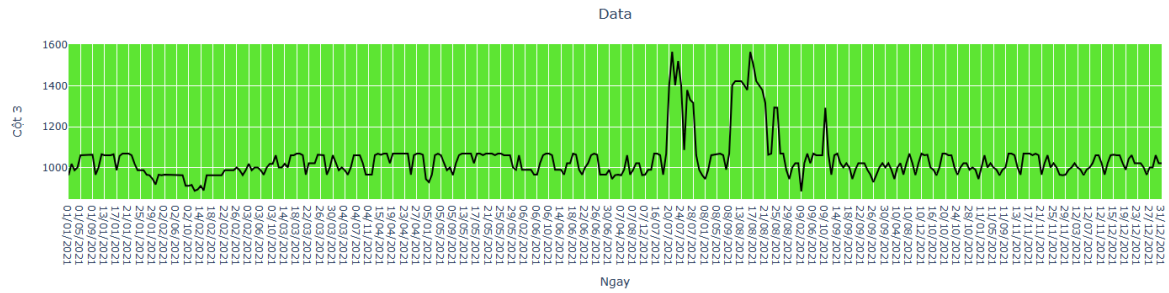
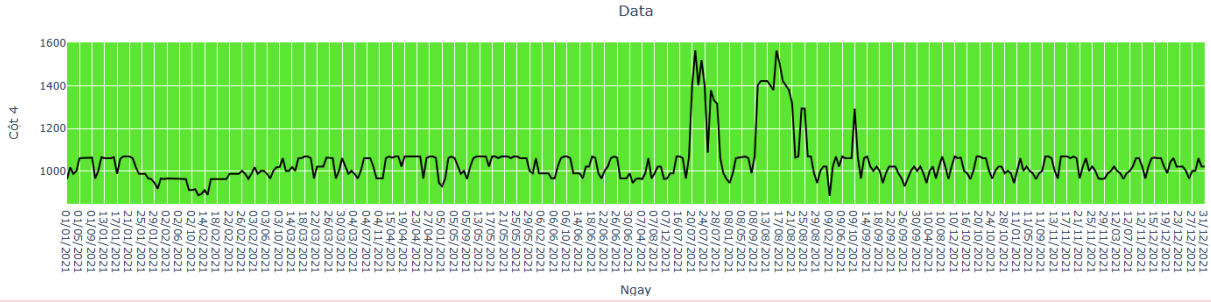
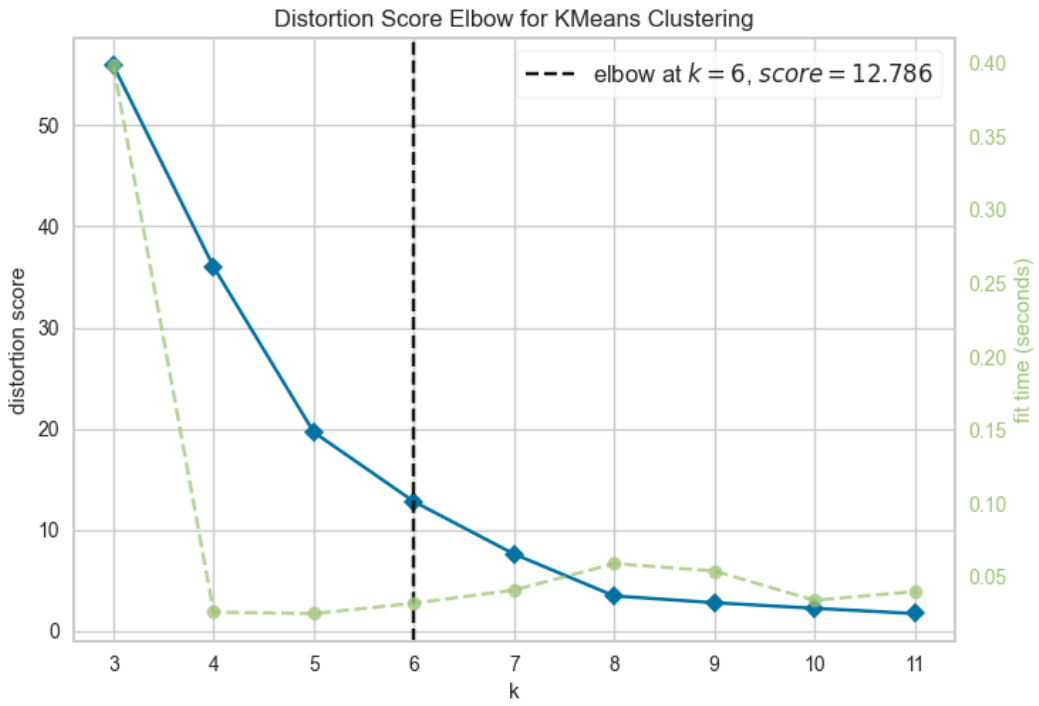
```
import plotly.express as px
fig = px.line(y=df['3'], x=df['Ngay'])
fig.update_traces(line_color='black')
fig.update_layout(xaxis_title="Ngay",
                  yaxis_title="Cột 3",
                  title={'text': "Data", 'y':0.95, 'x':0.5, 'xanchor':'center', 'yanchor':'top'},
                  plot_bgcolor='rgba(53,223,0,0.8)')
```



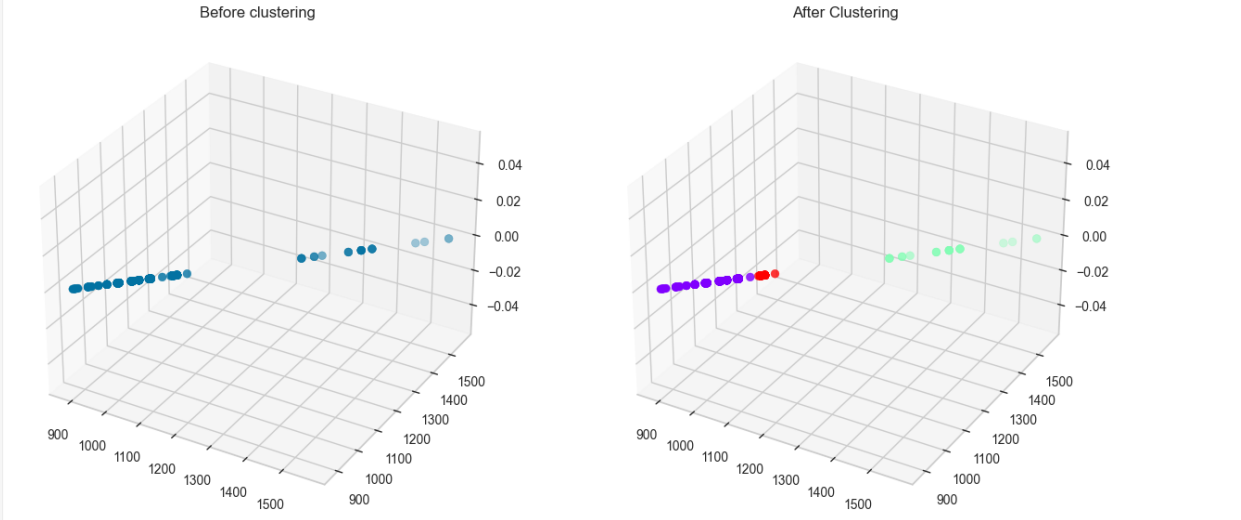

```
fig = px.line(y=df['4'], x=df['Ngay'])
fig.update_traces(line_color='black')
fig.update_layout(xaxis_title="Ngay",
                  yaxis_title="Cột 4",
                  title={'text': "Data", 'y':0.95, 'x':0.5, 'xanchor':'center', 'yanchor':'top'},
