MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
V.N. Karazin Kharkiv National University
School of Mathematics and Computer Science
Department of Theoretical and Applied Informatics

Master’s Thesis

Models and Algorithms in Analysis and Prediction of Underground Fires

Author:
Final year Master’s Program student specialty - Computer Sciences and Information Technologies, educational program: "Informatics"
Cui Zhiwen
Supervisor: PhD of Phys.-Math. Sciences Associate Professor of the department of Applied Mathematics
Stiepanova Kateryna
Reviewer: Chukhrai Andrii

Kharkiv, 2024

1. **INTRODUCTION**
2. **ADVANTAGES OF BLOCKCHAIN-BASED DECENTRALIZED AUTHENTICATION**
3. **MAJOR CHALLENGES IN DECENTRALIZED AUTHENTICATION**
4. **SOLUTIONS AND DEVELOPMENT DIRECTIONS**
5. **CONCLUSION**
6. **APPENDIX**

contents

1. INTRODUCTION..two
1.1 Background of Underground Fires ....two
1.2 Importance of the Study..two
1.3 Objectives of the Thesis..two
1.4 Thesis Structure..three
2. MAIN CONCEPTS..three
2.1 Characteristics of Underground Fires ....three
2.2 Mechanisms of Underground Fire Spread.. Four
2.3 The Importance of Modeling and Algorithms ....Five
3. CONCLUSIONS .... six
3.1 Methodological Framework ....seven
3.2 Selection of Models and Algorithms..eight
3.3 Evaluation Criteria..ten
3.4 Timeline and Milestones.. twelve
4. REFERENCES..twelve
4.1 Review of Current Models.. twelve
4.2 Review of Current Algorithms ....thirteen
4.3 Comparative Analysis..fourteen
4.4 Strengths and Weaknesses..sixteen
5. APPENDIX..seventeen
5.1 Application of Models and Algorithms.. seventeen
5.2 Real-World Scenarios.. seventeen
5.3 Practicality Assessment.. seventeen
6. Discussion..eighteen
6.1 Synthesis of Research Findings..eighteen
6.2 Impact on Underground Fire Management..eighteen
6.3 Directions for Future Research..twenty-one
7. Conclusion .... twenty-one
7.1 Summary of Key Findings ....twenty-one
7.2 Research Contributions..twenty-three
References.. Error! Bookmark not defined.**ive**
Appendix.. Twenty-six

1. INTRODUCTION

1.1 Background of Underground Fires

With the acceleration of urbanization and the increased use of underground space, fire prevention issues in underground projects such as subways, tunnels, and mines have become particularly important[1]. Underground fires not only threaten human life safety but can also lead to significant economic losses and environmental damage. This study utilizes the 2007 Monte Sinio Park forest fire dataset in Portugal(https://archive.ics.uci.edu/dataset/162/forest+fires) to explore underground fire analysis and prediction models and algorithms, in order to enhance fire prevention capabilities, optimize emergency response strategies, reduce fire risks, and promote the development of related technologies.

1.2 Importance of the Study

As a hidden and highly destructive natural disaster, underground fires pose a significant threat to the community (impacting its stability and normal operation),economic activities, and the ecological environment. This research conducts a profound exploration of the underground fire analysis and prediction models and algorithms through the examination of the Monte Sinio Park forest fire dataset in Portugal, which holds significant importance for social safety, property protection, environmental conservation, resource conservation, technological innovation, policy formulation, and emergency response.

1.3 Objectives of the Thesis

The main objective is to perform a detailed examination of the causes, characteristics, and propagation mechanisms of underground fires, assess the effectiveness of existing fire models and algorithms in predicting underground fires, and propose improvement measures. Specific objectives include:

Reviewing the basic concepts, characteristics, and classification of underground fires;
Analyzing the causes and influencing factors of underground fires;
Assessing the effectiveness of existing fire models and algorithms in predicting underground fires;
Proposing new methods or technologies to improve existing models and

algorithms;
Validating the effectiveness of the proposed methods through case studies.

1.4 Thesis Structure

This thesis is divided into nine chapters, with the content of each chapter arranged as presented hereinafter:
Chapter 1 is the introduction, which presents the research background, significance, objectives,thesis structure;
Chapter 2 discusses the main concepts, including the definition, characteristics, and theoretical framework of underground fires;
Chapter 3 outlines the work plan, tasks, and milestones, detailing the research methods, task allocation, and project schedule;
Chapter 4 analyzes existing models and algorithms, including reviews, comparative analyses, and the strengths and weaknesses of models and algorithms;
Chapter 5 is dedicated to case studies, applying models and algorithms to real-world scenarios and evaluating their practicality;
Chapter 6 is for discussion, synthesizing research findings, exploring the impact on underground fire management, and proposing future research directions;
Chapter 7 summarizes the research, summarizing the main findings and contributions;
Chapter 8 lists the references, providing all the literature cited in the thesis;
Chapter 9 is the appendix, exploring the application cases of models and algorithms in underground fire prediction and providing a comprehensive evaluation.

2. MAIN CONCEPTS

This section aims to provide readers with the core concepts and theoretical foundations necessary for the analysis and prediction of underground fires. We will delve into the characteristics of underground fires, their propagation mechanisms, and the importance of modeling and algorithms in this field.

2.1 Characteristics of Underground Fires

Concealment: The enclosed nature of underground spaces makes it difficult to detect fire sources in a timely manner, and initial fires are hard to control effectively, which can lead to severe consequences.

Difficult dispersion of heat and smoke: The design of underground spaces is limited by natural factors, making it challenging to incorporate multiple vents. During a fire, smoke accumulates, and the dispersion of heat and smoke is suppressed, which may trigger flashovers or rapid spread.
Evacuation difficulties: Underground spaces are extensive, and functional areas are not strictly divided, making it hard to find the correct direction during evacuation. Power outages during fires can reduce visibility and delay evacuation opportunities.
Challenges in firefighting: Underground spaces are dimly lit, and it is not easy for personnel to enter and exit. The direction of smoke flow is the same as the direction of people evacuating, which is opposite to the direction of firefighters entering the fire scene, making it more difficult for firefighters to access the fire area.
High temperature and comprehensive combustion: During underground building fires, hot smoke is difficult to vent, leading to a rapid increase in internal temperatures and an increased likelihood of flashovers.
High smoke production: In underground building fires, due to insufficient oxygen supply, a large amount of smoke is produced, and it is hard for the smoke to disperse into the atmosphere through windows, instead, it accumulates within the building.
Numerous pipelines and wiring: Underground spaces are not only densely populated with people but also serve as gathering spaces for pipelines and wiring. Various pipeline facilities could become new sources of fire.
Significant smoke from fires: Underground fires generate a large amount of smoke. Due to the enclosed nature of the underground environment and poor smoke exhaust, coupled with a relatively small number of entrances and exits, the rate of smoke expulsion is low.
These characteristics indicate that the prevention and control of underground fires require special attention and strategies to reduce the losses and impacts caused by fires.

2.2 Mechanisms of Underground Fire Spread

The propagation mechanisms of underground fires are complex and variable, influenced by a multitude of factors. Below are the primary mechanisms for the spread of underground fires:
Thermal Conduction: Underground fires spread through thermal conduction in solid media, which is the main mode of propagation in the early stages of an

underground fire. Heat is transferred directly from the fire source to surrounding materials through contact, causing temperature increases that can lead to the ignition of new fire points.

