Type of the Paper (Article)

TPE-RBF-SVM Model for Soybean Categories Recognition in Selected Hyperspectral Bands Based on Extreme Gradient Boosting Feature Importance Values

Qinghe Zhao, Zifang Zhang, Yuchen Huang, Junlong Fang*

Electrical Engineering and Information College, Northeast Agricultural University;
* Correspondence: jlfang@neau.edu.cn; Tel.: (+086) 189-4505-5858 / (+852) 5513-7094

Abstract: Soybean with insignificant differences in appearance have large differences in their internal physical and chemical components, therefore follow-up storage, transportation and processing require targeted differential treatment. A fast and effective machine learning method based on hyperspectral data of soybean for pattern recognition of categories is designed as a non-destructive testing method in this paper. A hyperspectral-image dataset with 2299 soybean seeds in 4 categories is collected; Ten features is selected by extreme gradient boosting algorithm from 203 hyperspectral bands in range 400 to 1000 nm; A Gaussian radial basis kernel function support vector machine with optimization by the Tree-structured Parzen Estimator algorithm is built as TPE-RBF-SVM model for pattern recognition of soybean categories. The metrics of TPE-RBF-SVM are significantly improved compared with other machine learning algorithms. The accuracy is 0.9165 in the independent test dataset which is 9.786% higher for vanilla RBF-SVM model and 10.02% higher than the extreme gradient boosting model.

Keywords: Hyperspectral Technology; Non-destructive Testing; Soybean; Machine Learning; Support Vector Machine; Extreme Gradient Boosting; Tree-structured Parzen Estimator;

1. Introduction

Soybean (scientific name: Glycine max), is an East Asian native the legume family whose seeds are an excellent source of plant protein and lipids[1]. Soybean has the dual properties of a food crop and an economic crop. About 85% of the global soybean crop is processed into soybean oil or soybean meal, and the rest is processed in other ways or eaten directly[1,2]. Soybean with insignificant differences in appearance have large differences in their internal physical and chemical components, so follow-up storage, transportation and processing require targeted differential treatment[3]. Therefore, there is urgent need for a fast automagical non-destructive testing technology to classification of soybean varieties in the breeding, sorting and subsequent processing.

Hyperspectral technology, a non-destructive testing technology, is widely used in agriculture, food industry and other many fields. Spectral data, compared with visual image based on traditional machine vision, can provide richer information from data sources. However, it is necessary for high-throughput spectral data to cooperate with effective analysis method or advanced model to make the most of its rich amount of information of data in pattern recognition or outlier detection. Partial least squares regression and PLS-DA model are traditional analysis methods for modeling and prediction: E.M. Abdel-Rahman et al. predicted chard yield grown under different irrigation water sources from hyperspectral data by improved sparse PLS regressions[4]; A. Folch-Fortuny et al. detected decay lesions in citrus fruits by N-way PLSR discriminant analysis model in SWIR bands of spectrum[5]; T. Rapaport et al. assessed grapevine water status by fusing the both hyperspectral imaging and leaf physiology with PLSR model[6]. With the development of new machine learning technologies and the decline of computing power costs

doi:10.20944/preprints202207.0280.v1

in recent years, the intersection of hyperspectral technology and machine learning is getting more popular: L.P. Osco et al. proposed that random forest method performed well on predicted main elements and trace elements by hyperspectral data of Valencian orange leaves[7]； C. Erkinbaev et al. applied perceptron model with backpropagation in artificial neural networks in the SWIR bands of wheat that tended to perform better than PLSR models in the hardness testing task of seed single grain[8]； ZHANG X. et al. applied GA-SVM as the main algorithm with selected features in hyperspectral data by continuous projection of the NIR bands to complete adulteration identification for saline holothurian with explanation on spectroscopic analysis[9].

Support vector machine, a classic mature machine algorithm widely used in supervised learning tasks in various fields, has the advantages of both high robustness and great performance in most usage [10–12]. But there is huge computing resource required when fitted and predicted by high-dimensional dataset like hyperspectral data, so that a reasonable dimensionality reduction method is required to necessary as pipeline before the data input. Direct dimensionality reduction based on mathematical calculation and sequential feature selection based on modelling are two more commonly used method. LI Y. et al. applicated the principal component analysis in frequency domain spectral data for SVM model to distinguish heavy metal pollution in crops[13]; M. Pal et al. compared and discussed several feature selection algorithms such as recursive feature elimination and correlation-based feature selection for SVM method in AVIRIS, a remote sensor of hyperspectral dataset[14]; In addition, the hyperparameter configuration in the modelling of SVM is a key for better performance as well, and it is necessary to achieve a limited number of model self-optimized iterations in an effective way. It is not only to improve the performance of the machine learning model itself, but also a crux link to realize the auto machine learning in a specific actual production environment in the future. In recent years academic research, meta-heuristic algorithms such as particle swarm optimization[15,16], simulated annealing algorithm[17], and genetic algorithm[16] are more concerned as the model optimization tricks, but random search or grid search is still widely used in practical deployed application[18,19]. The sequential model-based global optimization is an optimization method that has been applied to large-scale neural networks yet with better performance than traditional method in practical engineering[19,20]. Further, it has been verified by this paper that it can also effectively improve performance in an SVM model with the hyperspectral sub-band dataset.

This paper proposes a multi-classification method for soybean seed by hyperspectral data based on support vector machine with Gaussian radial basis function kernel (RBF-SVM) optimized by tree-structured Parzen Estimator (TPE) after feature selection as follows：

- Dataset construction: the hyperspectral images range from 400nm to 1000nm collection from 2299 soybean seeds with 4 categories was completed;
- Feature selection by a boosting algorithm: an extreme gradient boosting algorithm was introduced to reduce redundancy dimensionality of hyperspectral data. Ten feature-bands were determined from the original 203 hyperspectral bands for a subset;
- Optimized RBF-SVM model with TPE: a support vector machine with Gaussian radial basis kernel function is built for the multi-classification pattern recognition task of soybean datasets, and the tree-structured Parzen estimator method was introduced to improve the model as TPE-RBF-SVM;

The four categories multi-classification accuracy of the above method, TPE-RBF-SVM, in the independent test dataset is 0.9165 (F1=0.9052), which is 9.786% higher for vanilla RBF-SVM model without TPE and 10.02% higher than the extreme gradient boosting model. Compared with other machine learning algorithms, the metrics are significantly improved as well.

2. Gaussian Radial Basis Kernel Support Vector Machine Optimized by Tree-structured Parzen Solution

2.1. Support Vector Machine with Gaussian Radial Basis Kernel

Support vector machine (SVM) is a robust supervised learning algorithm based on statistical machine learning theory jointly proposed by V. N. Vapnik and A. Y. Chervonenkis[21] The SVM algorithm iteratively searches the hyperplane in the sample space of data, and obtains a lossy partitioned hyperplane interval composed of support vector (SV) to complete the classification task in machine learning; B. E. Boser and V. N. Vapnik thereafter introduced the kernel function into the SVM to map the original sample space in the high-dimensional Hilbert space for further optimization of the model application[22]; Chih-Jen Lin et al. programed libsvm and liblinear, an efficient implementation of the quadratic programming (QP) in SVM, and then algorithm got widely used in pattern recognition and other fields in machine learning[23].

