Problem Chosen	2023 ShuWei Cup Summary Sheet	Team Control Number
B		2023111421454

Summary

This text discusses the impact and catalytic reaction mechanisms of the pyrolysis of cotton stalks, cellulose, and lignin catalyzed by desulfurized ash at different pyrolysis composition ratios. Additionally, predictive models are established based on limited data to provide accurate guidance for the trends in pyrolysis product distribution changes.

For the first question, Attachment 1 presents the pyrolysis product yields of CS, CE, and LG at different desulfurized fly ash mixing ratios. Data preprocessing, including handling missing values and outliers, was performed. Visualizations such as scatter plots, line charts, and stacked charts were utilized to observe the trends in the yields of various pyrolysis products at different mixing ratios.

Then, based on the initial reaction principles, a univariate linear regression model was established between the pyrolysis product yields and the corresponding pyrolysis composition mixing ratios. Correlation and causation analyses were conducted, revealing that the desulfurized fly ash catalyst significantly promoted the pyrolysis processes of CS, CE, and LG.

For the second question, Attachment 2 displays the yields of pyrolysis gas components of CS, CE, and LG at different desulfurized fly ash mixing ratios. Data preprocessing, including handling missing values and outliers, was carried out. Line charts were used to visualize the relationship between the yields of pyrolysis gas components and the corresponding pyrolysis composition mixing ratios. Correlation and causation analyses were conducted to explore the impact of mixing ratios on the yields of pyrolysis gas components.

For the third question, further visualizations of the data in Attachments 1 and 2 were performed to obtain the pyrolysis product distribution and gas distribution catalyzed by DFA for different substances. Differential analysis was then conducted, and significant differences in the yields of pyrolysis products and gas components for CE and LG were observed under the catalysis of desulfurized fly ash at the same ratio.

For the fourth question, a catalytic reaction mechanism model for CE and LG catalyzed by desulfurized fly ash was established based on literature review, analysis of the obtained images from the first three questions, and polynomial fitting. Reaction kinetics modeling was then conducted.

For the fifth question, due to the limited amount of data, mathematical prediction models were established. The fitting model obtained from the fourth question was initially used for prediction. Subsequently, a grey prediction model was established. Finally, a comparison was made between the two methods, and the grey prediction model was determined to be the model with the highest fitting degree.

Key word: Biomass Pyrolysis,Desulfurized Ash,Linear Regression,Polynomial Regression,Grey Prediction,Reaction Kinetics

Content

1. Introduction

1.1 Problem Background

Cotton stalks, rich iiomass components such as cellulose and lignin, are considered important biomass resources. The study aims to explore the mechanism and properties of the pyrolysis products of cotton stalks, as well as the mechanism and effects of catalysts during the pyrolysis process. The selection of model compounds CE (cellulosic oligosaccharide) and LG (lignin) represents cellulose and lignin, respectively, in cotton stalks. This choice facilitates a more precise control of experimental conditions to investigate the catalytic effects of sulfide ash on different biomass components. The experiments were conducted using a fixed-bed pyrolysis system under various mixing ratios, aiming to explore the catalytic mechanisms and effects of sulfide ash on the pyrolysis products of cotton stalks.

The purpose of our study is to explore the catalytic mechanisms and effects of desulfurized fly ash on different biomass components of cotton stalks. The aim is to provide precise guidance for understanding the trends in the distribution of pyrolysis products.

1.2 Restatement of the problem

Question1 Analyze the relationship between the yields of pyrolysis products (tar, water, char, syngas) and the corresponding pyrolysis blend ratios. Elucidate whether desulfurized fly ash, acting as a catalyst, promotes the pyrolysis processes of cotton stalks, cellulose, and lignin.

Question2 Discuss the influence of blending ratios on the yields of pyrolysis gas components through visual representations such as graphs

Question3 Under the catalytic effect of desulfurized fly ash at the same proportion, whether there are significant differences in the pyrolysis products and the yields of pyrolysis gas components between CE and LG.

Question4 Establish a catalytic reaction mechanism model for desulfurized fly ash on model compounds of CE and LG, and perform reaction kinetics modeling.

Question5 Predicting the yields or quantities of pyrolysis products under limited data conditions.

1.3 Our work of five questions

Question1 Appendix One displays the pyrolysis product yields of CS, CE, and LG under different desulfurized fly ash blending ratios. We began by preprocessing the data, addressing missing values and outliers. Subsequently, we employed scatter plots, line graphs, and stacked plots to visualize the yields of various pyrolysis products, observing the trends in product variation under different blending ratios. Following this, based on initial reaction principles, we established a univariate linear regression model for the relationship between pyrolysis product yields and the corresponding blending ratios of the catalyst. Finally, we conducted correlation analysis and causal analysis to explore whether the desulfurized fly ash catalyst promotes the pyrolysis processes of CS, CE, and LG.

Question2 Appendix Two presents the yields of pyrolysis gas components of CS, CE, and LG under different desulfurized fly ash blending ratios. We started by preprocessing the data, addressing missing values and outliers. Subsequently, we used line graphs to visualize the yields of various pyrolysis gases, aiming to observe the relationship between pyrolysis gas yields and the corresponding blending ratios. Finally, we conducted correlation analysis and causal analysis to explore the impact of blending ratios on the yields of pyrolysis gas components.

Question3 First, we conducted further visualization of the data in Appendices One and Two to obtain the distribution of pyrolysis products and pyrolysis gas components catalyzed by desulfurized fly ash (DFA) for different substances. Through initial observations, we examined whether there were significant differences in the yields of pyrolysis products and pyrolysis gas components between CE and LG under the catalytic effect of desulfurized fly ash at the same proportion. Next, we performed differential analysis using analysis of variance (ANOVA) to further statistically analyze the existence of significant differences.

Question4 Firstly, through literature review, we analyzed the images obtained from the first three questions, explained the reasons for the variations in pyrolysis product and pyrolysis gas component yields, and established a catalytic reaction mechanism model of desulfurized fly ash for CE and LG. Then, by consulting the literature, we conducted exponential fitting and polynomial fitting for reaction kinetics modeling.

Question5 Due to the limited amount of data, we established mathematical prediction models. Firstly, we utilized the fitting model obtained from the fourth question for prediction. Subsequently, we employed the grey forecasting method to establish another prediction model. Finally, we compared the fitting degrees of the two methods and identified the prediction model with the highest fitting degree.

2. Symbol and Assumptions

2.1 Symbol Description

Sign	Instructions
β_i	regression coefficient
X	Pyrolytic combination mixing
Y	Product yield
ε	error
P_{xy}	Pearson's correlation coefficient
k	reaction rate constant
A	Preexponential factor
E	activation energy
R	perfect gas constant
T	temperature

2.2 Assumptions

In order to construct a more accurate mathematical model, this paper makes the following reasonable assumptions and conditions based on the actual circumstances.

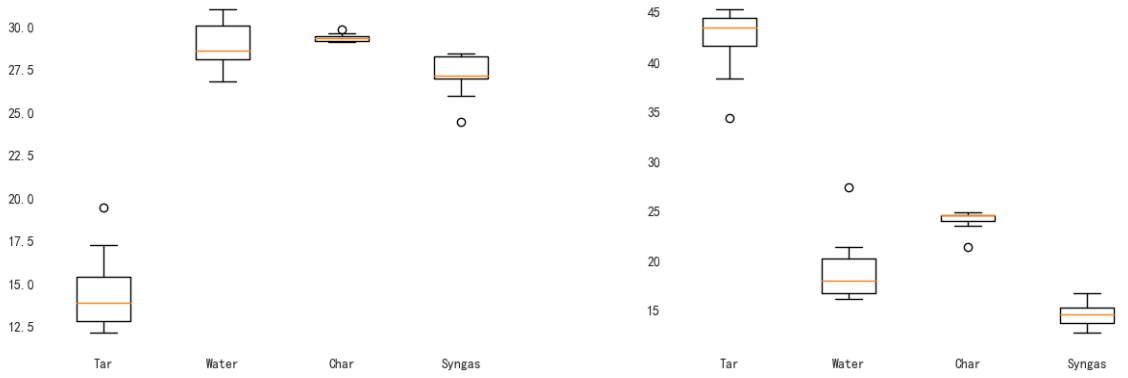
* In the catalytic pyrolysis process of cotton stalks, cellulose, and lignin using DFA, it is assumed that, aside from the specified catalyst proportion, temperature, and reaction time, the influence of other conditions on the experimental results can be considered negligible.
* During the reaction, it is assumed that the ambient temperature and pressure in the laboratory will not have an impact on the reaction.

3. Solution of Question1

3.1 Data Preprocessing and Visualization

We began by preprocessing the data in Appendix 1. Upon analyzing the data, we did not identify any missing values. Next, we employed box plots to detect outliers. The results of the outlier detection are shown in the following figure: Upon observation, we found no missing

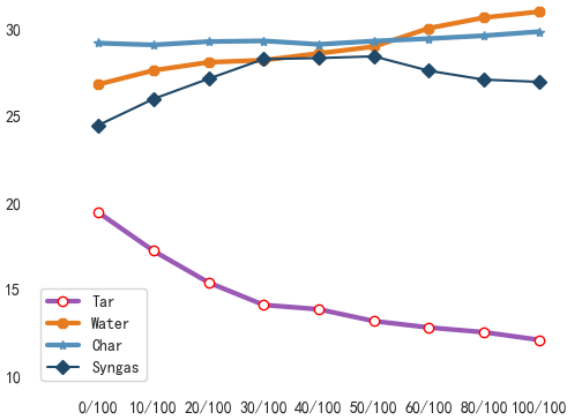
values in the data. We obtained box plots for the yields of various pyrolysis products. We observed that the outliers in CS pyrolysis product data were mainly generated when not catalyzed by desulfurized fly ash. The outliers for CE and LG were mostly generated when the pyrolysis blend ratio was 0.1. Given the complexity of chemical reactions and the limitations in data volume, we did not address these outliers.



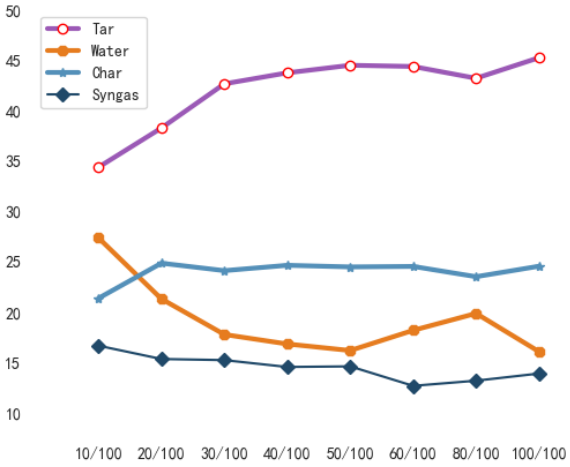
(a) DFA/CS (b) DFA/CE

Figure 1 Box plot

In the second step, we visualized the data. Figure 2 and Figure 3 illustrates the variation trends of various pyrolysis products for CS, CE, and LG at different DFA blending ratios. We observed that With the increase in blending ratio, the tar yield of CS significantly decreases, and the syngas yield initially increases and then decreases, while the water yield notably increases. For CE, the tar and char yields significantly increase, inhibiting the generation of water and syngas. In the case of LG, the production of tar is suppressed, while the generation of water and syngas is promoted.



