into errors. The larger C is, the more errors can not be tolerated and it is easy to overfit. The smaller C is, the easier it is to underfit. If C is too large or too small, the generalization ability becomes poor.

- Nuclear coefficient γ

  γ is a hyperparameter for nonlinear support vector machines. One of the most common nonlinear kernel functions is the radial basis function (rbf). The γ parameter of rbf controls the influence distance of a single training point. A lower γ value indicates a larger radius of similarity, which leads to more points being grouped together. For high γ values, the points must be very close to each other for them to be considered the same group (or class). Therefore, models with very large γ values tend to be overfitted.

3.1.4 Solution and Result

(1) Data preprocessing and feature extraction

First, we analyzed the total data set using pandas and found no missing or ill-formed values. After that, the maximum and minimum normalization of the training data is processed, and the characteristics shown in Figure 2 are found:

With 0.005 as the threshold, 16 groups of feature sets numbered x_6, x_7, x_8, x_{15}, x_{16}, x_{17}, x_{24}, x_{25}, x_{32}, x_{33}, x_{34}, x_{35}, x_{41}, x_{42}, x_{43}, x_{44} can be screened by low variance filtering method.

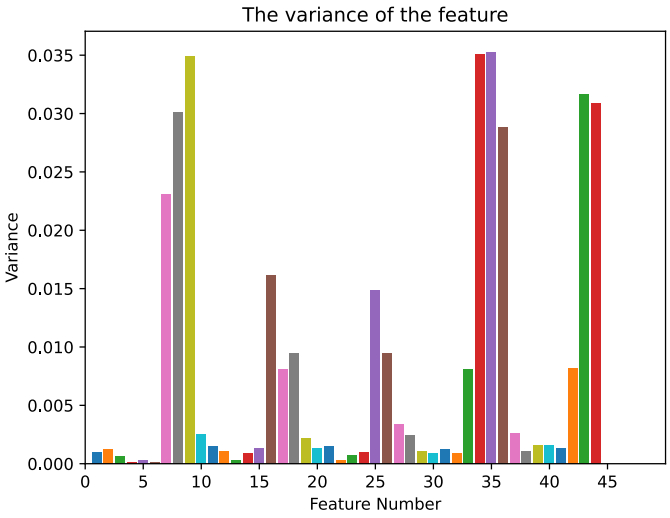

Figure 2: Low variance filtered results

Subsequently, high correlation filtering was used for further filtering. After many attempts and investigation of relevant cases, the threshold is determined to be 0.8. Pearson correlation coefficient of the above 16 feature sets was calculated, and the features shown in Figure 3 were obtained:

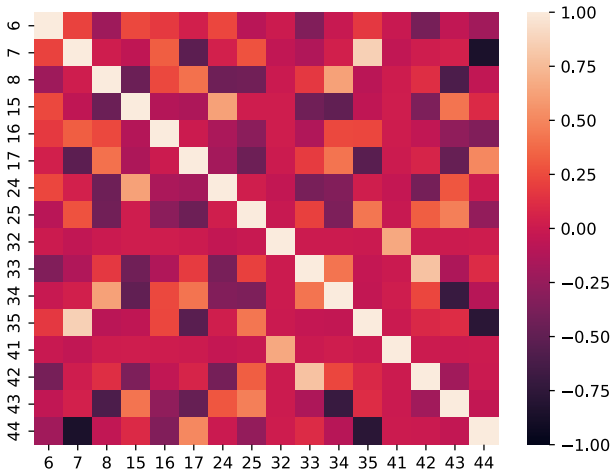

Figure 3: High correlation filter results

Finally, 13 groups of significant features were selected: x_6, x_7, x_8, x_{15}, x_{17}, x_{24}, x_{25}, x_{32}, x_{33}, x_{34}, x_{41}, x_{43}, x_{44}.

The 13 groups of significant features were selected from the training set and the test set respectively, and the mean and variance of the data in each time segment were extracted for data compression and feature refinement to construct the training set with 26 features and 6840 data and the test set with 26 features and 2280 data. And we conduct Z-score standardization for model training.