From the perspective of statistical learning, when the sample is like as the $\{(\boldsymbol{x}_i{}^{[m]}, y_i)$, i= 1, 2, 3, …, $n\}$, composed of n samples of m dimension and corresponding real values y , the support vector machine needs to solve the follows (1) iteratively:

$$\underset{\omega,b,\boldsymbol{loss}}{argmin}:\frac{1}{2}\omega^{\mathrm{T}}\omega+\mathrm{C}\sum_{i=1}^{n}\boldsymbol{loss}_i \qquad (1)$$
$$\text{s.t. } y_i\left(\boldsymbol{\omega}^{\mathrm{T}}\psi(\boldsymbol{x}_i)+b\right)\geq 1\text{-}\boldsymbol{loss}_i,\ \boldsymbol{loss}_i\geq 0$$

where $\boldsymbol{\omega}$ and b is the parameter of point-normal equation of a hyperplane, $y = \boldsymbol{\omega x} + b$; $\boldsymbol{loss}_i$ = max (0, 1-$y_i(\boldsymbol{\omega}^{\mathrm{T}}\psi(\boldsymbol{x}_i)$-$b$)) is the hinge loss of true values and predicted ones in sample space; **C** is the strength of the regularization of objective function; $\psi(x_i)$ is the map function between sample space of $\boldsymbol{x}_i$ and high-dimensional Hilbert space that introduces kernel function method later.

The objective function (1) is a convex optimization problem and it satisfies the Karush-Kuhn-Tucker conditions that is commonly solved by the Lagrangian dual method[21,24]. The objective function after conversion of the parameters to the dual problem is as follows (2):

$$\underset{\omega,b,\boldsymbol{loss}}{argmin}:\frac{1}{2}\boldsymbol{\alpha}^{\mathbf{T}}\mathbf{Q}\boldsymbol{\alpha}\ \text{-}\ \mathbf{e}^{\mathbf{T}}\boldsymbol{\alpha} \qquad (2)$$
$$\text{s.t. } \boldsymbol{y}^{\mathrm{T}}\boldsymbol{\alpha}=0,\ 0\leq\boldsymbol{\alpha}_{\mathrm{i}}\leq\mathrm{C}$$

where **Q** is the transformed positive semi-definite Hermitian Matrix, whose elements is $\mathrm{Q}_{\mathrm{ij}} = y_iy_j\psi(\boldsymbol{x}_i)^T\psi(\boldsymbol{x}_j)$; $\boldsymbol{e}$ is the ones-vector full with the number 1 to ensure the validity of the calculation；$\boldsymbol{\alpha}$ is the dual coefficient vector.

The dual objective function (2) uses Quadratic Programming to complete the iterative solution of $\boldsymbol{\alpha}$ and the support vectors $\boldsymbol{SV} = \{\boldsymbol{x}_k, k \in \mathrm{QP}(\alpha)^*\}$. In the solution and application process, the Gaussian radial basis kernel function (RBF) is introduced for mapping the samples into the Hilbert space. For the prediction $\hat{y}$ of a new sample ($\boldsymbol{x}$, y), it can be expressed in the form as the follows (3):

$$\hat{y} = sgn\sum_{\mathrm{i}\in\boldsymbol{SV}} y_i\boldsymbol{\alpha}_i\kappa(\boldsymbol{x}_i,\boldsymbol{x})+b \qquad (3)$$

where $\kappa(\boldsymbol{x}_i, \boldsymbol{x})= \psi(\boldsymbol{x}_i)^T\psi(\boldsymbol{x})= exp(\text{-}\frac{||\boldsymbol{x}_i\text{-}\boldsymbol{x}||}{2\sigma^2})$ and the σ is the free parameter to control the mapping process in Gaussian radius basis function[24].

That is the general or vanilla Gaussian radial basis kernel function support vector machine short for RBF-SVM.

2.2. Optimization of SVM with Tree-structed Parzen Estimator

Based on the derivation (1) to (3) of the above mathematical principles, the proven SVM model needs to artificially determine the penalty scaling strength C in equation (1) and the free parameter σ of the kernel function control mapping in equation (3). It is customary to express σ with $\gamma = \frac{1}{2\sigma^2}$ and γ is usually set as $\frac{1}{m}$ or $\frac{1}{m} \times \sum \frac{(x_i - \bar{x})^2}{n}$ according to the empirical formula for dataset with *n* samples and *m* features[25]. Penalty scaling strength C generally is determined as 10^k ($k \in \boldsymbol{N}$) experimentally in the validation dataset or cross-validation[25].

Optimization for hyperparameters can be further determined to verify the prediction performance of the SVM as the follows (4):

$$\underset{C,\,\gamma}{argmin}{:}L\big(\, y_i,\ \hat{y}{=}f(\boldsymbol{x}_i{,}C{,}\gamma)\ \big),\big(\boldsymbol{x}_i{,}y_i\big){\in}\textbf{valid} \tag{4}$$

where *L* is the metrics (the accuracy in this paper) or a loss function to evaluate model performance in validation dataset; *f* is the fitted RBF-SVM model with corresponding hyperparameters, C and γ, in optimization.

Objective function (4) can be seen as an optimization issue of best hyperparameters pair **z** = (C, γ) in the sample space in essence of black-box process[19]. We will implement this by a Bayesian optimization method, the tree-structured Parzen estimator (TPE). It is one of sequential model-based optimization (SMBO) method proposed by Ozaki et al. in 2020 that originally optimizes for large-scale neural networks[20]. The pseudo code of Tree-structured Parzen Estimator method is as follows:

Table 1. The pseudo-code of tree-structured Parzen estimator algorithms

Algorithm. TPE algorithm (for RBF-SVM)

```
Initialization $\mathbf{H}_0 = \varnothing$, $\mathbf{z} = \mathbf{z}_0$
For: k = 1 to $I_{\max}$
    Update hyperparameters: $\mathbf{z}^* = argmin(\ EI_k(\mathrm{P}, \mathbf{z}_k[H_{k-1}])\ )$
    RBF-SVM  repeat fitting and evaluating: $\boldsymbol{L}_k$
    Update optimization history: $\mathbf{H}_k = \mathbf{H}_{k-1} \cup < EI_k, \boldsymbol{L}_k >$
End
Return $\mathbf{z}^* = argmin{:}\boldsymbol{L}(y_i, \hat{y} = f(\boldsymbol{x}_i, C, \gamma))$
```

TPE algorithms would build the probabilistic model for optimization objective function targeting **z** by the surrogate function, *EI* values, and threshold *P* in iteration history and evaluate the results of tuning values ***L*** in validation dataset or cross-validation method. The surrogate function is as follows[20]:

$$EI = \frac{\int_{-\infty}^{+\infty} \max(\boldsymbol{L} * - \boldsymbol{L}[H_{i-1}], 0) \times p(\boldsymbol{L}[H_{i-1}]) dL}{\mathrm{P} + (1 - \mathrm{P}) \times \frac{g(\boldsymbol{z}[H_{i-1}])}{h(\boldsymbol{z}[H_{i-1}])}} \tag{5}$$

where *g(z)* and *h(z)* are the probability distributions when the value of ***L*** is greater than or less than the threshold *P* respectively whose distributions come from the historical information ***H*** accumulated in the previous *k-1* iterations.

The combination of TPE method and RBF-SVM algorithm is going to realize hyperparameters tuning and model optimization based on SMBO of Bayesian optimization according to the above theory. When the training, validation and test dataset meet the independent and identically distributed (IID), this method will balance the empirical risk and generalization risk of the model to the greatest extent, and finally complete a more accurate modeling process through the automatic configuration of controlled hyperparameters.

3. Crux spectrum feature selection based on extreme gradient boosting

The spectral data is high-throughput sequencing that would causes dimensional explosion when support vector machine directly applicates the kernel function to map raw data into higher dimensional space that is heavily depended on computing power and time however. In addition, there is high redundancy for hyperspectral information itself and not all bands have a positive contribution to fit the model in classification task. Key feature selection of raw data is necessary to extract more crux information than RGB image and less redundancy for model building for precision fitting, therefore a method based on boosting algorithm is adopted in this paper.

Gradient Boosting, a boosting branch strategy of ensemble learning methods, is originally proposed by J.H. Friedman, who combined series of weak learners called meta learners with the specific strategy for single machine learning task[26]. And the final model will have better performance than single meta learners ensembled. Extreme Gradient Boosting is a gradient boosting algorithm proposed by Tianqi Chen in 2011[27]. The algorithm is constructed by CART trees as the main meta learner and set the the quadratic Taylor expansion of the loss function in the boosting iteration. Model based on trees has the better interpretability and ensemble process effectively improves the performance than single tree model so that it has been widely applicated deployed in various fields requiring reliance on feature interpretability.