(a) DFA/CS



(b) DFA/CE

Figure 2 Plot

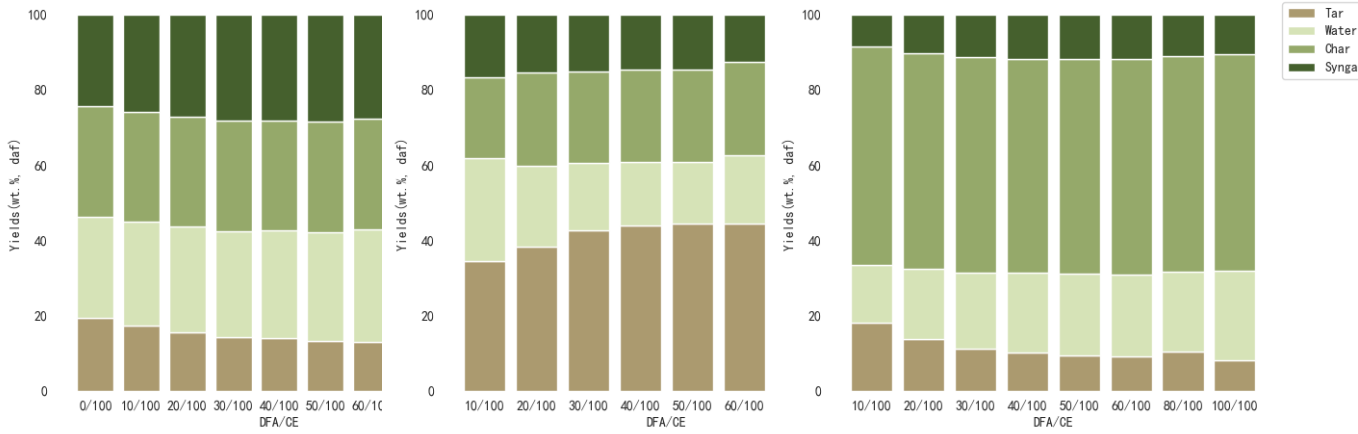
3.2 Multiple Linear Regression Model

In the third step, we established a multiple linear regression model to further explore the relationship between blending ratios and the yields of pyrolysis products. For each pyrolysis combination, we constructed a multiple linear regression model, with the blending ratio as the independent variable and the yield of pyrolysis products as the dependent variable. The model can be represented as:

$$Y_i = \beta_{0i} + \beta_{1i}X + \beta_{2i}D + \beta_{3i}XD + \varepsilon_i$$

Y_i represents the yield of pyrolysis products.

X represents the blending ratio.



(a) DFA/CS (b) DFA/CE (c) DFA/LG

Figure 3 Stack Bar

D represents whether a catalyst is added.

DX represents the degree of interaction between the catalyst and the raw material.

β_{0i} represents the intercept term.

β_{1i} represents the blending ratio coefficient.

β_{2i} represents the catalyst coefficient.

β_{3i} represents the coefficient of interaction between the catalyst and the raw material.

ε_i represents the Residual

The goal of the least squares method is to minimize the sum of squared differences between the actual values Y_i and the estimated values $\hat{Y}_i$. In this problem, the sample regression function is given by:

$$Y_i = \beta_{0i} + \beta_{1i}X + \beta_{2i}D + \beta_{3i}XD + \varepsilon_i$$

For the i-th sample, the difference between the actual and estimated values is:

$e_i = Y_i - Y_i = Y_i - (\beta_0 + \beta_1 X_i + \beta_2 D_i + \beta_3 XD_i)$

The sum of squared residuals for all i (with a sample size of n) is:

$Q = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} [Y_i - (\beta_0 + \beta_1 X_i + \beta_2 D_i + \beta_3 XD_i)]^2$

Let R_{ji} denote the residuals of the j-th explanatory variable X_j after regressing on other explanatory variables using the Ordinary Least Squares (OLS) method:

$\beta_j = \frac{\sum_{i=1}^{n} R_{ji} Y_i}{\sum_{i=1}^{n} R_{ji}^2}$

The multiple regression equations and corresponding R-squared values for each pyrolysis product obtained are shown in the following table. Additionally, we conducted t-tests to examine the significance of the regression coefficients.

Partial fit images are shown in Figure Four and Figure Five.

Category	Equation	R-square
CS_Tar	y = -2.4915x-3.0983D-2.4915Dx+19.46	0.915
CS_Water	y = 2.0356x+0.3477D+2.0356Dx+26.84	0.973
CS_Char	y = 0.3838x-0.1842D+0.3838Dx+29.21	0.853
CS_Syngas	y = 0.072x-2.9347D+0.0721Dx+24.49	0.853
CE_Tar	y = 4.6892x+37.5105D+4.6892Dx	0.575
CE_Water	y = -3.7277x+22.8971D+0.6879Dx	0.365
CE_Char	y = 0.6879x+23.3943D+0.6879Dx	0.135
CE_Syngas	y = -1.6493x+16.1981D-1.6493Dx	0.642
LG_Tar	y = -4.1803x+15.3758D-4.1803Dx	0.630
LG_Water	y = 3.6112x+16.9141D+3.6112Dx	0.750
LG_Char	y = -0.2415x+57.5717D-2.2451Dx	0.156
LG_Syngas	y = 0.8107x+10.1383D+0.8107Dx	0.183

Table 1

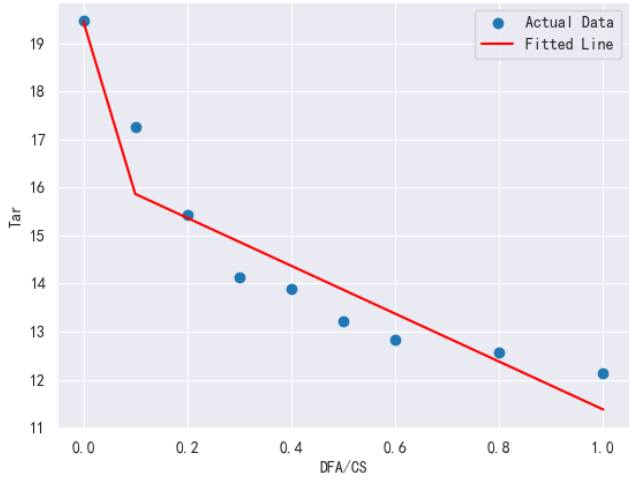


Figure 4 CS/Tar fitting figure

3.3 Correlation Analysis and Causal Analysis

We first obtained the Pearson correlation coefficients between each pyrolysis product and the blending ratio. Then, we visualized the correlations between variables by generating a correlation heatmap. Figure 4 is only the heatmap for CS, while the correlation images for CE and

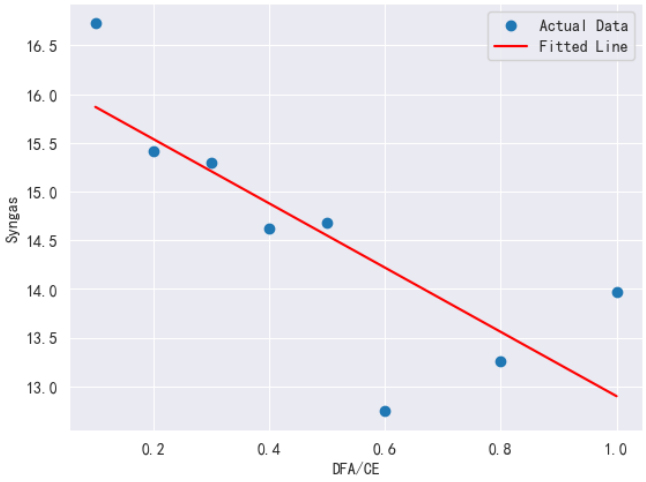


Figure 5 CE/Syngas fitting figure

LG are shown in the appendix.

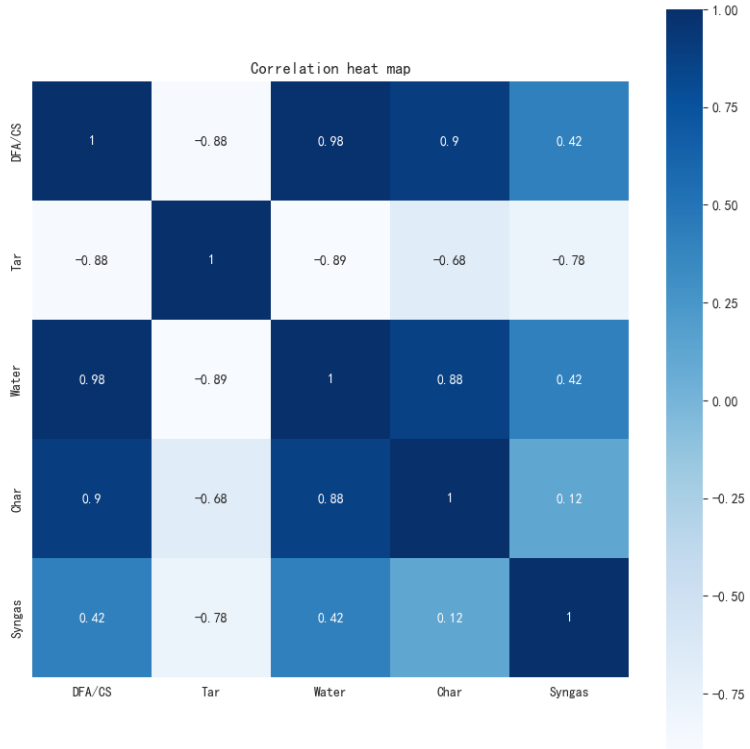


Figure 6 CS Pearson correlation coefficients Heap

For CS, we observed that the Tar, Water, and Char yields have a higher correlation with the blending ratio, while the correlation with Syngas is relatively low. For CE, we found that the Tar,

Water, and Syngas yields have a higher correlation with the blending ratio, while the correlation with Char is relatively low. For LG, we observed that the Tar and Water yields have a higher correlation with the blending ratio, while the correlations with Char and Syngas are relatively low.