(2) Model training

SVM classifier in scikit-learn database was used to construct the model. Since the number of characteristic dimensions of data is much smaller than the number of samples, the kernel function of "rbf" is selected to construct a classification model biased towards nonlinear and high dimensions. With the parameters C = 64, γ = 0.0078125, the classification results as shown in Table 2 were obtained:

Table 2: Classification result

training set score	0.9931286549707602
test set score	0.823245614035087
The number of errors in the test set	403
Test set error percentage	0.17675438596491228

3.1.5 Result Analysis

The score of the training set is obtained based on the accuracy index of the prediction results, indicating that on the given training set, the accuracy of our classification model reaches 99.3 %, which can effectively classify the given data with a large number of features. In the test set, the classification accuracy of the model is up to 82.3 %, which has excellent performance. Among the error classification results on the test set, the error data numbering distribution after being classified into the error category is shown in Figure 4:

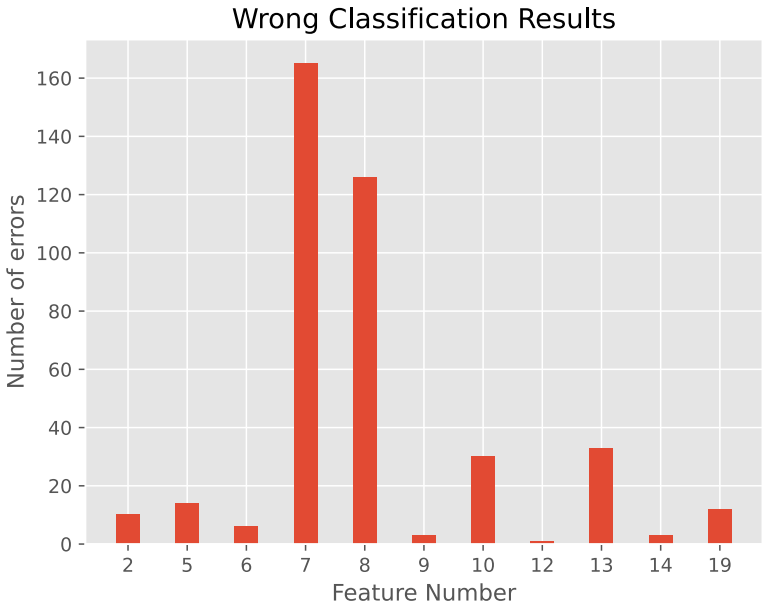

Figure 4: Error data number distribution

It can be seen that category 7 and Category 8 account for a high proportion of errors, and the model has a high misidentification rate for category 7 and category 8.

3.1.6 Strength and Weakness

- Strength

  The model adopts SVM multi-classification algorithm, which has the characteristics of good generalization, unique global optimal solution and robustness, and shows unique advantages in solving nonlinear and finite sample classification problems. In addition, SVM has few parameters, so the packet tuning and optimization of SVM are relatively simple. Therefore, SVM method is used to learn the data read and output the classification model.

- Weakness

  Since the SVM algorithm is not applicable to a large amount of data, we compress the features of the data, which is bound to lose some effective information. In addition, the SVM algorithm has a high time complexity, which is between $O(n_{features} \times n_{samples}^2)$ and $O(n_{features} \times n_{samples}^3)$. Therefore, this model is suitable for general configuration of computers and machine learning models. When the data volume is large, the operation efficiency will be reduced.

3.2 Model Improvements for Generalization

3.2.1 Extra Definitions

Due to the high cost of data, we need to make the model have good generalization ability under the limited data set. Therefore, we need to study and evaluate this issue in terms of specific metrics.

Extra Definition 1: Confusion Matrix

Take the 2×2 confusion matrix as an example, as shown in Figure 5.

Confusion Matrix		Actual Value	
		P	N
Predictive Value	P'	TP	FP
	N'	FN	TN

Figure 5: Confusion Matrix Example Diagram

- P indicates positive example;
- N indicates negative example;
- FP indicates the number of samples that are actually negative but are predicted to be positive;
- TN indicates the number of samples that are actually negative and are predicted to be negative;
- TP represents the number of samples that are actually positive and are predicted to be positive;
- FN indicates the number of samples that are actually positive but are predicted to be negative;
- $P^{'}$ indicates the number of samples that are predicted to be positive;
- $N^{'}$ indicates the number of all samples that are predicted to be negative.

3.2.2 Solution and Result

For the evaluation of the model, the data used in this model was standardized by Z-score. We use the following three indicators to evaluate the model.

- Accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

- Precision: (Here, the macro average accuracy is used for calculation)

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

- Recall rate:

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

The accuracy, precision and recall rate of the model are shown in Table 3:

Table 3: Model Evaluation Results

Evaluation Index	**Data Result**
Accuracy	Accuracy = 82.325 %
Precision	Precision = 88.335 %
Recall rate	Recall = 82.325 %

3.2.3 Generalization ability comparison

(1) We use different kernel functions for model training and evaluate the model according to the indicators. Here we only cite the results of training with the linear kernel function kernel='linear'.