                  plot_bgcolor='rgba(53,223,0,0.8)')
```



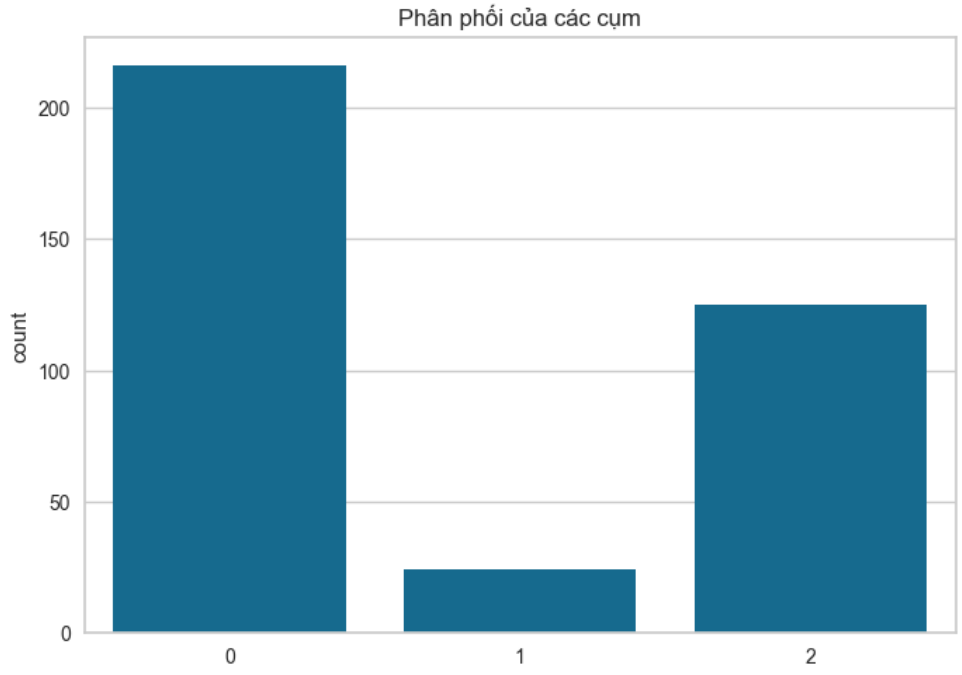


```
fig = plt.figure(figsize=(15,8))
ax = plt.subplot(1,2,1, projection='3d', label="bla")
ax.scatter(df['3'], df['4'], s=40, marker='o', cmap = 'rainbow' )
ax.set_title("Before clustering")
ax = plt.subplot(1,2,2, projection='3d', label="bla")
ax.scatter(df['3'], df['4'], s=40, c=df["Clusters"], marker='o',cmap="rainbow")
ax.set_title("After Clustering")
plt.show()
```

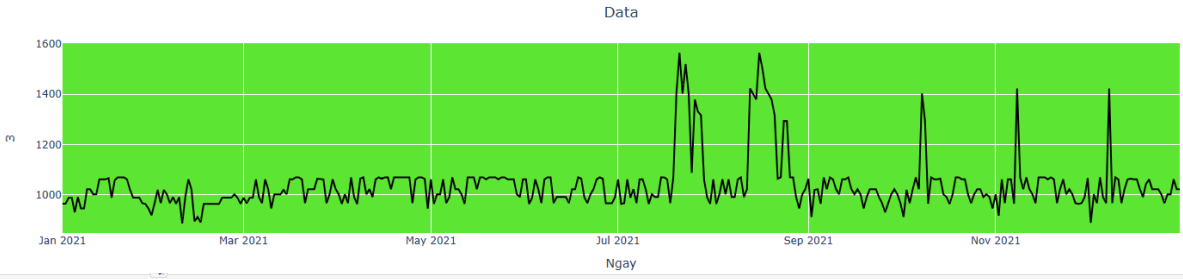
```
C:\Users\WELCOME\AppData\Local\Temp\ipykernel_20576\2380418981.py:3: UserWarning:

No data for colormapping provided via 'c'. Parameters 'cmap' will be ignored
```



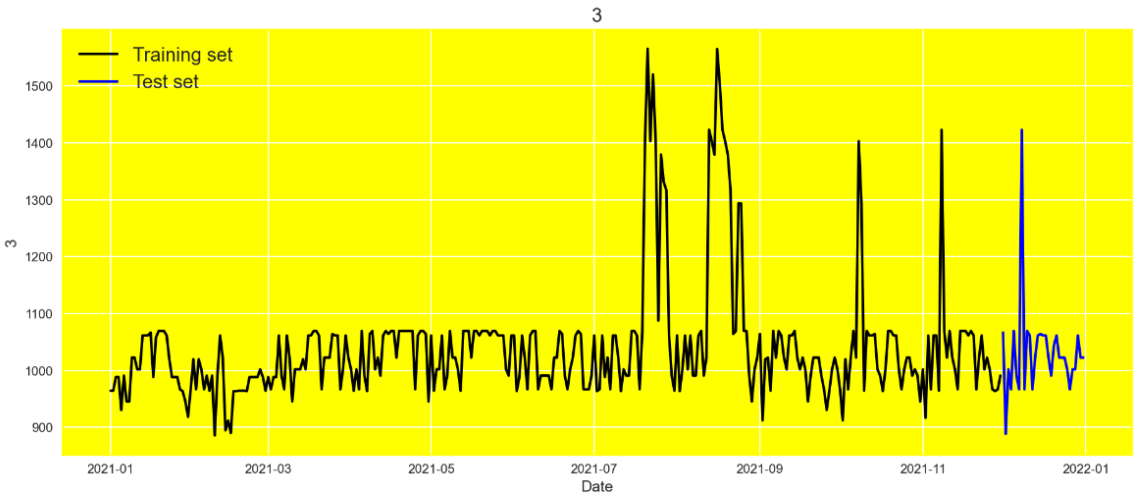

```