We also conducted causal analysis to further investigate whether desulfurized fly ash, as a catalyst, promotes the pyrolysis processes of cotton stalks, cellulose, and lignin. Causal inference is a scientific discipline that aims to estimate counterfactuals based on observational data, analyzing the causal relationships between interventions and outcomes. In causal inference, researchers seek to understand the impact of a particular event or treatment on observed outcomes, going beyond mere description of associations We utilized the Python library DoWhy, released by Microsoft for end-to-end causal inference, to conduct causal analysis. The workflow is as follows:

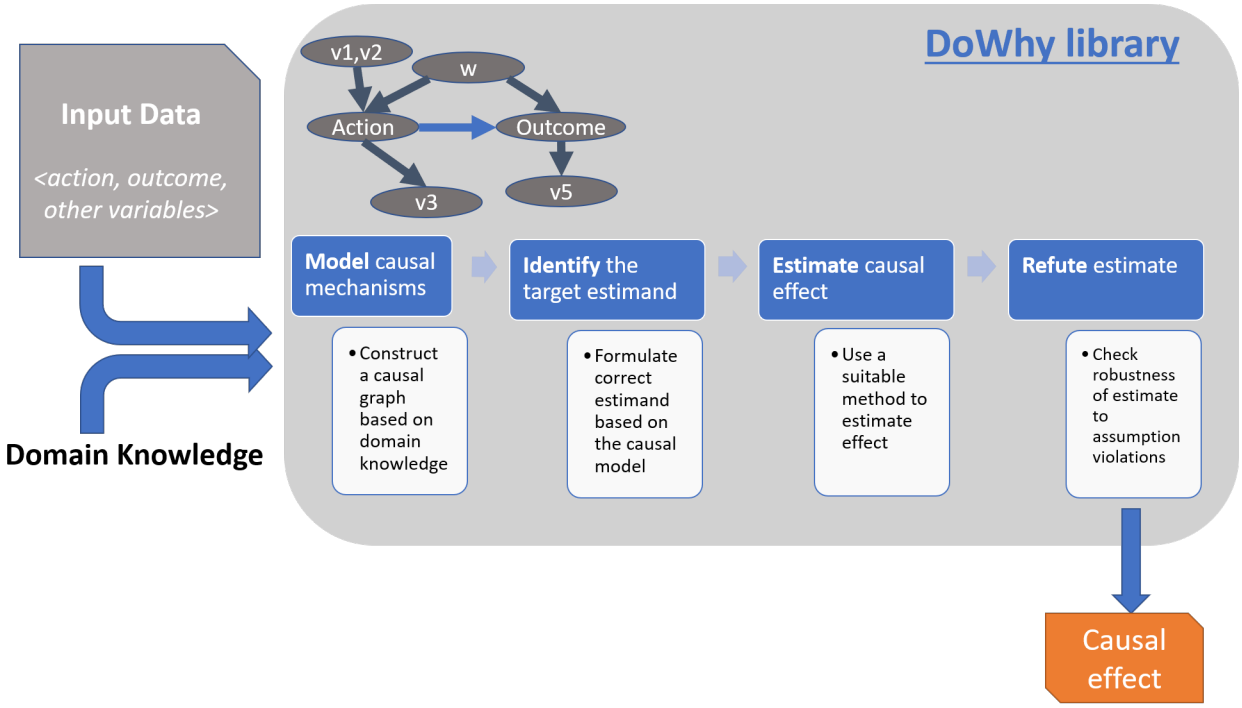


Figure 7 The overall process of DoWhy

We can observe that the blending ratio does have a significant impact on the yields of various products.Additional causal inference diagrams are provided in the appendix.

In conclusion, we have found that desulfurized fly ash, acting as a catalyst, indeed promotes the pyrolysis processes of cotton stalks, cellulose, and lignin.

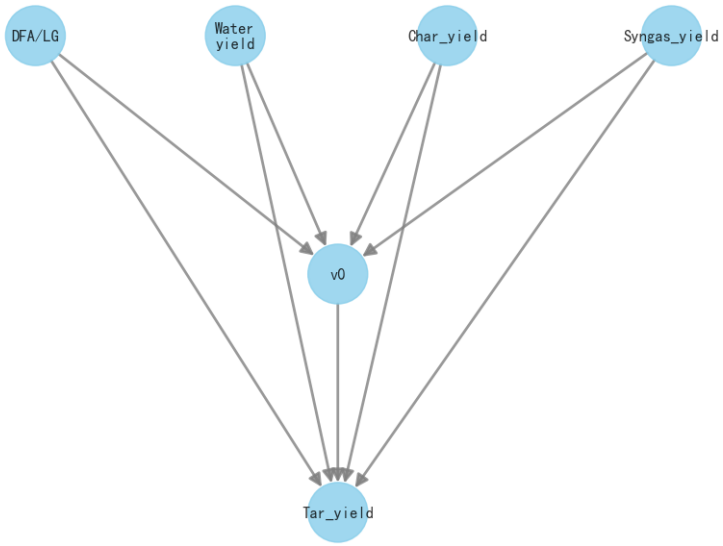


Figure 8 Causal inference of LG

4. Solution of Question2

Data Preprocessing and Visualization

We first conducted data preprocessing and found no missing values. Regarding outliers, considering the complexity of the chemical mechanisms and limitations in data quantity, we did not handle them. Next, we visualized the data using line charts, plotting the yields of various pyrolysis gases. This was done to observe the relationship between the yields of pyrolysis gases and the corresponding blend ratios in the pyrolysis combinations.

We find that as the blending ratio increases:

For CS, we observe a decrease in the content of CO and CO2, a significant increase in H2 content, an initial increase followed by a decrease in CH4 content, and relatively stable content for other gases. For CE, a significant increase in H2 content is noted, along with a significant decrease in CO and CO2 content. For LG, pyrolysis almost does not generate H2 and C2H6, while there is a significant increase in CO and CO2 content.

4.1 Correlation Analysis and Causal Analysis

We first obtained the Pearson correlation coefficients between each pyrolysis product and the blending ratio. Then, we visualized the correlations between variables by generating a correlation heatmap. Figure 8 is only the heatmap for CS, while the correlation images for CE and LG are shown in the appendix.

We observed that:

For CS, the blending ratio shows the highest correlations with H2, C3H8, and CO2, reaching

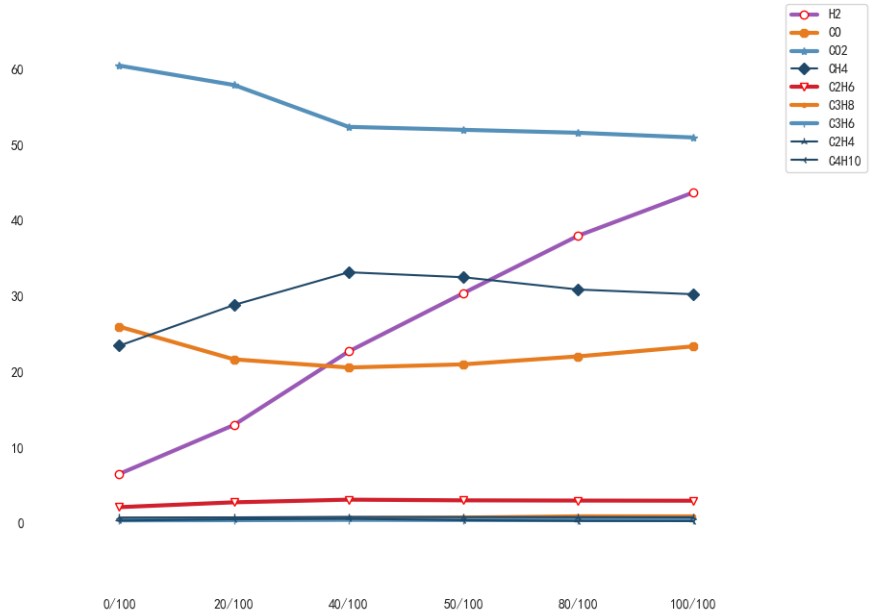


Figure 9 CS gas Plot. The trend charts for other CE and LG are provided in the appendix.

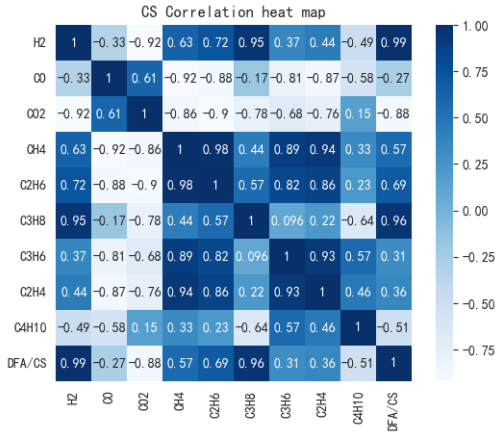


Figure 10 CS gas Heap.

up to 0.99, 0.96, and 0.88, respectively. It exhibits negative correlations with CO, CO2, and C4H10, while the correlation with C3H6 and C2H4 is not strong. For CE, the blending ratio has a strong correlation with the yield of each gas. For LG, the blending ratio has a strong correlation with H2, CO, and CO2, but almost no correlation with CH4 content.

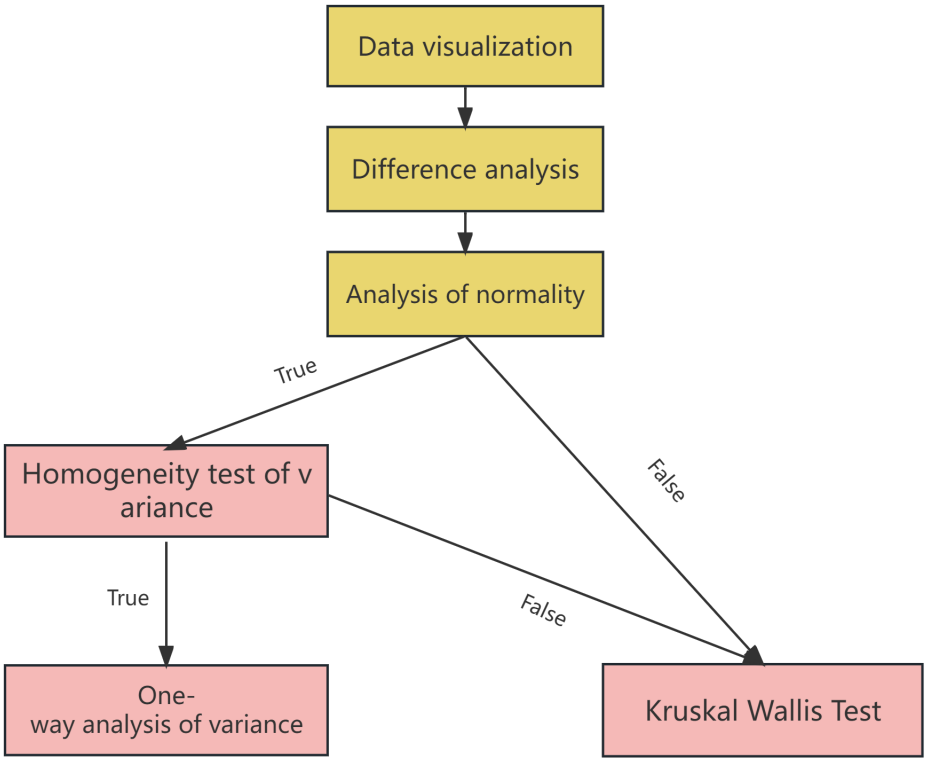


Figure 11 Progress

5. Solution of Question3

5.1 Data Visualization

DFA catalyzes the pyrolysis product distribution and pyrolysis gas distribution of different substances. Preliminary observations suggest that under the catalytic effect of desulfurized fly ash at the same proportion, there may be significant differences in the pyrolysis products and components of pyrolysis gases between CE and LG.