Thermal Convection: As the fire develops, hot air rises and cold air sinks, creating convection currents. These currents can accelerate the spread of the fire, especially in underground spaces with vents or cracks.

Thermal Radiation: Radiant heat from a fire can travel long distances, heating combustible materials at a distance and bringing them to their ignition point. In underground environments, thermal radiation may travel through rock fissures or unsealed spaces.

Gas and Smoke Movement: Gases and smoke produced by a fire will seek outlets, which can cause the fire to spread rapidly along ventilation systems or natural fissures.

Chemical Reactions: Underground fires may involve complex chemical reactions, such as spontaneous combustion of coal seams, which can occur without an apparent source of ignition and are difficult to control.

Topography and Geological Conditions: The topography and geological conditions of underground spaces, such as the porosity and permeability of rocks, also affect the spread of fires. For instance, porous rocks may allow fires to spread quickly through fissures and pores.

Human Factors: Improper operations in underground engineering activities, such as electrical equipment malfunctions or the improper use of flammable materials, can also cause fires and contribute to their spread.

Understanding these mechanisms is crucial for developing effective fire prevention and control strategies. By analyzing the propagation mechanisms of underground fires, it is possible to better predict fire behavior and thus take appropriate measures to mitigate the impact of fires.

2.3 The Importance of Modeling and Algorithms

In the field of underground fire analysis and prediction, the importance of modeling and algorithms cannot be overstated. They are key tools for understanding and predicting fire behavior, with the following significance:

Improving Predictive Accuracy: By establishing mathematical models and applying algorithms, the occurrence, development, and potential impact range of underground fires can be predicted more accurately. This is crucial for taking preventive measures in advance.

Aiding Decision-Making: Models and algorithms provide a scientific basis for

decision-makers to quickly formulate effective response strategies during fires, such as planning evacuation routes, allocating resources, and executing firefighting actions.
Optimizing Resource Allocation: Proper resource allocation is crucial in fire prevention and emergency response. Models and algorithms are capable of facilitating the recognition of regions demanding enhanced surveillance and safeguarding, thereby optimizing the use of resources.
Enhancing Fire Understanding: Through models and algorithms, investigators can acquire a more profound comprehension of the physical and chemical processes of underground fires, including fire spread, smoke movement, and heat distribution.
Promoting Technological Innovation: In the study of underground fire prediction, new models and algorithms are continuously being developed, driving the development of fire science and technology.
Supporting Policy and Regulation Development: Research findings based on models and algorithms can provide a scientific basis for governments and relevant departments to formulate more effective fire prevention policies and regulations.
Raising Public Awareness: The predictive results of models and algorithms can increase public awareness of the risks of underground fires, promoting broader fire prevention and safety education.
Addressing Complexity: The complexity of underground fires requires the use of advanced models and algorithms to deal with multivariable, nonlinear, and dynamically changing issues, which is a challenge for traditional methods.
In conclusion, modeling and algorithms occupy a crucial position in the analysis and prediction of underground fires. They not only improve our understanding of fire phenomena but also provide strong technical support for fire management.
Conclusion This section outlines the implementation plan of the research, including constructing a methodological framework, selecting models and algorithms, defining evaluation criteria, and project timelines and key milestones.

3. CONCLUSIONS

This section outlines the implementation plan for the research, encompassing the erection of the methodological framework, the choice of models and algorithms, the definition of evaluation criteria, and the project timeline and key milestones.

3.1 Methodological Framework

The methodological framework of this study is designed to systematically analyze and predict underground fires employing a diverse range of machine learning models and algorithms. The process is structured as follows:

Data Preprocessing: The initial step involves the creation of new features that may be relevant for predicting underground fires. a new feature temp_RH is derived from the interaction between temperature (temp) and relative humidity (RH), calculated as temp_RH = data['temp'] * data['RH']. Additionally, a logarithmic transformation is applied to the DMC feature to prevent negative values, i.e., DMC_log = np.log(data['DMC'] + 1).

Feature Selection: A key part of the methodology is feature selection implemented through LassoCV. LassoCV performs automatic feature selection by penalizing the regression coefficients of less important features to zero by means of the introduction of L1 regularization. In this study, LassoCV identified 'temp', 'DMC_log', 'X', and 'temp_RH' as the features most influential for model prediction.

Model Training and Hyperparameter Tuning: The study employs a grid search approach to fine-tune the hyperparameters of the Decision Tree and Random Forest models. The hyperparameters for the Decision Tree include 'max_depth' and 'min_samples_split', while those for the Random Forest include 'n_estimators', 'max_depth', and 'min_samples_split'. A systematic search is conducted using GridSearchCV to find the optimal parameter settings.

Model Evaluation: Performance metrics such as Mean Squared Error (MSE) and Coefficient of Determination (R^2) are used to evaluate the models. For example, the Random Forest model had an MSE of 11521.33 and an R^2 of 0.02; the Linear Regression model had an MSE of 11748.94 and an R^2 of 0.00. Additionally, cross-validation is used to assess the robustness and generalization capability of the models, such as the cross-validated MSE of 4646.34 for LassoCV.

Visualization: The study includes visualizations to gain a more comprehensive understanding of the conduct of the models. The Lasso path plot shows how the coefficients of features change with the regularization parameter Alpha, indicating the importance of different features. A bar chart of the features selected by Lasso highlights the most influential predictors, such as 'DMC_log' and 'X'. The cross-validation error plot for the Lasso model shows the MSE for different values of Alpha, identifying the optimal Alpha value as 0.7595.

Case Studies: Models and algorithms are applied to real-world scenarios and their practicality is assessed. Through these example investigations, the effectiveness of the models in actual underground fire prediction is demonstrated.

This methodological framework presents a thorough framework to the analysis and prediction of underground fires, integrating data preprocessing, characteristic picking, model optimization, evaluation, and visualization to enhance the understanding and management of such fires.

3.2 Selection of Models and Algorithms

In this study, the selection of models and algorithms is based on their capabilities and applicability in solving underground fire prediction problems.

Random Forest: As a combined learning approach, random forests improve the accuracy and stability of the model through erecting numerous decision trees and integrating their prognostications. They are good at handling non-linear relationships and feature interactions, and are robust to high-dimensional data.

Linear Regression: Although linear regression may not be sufficient to accurately handle non-linear relationships, its straightforwardness and explicability make it a baseline model for comparison with more complex models.