As an additive model, objective function of the meta learner *f* of the single round in the dataset ($\boldsymbol{x}_i$, y) is as follows:

$$obj(t) = \sum_{i=1}^{n} \boldsymbol{loss}\left[y_i, \hat{y}_i^{t-1} + f_t(\boldsymbol{x_i})\right] + \sum_{i=1}^{t} \Omega\left(f_i\right) \tag{6}$$

where t is the boosting rounds in model and Ω is the regularization part of the tree.

After the quadratic Taylor expansion of the loss function and bring the structural parameters of the CART tree, the objective function is as the follows (7):

$$obj(t) = \sum_{i=1}^{n} \left[g_i^t \cdot f_t(\boldsymbol{x_i}) + \frac{1}{2} \cdot h_i^t \cdot f_t(\boldsymbol{x_i})^2\right] + \gamma \cdot T + \frac{1}{2} \cdot \lambda \cdot \sum_{j=1}^{\mathrm{T}} \omega_j^2 \tag{7}$$

where g and h are first and second derivatives of loss function, T is the number of the leaves of tree, ω is the weight of tree leaf and γ is the minimize gain for node to split. All three are the construction of CART meta model *f*, λ is the L2 regularization from Ω.

Reindex to the objective function (7) from rounds' *t* to nodes' *j*, solve it by the greedy strategy to get best solution $\omega_j^{[\mathrm{best}]} = -\frac{\mathrm{G}_j}{\lambda + \mathrm{H}_j}$ where G and H is the sum of the Derivatives in the loss functions, and bring back to (7) after simplification as the follows (8):

$$best_{obj(t)} = -\frac{1}{2} \cdot \sum_{j=1}^{T} \left(\frac{G_j^2}{H_j + \lambda}\right) + \gamma \cdot T \tag{8}$$

The loss or the metrics of the training dataset will continue improving as the number of rounds increases if the ensemble model is trained as (8). However, it is such common with too many rounds to get overfitting. Early-stopping can control the over train of the boosting that would stop the boosting iterating when cross-validation or validation dataset cannot improve after the rounds we set and output the best numbers of iteration as the final ensemble.

For each meta CART tree learner in boosting, the gain in feature splitting is calculated as follows:

$$\mathrm{Gain} = \frac{1}{2}\left[\frac{\mathrm{G_L^2}}{\mathrm{H_L} + \lambda} + \frac{\mathrm{G_R^2}}{\mathrm{H_R} + \lambda} - \frac{(\mathrm{G_L} + \mathrm{G_R})^2}{\mathrm{H_L} + \mathrm{H_R} + \lambda}\right] - \gamma \tag{9}$$

Where L and R mean the splitting path and the calculation method is the same as above (7-8); Gain is the splitting indicators or references to grow a tree with Dependence Maximization of features in samples space. The ensemble model makes a mean calculation of all *t** estimators for macro-Gain as ***FIV*** to measure feature importance in the model[28]. And the FIVs is as follows (10):

$$\mathrm{FIV}(s) = \frac{\sum_{i=1}^{t^*}(\mathrm{Gain}_s)}{\sum_{i=1}^{t^*}\sum_{j=1}^{m}\left(\mathrm{Gain}_j\right)} \tag{10}$$

The value of ***FIV***(*s*) represents the information gain of the feature *s* to the all meta learners or model during the iteration of the extreme gradient boosting. When the performance of the model reaches certain acceptable, ***FIV***s can be used in both feature selection and data interpretation. The range of this value is a floating number in [0.00, 1.00]. The closer it is to the value of the right boundary, the more important the feature is for the ensemble model. This paper will use the FIV value modeled based on the extreme gradient boosting algorithm for feature selection, compress the high -dimension spectral data into less crux bands, and then build RBF-SVM model with the TPE method mentioned above.

4. Materials and Methods

4.1. Soybean material and hyperspectral dataset

The soybean samples were from the Soybean Research Institute of Northeast Agricultural University (Harbin, China). The basic nutritional information of four categories of soybeans is shown in Table 2. About 1000 seeds of each category were randomly selected primarily. After removing the samples with obvious abnormal appearance, the hyperspectral information of the samples was taken by Headwall VNIR-A system as Figure 1. in Application Research Laboratory of Spectral Technology of Northeast Agricultural University. In order to avoid jamming from the harmonic current generated by the large AC motor in the power supply system, and further to limit the light source to the halogen light source of the experimental platform as much as possible, the spectral data was collected from 23:00 to 2:00 at night.

Table 2. Basic soybeans information of nutrition and experiment dataset construction

Category	Crude Protein	Crude Fat	Shape	Seed Coat Luster	Seed Hilum	Train & Valid	Test Dataset	Label	Sum
Dongsheng-1	41.30%	19.97%	spherical	shiny	yellow	375	125	0	500
Changnong-33	37.57%	23.00%	ellipsoid	shiny	yellow	374	125	1	499
Changnong-38	37.26%	21.33%	ellipsoid	slight	yellow	450	150	2	600
Changnong-39	40.91%	20.15%	spherical	dull	brown	525	175	3	700

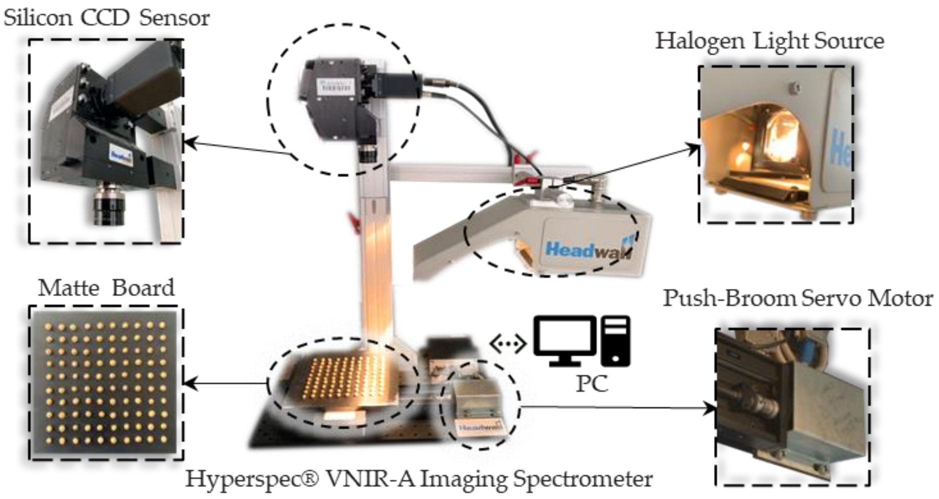


Figure 1. Headwall Photonics Hyperspec® VNIR-A system and customized matte board

The shape of soybeans is spherical or ellipsoid so a push-broom board matching the shape of the seeds was designed to maintain the stability before taking images. The customized board is a square with a side length of 200 mm and a thickness of 10 mm made by polypropylene. The surface of the board is subjected to diameter of 10mm and depth of 10mm drilling to ensure that the soybeans can be fixed in the holes. In order to avoid the scattering of the halogen light source by the polypropylene material, the surface was further sprayed with water-based acrylic paint (Botny-B1924-#4) for matte treatment, and then finish the dark-white correction in blank board.

The sample collection wavelength is configured from 400nm to 1000nm, the ultraviolet-visible-near-infrared spectral band, the width of bands is about 3nm, and a total of 203 bands are collected. During acquisition, the moving speed of the push-broom board is 5mm/s with 38.84ms exposure time. The imaging range is controlled within the extinction plate and the frame period is 0.04ms. And the final hyperspectral cubic is obtained as the size of (1004, 812, 203).