By observing the distribution charts of pyrolysis products and pyrolysis gas components under the catalytic effect of desulfurized fly ash at the same proportion, we found significant differences in the trends and quantities of pyrolysis products for CE and LG model compounds. The variations in the yields of pyrolysis products indicate a notable disparity in content between the two model compounds.

5.2 Analysis of variance

In order to further explore the differences in the content of pyrolysis products and pyrolysis gas components between CE and LG, we conducted a differential analysis. For the differential analysis, we first conducted a normality test and found that the data satisfied normality. Next,

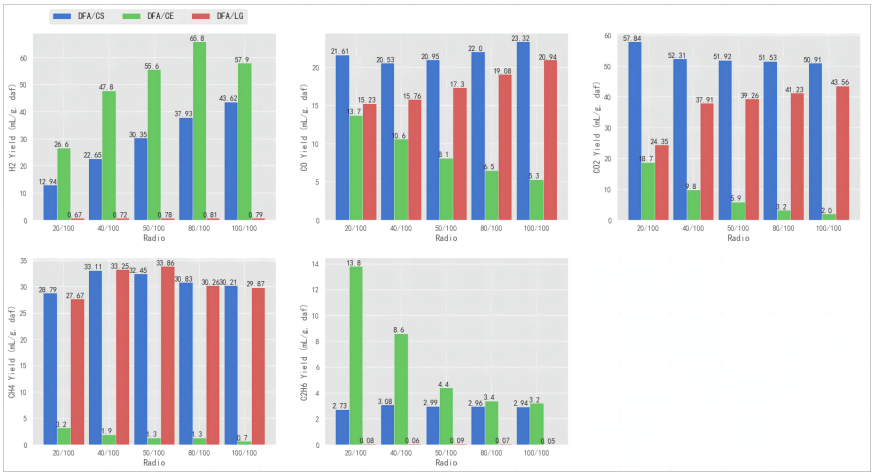


Figure 12 DFA catalyzes the distribution of pyrolysis gases for different substances.

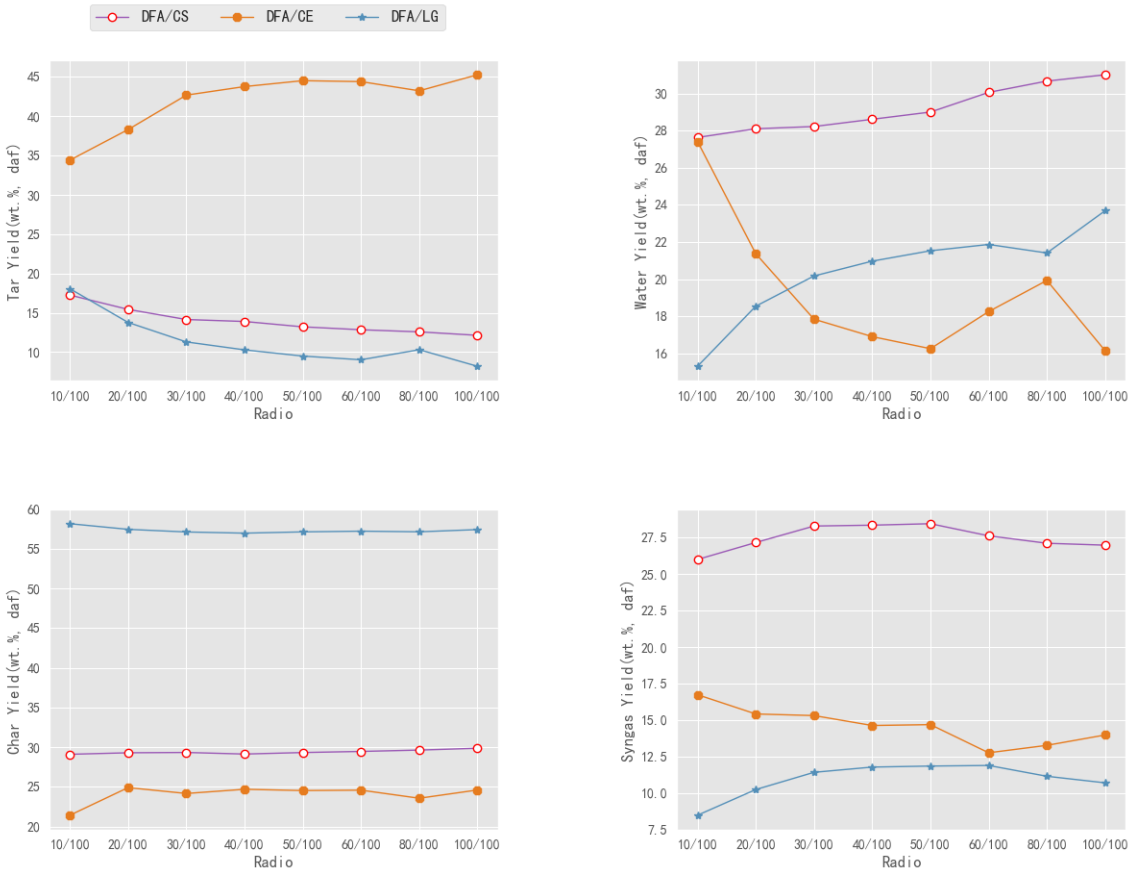


Figure 13 DFA catalyzes the distribution of pyrolysis products for different substances.

we performed a homogeneity of variance test. If there was no significant difference in variance, we used one-way analysis of variance (ANOVA). If variance was not homogeneous, we employed the Kruskal-Wallis test for variance analysis. The homogeneity of variance test results are presented in the table below:

	Levene	P-value	Reject
H_2	7.59184	0.02485	True
CO	0.79633	0.39824	False
CO_2	0.00139	0.97116	False
CH_4	6.03155	0.03957	True
C_2H_6	15.21740	0.00454	True
Tar	0.26936	0.61187	False
Water	0.80436	0.38522	False
Char	3.26165	0.09246	False
Syngas	0.07080	0.79406	False

We observed that there were no significant differences in the homogeneity of variance for H2, CH4, and C2H6. Hence, we conducted one-way analysis of variance (ANOVA) for these variables. However, for CO, CO2, Tar, Water, Char, and Syngas, where variance was not homogeneous, we performed variance analysis using the Kruskal-Wallis test. The results of the variance analysis are as follows:

	P-value	Reject
CO	0.00136	True
CO_2	0.00018	True
Tar	6.00138	True
Water	0.47674	True
Syngas	3.06742	True

	P-value	Reject
H_2	0.0001	True
CH_4	0.0	True
C_2H_6	0.0116	True

We found significant differences in all pyrolysis products and pyrolysis gas components between CE and LG.

6. Solution of Question4

6.1 Catalytic reaction mechanism

We first analyzed the pyrolysis product distribution and pyrolysis gas distribution of different substances catalyzed by DFA in Figures 9 and 10. We observed: Observing the gas distribution chart demonstrates the variations in gas composition during pyrolysis at different blending ratios for DFA/CS, DFA/CE, and DFA/LG. The results indicate that CE and LG in CS have a noticeable impact on the gas composition distribution of DFA/CS pyrolysis, achieved through the modeling of compounds.

As shown in Subfigure 1, DFA/LG almost does not produce any H2, with yields significantly lower than those of DFA/CS pyrolysis. With increasing DFA content, the H2 production in DFA/CE significantly increases, suggesting that DFA promotes the degradation of cellulose during CE pyrolysis. This indicates that DFA accelerates cellulose degradation in the DFA/CS pyrolysis process. In the early stages of LG pyrolysis, aromatic ring-opening reactions and subsequent condensation or cross-linking reactions occur in the residual material, resulting in the generation of a considerable amount of hydrogen. This suggests that DFA significantly inhibits H2 generation in LG.[1]

As shown in Subfigures 2 and 3, the yield sequence of CO and CO2 is DFA/CS > DFA/LG > DFA/CE. When comparing the yields of DFA/CS and DFA/LG, we observe that CO and CO2 are primarily produced from the carbonyl and carboxyl groups in LG of CS.[2]

As shown in Subfigure 4, the production of CH4 during DFA/CS pyrolysis is mainly attributed to the breakage of functional groups at the partially terminal ends and side chains of lignin polymers. This includes the breakage of terminal -OH bonds, -COOH bonds, and phenyl C-C bonds, resulting in the release of small amounts of C1-C3 hydrocarbon gases and a significant amount of oxygen-containing compounds (water, CO, CO2).[3]

The entire pyrolysis process may be in a state of mutual inhibition. This is because the pyrolysis process of lignin is a slow and wide-ranging decomposition process over a broad temperature range, whereas cellulose is just the opposite, undergoing a rapid and intense pyrolysis process. This results in the entire low-temperature pyrolysis zone (300–390 °C), where the rapid decomposition of cellulose requires chain-breaking reactions involving entities such as H radicals. However, some of these initiating groups are occupied by lignin, and activated cellulose does not receive enough chain-breaking initiators. As a result, the pyrolysis reaction is hindered, leading to a significant occurrence of side reactions such as coking, an increase in residual coke, and incomplete reaction.

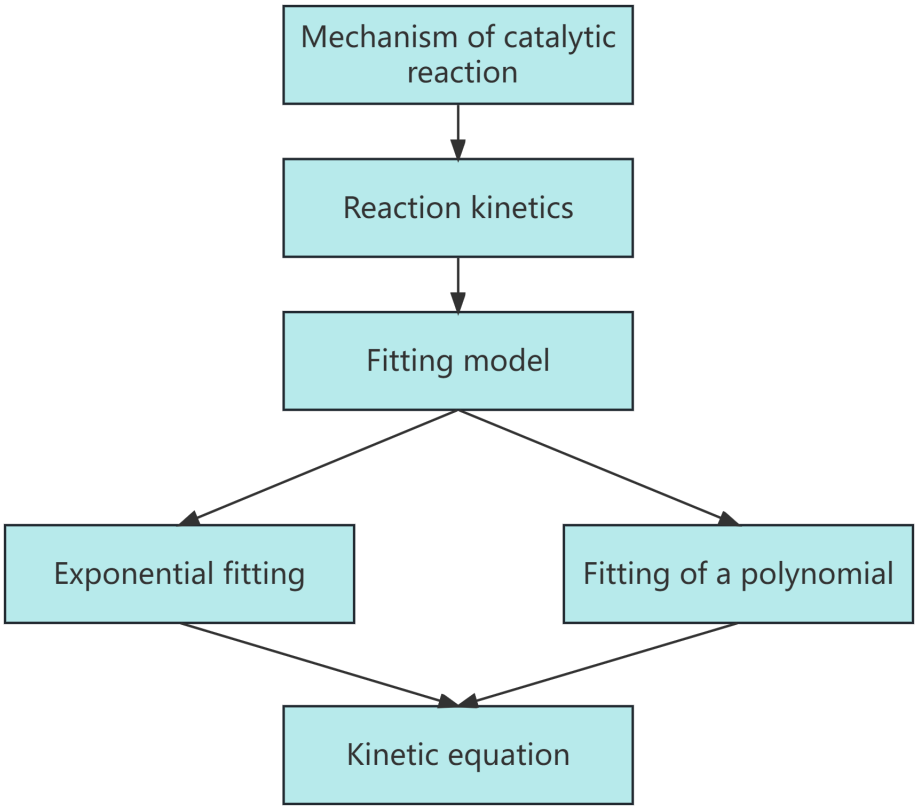


Figure 14 Progress

6.2 Model fitting and establishment

Research indicates that the pyrolysis process of cellulose can be described as a first-order reaction, while the pyrolysis of lignin can be considered as two first-order reactions. Therefore, the co-pyrolysis process of cellulose and lignin can be hypothesized as a combination of first-order reaction kinetics.[4]

Due to the absence of data regarding time and temperature, we are unable to establish a precise reaction rate equation. Therefore, we employed exponential models, first-order and second-order polynomial regression models to compare the fitting effects and determine the reaction kinetics equation.