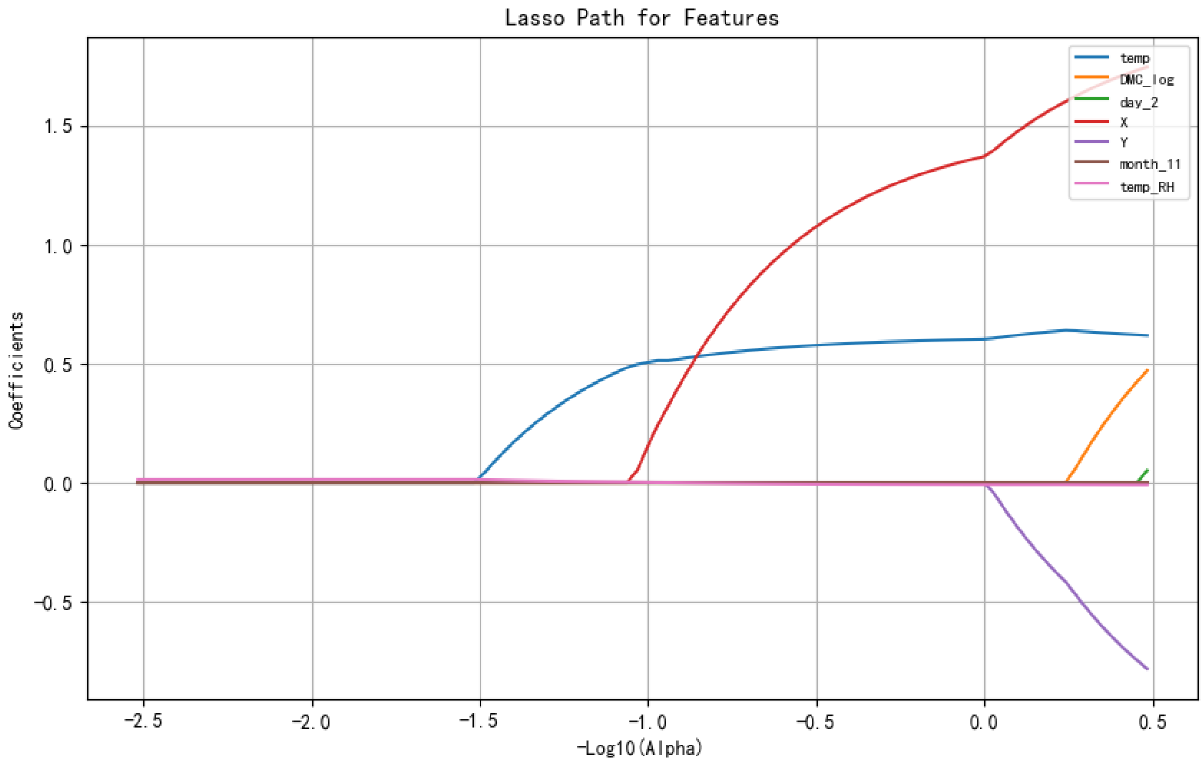
Support Vector Regression (SVR): SVR is the regression version of support vector machines, designed to find the optimal hyperplane that minimizes errors while maintaining the model's generalization ability. SVR is particularly suitable for high-dimensional data.

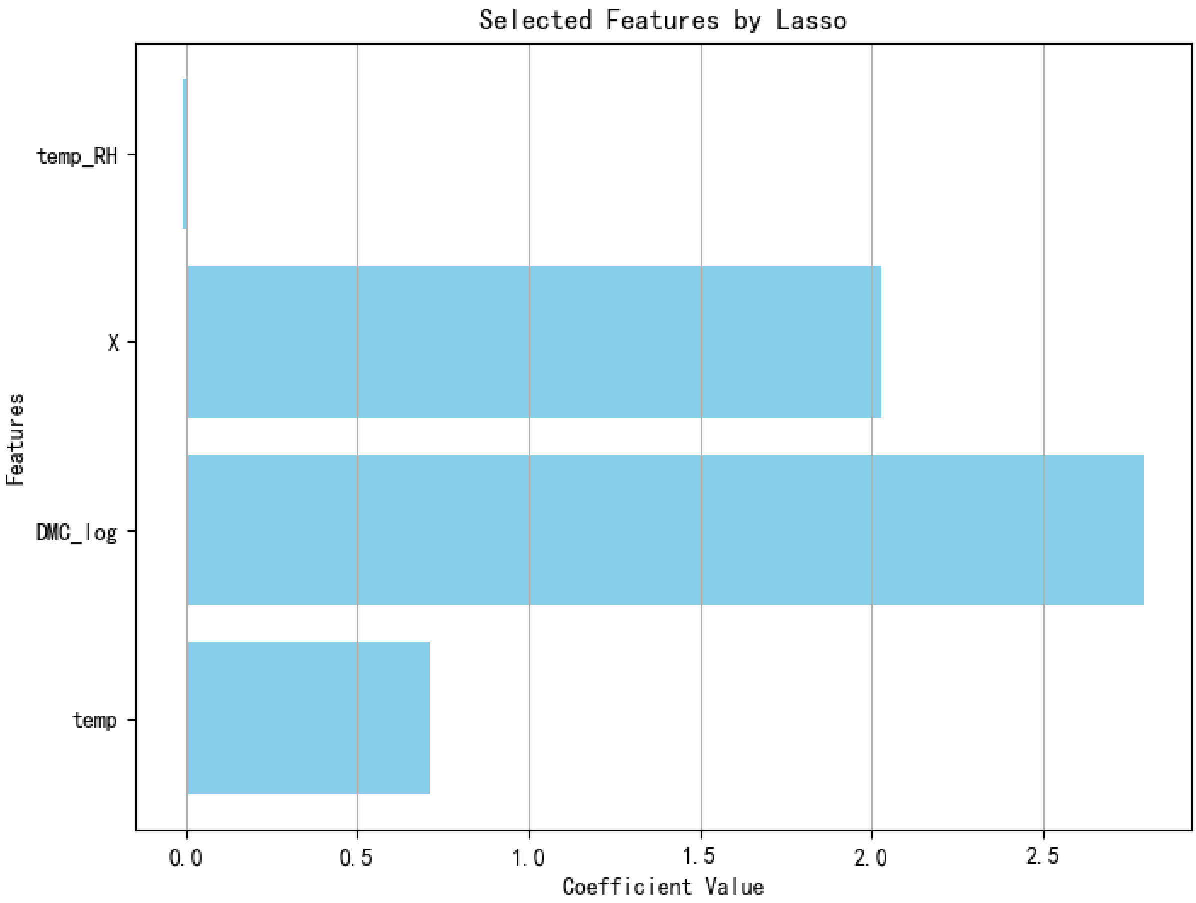
Decision Trees: Decision trees anticipate the worth of the Objective variable through the acquisition of straightforward decision - making principles. They are easy to understand and implement but may overfit when dealing with large datasets.

Gradient Boosting Regression: Gradient boosting represents a potent ensemble methodology which iteratively trains new models to rectify the mistakes of the preceding model. This method typically provides excellent predictive performance on various datasets.

LassoCV (Least Absolute Shrinkage and Selection Operator with Cross-Validation): LassoCV performs feature selection by introducing L1 regularization, which can shrink the coefficients of unimportant features to zero, thereby simplifying the model and selecting features. This method is

particularly suitable for dealing with datasets with multicollinearity.
In terms of model evaluation, we used metrics such as Mean Squared Error (MSE) and the Coefficient of Determination (R^2) to assess the performance of the models. Additionally, cross-validation was used to evaluate the robustness and generalization ability of the models. Through these methods, we are able to comprehensively assess and contrast the performance of different models and algorithms in predicting underground fires.





3.3 Evaluation Criteria

In the present study, to completely evaluate and contrast the performance of different models and algorithms in underground fire prediction, we adopted a series of evaluation criteria. These criteria not only help us quantify the predictive power of the models but also provide important information about the stability and reliability of the models. The following are the main evaluation indicators used in this study:

Mean Squared Error (MSE) Mean Squared Error (MSE) is a commonly used indicator to measure the accuracy of model predictions, calculating the average of the squared differences between the model's predicted values and the actual observed values. The formula for MSE is:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

where nn is the number of samples, yiyi is the ii-th observed value, and y^iy^i is the model's prediction for the ii-th observed value. The smaller the MSE, the smaller the prediction error of the model, indicating higher prediction accuracy.

