

1. Kịch bản

2 Ảnh training

```
df=pd.read_csv("GiaSMPvaSMPcap2021_GiathitruongSMP.csv", encoding='ISO-8859-1',sep =';')
df.head()
```

```
new_df= pd.concat([df['Ngày'],df['12']],axis=1)
new_df.head(10)
```

```
[4] feats = ['12']
```

```
[5] from sklearn.preprocessing import PowerTransformer
X = df['12'].values.reshape(-1,1)

transform = PowerTransformer()
X_transformed = transform.fit_transform(X)
```

```
BGM = BayesianGaussianMixture(n_components=7,covariance_type='full',random_state=1,n_init=15)
# mô hình phù hợp và dự đoán cụm
preds = BGM.fit_predict(X)

# thêm tính năng cụm vào khung dữ liệu gốc
df["Clusters"]= preds
```

```
[7] pp=BGM.predict_proba(X)
# Calculating the probabilities of each prediction
df_new=pd.DataFrame(X,columns=feats)
df_new[[f'predict_proba_{i}' for i in range(7)]] = pp # creating new dataframe columns of probabilities
df_new['preds']=preds
df_new['predict_proba']=np.max(pp,axis=1)
df_new['predict']=np.argmax(pp,axis=1)

train_index=np.array([])
for n in range(7):
    n_inx=df_new[(df_new.preds==n) & (df_new.predict_proba > 0.68)].index
    train_index = np.concatenate((train_index, n_inx))
```

```
[9] lgb_preds=0
    for model in model_list:
        lgb_preds+=model.predict(df_new[feats])

[10] labels=np.argmax(lgb_preds,axis=1)

[11] import numpy as np
    # Tính chuỗi return
    r_t = np.log(new_df['12'] / new_df['12'].shift(1)).values

[12] mean = np.nanmean(r_t)
    r_t[0]=mean
    r_t[:5]

array([ 0.00016098,  0.0557576 , -0.03117627,  0.0224099 ,  0.04894091])

[13] from statsmodels.tsa.stattools import adfuller
    result = adfuller(r_t)
    print('ADF Statistic: %f' % result[0])
    print('p-value: %f' % result[1])
    print('Critical Values:')
    for key, value in result[4].items():
        print('\t%s: %.3f' % (key, value))
```

```
[8] from sklearn.model_selection import StratifiedKFold
    X_new=df_new.loc[train_index][feats]
    y=df_new.loc[train_index]['preds']
    S
    params_lgb = {'learning_rate': 0.06,'objective': 'multiclass','boosting': 'gbdt','n_jobs': -1,'verbosity': -1, 'num_classes':7}

    model_list=[]

    gkf = StratifiedKFold(12)
    for fold, (train_idx, valid_idx) in enumerate(gkf.split(X_new,y)):

        tr_dataset = lgb.Dataset(X_new.iloc[train_idx],y.iloc[train_idx],feature_name = feats)
        vl_dataset = lgb.Dataset(X_new.iloc[valid_idx],y.iloc[valid_idx],feature_name = feats)

        model = lgb.train(params = params_lgb,
            train_set = tr_dataset,
            valid_sets = vl_dataset,
            num_boost_round = 5000,
            callbacks=[ lgb.early_stopping(stopping_rounds=300, verbose=False), lgb.log_evaluation(period=200)])

        model_list.append(model)
```

```
[14] from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
    import matplotlib.pyplot as plt
    plt.figure(figsize = (8, 6))
    ax1 = plot_acf(r_t,color='orange')
```

```
[15] from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
    import matplotlib.pyplot as plt
    plt.figure(figsize = (8, 6))
    ax2 = plot_pacf(r_t)
```

```
from statsmodels.tsa.arima.model import ARIMA

# Khởi tạo và phù hợp với mô hình ARIMA
model_arima = ARIMA(r_t, order=(2, 0, 2))
model_fit = model_arima.fit()

# In ra tóm tắt của mô hình
print(model_fit.summary())

df['Ngày'] = pd.to_datetime(df['Ngày'], format='%d/%m/%Y')
df.sort_values(by='Ngày', ascending=True, inplace=True)
df.reset_index(drop=True, inplace=True)

test_size = df[df['Ngày'].dt.month==12].shape[0]
test_size

31

plt.figure(figsize=(15, 6), dpi=150)
plt.rcParams['axes.facecolor'] = 'gray'
plt.rc('axes', edgecolor='white')
plt.plot(df['Ngày'][:-test_size], df['10'][:-test_size], color='red', lw=2)
plt.plot(df['Ngày'][-test_size:], df['10'][-test_size:], color='pink', lw=2)
plt.title('10', fontsize=15)
plt.xlabel('Date', fontsize=12)
plt.ylabel('10', fontsize=12)
plt.legend(['Training set', 'Test set'], loc='upper left', prop={'size': 15})
plt.grid(color='white')
plt.show()
```