Through analysis, we found that the quadratic polynomial provides the best fit. We used a quadratic regression model to establish the reaction kinetics equation. The results of the quadratic polynomial fit are shown in the table below, while the linear and exponential fit data can be found in the appendix.

Partial fitting curves and residual plots are shown in the figure below. Other fitting images can be found in the appendix.

The final kinetic fitting equations obtained are in Tabel 4,5:

	Tar	Water	Char	Syngas	H_2	CO	CO_2	CH_4	C_2H_6
RMSE	1.65895	4.43903	0.71294	0.20209	0.51277	0.17291	0.53685	0.06020	0.57408
R-squared	0.86617	0.64088	0.37426	0.85698	0.99713	0.98090	0.98518	0.91657	0.96519

Table 2 CE

	Tar	Water	Char	Syngas	H_2	CO	CO_2	CH_4	C_2H_6
RMSE	1.13232	0.69048	0.02862	0.12903	0.00013	0.08504	3.17526	1.56059	0.00011
R-squared	0.87416	0.87757	0.76475	0.88912	0.95183	0.98099	0.92992	0.70165	0.47449

Table 3 LG

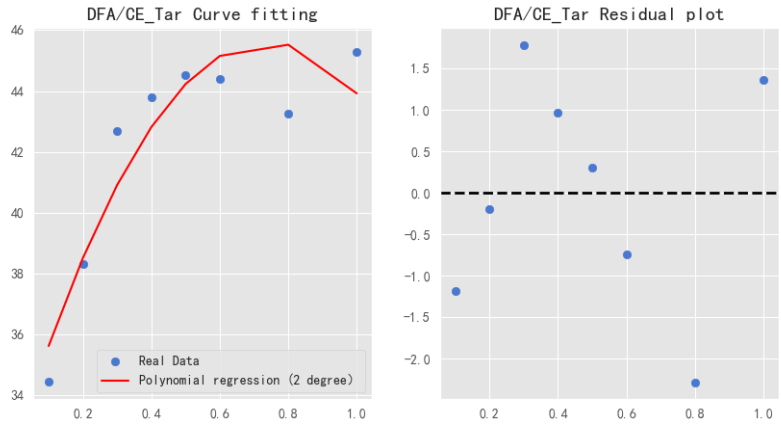


Figure 15 CE Tar

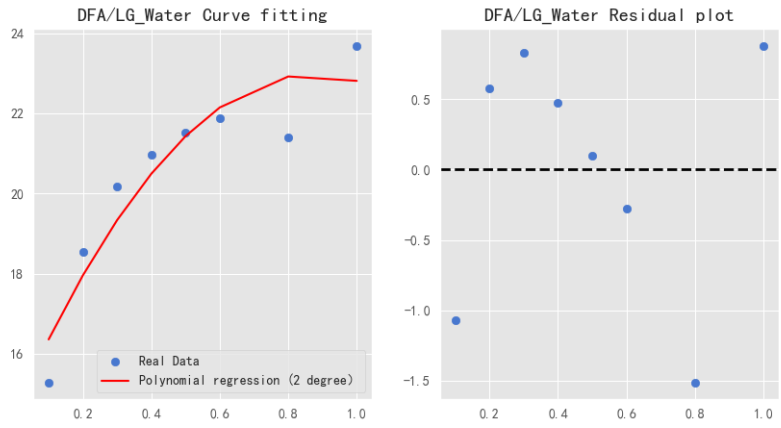


Figure 16 LG Water

CE Fit Equations	
Tar	$y = 36.33039x^2 - 24.63122x + 32.22637$
Water	$y = -33.67811x^2 - 23.97290x + 28.04360$
Char	$y = 8.79220x^2 - 6.77787x + 21.93924$
Syngas	$y = -11.43454x^2 + 7.43619x + 17.79451$
H_2	$y = 182.61394x^2 - 118.50301x - 5.64167$
CO	$y = -24.91156x^2 + 12.19388x - 18.19167$
CO_2	$y = -64.96088x^2 + 37.65306x + 29.85833$
CH_4	$y = -8.24319x^2 + 4.59184x + 4.54167$
$C2H6$	$y = -47.25510x^2 + 28.63945x + 22.11667$

Table 4 CE fitting Tabel

LG Fit Equations	
Tar	$y = -29.40432x^2 + 19.23163x + 19.50452$
Water	$y = 19.26347x^2 - 11.00430x + 14.55167$
Char	$y = -4.34767x^2 + 3.53186x + 58.32996$
Syngas	$y = 14.48852x^2 - 11.75919x + 7.61383$
H_2	$y = 0.60476x^2 - 0.36904x + 0.55750$
CO	$y = 3.82950x^2 + 2.8545x + 14.24667$
CO_2	$y = 78.49268x^2 - 48.06122x + 11.82583$
CH_4	$y = 36.19421x^2 - 29.73639x + 22.41917$
$C2H6$	$y = 0.06377x^2 - 0.076530x + 0.0650$

Table 5 LG fitting Tabel

7. Solution of Question5

We first use the fitted models obtained in the fourth question for prediction.The obtained results are as follows:

Grey Predicting Model

The algorithmic process of the Grey Prediction Model is illustrated in Figure 17. The pre-

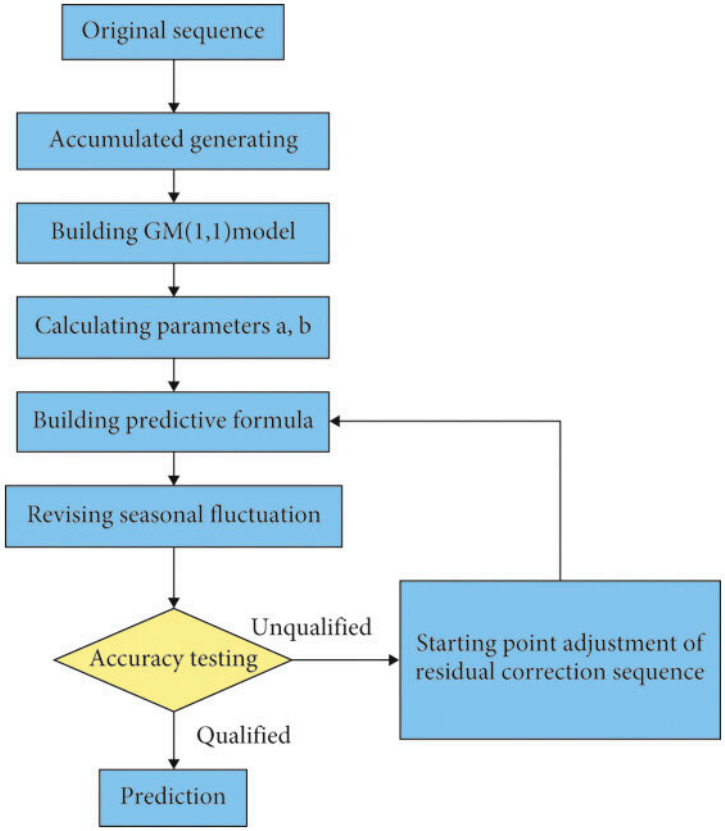


Figure 17 Grey prediction model

diction results of the grey prediction model are shown in Table 6 ,Table 7 and Table 8.

Table 6 CE

	Tar	Water	Char	Syngas
0	34.42	27.42	21.43	16.73
1	38.31	21.37	24.91	15.41
2	42.69	17.84	24.17	15.3
3	43.78	16.9	24.7	14.62
4	44.53	16.25	24.54	14.68
5	44.41	18.25	24.59	12.75
6	45.341	17.461	24.775	12.566
7	45.668	18.153	24.720	11.754
8	46.086	18.773	24.838	10.810
9	46.727	18.740	24.904	10.420
10	47.113	19.413	24.936	9.644
RMSE	0.388	2.013	0.110	2.216

Table 7 LG

	Tar	Water	Char	Syngas
0	18.06	15.3	58.17	8.47
1	13.77	18.54	57.46	10.23
2	11.29	20.17	57.13	11.41
3	10.28	20.97	56.98	11.77
4	9.49	21.53	57.14	11.84
5	9.02	21.87	57.23	11.88
6	8.240	22.580	57.235	12.097
7	7.722	23.062	57.360	12.155
8	7.202	23.619	57.408	12.287
9	6.650	24.248	57.474	12.427
10	6.222	24.802	57.560	12.525
RMSE	0.468	0.628	0.070	1.465

Table 8 CS

	Tar	Water	Char	Syngas
0	19.46	26.84	29.21	24.49
1	17.25	27.64	29.11	26.0
2	15.43	28.11	29.3	27.16
3	14.14	28.23	29.34	28.29
4	13.89	28.62	29.14	28.35
5	13.21	29.01	29.33	28.45
6	12.84	30.07	29.47	27.62
7	12.414	30.496	29.466	27.706
8	11.927	31.258	29.631	27.348
9	11.566	32.039	29.700	26.982
10	11.146	32.672	29.781	26.852
RMSE	0.156	0.184	0.174	0.596

8. Strengths and Weakness

Strengths

* Multiple fitting models were established, and the best-fitting model was selected as the reaction kinetics model, improving the accuracy and generalization ability of the model.
* We searched the literature for a deeper understanding of the reaction mechanism.
* The gray prediction model is used for small sample data, which effectively improves the generalization ability of the model .

Weakness

* We don't deal with outliers in complex chemical reactions, which can have an impact on model accuracy.
* Due to the lack of information such as time and temperature, we did not build the reaction rate equation and the reaction kinetics equation more precisely.

References

[1] Zhongyang Luo, Shurong Wang, and Xiujuan Guo. Selective pyrolysis of organosolv lignin over zeolites with product analysis by tg-ftir. Journal of Analytical and Applied Pyrolysis, 95:112–117, 2012.