pl = sns.countplot(x=np.argmax(lgb_preds,axis=1))
pl.set_title("Phân phối của các cụm")
plt.show()
```



```
import plotly.express as px
fig = px.line(y=df['3'], x=df['Ngay'])
fig.update_traces(line_color='black')
fig.update_layout(xaxis_title="Ngay",
                  yaxis_title="3",
                  title={'text': "Data", 'y':0.95, 'x':0.5, 'xanchor':'center', 'yanchor':'top'},
                  plot_bgcolor='rgba(53,223,0,0.8)')
```




```
plt.figure(figsize=(15, 6), dpi=150)
plt.rcParams['axes.facecolor'] = 'yellow'
plt.rc('axes',edgecolor='yellow')
plt.plot(df['Ngay'][:-test_size], df['3'][:-test_size], color='black', lw=2)
plt.plot(df['Ngay'][-test_size:], df['3'][-test_size:], color='blue', lw=2)
plt.title('3', fontsize=15)
plt.xlabel('Date', fontsize=12)
plt.ylabel('3', fontsize=12)
plt.legend(['Training set', 'Test set'], loc='upper left', prop={'size': 15})
plt.grid(color='white')
plt.show()
```



```
plt.figure(figsize=(10, 8), dpi=150)
plt.rcParams['axes.facecolor'] = 'yellow'
plt.rc('axes',edgecolor='yellow')
plt.plot(df['Ngay'].iloc[:-test_size], scaler.inverse_transform(train_data), color='black', lw=2)
plt.plot(df['Ngay'].iloc[-test_size:], y_test_true, color='blue', lw=2)
plt.plot(df['Ngay'].iloc[-test_size:], y_test_pred, color='red', lw=2)
plt.title('Prediction', fontsize=15)
plt.xlabel('Date', fontsize=12)
plt.ylabel('3', fontsize=12)
plt.legend(['Training', 'Data thực tế', 'Data dự đoán'], loc='upper left', prop={'size': 15})
plt.grid(color='white')
plt.show()
```