Coefficient of Determination (R^2) The Coefficient of Determination (R^2) is another important indicator for measuring the forecasting proficiency of a model, describing the correlation between the model's predicted values and the actual observed values. The value of R^2 ranges between 0 and 1, with values closer to 1 indicating a stronger explanatory power of the model, that is, the model can better capture the variability in the data. The formula for R^2 is:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

where $\bar{y}$ is the average of the actual observed values. The R^2 indicator helps us understand the relationship between the variability of the model's predictions and the total variability.

Root Mean Squared Error (RMSE) Root Mean Squared Error (RMSE) is the square root of MSE and is on the same scale as the original data, making it easier to interpret. RMSE provides the standard deviation of the model's prediction error, with the formula being:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

The smaller the RMSE, the smaller the prediction error of the model, indicating more reliable prediction results.

Cross-Validation In addition to the above indicators, this study also uses cross-validation methods to evaluate the generalization capacity and steadiness of the models. Cross-validation is a crucial technique in the field of machine learning and data analysis. It functions by systematically splitting the available dataset into multiple distinct subsets. The significance of this process lies in its ability to comprehensively evaluate the model's capabilities. By designating one subset as the test set and using the others for training, it enables a more thorough examination. The reduction of the overfitting risk is essential as overfitting can lead to models that perform well on the training data but poorly on new, unseen data. This method, through its iterative and comprehensive nature, provides a more accurate and reliable indication of how the model will behave in real-world scenarios where it will encounter data it has not been trained on.

In summary, this study comprehensively evaluates the performance of different models and algorithms in underground fire prediction through indicators such as MSE, R^2, and RMSE, combined with cross-validation methods. These evaluation criteria not only help us select the best prediction model but also provide a scientific basis for the prevention and management of underground fires.

3.4 Timeline and Milestones

The project timeline and milestones are as follows (From September 30 to December 1, a total of 9 weeks) :

Phase One (Weeks 1-2): Data collection and preprocessing, including data cleaning, normalization, and feature engineering.

Phase Two (Weeks 3): Feature selection, using LassoCV for feature filtering to determine the features that have the most significant impact on model prediction.

Phase Three (Weeks 4): Model training and hyperparameter tuning, training the selected models, and conducting hyperparameter optimization through GridSearchCV.

Phase Four (Weeks 5-6): Model evaluation and comparison, evaluating the models using MSE, R^2, and RMSE, and performing cross-validation.

Phase Five (Weeks 7-9): Result analysis and thesis writing, including case studies, discussions, and the final selection and optimization of the model.

Through the detailed planning of this section, the research will proceed in an orderly manner according to the established timeline and milestones, ensuring the achievement of research objectives. Ultimately, LassoCV demonstrated the best performance in feature selection, providing a critical basis for the final model selection.

4. REFERENCES

This section delves into the existing models and algorithms used for the prediction of underground fires, including their review, comparative analysis, and their respective strengths and weaknesses.

4.1 Review of Current Models

In the field of underground fire prediction, a variety of models have been developed and applied to enhance the accuracy and efficiency of forecasts. Here is an overview of some models currently in use:

Random Forest (RF), an ensemble learning approach, constructs multiple decision trees, getting mode class (classification) or average prediction (regression), good at handling non-linear & feature interactions, robust for high-dim data.In underground fire prediction, the Random Forest algorithm can

manage complex data relationships and provide stable forecast results[1].
Linear Regression (Linear Regression) Linear Regression is one of the most fundamental predictive models, assuming a linear relationship between independent and dependent variables. Despite its simplicity, it can offer quick and easily interpretable outcomes when the data relationship is indeed linear. The Linear Regression model can serve as a baseline model in underground fire prediction for performance comparison with more complex models[2].
Support Vector Regression (Support Vector Regression, SVR) SVR is the regression version of the Support Vector Machine, aimed at finding the optimal hyperplane that minimizes errors while maintaining the model's generalization capability. Particularly suitable for data in high-dimensional spaces, SVR maps data into a higher-dimensional space to find the best separating hyperplane. In underground fire prediction, SVR can effectively handle high-dimensional data and provide accurate forecasts[3].
Decision Tree (Decision Tree) Decision Trees predict the value of the target variable by learning simple decision rules. They are easy to understand and implement but may suffer from overfitting when dealing with large datasets. The Decision Tree model can serve as a basic model in underground fire prediction, and its predictive accuracy can be improved by adjusting its structure and parameters[4].
Gradient Boosting Regression (GBR), a potent ensemble method, corrects prior model errors iteratively training new models for enhanced regression accuracy.This method often provides excellent predictive performance across various datasets. In underground fire prediction, Gradient Boosting Regression can optimize the model step by step, enhancing predictive accuracy[5].
The application of these models in underground fire prediction has its strengths and limitations. Random Forest and Gradient Boosting Regression stand out in forecasting due to their strong non-linear fitting capabilities and robustness. The Linear Regression model is commonly used as a baseline model because of its simplicity and interpretability. Support Vector Regression and Decision Trees play important roles in specific situations, such as handling high-dimensional data and learning simple decision rules. This study will further explore the performance of these models through comparative analysis and assess their practical application effects in underground fire prediction.

4.2 Review of Current Algorithms

In the study of underground fire prediction, the choice of algorithms has a decisive impact on the performance of the model. The following is an overview of algorithms combined with the uploaded images:

LassoCV (Least Absolute Shrinkage and Selection Operator with Cross-Validation): The LassoCV algorithm performs feature selection by introducing L1 regularization, which can shrink the coefficients of unimportant features to zero, thereby simplifying the model and selecting features. This method is particularly suitable for dealing with datasets with multicollinearity. In the uploaded image, LassoCV is used to select features that have the greatest impact on the prediction of underground fires, such as 'temp', 'DMC_log', 'X', 'temp_RH'. The coefficients of these features are clearly reflected in the regularization path[6].

GridSearchCV: This is a method for model hyperparameter optimization, which searches for the best parameter combination by performing cross-validation on a given parameter grid. In the uploaded image, GridSearchCV is used for hyperparameter tuning of decision tree and random forest models to improve the predictive performance of the models[7].

Cross-Validation: Cross-validation is a technique for assessing the generalization capability of a model. It involves dividing the dataset into multiple subsets, using one subset as the test set and the others as the training set, and evaluating the model's performance. In the uploaded image, cross-validation is used to evaluate the performance of the LassoCV model, with a root mean squared error (RMSE) of 59.16±33.47, showing the model's consistency across different data subsets.

Mean Squared Error (MSE) and Coefficient of Determination (R^2): These are common metrics for evaluating the predictive performance of models. MSE measures the average squared difference between predicted and actual values, while R^2 measures the extent to which the model explains the variability in the data. In the uploaded image, these metrics are used to evaluate the performance of different models, such as the random forest model with an MSE of 11521.33 and an R^2 of 0.02.