[2] Jinbao Huang, Chao Liu, Hong Tong, Weimin Li, and Dan Wu. A density functional theory study on formation mechanism of co, co2 and ch4 in pyrolysis of lignin. Computational and Theoretical Chemistry, 1045:1–9, 2014.

[3] JIN Pen and LI Bao-xia. Experiments and kinetic analysis on the co-py- rolysis of cellulose and lignin. Chemical Industry and Engineer- ing Progress, 32(2):303, 2013.

[4] Ramakrishna Gottipati and Susmita Mishra. A kinetic study on pyrolysis and combustion characteristics of oil cakes: Effect of cellulose and lignin content. Journal of fuel chemistry and technology, 39(4):265–270, 2011.

Appendix

Listing 1: The python Source code of Algorithm

```
import numpy as np
import pandas as pd
pd.set_option('display.float_format', lambda x: '{:.2f}'.format(x))
import seaborn as sns
color = sns.color_palette()
sns.set_style('darkgrid')
import matplotlib.pyplot as plt
plt.rcParams["font.sans-serif"]=["SimHei"] #
plt.rcParams["axes.unicode_minus"]=False #      "" -
%matplotlib inline
from scipy import stats
from scipy.special import boxcox1p
from scipy.stats import norm, skew
import warnings
def ignore_warn(*args, **kwargs):
    pass
warnings.warn = ignore_warn
##
DFA_CS_data = pd.read_excel("../Data/DFA-CS.xlsx")
DFA_CE_data = pd.read_excel("../Data/DFA-CE.xlsx")
DFA_LG_data = pd.read_excel("../Data/DFA-LG.xlsx")
###
fig, ax = plt.subplots()
data1 = DFA_CE_data.loc[0].values[1:]
data2 = DFA_CE_data.loc[1].values[1:]
data3 = DFA_CE_data.loc[2].values[1:]
data4 = DFA_CE_data.loc[3].values[1:]
data = [data1,data2,data3,data4]
ax.boxplot(data)
ax.set_xticklabels(['Tar', 'Water', 'Char', 'Syngas'])
plt.savefig("../Figure /1/DFA_CE_box.png")

fig, ax = plt.subplots()
data1 = DFA_CS_data.loc[0].values[1:]
data2 = DFA_CS_data.loc[1].values[1:]
data3 = DFA_CS_data.loc[2].values[1:]
```

```
data4 = DFA_CS_data.loc[3].values[1:]
data = [data1,data2,data3,data4]
ax.boxplot(data)
ax.set_xticklabels(['Tar', 'Water', 'Char', 'Syngas'])
plt.savefig("../Figure /1/DFA_CS_box.png")
fig, ax = plt.subplots()
data1 = DFA_LG_data.loc[0].values[1:]
data2 = DFA_LG_data.loc[1].values[1:]
data3 = DFA_LG_data.loc[2].values[1:]
data4 = DFA_LG_data.loc[3].values[1:]
data = [data1,data2,data3,data4]
ax.boxplot(data)
ax.set_xticklabels(['Tar', 'Water', 'Char', 'Syngas'])
plt.savefig("../Figure /1/DFA_LG_box.png")
plt.show()

data_index = DFA_LG_data.columns.values[1:]
fig, axs = plt.subplots(nrows=2, ncols=2, figsize=(10, 8))

#
axs[0, 0].scatter(x=data_index, y=DFA_LG_data.loc[0].values[1:])
axs[0, 0].set_title('Tar')

axs[0, 1].scatter(x=data_index, y=DFA_LG_data.loc[1].values[1:])
axs[0, 1].set_title('Water')

axs[1, 0].scatter(x=data_index, y=DFA_LG_data.loc[2].values[1:])
axs[1, 0].set_title('Char')

axs[1, 1].scatter(x=data_index, y=DFA_LG_data.loc[3].values[1:])
axs[1, 1].set_title('Syngas')

#
plt.tight_layout()

#
plt.savefig("../Figure /1/Scatter/DFA_LG_scatter_subplots.png")

#
```

```
plt.show()
#
plt.figure(figsize=(15, 8))
categories = DFA_CE_data.columns.values[1:]
Tar = DFA_CE_data.loc[0].values[1:]
Water = DFA_CE_data.loc[1].values[1:]
Char = DFA_CE_data.loc[2].values[1:]
Syngas = DFA_CE_data.loc[3].values[1:]
#
fig, ax = plt.subplots()

#          label
bar1 = ax.bar(categories, Tar, label='Tar',color="#AB9A6F")

#          bottom
bar2 = ax.bar(categories, Water, bottom=Tar, label='Water',color="#D6E3B7")

#          bottom
bar3 = ax.bar(categories, Char, bottom=[i+j for i, j in zip(Tar,Water)],
    label='Char', color='#95A96A')

bar4 = ax.bar(categories, Syngas, bottom=[i+j+k for i,j,k in
    zip(Tar,Water, Char)], label='Syngas', color="#45602D")
#
ax.legend(loc='upper left', bbox_to_anchor=(1, 1))
#
ax.set_xlabel('DFA/CE')
ax.set_ylabel('Yields(wt.%, daf)')
#
plt.tight_layout()
#
plt.savefig("../Figure/DFA_CE_bar.png")

plt.figure(figsize=(15, 8))
categories = DFA_CS_data.columns.values[1:]
Tar = DFA_CS_data.loc[0].values[1:]
Water = DFA_CS_data.loc[1].values[1:]
Char = DFA_CS_data.loc[2].values[1:]
Syngas = DFA_CS_data.loc[3].values[1:]
```

```
#
fig, ax = plt.subplots()

#          label
bar1 = ax.bar(categories, Tar, label='Tar',color="#AB9A6F")

#          bottom
bar2 = ax.bar(categories, Water, bottom=Tar, label='Water',color="#D6E3B7")

#          bottom
bar3 = ax.bar(categories, Char, bottom=[i+j for i, j in zip(Tar,Water)],
    label='Char', color='#95A96A')

bar4 = ax.bar(categories, Syngas, bottom=[i+j+k for i,j,k in
    zip(Tar,Water, Char)], label='Syngas', color="#45602D")
#
ax.legend(loc='upper left', bbox_to_anchor=(1, 1))
#
ax.set_xlabel('DFA/CE')
ax.set_ylabel('Yields(wt.%, daf)')
#
plt.tight_layout()
#
plt.savefig("../Figure/DFA_CS_bar.png")
plt.figure(figsize=(15, 8))
categories = DFA_LG_data.columns.values[1:]
Tar = DFA_LG_data.loc[0].values[1:]
Water = DFA_LG_data.loc[1].values[1:]
Char = DFA_LG_data.loc[2].values[1:]
Syngas = DFA_LG_data.loc[3].values[1:]
#
fig, ax = plt.subplots()

#          label
bar1 = ax.bar(categories, Tar, label='Tar',color="#AB9A6F")

#          bottom
bar2 = ax.bar(categories, Water, bottom=Tar, label='Water',color="#D6E3B7")
```

```
#          bottom
bar3 = ax.bar(categories, Char, bottom=[i+j for i, j in zip(Tar,Water)],
    label='Char', color='#95A96A')

bar4 = ax.bar(categories, Syngas, bottom=[i+j+k for i,j,k in
    zip(Tar,Water, Char)], label='Syngas', color="#45602D")
#
ax.legend(loc='upper left', bbox_to_anchor=(1, 1))
#
ax.set_xlabel('DFA/CE')
ax.set_ylabel('Yields(wt.%, daf)')
#
plt.tight_layout()
#
plt.savefig("../Figure/DFA_LG_bar.png")
from scipy.stats import levene

#
group1 = DFA_CS_data.loc[0].values[1:]
group2 = DFA_CS_data.loc[1].values[1:]
group3 = DFA_CS_data.loc[2].values[1:]
group4 = DFA_CS_data.loc[3].values[1:]

#
statistic, p_value = levene(group1, group2, group3, group4, center='mean')

#
print("  Levene:", statistic)
print(" p:", p_value)

#             p0 .05
alpha = 0.05
if p_value < alpha:
    print("        ")
else:
    print("           ")
from scipy.stats import levene

#
```

```
group1 = DFA_CE_data.loc[0].values[1:]
group2 = DFA_CE_data.loc[1].values[1:]
group3 = DFA_CE_data.loc[2].values[1:]
group4 = DFA_CE_data.loc[3].values[1:]

#
statistic, p_value = levene(group1, group2, group3, group4, center='mean')

#
print(" Levene:", statistic)
print(" p:", p_value)

#             p0 .05
alpha = 0.05
if p_value < alpha:
    print("        ")
else:
    print("           ")
from scipy.stats import levene

#
group1 = DFA_CS_data.loc[0].values[1:]
group2 = DFA_CS_data.loc[1].values[1:]
group3 = DFA_CS_data.loc[2].values[1:]
group4 = DFA_CS_data.loc[3].values[1:]

#
statistic, p_value = levene(group1, group2, group3, group4, center='mean')

#
print(" Levene:", statistic)
print(" p:", p_value)

#             p0 .05
alpha = 0.05
if p_value < alpha:
    print("        ")
else:
    print("           ")
```

```
group1 = DFA_CS_data.loc[0].values[1:]
group2 = DFA_CS_data.loc[1].values[1:]
group3 = DFA_CS_data.loc[2].values[1:]
group4 = DFA_CS_data.loc[3].values[1:]

from scipy.stats import kruskal

#  Kruskal-  Wallis
statistic, p_value = kruskal(group1, group2, group3, group4)

#
print("Kruskal-  Wallis:", statistic)
print(" p:", p_value)

#             p0 .05
alpha = 0.05
if p_value < alpha:
    print("            ")
else:
    print("          ")


group1 = DFA_CE_data.loc[0].values[1:]
group2 = DFA_CE_data.loc[1].values[1:]
group3 = DFA_CE_data.loc[2].values[1:]
group4 = DFA_CE_data.loc[3].values[1:]

#  Kruskal-  Wallis
statistic, p_value = kruskal(group1, group2, group3, group4)

#
print("Kruskal-  Wallis:", statistic)
print(" p:", p_value)

#             p0 .05
alpha = 0.05
if p_value < alpha:
    print("            ")
else:
```

```
    print("          ")

group1 = DFA_LG_data.loc[0].values[1:]
group2 = DFA_LG_data.loc[1].values[1:]
group3 = DFA_LG_data.loc[2].values[1:]
group4 = DFA_LG_data.loc[3].values[1:]

statistic, p_value = kruskal(group1, group2, group3, group4)

#
print("Kruskal-  Wallis:", statistic)
print(" p:", p_value)

#             p0 .05
alpha = 0.05
if p_value < alpha:
    print("            ")
else:
    print("          ")

group1 = DFA_CS_data.loc[0].values[1:]
group2 = DFA_CS_data.loc[1].values[1:]
group3 = DFA_CS_data.loc[2].values[1:]
group4 = DFA_CS_data.loc[3].values[1:]

df_CS = pd.DataFrame(data={'DFA/CS':[0,0.1,0.2,0.3,0.4,0.5,0.6,0.8,1],
    'Tar':group1,'Water':group2,'Char':group3,'Syngas':group4})
df_CS.corr()
fig, ax = plt.subplots(figsize=(10, 10),facecolor='w')
#
sns.heatmap(df_CS.corr(),annot=True, vmax=1, square=True, cmap="Blues",
    fmt='.2g')
plt.title('Correlation heat map')
plt.savefig("../Figure   /1//CS")
plt.show()
group1 = DFA_CE_data.loc[0].values[1:]
group2 = DFA_CE_data.loc[1].values[1:]
group3 = DFA_CE_data.loc[2].values[1:]
group4 = DFA_CE_data.loc[3].values[1:]
```

```
df_CE = pd.DataFrame(data={'DFA/CE':[0.1,0.2,0.3,0.4,0.5,0.6,0.8,1],
    'Tar':group1,'Water':group2,'Char':group3,'Syngas':group4})
df_CE.corr()
fig, ax = plt.subplots(figsize=(10, 10),facecolor='w')
#
sns.heatmap(df_CE.corr(),annot=True, vmax=1, square=True, cmap="Blues",
    fmt='.2g')
plt.title('Correlation heat map')
plt.savefig("../Figure   /1//CE")
plt.show()
group1 = DFA_LG_data.loc[0].values[1:]
group2 = DFA_LG_data.loc[1].values[1:]
group3 = DFA_LG_data.loc[2].values[1:]
group4 = DFA_LG_data.loc[3].values[1:]

df_LG = pd.DataFrame(data={'DFA/LG':[0.1,0.2,0.3,0.4,0.5,0.6,0.8,1],