Table 4: The training results of the linear model

Evaluation Index	**Data Result**
Accuracy	Accuracy = 81.271 %
Precision	Precision = 83.282 %
Recall rate	Recall = 81.272 %

It can be seen from Table 4 that choosing kernel='rbf' improves the generalization ability of this model.

(2) We use different algorithms to build classification models, and here we only cite the results of model building using random forest classifiers.

Table 5: Training Results of Random Forest Classifier

Evaluation Index	**Data Result**
Accuracy	Accuracy = 82.105 %
Precision	Precision = 85.861 %
Reacall rate	Recall = 82.105 %

It can be seen from Table 5 that multi-classification svm has good generalization ability when classifying biased nonlinear and multi-feature data.

(3) In the original test set without Z-score normalization, we randomly add 2 % noise for verification, and the accuracy rate was 81.172 %. It can be seen that the model has good robustness.

It should be pointed out that although there are differences in the evaluation indicators of different classification models, the differences are small, reflecting that the given data to be classified has category impact factors that are biased towards linear correlation.

In addition, in order to improve the generalization ability of the model, we adopt an optimized method when constructing the model for parameter adjustment, which will be introduced together with the improved method for the overfitting problem in 3.3.

3.3 Model Improvements for Overfitting Issues

3.3.1 Solution and Result

SVM processing data requires standardization. We adopt the Z-score standardization method, which has a certain correction ability for overfitting. In addition, when performing feature extraction, we first perform low-variance filtering, and then perform high-correlation filtering, which improves the generalization ability of the model's linear and non-linear data distribution methods, and prevents overfitting to a certain extent. But these two items are not the main improvement measures.

For multi-class SVM, under the completely linearly separable data set, the support vector machine has no overfitting problem because its solution is unique. In the case of non-linear inseparability, although the objective function of SVM adopts a structural risk minimization strategy, the SVM still has the problem of overfitting due to the kernel function introduced by the kernel that allows misclassification.

The SVM model we built is a support vector machine with kernel function + soft interval. Then, the following reasons lead to SVM overfitting:

(1) The kernel function leads to overfitting. We can correct it by adjusting the kernel coefficient parameter γ of the nonlinear kernel function rbf;

(2) The interval we require is too large, that is, when the parameter C in the soft-margin support vector machine is too large, it means that we pay more attention to the interval and insist on completely separating the data. When C tends to infinity, it is equivalent to hard-margin SVM. When C is too small, the model is prone to underfitting.

In machine learning, we can draw a verification curve to obtain the optimal hyperparameters, but the verification curve can only obtain one optimal hyperparameter at a time. If there are many permutations of multiple hyperparameters, we can use grid search to find the optimal combination of hyperparameters.

To sum up, we adopt the method of grid search for parameter adjustment to determine the optimal parameter (C, γ) suitable for a given data classification, using the GridSearchCV tool in scikit-learn, in the interval Search for C in $[2^{-5}, 2^{15}]$, and search for γ in the interval $[2^{-9}, 2^{3}]$. The process is shown in Figure 6.

In the end, we find that the optimal kernel is rbf, and the optimal (C, γ) parameter is (64, 0.0078125), which effectively overcomes the overfitting problem caused by overtraining.