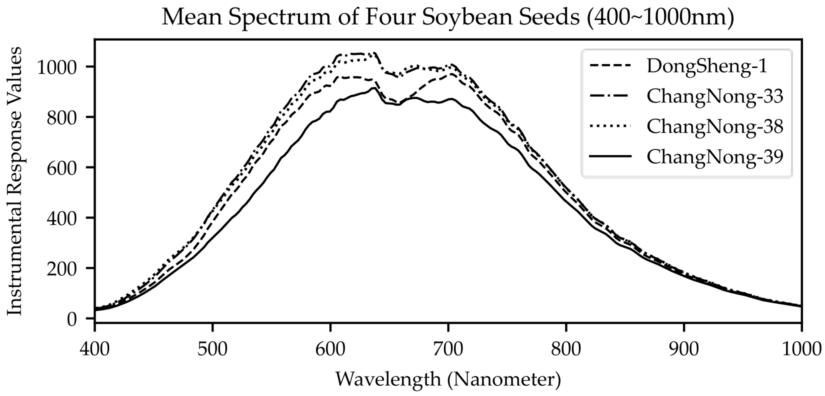


Figure 2. Mean spectral curves of four categories soybean

The soybean seed information was exported from the cube and transform to data with instrument response for further processing by ENVI Classical 5.3. The ROI area is defined by masked-method, and the average sample hyperspectral data is obtained by taking a single soybean seed as an independent sample. Figure 2. shows the average spectral curve of the four categories of soybean. The vertical axis is the instrument response value that is proportional to the reflectance of the spectral data. Because data preprocessing and format conversion are involved in the later stage, the instrument response is directly applicated as raw data. Finally, the dataset is splitted into training, validation and test dataset according to a certain ratio (2:1:1) as shown in Table 2.

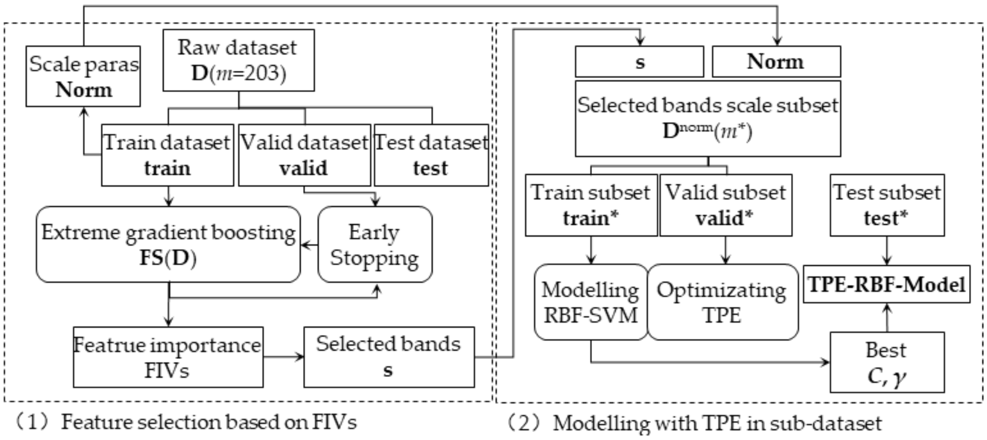


Figure 3. Flow chart of feature selection and optimized RBF-SVM modelling

4.2. Feature selection and optimized RBF-SVM model

The model design is as shown in Figure 3. The TPE-RBF-SVM model for the feature-band subset is designed into two steps:

4.2.1. Feature selection based on feature importance

Feature selection transforms the 203-band UV-Vis-NIR spectral dataset **D**(m=203) into a wavelength bands subset **s** with 10 crux spectral bands.

Firstly, the extreme boosting model is built and fitted by full-band dataset train of **D**(m). The fitting process would have a validation in **valid** dataset of **D**(m) to control possible overfitting and underfitting, then a full-band extreme boosting model $\mathbf{FS}_{\mathbf{D}(m=203)}$ was finished. After performance evaluation, the **FIV**s of acceptable model, $\mathbf{FS}_{\mathbf{D}(m=203)}$, will be extracted and mask for a sub-set by the sorted descending top 10 spectral band wavelengths as **s** with feature interpretation.

4.2.2. Modelling and optimizing RBF-SVM with TPE in sub-dataset

According to the characteristics of the SVM algorithm, the original data **D**(m=203) will be scaled as the follows:

$$x_i^{norm}(j) = \frac{x_i(j)\text{-}\overline{x(j)}}{x_{\text{std}}(j)},\ i=1,2,\ 3,\ \ldots,\ n;\ i=1,2,\ 3,\ \ldots,\ m \tag{11}$$

where $\bar{x}$ and x_{std} are the average and variance value of the dataset train correspondingly, and these two data will be used directly as constant values in normalized scaling for **valid** and **test** to avoid data leaks that pollute train dataset by accident information.

After the scaling, $\mathbf{D}^{norm}$(m*=10) is constructed according to sub-dataset in (11) above. The RBF-SVM is going to build and be fitted by **train*** of $\mathbf{D}^{norm}$(m*=10) with optimisation searching by TPE algorithms in **valid*** datasets for better performance metrics.

The RBF-SVM algorithm here is built by the libsvm with OVR (one vs rest samples) for multi-classification. That is transformed into 4 two-class sub-task, and each two-class task would recognize the single category with other all categories. In order to avoid overfitting from extreme or outlier samples in the modelling, the maximize number of searching support vectors is constrained to 5000 times.

The search space of the TPE-RBF-SVM includes the hyperparameter C (Penalty scaling strength of SVM) and γ (kernel coefficient of RBF) as follows Table 3. shown:

Table 3. Hyperparameters to tune and search space

Hyperparameter	Data type	Search space	Minimize step
C	float	1e-2,1e5	1e-8
γ	float	1e-8,1	1e-10

4.3 Algorithm of control group

The extreme gradient boosting model itself is a supervised learning algorithm, which will be further tested as a comparison algorithm (xgbc) in the subset **D**(m*=10); Also, Vanilla RBF-SVM (svc2) will be considered to compare the effect of the TPE; And in addition, four other machine learning algorithms as comparative models below:

- CART Tree(tree)

Decision tree is a non-parametric supervised learning model commonly applicated in machine learning. CART, one of decision tree, is also a meta learner in the most boosting model. In this research, the Gini-index is set as the information gain indices for the comparison model, whose maximized depth of the tree is not constrained, node splitting method is the greedy and all features are considered during the splitting nodes.

- Random Forest(rdrf)

Random forest is another popular ensemble model based on tree model in engineering application. By controlling the hyperparameters of the meta learners, random forest randomly splits the feature of the samples, and uses the random sampling for the subsample to realize the dataset expansion. To control model's variables, hyperparameters were configured the same type of boosting model, but the maximized depth of the tree is not limited here.

- Logistic Regression(lgst)

Logistic regression is a probabilistic classification method that maps a function of dataset features to a target to predict that a new example belongs to one of the target categories models learnt. It is a classic linear classifier in machine learning. In the model we built, the L2 regular term is added to the iterative objective, and the maximized number of iterations for convergence is limited to 100. The solution method is configured as LBFGS. Logistic regression model is sensitive to the size of the data by pre-experimented, so the standardized $\mathbf{D}^{norm}(m^{*}=10)$ is used for modeling.

- Multilayer Perceptron (mlp2)

Perceptron is a widely used model in deep neural network, which adopts back-propagation to realize iterated learning by given data features. Perceptron has excellent learning ability for nonlinear tasks or datasets. In this paper, a 2-layer perceptron compiled by Adam optimizer is used as a comparison of neural network. The activation function of the hidden layer is configured as RELU function, the number of hidden nodes is (100, 50), the output layer function is SoftMax function, and the learning rate is configured as 0.001. We limited the maximum number of iterations is 200.