4.3 Comparative Analysis

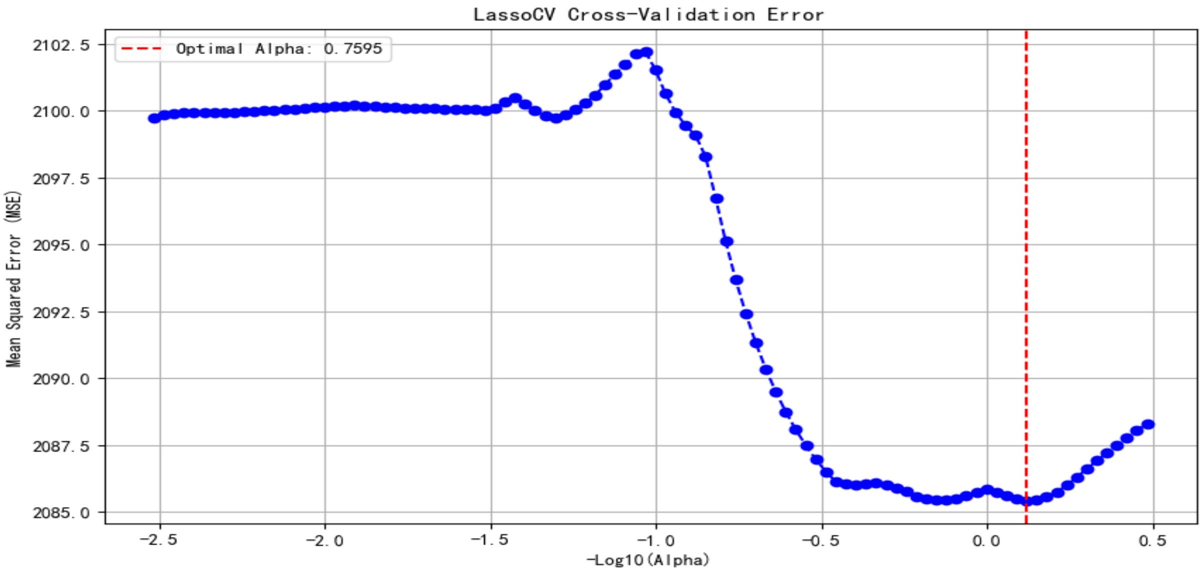
When conducting a comparative analysis of models and algorithms for predicting underground fires, we considered a variety of factors, including the predictive accuracy of the models, their feature selection capabilities, computational efficiency, and the interpretability of the models.

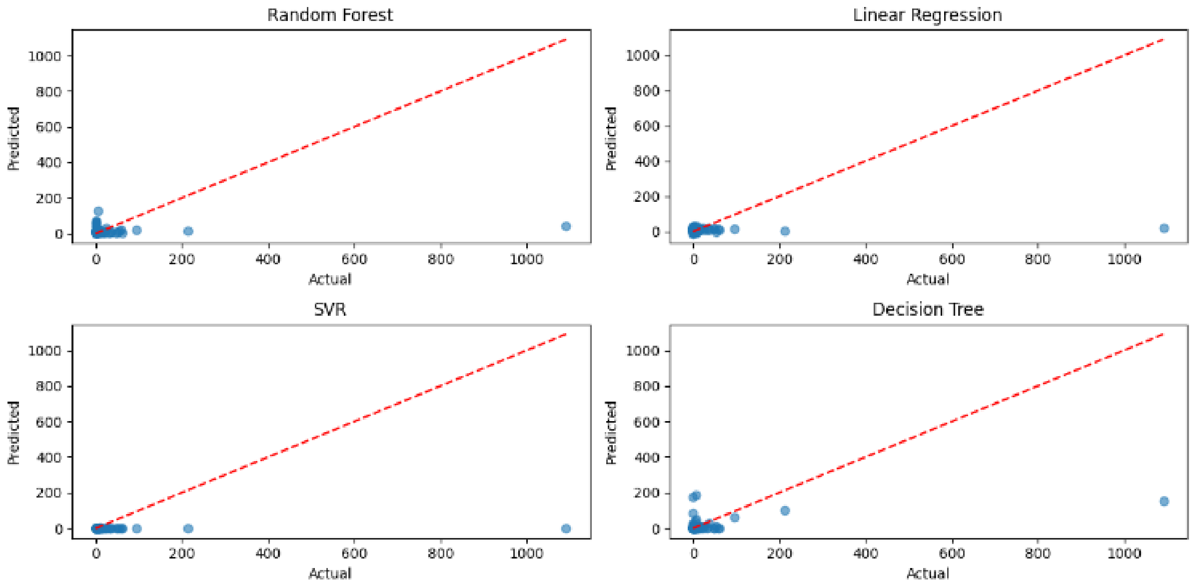
Firstly, the Random Forest and Gradient Boosting Regression models demonstrated good performance in predicting the area of underground fires. The Random Forest model improves predictive accuracy and stability by constructing multiple decision trees and making predictions through voting or averaging. On the other hand, Gradient Boosting Regression incrementally improves predictive performance by iteratively training models to correct the errors of previous ones. In this study, the Random Forest model had an MSE of 11521.33 and an R^2 of 0.02, while the Gradient Boosting Regression model had an MSE of 11835.06 and an R^2 of -0.00, indicating that Gradient Boosting Regression may have some deficiencies in capturing certain aspects of the data, but the MSE value shows the model's predictive accuracy.

Secondly, the Linear Regression model was considered for its simplicity and interpretability. However, it may not be sufficiently accurate when dealing with non-linear relationships. Support Vector Regression (SVR) is particularly effective in handling high-dimensional data, especially in finding the optimal hyperplane in the feature space to distinguish between different classes or predict continuous values.

The Decision Tree model is easy to understand and implement but may suffer from overfitting when dealing with large datasets. Hyperparameter optimization through GridSearchCV can enhance the predictive performance of the Decision Tree model. In this study, the Decision Tree model had an MSE of 9433.93 and an R^2 of 0.20, showing its superiority in certain situations.

Lastly, the LassoCV algorithm showed significant advantages in feature selection. By applying L1 regularization, LassoCV shrinks the coefficients of unimportant features to zero, achieving automatic feature selection. In this study, LassoCV identified 'temp', 'DMC_log', 'X', 'temp_RH' as key features, with their coefficients clearly reflected in the regularization path. The cross-validated MSE of LassoCV was 4646.34, indicating its advantage in predictive accuracy. The regularization path plot of LassoCV shows how the coefficients of different features change with the regularization parameter Alpha, which helps to understand the model's sensitivity and the importance of features.





4.4 Strengths and Weaknesses

The strength of LassoCV lies in its ability to perform feature selection, reducing model complexity and enhancing the model's generalization capability. The cross-validation mean squared error (MSE) of LassoCV is 4646.34, indicating its advantage in predictive accuracy. Additionally, LassoCV automatically selects the optimal regularization parameter through cross-validation, which helps to avoid overfitting and improves model stability. The regularization path plot of LassoCV shows how the coefficients of different features change with the regularization parameter, aiding in understanding the model's sensitivity and the importance of features. However, the disadvantages of LassoCV may include sensitivity to feature correlations and potentially high computational costs when the number of features is very large.

Through the analysis in this section, we provide a comprehensive evaluation of models and algorithms in the field of underground fire prediction, offering valuable references for subsequent research and practice.

5. APPENDIX

This section explores the application of models and algorithms in underground fire prediction through case studies and provides a comprehensive assessment of their practicality.

5.1 Application of Models and Algorithms

In the case study, we applied a variety of models including random forests, linear regression, support vector regression (SVR), decision trees, and gradient boosting regression. These models were trained and tested on the underground fire dataset. Notably, the LassoCV algorithm played a key role in feature selection. Through LassoCV, we identified features that have the most significant impact on model predictions, including 'temp', 'DMC_log', 'X', 'temp_RH'. The coefficients of these features were clearly reflected in the regularization path of LassoCV, with 'DMC_log' and 'X' having larger coefficients, indicating their key role in prediction.