    'Tar':group1,'Water':group2,'Char':group3,'Syngas':group4})
df_LG.corr()
fig, ax = plt.subplots(figsize=(10, 10),facecolor='w')
#
sns.heatmap(df_LG.corr(),annot=True, vmax=1, square=True, cmap="Blues",
    fmt='.2g')
plt.title('Correlation heat map')
plt.savefig("../Figure   /1//LG")
plt.show()
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.api as sm
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.preprocessing import PolynomialFeatures

#
#
name = ['Tar', 'Water', 'Char', 'Syngas']

for i,key in enumerate(name):
    x = np.array([0,0.1,0.2,0.3,0.4,0.5,0.6,0.8,1])
```

```
    y = DFA_CS_data.loc[i].values[1:]
    x = x.reshape(-1, 1)
    y = y.reshape(-1, 1)
    #
    degree = 2 #
    poly_features = PolynomialFeatures(degree=degree)
    x_poly = poly_features.fit_transform(x)

    #
    model = LinearRegression()
    model.fit(x_poly, y)

    #
    # x_new = np.linspace(min(x), max(x), 100).reshape(-1, 1)
    # x_new_poly = poly_features.transform(x_new)
    y_pred = model.predict(x_poly)

    mse = mean_squared_error(y, y_pred)
    r2 = r2_score(y, y_pred)

    #
    print(f"   (MSE): {mse}")
    print(f"   (R-squared): {r2}")
    coefficients = model.coef_[0]
    print(f"    : {coefficients}")
    intercept = model.intercept_
    print(f"      (intercept): {intercept}")

    fig, axs = plt.subplots(nrows=1, ncols=2, figsize=(10, 5))
    #
    axs[0].scatter(x, y, label='Real Data')
    axs[0].plot(x, y_pred, color='red', label=f'Polynomial regression
        ({degree}  degree')
    axs[0].set_title('DFA/CS_{} Curve fitting'.format(key))
    axs[0].legend()
    # plt.savefig("../Figure   /1//")
    residuals = y - y_pred
    axs[1].scatter(x, residuals)
    axs[1].axhline(y=0, color='black', linestyle='--', linewidth=2)
```

```
    axs[1].set_title('DFA/CS_{} Residual plot'.format(key))
    plt.savefig("../Figure   /1//DFA_CS/{}.png".format(key))
    plt.show()
    DFA_CS_gas = pd.read_excel("../Data/DFA_CS_gas.xlsx")
    DFA_CE_gas = pd.read_excel("../Data/DFA_CE_gas.xlsx")
    DFA_LG_gas = pd.read_excel("../Data/DFA_LG_gas.xlsx")
    DFA_CE_gas.set_index(DFA_CE_gas['DFA/CE'], inplace=True)
    DFA_CE_gas.drop(columns='DFA/CE', inplace=True)
    DFA_CS_gas.set_index(DFA_CS_gas['DFA/CS'], inplace=True)
    DFA_CS_gas.drop(columns='DFA/CS', inplace=True)
    DFA_LG_gas.set_index(DFA_LG_gas['DFA/LG'], inplace=True)
    DFA_LG_gas.drop(columns='DFA/LG', inplace=True)
    sns.heatmap(DFA_LG_gas.corr(),annot=True, vmax=1, square=True,
        cmap="Blues", fmt='.2g')
    plt.title('CE Correlation heat map')
    #plt.savefig("../Figure   /1//CS")
    plt.show()
    fig, ax = plt.subplots()
    ax.boxplot(DFA_CS_gas)
    ax.set_xticklabels(['H2', 'CO', 'CO2', 'CH4', 'C2H6', 'C3H8', 'C3H6',
        'C2H4', 'C4H10'])
    plt.savefig("../Figure /2/DFA_CS_gas_box.png")
    plt.show()

    fig, ax = plt.subplots()
    ax.boxplot(DFA_CE_gas)
    ax.set_xticklabels(['H2', 'CO', 'CO2', 'CH4', 'C2H6'])
    plt.savefig("../Figure /2/DFA_CE_gas_box.png")
    plt.show()

    fig, ax = plt.subplots()
    ax.boxplot(DFA_LG_gas)
    ax.set_xticklabels(['H2', 'CO', 'CO2', 'CH4', 'C2H6'])
    plt.savefig("../Figure /2/DFA_LG_gas_box.png")
    plt.show()

    x1 = DFA_LG_gas.index.values
    x = np.arange(5)
    width = 0.3
```

```
f, axs = plt.subplots(2, 3,figsize=(20, 10))
bars1 = axs[0,0].bar(x - width, DFA_CS_gas['H2'], width, label='DFA/CS')
bars2 = axs[0,0].bar(x, DFA_CE_gas['H2'], width, label='DFA/CE')
bars3 = axs[0,0].bar(x + width, DFA_LG_gas['H2'], width, label='DFA/LG')

def add_labels(bars):
    for bar in bars:
        yval = bar.get_height()
        axs[0,0].text(bar.get_x() + bar.get_width()/3, yval, round(yval,
            2), ha='center', va='bottom')

add_labels(bars1)
add_labels(bars2)
add_labels(bars3)
axs[0,0].set_xticks(x, x1)
axs[0,0].set_xlabel("Radio")
axs[0,0].set_ylabel("H2 Yield (mL/g, daf)")

bars1 = axs[0,1].bar(x - width, DFA_CS_gas['CO'], width, label='1')
bars2 = axs[0,1].bar(x, DFA_CE_gas['CO'], width, label='2')
bars3 = axs[0,1].bar(x + width, DFA_LG_gas['CO'], width, label='3')

def add_labels(bars):
    for bar in bars:
        yval = bar.get_height()
        axs[0,1].text(bar.get_x() + bar.get_width()/3, yval, round(yval,
            2), ha='center', va='bottom')

add_labels(bars1)
add_labels(bars2)
add_labels(bars3)

axs[0,1].set_xticks(x, x1)
axs[0,1].set_xlabel("Radio")
axs[0,1].set_ylabel("CO Yield (mL/g, daf)")


bars1 = axs[0,2].bar(x - width, DFA_CS_gas['CO2'], width, label='1')
```

```
bars2 = axs[0,2].bar(x, DFA_CE_gas['CO2'], width, label='2')
bars3 = axs[0,2].bar(x + width, DFA_LG_gas['CO2'], width, label='3')

def add_labels(bars):
    for bar in bars:
        yval = bar.get_height()
        axs[0,2].text(bar.get_x() + bar.get_width()/3, yval, round(yval,
            2), ha='center', va='bottom')

add_labels(bars1)
add_labels(bars2)
add_labels(bars3)

axs[0,2].set_xticks(x, x1)
axs[0,2].set_xlabel("Radio")
axs[0,2].set_ylabel("CO2 Yield (mL/g, daf)")


bars1 = axs[1,0].bar(x - width, DFA_CS_gas['CH4'], width, label='1')
bars2 = axs[1,0].bar(x, DFA_CE_gas['CH4'], width, label='2')
bars3 = axs[1,0].bar(x + width, DFA_LG_gas['CH4'], width, label='3')

def add_labels(bars):
    for bar in bars:
        yval = bar.get_height()
        axs[1,0].text(bar.get_x() + bar.get_width()/3, yval, round(yval,
            2), ha='center', va='bottom')

add_labels(bars1)
add_labels(bars2)
add_labels(bars3)

axs[1,0].set_xticks(x, x1)
axs[1,0].set_xlabel("Radio")
axs[1,0].set_ylabel("CH4 Yield (mL/g, daf)")


bars1 = axs[1,1].bar(x - width, DFA_CS_gas['C2H6'], width,
    label='DFA/CS')
```

```
bars2 = axs[1,1].bar(x, DFA_CE_gas['C2H6'], width, label='DFA/CE')
bars3 = axs[1,1].bar(x + width, DFA_LG_gas['C2H6'], width,
    label='DFA/LG')

def add_labels(bars):
    for bar in bars:
        yval = bar.get_height()
        axs[1,1].text(bar.get_x() + bar.get_width()/3, yval, round(yval,
            2), ha='center', va='bottom')

add_labels(bars1)
add_labels(bars2)
add_labels(bars3)
axs[1,1].set_xlabel("Radio")
axs[1,1].set_ylabel("C2H6 Yield (mL/g, daf)")

axs[1,1].set_xticks(x, x1)
axs[1,2] = 0
axs[1,1].legend(loc='best', bbox_to_anchor=(0, 2.35, -0.3,
    0),ncol=4,fontsize='large')
#plt.savefig("../Figure /3/Production distribution/gas")
plt.show()

class GrayForecast():
def __init__(self, data, datacolumn=None):

    if isinstance(data, pd.core.frame.DataFrame):
        self.data=data
        try:
            self.data.columns = [' ']
        except:
            if not datacolumn:
                raise Exception('    dataframe')
            else:
                self.data = pd.DataFrame(data[datacolumn])
                self.data.columns=[' ']
    elif isinstance(data, pd.core.series.Series):
        self.data = pd.DataFrame(data, columns=[' '])
    else:
```

```
            self.data = pd.DataFrame(data, columns=[' '])

        self.forecast_list = self.data.copy()

        if datacolumn:
            self.datacolumn = datacolumn
        else:
            self.datacolumn = None
        #save arg:
        #       data              DataFrame
        #       forecast_list     DataFrame
        #       datacolumn        string
    def level_check(self):
        #
        n = len(self.data)
        lambda_k = np.zeros(n-1)
        for i in range(n-1):
            lambda_k[i] = self.data.ix[i][" "]/self.data.ix[i+1][" "]
            if lambda_k[i] < np.exp(-2/(n+1)) or lambda_k[i] >
                np.exp(2/(n+2)):
                flag = False
        else:
            flag = True

        self.lambda_k = lambda_k

        if not flag:
            print("     X(0)   ")
            return False
        else:
            print("      ")
            return True

    #save arg:
    #       lambda_k          1-d list


    def GM_11_build_model(self, forecast=5):
        if forecast > len(self.data):
```

```
            raise Exception('    ')
        X_0 = np.array(self.forecast_list[' '].tail(forecast))
    #       1-AGO
        X_1 = np.zeros(X_0.shape)
        for i in range(X_0.shape[0]):
            X_1[i] = np.sum(X_0[0:i+1])
    #
        Z_1 = np.zeros(X_1.shape[0]-1)
        for i in range(1, X_1.shape[0]):
            Z_1[i-1] = -0.5*(X_1[i]+X_1[i-1])

        B = np.append(np.array(np.mat(Z_1).T),
            np.ones(Z_1.shape).reshape((Z_1.shape[0], 1)), axis=1)
        Yn = X_0[1:].reshape((X_0[1:].shape[0], 1))

        B = np.mat(B)
        Yn = np.mat(Yn)
        a_ = (B.T*B)**-1 * B.T * Yn

        a, b = np.array(a_.T)[0]

        X_ = np.zeros(X_0.shape[0])
        def f(k):
            return (X_0[0]-b/a)*(1-np.exp(a))*np.exp(-a*(k))

        self.forecast_list.loc[len(self.forecast_list)] = f(X_.shape[0])

    def forecast(self, time=5, forecast_data_len=5):
        for i in range(time):
            self.GM_11_build_model(forecast=forecast_data_len)

    def log(self):
        res = self.forecast_list.copy()
        if self.datacolumn:
            res.columns = [self.datacolumn]
        return res
    def reset(self):
        self.forecast_list = self.data.copy()
```

```
    def plot(self, data):
        plt.scatter([2,4,5,8], data, label='Original Data', color='red')
        plt.plot(range(0,len(self.forecast_list)),self.forecast_list,
            label='Forecast Data')
        if self.datacolumn:
            plt.ylabel(self.datacolumn)
            plt.legend()
```