4.4. Evaluation metrics and analysis environment

The identification task among soybean categories is a supervised machine learning multi-class task, therefore, the multi-class accuracy is used as the main evaluation metrics firstly for the model in this paper. The multi-class accuracy ACC is calculated as below:

$$\mathrm{ACC}=\frac{1}{n_{\mathrm{sample}}}\sum_{i=1}^{n_{\mathrm{sample}}}\boldsymbol{I}\left(\widehat{y}_i=y_i\right) \qquad (12)$$

where $\widehat{y}_i$ is the predicted value of corresponding true value y_i, the n_{sample} is the number of samples in dataset. Further, an ACC-based confusion matrix will be introduced for detailed category predict evaluation.

On the other hand, the research will additionally use F1 score as the second evaluation metric. The F1 score is the harmonic mean of the precision and recall from the dataset. The calculation method is as follows:

$$F_{\beta}=\left(1+\beta^{2}\right)\times\frac{precision\times recall}{\beta^{2}\times precision+recall} \qquad (13)$$

$$precision=\frac{TP}{TP+FP},\ recall=\frac{TP}{TP+FN} \qquad (14)$$

where β=1.00 means the harmonic mean calculation, the *TP* is the counts of correctly predicting as True Positive and the *FN*, False Negative is the counts of misrecognition.

All the codes designed in this research have been open sourced under the MIT license. The hardware environment that the analysis depends on is shown in the Table 4. below, and the compile environment is based on Python 3.9 in Windows 10 LTSC. To make sure reproducibility of the all experimental and analysis results, all random seeds involved in this paper are set as 615.

Table 4. Environment and tools of analysis and model building in paper

Compute Environment		Analysis Tools
CPU	Intel® Core™ i5-10400F （2.90GHz）	Scikit-learn v 0.24.1， Pandas 1.3.3，Numpy 1.19.3， Scipy 1.6.2，xgboost 1.4.2， liblinear 3.23.0.4
RAM	DDR4 3000Mhz 48GB = 2x8GB + 2x16GB	
Operating System	Windows LTSC 21H2	
Random Seed	615	

5. Results

5.1. Feature selection based on FIVs

The train dataset, **train**, in the range of 400nm to 1000nm fits extreme gradient boosting model cited above, which applicated **valid** in separate validation dataset for early-stopping. The boosting was set of the max number of estimators of 1000 rounds and finally the iterations stopped in 880th round. At this time the model with validation dataset would not have been being improved the loss metrics in 10 rounds so it is deemed to fitted fully. $\mathbf{FS}_{\mathrm{D}(m=203)}$ and corresponding ***FIV*** can be pick up from fitted boosting model.

Figure 4. shows the distribution of bands and ***FIV***, whose horizontal axis represents the band wavelength of the spectral data; the vertical axis is the ***FIV*** numerical distribution with a sum of 1.00 from the extreme gradient boosting model. According to the ***FIV*** value, the top 10 wavelengths are extracted as the selected crux wavelength band to **s**, whose distribution is shown covered in the band marked with the dotted line.

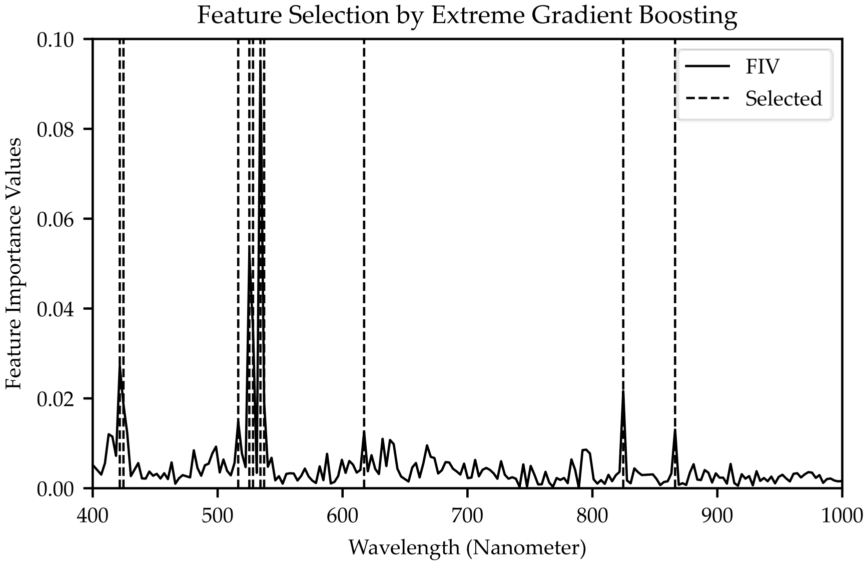


Figure 4. All bands' FIV values and the selected crux ten bands

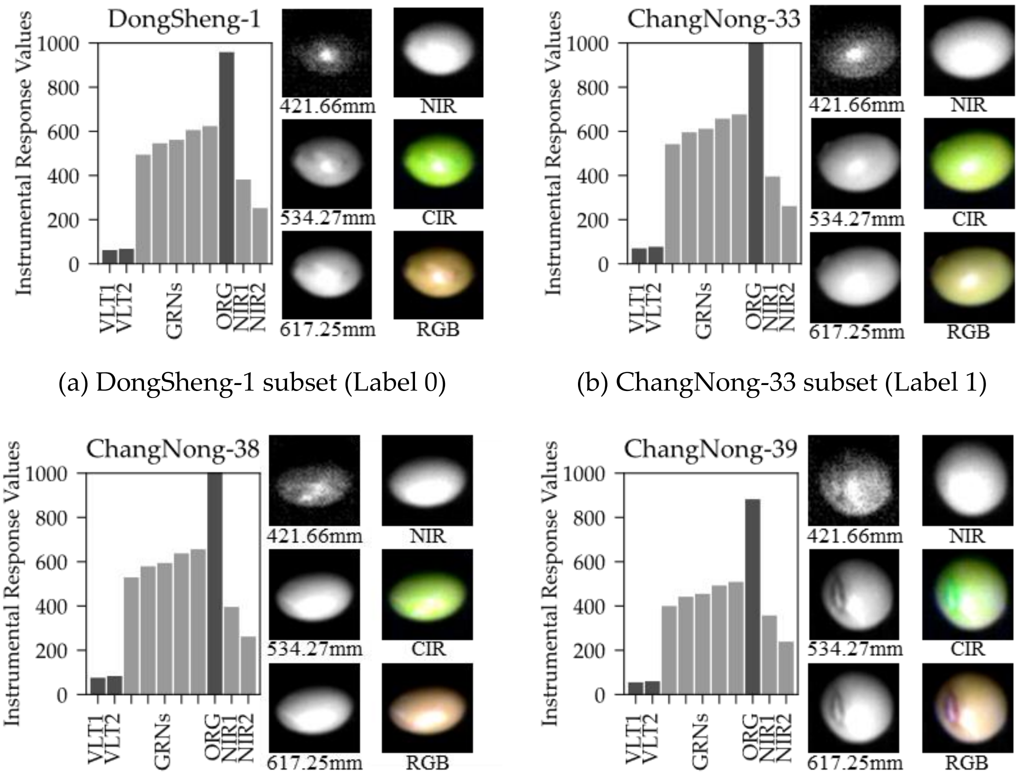
Table 5. shows the central wavelengths of 10 selected bands and their **FIV** values. 10 selected bands contribute 30.9631% of the all in the model. And calculate average weight of each band to transform them into the relative FIVs based on the sum of selected bands' values.