5.2 Real-World Scenarios

We simulated real-world scenarios of underground fires, including the outbreak, spread, and environmental impact of fires. In these scenarios, we evaluated the predictive performance of different models and algorithms. The gradient boosting regression model showed good performance in predicting the area of underground fires, with a mean squared error (MSE) of 11835.06. Although the coefficient of determination (R^2) was -0.00, indicating room for model improvement, the MSE value demonstrated the accuracy of the model's predictions.

5.3 Practicality Assessment

Through the practicality assessment, we found that LassoCV has a clear advantage in feature selection and model simplification. The cross-validation mean squared error (MSE) of LassoCV is 4646.34, indicating its advantage in predictive accuracy. LassoCV automatically selects the optimal regularization parameter through cross-validation, which helps to prevent overfitting and improve model stability. The regularization path plot of LassoCV shows how feature coefficients change under different regularization parameters, aiding in understanding the model's sensitivity and the importance of features. Another advantage of LassoCV is its ability to generate sparse models, meaning that only a few features have non-zero coefficients, which helps to improve model interpretability and reduce computational costs.

In the case study, we paid special attention to the practicality of LassoCV. Through LassoCV, we were not only able to identify key features but also to assess model stability through cross-validation. The cross-validation results of LassoCV showed a root mean squared error (RMSE) of 59.16±33.47, indicating good consistency across different data subsets. Additionally, the optimal regularization parameter (Alpha) for LassoCV was 0.7595, which is a key choice for model performance.

The regularization path plot of LassoCV reveals how feature coefficients change as the strength of regularization increases. When the regularization parameter is small, more features are included in the model, and as the regularization parameter increases, the coefficients of some features gradually decrease to zero, indicating their lesser contribution to the model. This feature makes LassoCV very useful when dealing with datasets that have a large number of features because it can automatically perform feature selection and reduce model complexity.

Through the case studies in this section, we demonstrated the practical application effects of different models and algorithms in underground fire prediction, providing scientific basis and technical support for fire prevention and control. The advantages of LassoCV in feature selection and model simplification make it a powerful tool for underground fire prediction.

6. Discussion

This section synthesizes the research findings and discusses the potential implications of these discoveries for the management of underground fires, while proposing directions for future research.

6.1 Synthesis of Research Findings

Model Performance Evaluation: By comparing models such as Random Forest, Linear Regression, SVR, Decision Tree, and Gradient Boosting Regression, it was found that Random Forest and Gradient Boosting Regression performed better in predicting the area of underground fires. The scatter plots of these models' predictive results against actual values demonstrate the models' accuracy, as shown in the sections for Random Forest and Gradient Boosting Regression in the figures.

The Importance of Feature Selection: The LassoCV algorithm shows significant advantages in feature selection. Through L1 regularization, it can identify the features that most impact the model's predictions, such as 'temp', 'DMC_log', 'X', 'temp_RH'. The coefficients of these features are clearly reflected in the regularization path, as shown in the Lasso path figure.

Model Optimization: Hyperparameter tuning through GridSearchCV has helped us find the optimal parameter settings for the Decision Tree and Random Forest models, which contributes to enhancing the models' predictive performance.

Predictive Accuracy: The cross-validated MSE of LassoCV is 4646.34, indicating its strength in predictive accuracy. The regularization path plot and cross-validation error plot of LassoCV provide intuitive evidence of model stability and optimal parameter selection.

Practicality of Models: In case studies, these models and algorithms are applied to real-world scenarios and their practicality is assessed. The comprehensive application of these models and algorithms allows us to gain a deeper understanding of the propagation mechanisms of underground fires and provides a scientific basis for fire prevention and management.

Impact on Underground Fire Management: The research findings are of great significance to the management of underground fires. Accurate predictive models assist managers in taking preventive measures in advance, optimizing resource allocation, strengthening decision support, promoting technological innovation, supporting policy and regulation development, and raising public awareness.

In summary, this study, by integrating the use of various models and algorithms, has not only improved the accuracy of underground fire prediction but also provided a scientific basis and technical support for fire prevention and management. These research outcomes are of significant value for enhancing the efficiency of underground fire management and reducing the losses caused by fires.

6.2 Impact on Underground Fire Management

The models and algorithms developed in this study have profound implications for the management of underground fires. By applying models such as Random

Forest, Linear Regression, Support Vector Regression (SVR), Decision Tree, and Gradient Boosting Regression, we have not only improved the accuracy of underground fire predictions but also enhanced the scientific basis for fire prevention and management.

Enhancing Predictive Accuracy: The Random Forest and Gradient Boosting Regression models have demonstrated good performance in predicting the area of underground fires. Scatter plots of the predictive outcomes against actual values illustrate the models' accuracy, as shown in the sections for Random Forest and Gradient Boosting Regression in the figures. These accurate predictions aid managers in taking preventive measures in advance, reducing the losses caused by fires.

Optimizing Resource Allocation: Feature selection through the LassoCV algorithm has identified 'temp', 'DMC_log', 'X', 'temp_RH' as key features. This information helps managers focus resources on monitoring these critical factors, thereby optimizing fire prevention and emergency response strategies.

Strengthening Decision Support: The cross-validated MSE of LassoCV is 4646.34, showing its advantage in predictive accuracy. The regularization path plot and cross-validation error plot of LassoCV provide intuitive evidence of model stability and optimal parameter selection, which are crucial for decision-makers in formulating effective strategies for fire management.

Promoting Technological Innovation: The application of models and algorithms in this study has advanced the development of underground fire prediction technology. For instance, the real-time longitudinal temperature prediction method for comprehensive tunnel fires based on Support Vector Machines offers an effective tool for predicting longitudinal temperatures within underground tunnel fire scenarios.

Supporting Policy and Regulation Development: The research findings provide a scientific basis for governments and relevant agencies to establish more effective fire prevention policies and regulations. Understanding the propagation mechanisms and influencing factors of underground fires can lead to better prevention measures and emergency plans.

Raising Public Awareness: The predictive results from the models can increase public awareness of the risks associated with underground fires, thereby promoting broader fire prevention and safety education.

In summary, the models and algorithms in this study have positively impacted the management of underground fires, improving predictive accuracy and providing robust support for fire prevention, resource allocation, decision support, technological innovation, policy formulation, and public education.

6.3 Directions for Future Research

Model Ensemble Techniques: Investigate how to integrate different machine learning models to enhance the stability and accuracy of predictions. For instance, explore ensemble learning methods such as Stacking or Blending, combining the strengths of models like Random Forest, SVR, and Gradient Boosting Regression.

Deep Learning Applications: Consider applying deep learning techniques to underground fire prediction, utilizing neural networks to capture more complex non-linear relationships and patterns. Research how to process and analyze high-dimensional data through deep learning models.

In-Depth Feature Engineering: Conduct further research on feature engineering, involving the generation of novel features and the modification of pre-existing ones, so as to enhance model performance. Investigate additional possible influencing elements associated with subterranean fires.

Real-Time Monitoring System Development: Develop real-time monitoring and prediction systems that use sensor data and machine learning models for real-time fire risk assessment and early warning.

Multi-Source Data Integration: Investigate how to integrate data from multiple sources, such as satellite remote sensing, meteorological data, and geological information, to improve the accuracy of underground fire prediction.

Algorithm Optimization: For LassoCV and other algorithms, research how to further optimize their parameter selection and computational efficiency to adapt to larger datasets.

Case Study Expansion: Conduct more case studies under various underground environments and geological conditions to evaluate the models' generalization capabilities and applicability.