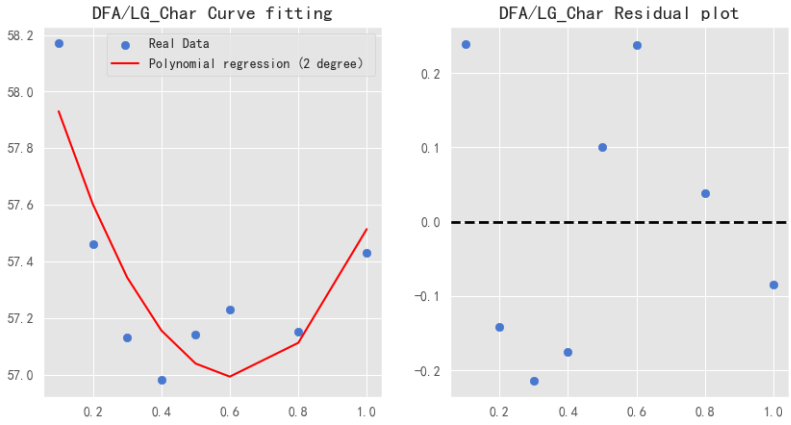


Figure 18

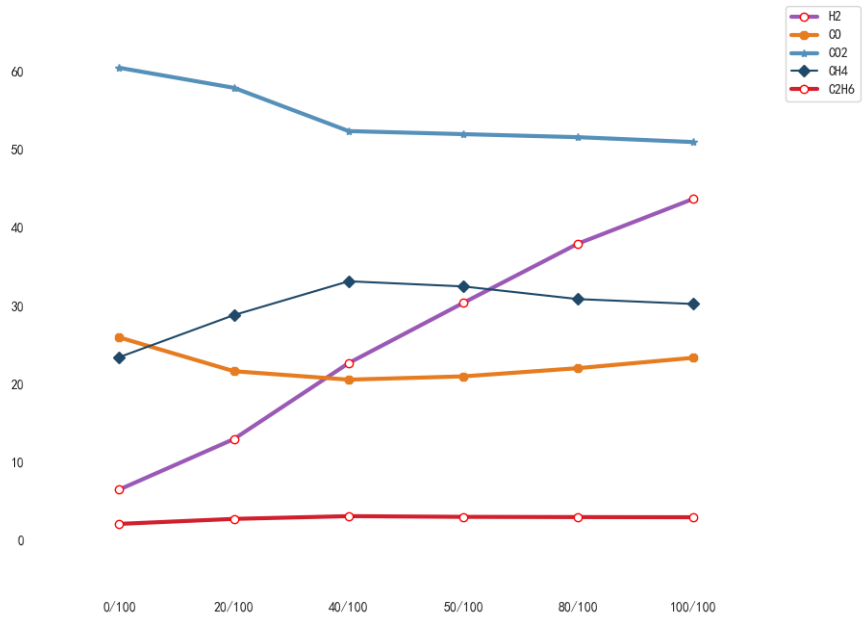


Figure 19

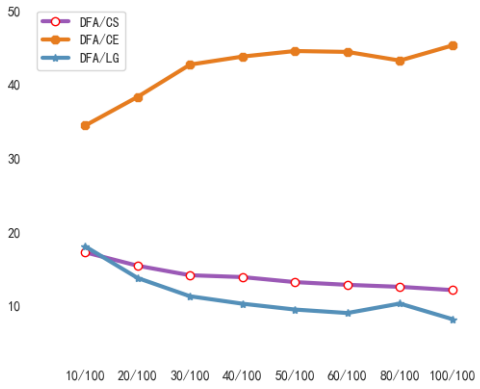


Figure 20

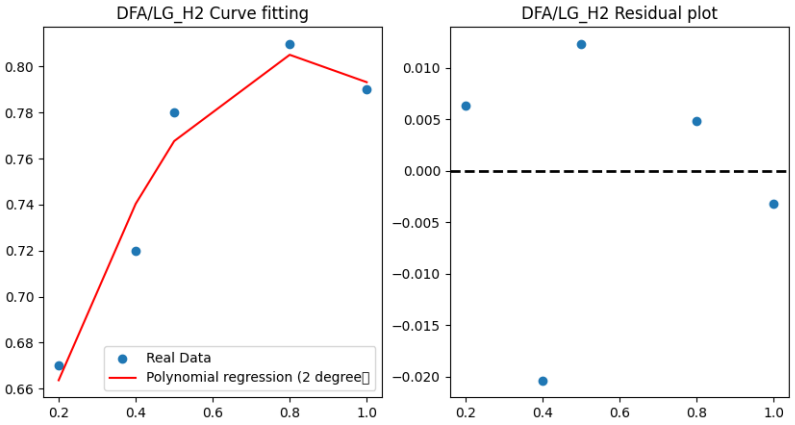


Figure 21

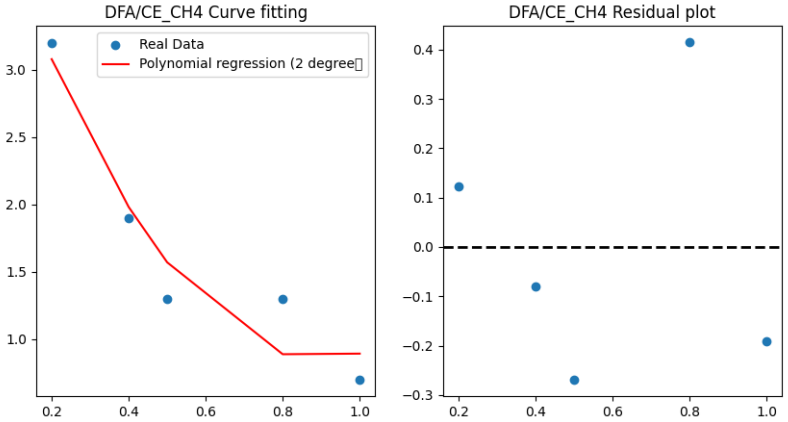


Figure 22

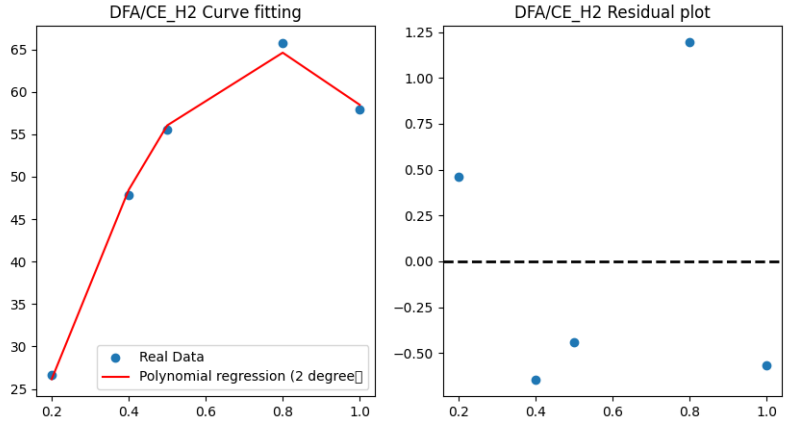


Figure 23

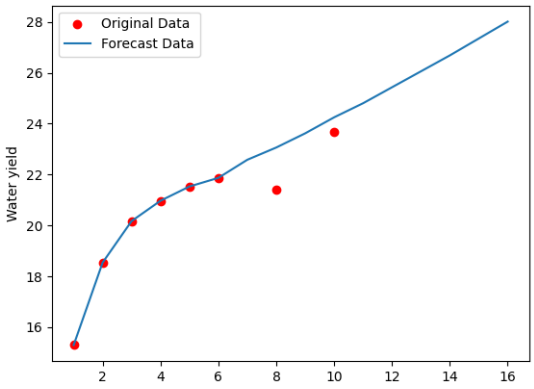


Figure 24

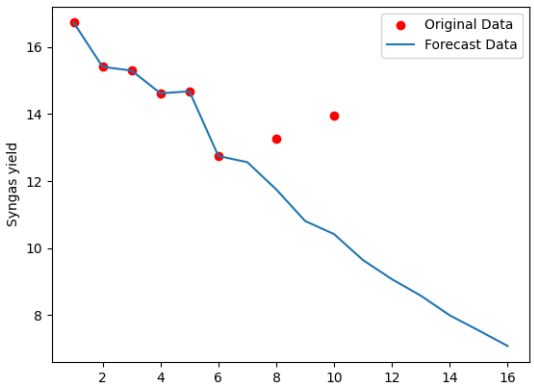


Figure 25