Table 5. Selected bands of center wavelength and relative values

Center	**Violet Bands**		**Green Bands**				**Orange**	**NIR Bands**		
wavelength	421.66	424.63	516.49	525.38	528.35	534.27	537.24	617.25	824.69	866.18
Full Bands' *FIV* values	2.73%	1.86%	1.48%	5.23%	3.52%	9.51%	1.92%	1.25%	2.16%	1.31%
Relative *FIV* values	8.80%	6.01%	4.77%	16.88%	11.38%	30.71%	6.19%	4.04%	6.98%	4.22%

Combining Table 5. and Figure 4., the selected crux bands include 2 violet bands (VLNs), 5 green bands (GRNs), 1 orange band (ORG) and 3 near-infrared bands (NIRs). Among them, the green bands (GRNs) accounts for 21.66% of the whole band (relative value is 69.93%), which is the dominant spectral band in the extreme gradient boosting model, and there are several obvious peaks in Figure 4.; The violet band (VLTs) and the near-infrared band (NIRs) accounted for 4.59% (14.81%) and 3.47% (11.20%), respectively, which can be considered as two spectral bands with the same secondary modeling importance.

The instrumental response subset bar plots of hyperspectral data from boosting model for 4 classes soybean are shown in Figure 5., where the right-side lists six visual picture in 421.66nm, 534.27nm, 617.25nm, NIR (824.69nm), color infrared (CIR, 534.27nm/824.69nm/421.66nm) and true color image (RGB). Through the image display of the first three bands, the hilum and other visual features of soybeans can be identified in the visible light band. The NIR will directly crosses the texture band, and further shows the refraction and projection of the light source by the intensity of the instrumental response value. CIR and RGB are indirectly or directly visible to human vision. It is difficult to distinguish the four classes of soybeans, but with the help of selected bands, it is more straight forward to complete the judgment of soybean types through data quantification.



(a) DongSheng-1 subset (Label 0)

(b) ChangNong-33 subset (Label 1)

(c) ChangNong-38 subset (Label 2)

(d) ChangNong-39 subset (Label 3)

Figure 5. Subset and visual features image for soybeans in this paper

5.2. Optimization of SVM by Tree-structed Parzen Estimator

In the scaled feature subset train*, keep the same 72 iterative search as grid search, and the optimized hyperparameter in the final saved iteration results are C = 1.7631×10^4 and γ=2.1056×10^{-4}, which appears during the 32nd iteration. At this time, the ACC in the validation dataset is 0.9072, and the ACC after testing in the independent dataset **test*** is 0.9165. The two values are similar so the generalization ability of TPE-RBF-SVM model with both structural risk and empirical risk is proofed.

Figure 6. is the distribution of the ACC metric in the validation process as the number of tuning iterations increases. Obviously, the ACC has a large variation range in the first 20 iterations of the search process. It can be explained as TPE is accumulating the researching statistics in the defined space as Chapter 2.2. cited, so the metric fluctuates wildly; In the subsequent 20 to 40 iterations, the fluctuation of the metric became slighter, but the overall effect was generally reduced, which showed that the ACC value diverged again. This is perhaps due to the over-search caused by further iteration after the enough statistical results. Then until the end of the 72nd iteration, the metric value is stable after several rounds of shocking, and the best result we want appears in 20~40 between the two mutations.

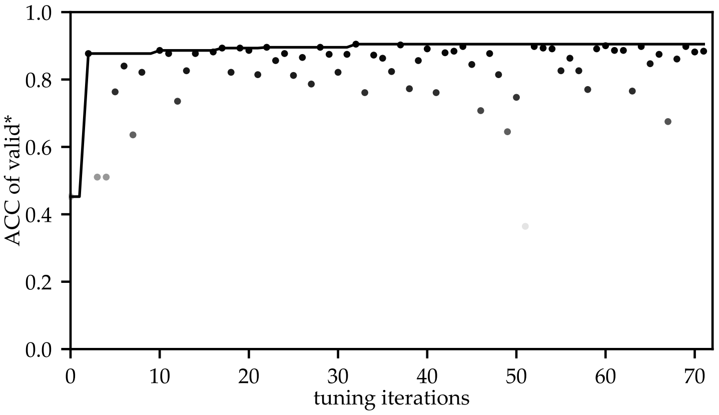
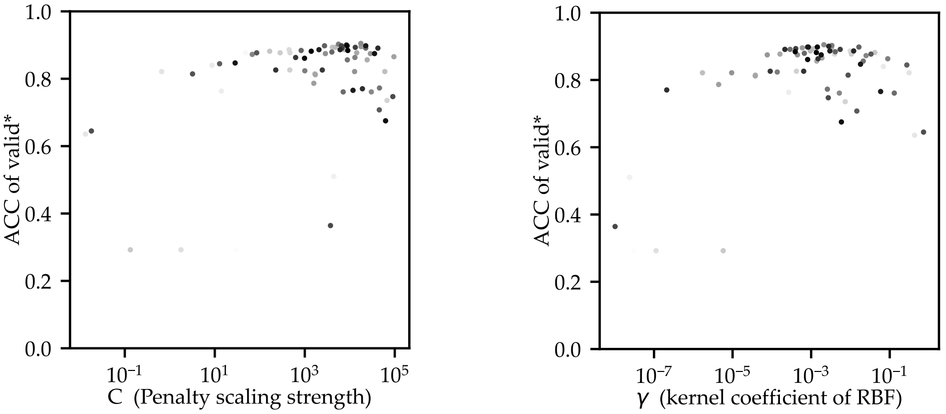


Figure 6. Accuracy and iterations of TPE optimization process

Figure 7. shows the optimising process tuned by C and γ, the color depth of point increases with the number of iterations. Both hyperparameters have a tendency to gather towards a certain center that best combination (C = 1.7631×10^{4}, γ=2.1056×10^{-4}) is closed to in search space. According to the algorithms theory of SVM, when the training dataset is complex multi-dimensional data, the relationship is complex between parameters of SVM and hyperparameters during training process. So, results of tuning by the metric will be one of multiple local optimal solutions in the search space. This is confirmed by the several iterations of convergence-mutation process of the metric from validation by TPE as the number of iterations increases.



(a)Change for accuracy with C and iterations (b) Change for accuracy with γ and iterations

Figure 7. Hyperparameters changing during iterations

The hyperparameters of the RBF-SVM model have been finally determined by the TPE algorithm in the sub-band spectral data set in 72 iterations. The model hyperparameter C = 1.7631×10^{4}, and the hyperparameter γ is determined to be 2.1056×10^{-4} for our datasets. The accuracy of the TPE-RBF-SVM model under this configuration is 0.9072 in the validation dataset and 0.9165 in test dataset.

5.3. Comparison with vanilla model and other algorithms

The sub-band dataset that obtained in Chapter 5.1. is used as training data fits the SVM with the tuned optimized hyperparameters obtained in Chapter 5.2 to build the TPE-RBF-SVM model (bst, short of "best"). And the same training data is used to fit other 6 models cited in Chapter 4.3. Then test the model with individual test dataset for metrics, ACC and F1 scores are shown in Figure 8.

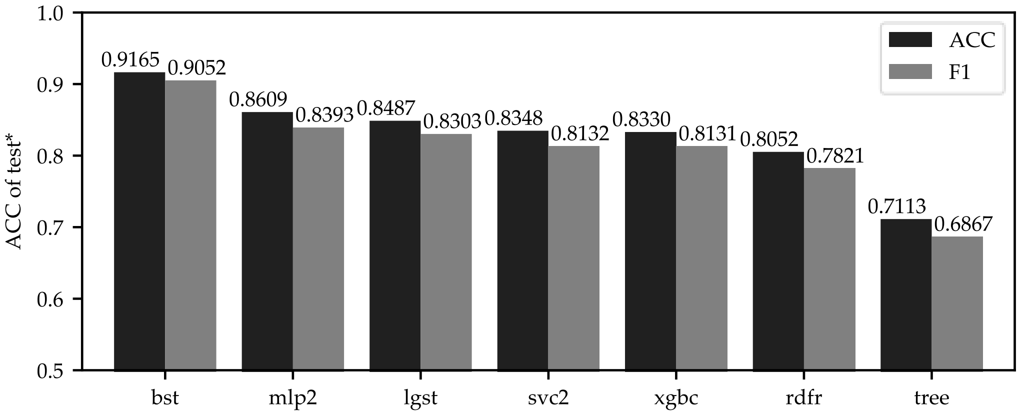


Figure 8. Accuracy and F1 score of all models in research

TPE-RBF-SVM model we recommended scores the highest in both metrics with four-classification accuracy of 0.9165 and F1 scores of 0.9052. Compared with unoptimized vanilla RBF-SVM, its accuracy improves by 0.0817 from 0.8348(9.786% increasing in percentage). Compared with the second-ranked multi-layer perceptron model(mlp2), the accuracy is increased by 0.0556 (6.458% increasing), and the F1 score is increased by 0.0659 (7.852% increasing). In the feature subset data, it has a very obvious improvement effect. It can be preliminarily considered that method we recommended has a very obvious improvement effect in selected sub-datasets for soybean pattern recognition.