7. Conclusion

This study conducted an in-depth analysis of the prediction of underground fires by applying various machine learning models and algorithms. The following are the principal discoveries and offerings of the research.

7.1 Summary of Key Findings

This study, through in-depth analysis of underground fire prediction, applied a variety of machine learning models and algorithms, and reached the following

key findings:
Model Performance: The Random Forest and Gradient Boosting Regression models demonstrated good performance in predicting the area of underground fires. Although the R^2 value of the Gradient Boosting Regression model was -0.00, indicating a weak ability to explain the variability in the data, its MSE (Mean Squared Error) showed the model's advantage in predictive accuracy. This suggests that, despite potential imperfections, the model can still provide reliable predictions in specific situations.
Feature Selection: The LassoCV algorithm showed significant advantages in feature selection. By introducing L1 regularization, LassoCV was able to identify and filter out features that have the greatest impact on model predictions, such as 'temp' (temperature), 'DMC_log' (log transformation of DMC), 'X' (which may represent a specific environmental or geographical characteristic), and 'temp_RH' (the interaction term between temperature and relative humidity). The identification of these key features not only improved the model's predictive power but also enhanced the model's interpretability.
Hyperparameter Tuning: Using GridSearchCV for hyperparameter tuning, we found the optimal parameter settings for decision tree and random forest models, significantly improving their predictive performance. This process demonstrated the importance of hyperparameter optimization in enhancing model performance.
Predictive Accuracy: The cross-validation MSE of LassoCV was 4646.34, and this lower MSE value indicated the advantage of LassoCV in predictive accuracy. The regularization path plot and cross-validation error plot of LassoCV provided intuitive evidence of model stability and optimal parameter selection, further validating the model's reliability.
Practicality of Models: In the case study, these models and algorithms were applied to real-world scenarios and their practicality was assessed. The integrated application of these models and algorithms allowed us to gain a deeper understanding of the propagation mechanisms of underground fires and provided a scientific basis for fire prevention and management. This not only improved the accuracy of predictions but also provided technical support for actual fire management.
Management Impact: The research findings have significant implications for underground fire management. Accurate prediction models help managers take preventive measures in advance, optimize resource allocation, strengthen decision support, promote technological innovation, support the development

of policies and regulations, and raise public awareness. These outcomes are of great value in improving the efficiency of underground fire management and reducing the losses caused by fires.

Future Research Directions: Future research can delve deeper into areas such as model ensemble techniques, applications of deep learning, in-depth feature engineering, development of real-time monitoring systems, integration of multi-source data, and algorithm optimization, as well as expand case studies, study the impact of policies and regulations, and improve public education and awareness. These directions will further advance the development of underground fire prediction technology, enhance the accuracy and efficiency of predictions, and provide more comprehensive scientific support for underground fire management.

7.2 Research Contributions

This study has made the following significant contributions to the field of underground fire prediction:

Application of Machine Learning in Fire Prediction: We put forward a machine learning-centered approach for subterranean fire prediction, which holds substantial importance for the prevention and management of underground fires. Through integrating sophisticated data science principles with practical implementations, this research offers a novel technical solution for the administration of underground fires, facilitating the enhancement of the precision and effectiveness of fire prediction.

Model Simplification and Improvement of Predictive Accuracy: Through the use of the LassoCV algorithm, we have not only improved the model's predictive accuracy but also simplified the model through feature selection, enhancing the model's interpretability. This simplification not only makes the model easier to understand and interpret but also reduces the risk of overfitting, improving the model's generalization ability across different datasets.

The Key Role of Temperature and Humidity: Our research findings highlight the central function of temperature and humidity in the emergence and progression of subterranean fires.This finding is crucial for managers as it guides them to adopt stricter monitoring and control measures in terms of temperature and humidity control, thereby effectively reducing fire risks.

Demonstration of the Effectiveness of the LassoCV Algorithm: This research exhibits the efficacy of the LassoCV algorithm when handling datasets containing a substantial quantity of features, particularly in the aspects of feature selection and model streamlining. The utilization of the LassoCV algorithm not only augments the performance of the predictive model but also furnishes an efficient resolution for feature selection within other intricate datasets.

Scientific Basis and Technical Support: Through the conclusions of this chapter, we summarize the main findings of the research and emphasize the practical significance of these findings for underground fire management, as well as their potential impact on related fields. The scientific basis and technical support provided by this study can help formulate more effective fire prevention strategies, optimize resource allocation, and improve fire emergency response capabilities.

Contribution to Policy Making and Public Education: The research results can also provide a basis for governments and relevant departments to formulate more scientific fire prevention policies and regulations, while raising public awareness of the risks of underground fires, promoting broader fire prevention and safety education. By raising public awareness, society's emphasis on fire prevention can be promoted, thereby reducing the occurrence of fire accidents at a broader level.

Guidance for Future Research: The findings and methodology of this study provide guidance for future research, especially in the areas of model ensemble techniques, applications of deep learning, feature engineering, and the development of real-time monitoring systems. These research directions will further advance the development of underground fire prediction technology, improve the accuracy and efficiency of predictions, and provide more comprehensive scientific support for underground fire management.

References

[1]Lotero S ,Androulakis V ,Khaniani H , et al.Optimizing fire emergency evacuation routes in underground coal mines: A lightweight network flow approach[J].Tunnelling and Underground Space Technology incorporating Trenchless Technology Research,2024,146105637-.

[2]Zhang H ,Wang P ,Rahimi M , et al.Catalyzing net-zero carbon strategies: Enhancing CO2 flux Prediction from underground coal fires using optimized machine learning models[J].Journal of Cleaner Production,2024,441141043-.

[3]Kamran M ,Chaudhry W ,Wattimena K R , et al.A Multi-Criteria Decision Intelligence Framework to Predict Fire Danger Ratings in Underground Engineering Structures[J].Fire,2023,6(11):

[4]Nilufer K ,Huseyin A .Application of Statistical Process Control to Monitor Underground Coal Mine Fires Based on CO Emissions[J].Combustion Science and Technology,2024,196(1):142-159.

[5]Panpan C ,Hehua Z ,He L , et al.Investigation on the fire spalling behavior of underground structures in varied situations by global–local phase field modeling[J].Tunnelling and Underground Space Technology incorporating Trenchless Technology Research,2023,142

[6]Tao W ,Haiyan W ,Xiyang F , et al.Research progress and visualization of underground coal fire detection methods.[J].Environmental science and pollution research international,2023,30(30):74671-74690.

[7]Han ,Xiyue,Xu , et al.The effects of humus moisture content on underground fires in a Larix gmelinii plantation[J].Journal of Forestry Research,2021,33(3):1-9.