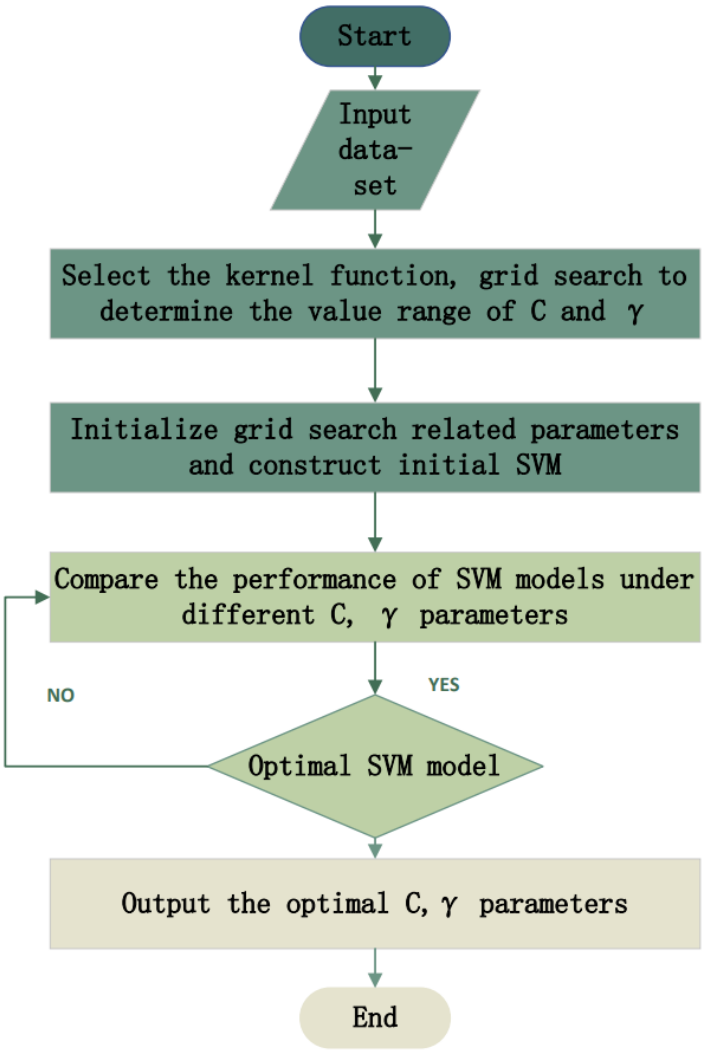

Figure 6: Model Tuning Flowchart

4 Conclusions

4.1 Conclusions of the Problem

In order to solve the classification problem of human behavior data, we carried out the following model construction:

- Feature extraction based on low variance filter and high correlation filter
- Classifier based on multi-class support vector machine
- Quantitative evaluation index and comparative evaluation of model generalization ability
- An effective solution to the problem of model overfitting

4.2 Methods Used in Our Models

- Algorithm:

  Low variance filtering

  High correlation filter

  Multi-Class support vector machines

  Confusion matrix, accuracy, precision and recall

  Grid search

- Tools:

  scikit-learn

  numpy,matplotlib,pandas

4.3 Application of Our Models

Our model can be applied to human action classification problems with multidimensional feature data, and can be generalized to classification problems given a dataset with tangible classifiable features.

5 Future Work

5.1 Weakness of the Model

Due to time constraints and limited availability of computer hardware, we still have deficiencies in model construction and tuning:

- No further feature extraction: we only extracted two features of mean and variance for each time period to increase the dimension;
- No separate classification based on each feature: we can sacrifice the amount of computation for better feature extraction;
- High complexity of the model: it takes up a lot of machine memory and takes a long time to run.

5.2 Feasible Improvement Plan

1. We can use some dimensionality reduction methods (such as pre-training for separate classification based on each feature) on the original basis to further delete some useless features, continue to reduce the complexity of the model, and improve the accuracy rate.
2. We can choose more statistics (such as range, peak value, skewness, root mean square frequency, etc.) to better compress and describe the data.
3. Stop in time. For the problem of overtraining, when the accuracy of the model does not change, stop training in time, which can effectively prevent overtraining.

4. We can use deep learning methods and use hardware and software environments that support deep learning (such as Pytorch, cuda, cudnn, etc.) to increase the amount of data for deep learning.

6 References

[1] Fan, Rong-En, et al., "LIBLINEAR: A library for large linear classification.", Journal of machine learning research 9.Aug (2008): 1871-1874.

[2] Bishop, Pattern recognition and machine learning, chapter 7 Sparse Kernel Machines

[3] Altun K, Barshan B, Tunçel O. Comparative study on classifying human activities with miniature inertial and magnetic sensors[J]. Pattern Recognition, 2010, 43(10): 3605-3620.

[4] A Tutorial on Support Vector Regression" Alex J. Smola, Bernhard Schölkopf - Statistics and Computing archive Volume 14 Issue 3, August 2004, p. 199-222.

[5] Platt "Probabilistic outputs for SVMs and comparisons to regularized likelihood methods"

[6] Ahmed M, Antar A D, Ahad M A R. An approach to classify human activities in real-time from smartphone sensor data[C]//2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR). IEEE, 2019: 140-145.

[7] Lu Junru. Research on binary classification method for high-dimensional unbalanced data based on support vector machine. Harbin Institute of Technology, 2018

[8] Crammer and Singer On the Algorithmic Implementation ofMulticlass Kernel-based Vector Machines, JMLR 2001.

[9] Liu Jia. Research and Application of Support Vector Machines in Imbalanced Data Classification. Xiamen University, 2021

[10] Liu Dongqi. Research on Imbalanced Data Classification Algorithm Based on Support Vector Machine. Zhejiang University, 2017

7 Appendix

7.1 Source Program

See attachment-demo.

7.2 Dataset

The given data set for the problem can be obtained in the original problem.