Compared with the vanilla extreme gradient boosting model, the accuracy of the TPE-RBF-SVM model has increased by 0.0835 from 0.8348 (10.02%). The results show that when the sub-band dataset is obtained by boosting model and model built as core classifier in this study, the accuracy is better than another ensemble model, random forest(accuracy=0.8157) and meta learner, decision tree(accuracy=0.7113). However, boosting model does not significantly outperform the experimental results of other types of algorithms, and is comparable to the unoptimized vanilla SVM model.

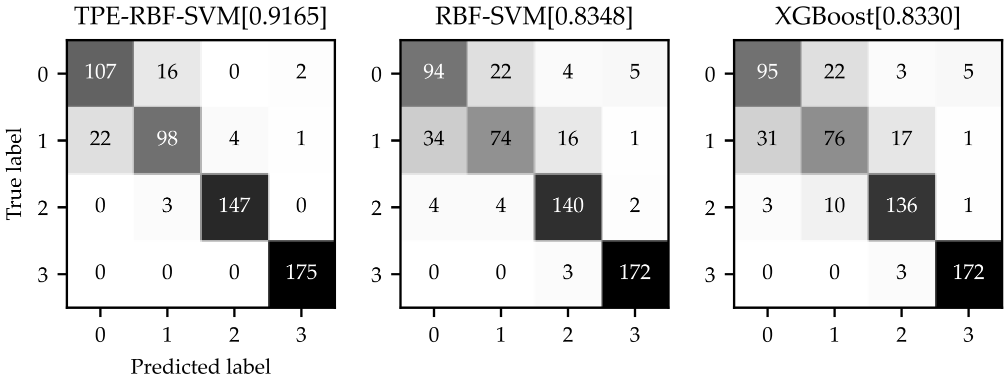


Figure 9. The confusion matrices of classification results of TPE-RBF-SVM and vanilla models

The confusion matrices of the TPE-RBF-SVM we recommended, the vanilla RBF-SVM and the extreme gradient boosting model are shown in Figure 9. The vertical axis of the confusion matrix is the real category and the horizontal axis is the predicted category from the model. The confusion matrices show that both the SVM models and the boosting model have excellent results for category 2(Changnong-38) and category 3(Changnong-39), and the TPE-RBF-SVM can almost recognize them totally. But on the other hand, for category 0 (Dongsheng-1) and category 1 (Changnong-33), which are more difficult to classify correctly, almost contribute all the negative effects of three models, while the Tree-structed Parzen Estimator algorithm significantly suppressed this effect with excellent results among models.

FIVs based on extreme gradient boosting method can select crux bands from bands ranged from 400nm to 1000nm for dimension-reduced sub-band dataset. And in the sub-band dataset by FIVs from extreme gradient boosting, compared with the vanilla RBF-SVM and XGBoost models, the SVM model optimized by TPE algorithms can effectively improve in the test dataset performance of soybean multi-classification tasks; Compared with other machine learning algorithms, the method still has high accuracy as well.

6. Discussion

(1) The extreme gradient boosting itself is a popular machine learning algorithm widely with wide applications, whose feature interpretability based on tree model is an unparalleled advantage. However, not all extreme gradient boosting models, or tree-based models are suitable for feature selection. And as a machine learning model, its final performance is not necessarily superior to traditional non-tree algorithms. We believe that not all types of data or engineering needs are suitable for extreme gradient boosting method by the FIVs for feature selection without conditions. In the preliminary stage of data exploration by the boosting model in the paper, only the model itself has an acceptable performance and a certain generalization ability, its own ***FIV*** value is meaningful for feature selection. That is why crux bands selection in this paper needs an early-stopping trick of **Fs** model for better performance. When the model for ***FIV***s is over-fitted or under-fitted, the subsequent feature selection method would further expand the error because the selection model has been based on inappropriate feature weights. After the corrected selection model in first stage, all we need is a useful learner or machine learning model to classify or regress for the task. The SVM algorithm is a high-performance machine learning algorithm, but when the data dimension is too high, a lot of computing resources are needed to iteratively find suitable support vectors and more negative effects from redundant information would jam the performance in individual test dataset. Hyperspectral data itself has rich multi-dimensional features, but at the same time there is a large amount of data collinearity and information redundancy. Therefore, no matter whether direct dimensionality reduction or feature selection selected in this paper is used, it is a necessary process to apply the SVM algorithm to compress the high-dimensional raw data through a certain method. Therefore, for dataset with less features, the SVM algorithms perform better than ensemble methods. So, the models built by two algorithms in this paper built and had a better performance than each single one.

(2) In the research process, the final choice is based on feature selection for dimensionality reduction, and direct dimensionality reduction methods such as principal component analysis are not considered for research and comparison, because it is hoped for a method which can directly determine the spectral band and there is no need to collect useless bands from device-side source. That is when we applicated the TPE-RBF-SVM method in a practical engineering issue, only several bands we concerned like Figure 5. or Table 5. should be collected as a cheaper way in agricultural field. For the same reason, this paper does not use conventional spectroscopy pre-processing such as SNV (standard normal variate), MSC (multiplicative scatter correction) or Savitzky-Golay filter smoothing. All cited before need batch information for the whole dataset and for single sample we cannot do it because full-band data still needs to be acquired. And there is a risk of data leakage when conducting independent test datasets in research before application stage.

7. Conclusions

A fast and effective machine learning method was designed based on less bands hyperspectral data of soybean for pattern recognition of categories was designed as a non-destructive testing method. ***FIV***s based on extreme gradient boosting method can select

10 crux bands from 203 bands ranged from 400nm to 1000nm for dimension-reduced sub-band dataset. The hyperparameters of the RBF-SVM model have been finally determined by the TPE algorithm in the sub-band spectral dataset in 72 iterations. For best model hyperparameter C is 1.7631×10^4, and the hyperparameter γ is determined to be 2.1056×10^{-4}. And in the sub-band dataset by ***FIV***s from extreme gradient boosting, the metrics of TPE-RBF-SVM are significantly improved compared with other machine learning algorithms. The accuracy is 0.9165 in the independent test dataset which is 9.786% higher for vanilla RBF-SVM model and 10.02% higher than the extreme gradient boosting model.

Author Contributions: Conceptualization, Qinghe Zhao and Junlong Fang; Methodology, Qinghe Zhao; Software, Qinghe Zhao and Yuchen Huang; Validation, Qinghe Zhao and Zifang Zhang; Writing – original draft, Qinghe Zhao; Writing – review & editing, Zifang Zhang, Yuchen Huang and Junlong Fang. All authors will be informed about each step of manuscript processing including submission, revision, revision reminder, etc. via emails from our system or assigned Assistant Editor.

Funding: This research was funded by the National Key Research and Development Program of China, grant number 2016YFD0300610.

Institutional Review Board Statement: Not applicable. For studies not involving humans or animals.

Data Availability Statement: The dataset and fitted models can be downloaded at the GitHub at: https://github.com/gniqeh/TPE-RBF-SVM-SOYBEANS.