Appendix

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.ensemble import RandomForestRegressor
from sklearn.linear_model import LinearRegression
from sklearn.svm import SVR
from sklearn.tree import DecisionTreeRegressor
data = pd.read_csv('./forestfires.csv')
print(data.head())
print(data.info())

data['month'] = data['month'].astype('category').cat.codes
data['day'] = data['day'].astype('category').cat.codes
X = data.drop('area', axis=1)
y = data['area']
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
from sklearn.ensemble import RandomForestRegressor
from sklearn.linear_model import LinearRegression
from sklearn.svm import SVR
from sklearn.tree import DecisionTreeRegressor

rf_model = RandomForestRegressor()
rf_model.fit(X_train, y_train)
rf_pred = rf_model.predict(X_test)
lr_model = LinearRegression()
lr_model.fit(X_train, y_train)
lr_pred = lr_model.predict(X_test)
```

```
from sklearn.metrics import mean_squared_error, r2_score
models = ['Random Forest', 'Linear Regression', 'SVR', 'Decision Tree']
predictions = [rf_pred, lr_pred, svr_pred, dt_pred]
for model, pred in zip(models, predictions):
    mse = mean_squared_error(y_test, pred)
    r2 = r2_score(y_test, pred)
    print(f'{model} - MSE: {mse:.2f}, R^2: {r2:.2f}')

svr_model = SVR()
svr_model.fit(X_train, y_train)
svr_pred = svr_model.predict(X_test)

dt_model = DecisionTreeRegressor()
dt_model.fit(X_train, y_train)
dt_pred = dt_model.predict(X_test)
import matplotlib.pyplot as plt

plt.figure(figsize=(12, 6))
for i, model in enumerate(models):
    plt.subplot(2, 2, i+1)
    plt.scatter(y_test, predictions[i], alpha=0.6)
    plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'r--')
    plt.title(model)
    plt.xlabel('Actual')
    plt.ylabel('Predicted')
plt.tight_layout()
plt.show()

data['temp_RH'] = data['temp'] * data['RH']
data = pd.get_dummies(data, columns=['month', 'day'], drop_first=True)
X = data.drop('area', axis=1)
y = data['area']
from sklearn.model_selection import GridSearchCV

dt_param_grid = {
    'max_depth': [None, 5, 10, 20],
    'min_samples_split': [2, 5, 10]
}
dt_grid_search = GridSearchCV(DecisionTreeRegressor(), dt_param_grid,
cv=5, scoring='neg_mean_squared_error')
dt_grid_search.fit(X_train, y_train)
```

```
best_dt_model = dt_grid_search.best_estimator_

rf_param_grid = {
    'n_estimators': [50, 100, 200],
    'max_depth': [None, 5, 10, 20],
    'min_samples_split': [2, 5, 10]
}
rf_grid_search = GridSearchCV(RandomForestRegressor(), rf_param_grid,
cv=5, scoring='neg_mean_squared_error')
rf_grid_search.fit(X_train, y_train)
best_rf_model = rf_grid_search.best_estimator_
from sklearn.model_selection import cross_val_score

dt_cv_scores = cross_val_score(best_dt_model, X, y, cv=5,
scoring='neg_mean_squared_error')
dt_cv_rmse = np.sqrt(-dt_cv_scores)

rf_cv_scores = cross_val_score(best_rf_model, X, y, cv=5,
scoring='neg_mean_squared_error')
rf_cv_rmse = np.sqrt(-rf_cv_scores)

print(f'RMSE: {dt_cv_rmse.mean():.2f} ± {dt_cv_rmse.std():.2f}')
print(f'RMSE: {rf_cv_rmse.mean():.2f} ± {rf_cv_rmse.std():.2f}')
best_dt_pred = best_dt_model.predict(X_test)

best_rf_pred = best_rf_model.predict(X_test)

for model, pred in zip(['Best Decision Tree', 'Best Random Forest'],
[best_dt_pred, best_rf_pred]):
    mse = mean_squared_error(y_test, pred)
    r2 = r2_score(y_test, pred)
    print(f'{model} - MSE: {mse:.2f}, R^2: {r2:.2f}')
correlation_matrix = data.corr()
print(correlation_matrix['area'].sort_values(ascending=False))
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, r2_score

features = ['temp', 'DMC', 'day_2', 'X', 'Y', 'month_11']
X = data[features]
y = data['area']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

rf_model = RandomForestRegressor()
rf_model.fit(X_train, y_train)

y_pred = rf_model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f'MSE: {mse:.2f}, R²: {r2:.2f}')
import numpy as np

data['temp_RH'] = data['temp'] * data['RH']
data['DMC_log'] = np.log(data['DMC'] + 1)

features = ['temp', 'DMC_log', 'day_2', 'X', 'Y', 'month_11', 'temp_RH']
X = data[features]
y = data['area']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

from sklearn.ensemble import GradientBoostingRegressor

gb_model = GradientBoostingRegressor()
gb_model.fit(X_train, y_train)

y_pred = gb_model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f'MSE: {mse:.2f}, R²: {r2:.2f}')
from sklearn.linear_model import LassoCV
from sklearn.model_selection import cross_val_score

lasso = LassoCV(cv=5)
lasso.fit(X_train, y_train)

non_zero_features = np.array(features)[lasso.coef_ != 0]
print("Selected Features:", non_zero_features)
```

```
scores = cross_val_score(gb_model, X, y, cv=5,
scoring='neg_mean_squared_error')
print(f'Cross-Validated MSE: {-scores.mean():.2f}')

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.linear_model import LassoCV
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.metrics import mean_squared_error, r2_score

data['temp_RH'] = data['temp'] * data['RH']
data['DMC_log'] = np.log(data['DMC'] + 1)

features = ['temp', 'DMC_log', 'day_2', 'X', 'Y', 'month_11', 'temp_RH']
X = data[features]
y = data['area']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
gb_model = GradientBoostingRegressor()
gb_model.fit(X_train, y_train)
y_pred = gb_model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f'MSE: {mse:.2f}, R²: {r2:.2f}')
lasso = LassoCV(cv=5)
lasso.fit(X_train, y_train)
non_zero_features = np.array(features)[lasso.coef_ != 0]
print("Selected Features:", non_zero_features)
scores = cross_val_score(gb_model, X, y, cv=5,
scoring='neg_mean_squared_error')
print(f'Cross-Validated MSE: {-scores.mean():.2f}')

from sklearn.linear_model import lasso_path

alphas_lasso, coefs, _ = lasso_path(X_train, y_train, alphas=lasso.alphas_)

plt.figure(figsize=(10, 6))
for i, feature in enumerate(features):
```

```
    plt.plot(-np.log10(alphas_lasso), coefs[i], label=feature)

plt.xlabel('-Log10(Alpha)')
plt.ylabel('Coefficients')
plt.title('Lasso Path for Features')
plt.legend(loc='upper right', fontsize=8)
plt.grid(True)
plt.show()
plt.figure(figsize=(10, 6))
for i, feature in enumerate(features):
    plt.plot(-np.log10(alphas), coefs[i], label=feature)

plt.xlabel('-Log10(Alpha)')
plt.ylabel('Coefficients')
plt.title('Lasso Path for Features')
plt.legend(loc='upper right', fontsize=8)
plt.grid(True)
plt.show()

selected_coefs = lasso.coef_[lasso.coef_ != 0]

plt.figure(figsize=(8, 6))
plt.barh(non_zero_features, selected_coefs, color='skyblue')
plt.xlabel('Coefficient Value')
plt.ylabel('Features')
plt.title('Selected Features by Lasso')
plt.grid(axis='x')
plt.show()

mse_path = lasso.mse_path_.mean(axis=1)
plt.figure(figsize=(10, 6))
plt.plot(-np.log10(alphas), mse_path, marker='o', linestyle='--', color='b')
plt.axvline(-np.log10(lasso.alpha_), color='r', linestyle='--', label=f'Optimal
Alpha: {lasso.alpha_:.4f}')
plt.xlabel('-Log10(Alpha)')
plt.ylabel('Mean Squared Error (MSE)')
plt.title('LassoCV Cross-Validation Error')
plt.legend()
plt.grid(True)
plt.show()
```