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

References

1. Fehily, A.M. SOY (SOYA) BEANS | Dietary Importance. In *Encyclopedia of Food Sciences and Nutrition*; Elsevier, 2003; pp. 5392–5398 ISBN 978-0-12-227055-0.
2. Lee, T.; Tran, A.; Hansen, J.; Ash, M. Major Factors Affecting Global Soybean and Products Trade Projections. *Amber Waves: The Economics of Food, Farming, Natural Resources, and Rural America* **2016**, 1, doi:10.22004/ag.econ.244273.
3. Zhao, G.; Quan, L.; Li, H.; Feng, H.; Li, S.; Zhang, S.; Liu, R. Real-Time Recognition System of Soybean Seed Full-Surface Defects Based on Deep Learning. *Computers and Electronics in Agriculture* **2021**, *187*, 106230, doi:10.1016/j.compag.2021.106230.
4. Abdel-Rahman, E.M.; Mutanga, O.; Odindi, J.; Adam, E.; Odindo, A.; Ismail, R. A Comparison of Partial Least Squares (PLS) and Sparse PLS Regressions for Predicting Yield of Swiss Chard Grown under Different Irrigation Water Sources Using Hyperspectral Data. *Computers and Electronics in Agriculture* **2014**, *106*, 11–19, doi:10.1016/j.compag.2014.05.001.
5. Folch-Fortuny, A.; Prats-Montalbán, J.M.; Cubero, S.; Blasco, J.; Ferrer, A. VIS/NIR Hyperspectral Imaging and N-Way PLS-DA Models for Detection of Decay Lesions in Citrus Fruits. *Chemometrics and Intelligent Laboratory Systems* **2016**, *156*, 241–248, doi:10.1016/j.chemolab.2016.05.005.
6. Rapaport, T.; Hochberg, U.; Shoshany, M.; Karnieli, A.; Rachmilevitch, S. Combining Leaf Physiology, Hyperspectral Imaging and Partial Least Squares-Regression (PLS-R) for Grapevine Water Status Assessment. *ISPRS Journal of Photogrammetry and Remote Sensing* **2015**, *109*, 88–97, doi:10.1016/j.isprsjprs.2015.09.003.
7. Osco, L.P.; Ramos, A.P.M.; Faita Pinheiro, M.M.; Moriya, É.A.S.; Imai, N.N.; Estrabis, N.; Ianczyk, F.; Araújo, F.F. de; Liesenberg, V.; Jorge, L.A. de C.; et al. A Machine Learning Framework to Predict Nutrient Content in Valencia-Orange Leaf Hyperspectral Measurements. *Remote Sensing* **2020**, *12*, 906, doi:10.3390/rs12060906.
8. Erkinbaev, C.; Derksen, K.; Paliwal, J. Single Kernel Wheat Hardness Estimation Using near Infrared Hyperspectral Imaging. *Infrared Physics & Technology* **2019**, *98*, 250–255, doi:10.1016/j.infrared.2019.03.033.
9. Zhang, X.; Sun, J.; Li, P.; Zeng, F.; Wang, H. Hyperspectral Detection of Salted Sea Cucumber Adulteration Using Different Spectral Preprocessing Techniques and SVM Method. *LWT* **2021**, *152*, 112295, doi:10.1016/j.lwt.2021.112295.
10. Jahed Armaghani, D.; Asteris, P.G.; Askarian, B.; Hasanipanah, M.; Tarinejad, R.; Huynh, V.V. Examining Hybrid and Single SVM Models with Different Kernels to Predict Rock Brittleness. *Sustainability* **2020**, *12*, 2229, doi:10.3390/su12062229.

doi:10.20944/preprints202207.0280.v1

11. Ahmad, A.S.; Hassan, M.Y.; Abdullah, M.P.; Rahman, H.A.; Hussin, F.; Abdullah, H.; Saidur, R. A Review on Applications of ANN and SVM for Building Electrical Energy Consumption Forecasting. *Renewable and Sustainable Energy Reviews* **2014**, *33*, 102–109, doi:10.1016/j.rser.2014.01.069.
12. Zeng, N.; Qiu, H.; Wang, Z.; Liu, W.; Zhang, H.; Li, Y. A New Switching-Delayed-PSO-Based Optimized SVM Algorithm for Diagnosis of Alzheimer's Disease. *Neurocomputing* **2018**, *320*, 195–202, doi:10.1016/j.neucom.2018.09.001.
13. Li, Y.; Yang, K.; Gao, W.; Han, Q.; Zhang, J. A Spectral Characteristic Analysis Method for Distinguishing Heavy Metal Pollution in Crops: VMD-PCA-SVM. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy* **2021**, *255*, 119649, doi:10.1016/j.saa.2021.119649.
14. Pal, M.; Foody, G.M. Feature Selection for Classification of Hyperspectral Data by SVM. *IEEE Trans. Geosci. Remote Sensing* **2010**, *48*, 2297–2307, doi:10.1109/TGRS.2009.2039484.
15. Kour, V.P.; Arora, S. Particle Swarm Optimization Based Support Vector Machine (P-SVM) for the Segmentation and Classification of Plants. *IEEE Access* **2019**, *7*, 29374–29385, doi:10.1109/ACCESS.2019.2901900.
16. Nader, A.; Azar, D. Searching for Activation Functions Using a Self-Adaptive Evolutionary Algorithm. In Proceedings of the Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion; ACM: Cancún Mexico, July 8 2020; pp. 145–146.
17. Tharwat, A.; Hassanien, A.E. Quantum-Behaved Particle Swarm Optimization for Parameter Optimization of Support Vector Machine. *J Classif* **2019**, *36*, 576–598, doi:10.1007/s00357-018-9299-1.
18. Young, S.R.; Rose, D.C.; Karnowski, T.P.; Lim, S.-H.; Patton, R.M. Optimizing Deep Learning Hyper-Parameters through an Evolutionary Algorithm. In Proceedings of the Proceedings of the Workshop on Machine Learning in High-Performance Computing Environments; ACM: Austin Texas, November 15 2015; pp. 1–5.
19. Bergstra, J.; Bengio, Y. Random Search for Hyper-Parameter Optimization. *J. Mach. Learn. Res.* **2012**, *13*, 281–305.
20. Ozaki, Y.; Tanigaki, Y.; Watanabe, S.; Onishi, M. Multiobjective Tree-Structured Parzen Estimator for Computationally Expensive Optimization Problems. In Proceedings of the Proceedings of the 2020 Genetic and Evolutionary Computation Conference; ACM: Cancún Mexico, June 25 2020; pp. 533–541.
21. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach Learn* **1995**, *20*, 273–297, doi:10.1007/BF00994018.
22. Boser, B.E.; Guyon, I.M.; Vapnik, V.N. A Training Algorithm for Optimal Margin Classifiers. In Proceedings of the Proceedings of the fifth annual workshop on Computational learning theory - COLT '92; ACM Press: Pittsburgh, Pennsylvania, United States, 1992; pp. 144–152.
23. Chang, C.-C.; Lin, C.-J. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–27, doi:10.1145/1961189.1961199.
24. Herrero-Lopez, S. Multiclass Support Vector Machine. In *GPU Computing Gems Emerald Edition*; Elsevier, 2011; pp. 293–311 ISBN 978-0-12-384988-5.
25. Abdiansah, A.; Wardoyo, R. Time Complexity Analysis of Support Vector Machines (SVM) in LibSVM. *IJCA* **2015**, *128*, 28–34, doi:10.5120/ijca2015906480.
26. Friedman, J.H. Stochastic Gradient Boosting. *Computational Statistics & Data Analysis* **2002**, *38*, 367–378, doi:10.1016/S0167-9473(01)00065-2.
27. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; ACM: San Francisco California USA, August 13 2016; pp. 785–794.
28. Adler, A.I.; Painsky, A. Feature Importance in Gradient Boosting Trees with Cross-Validation Feature Selection. *Entropy* **2022**, *24*, 687, doi:10.3390/e24050687.