3. Ảnh kết quả



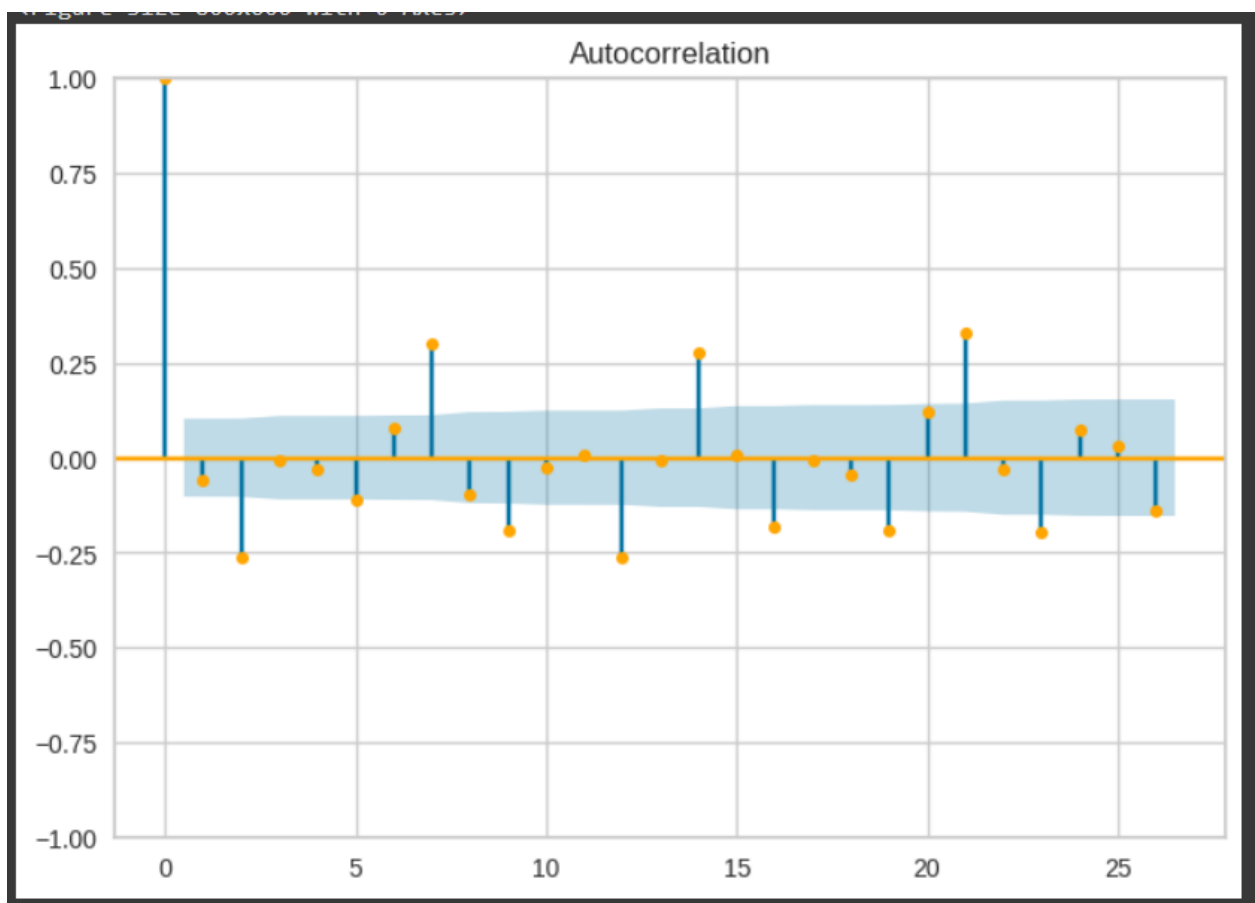
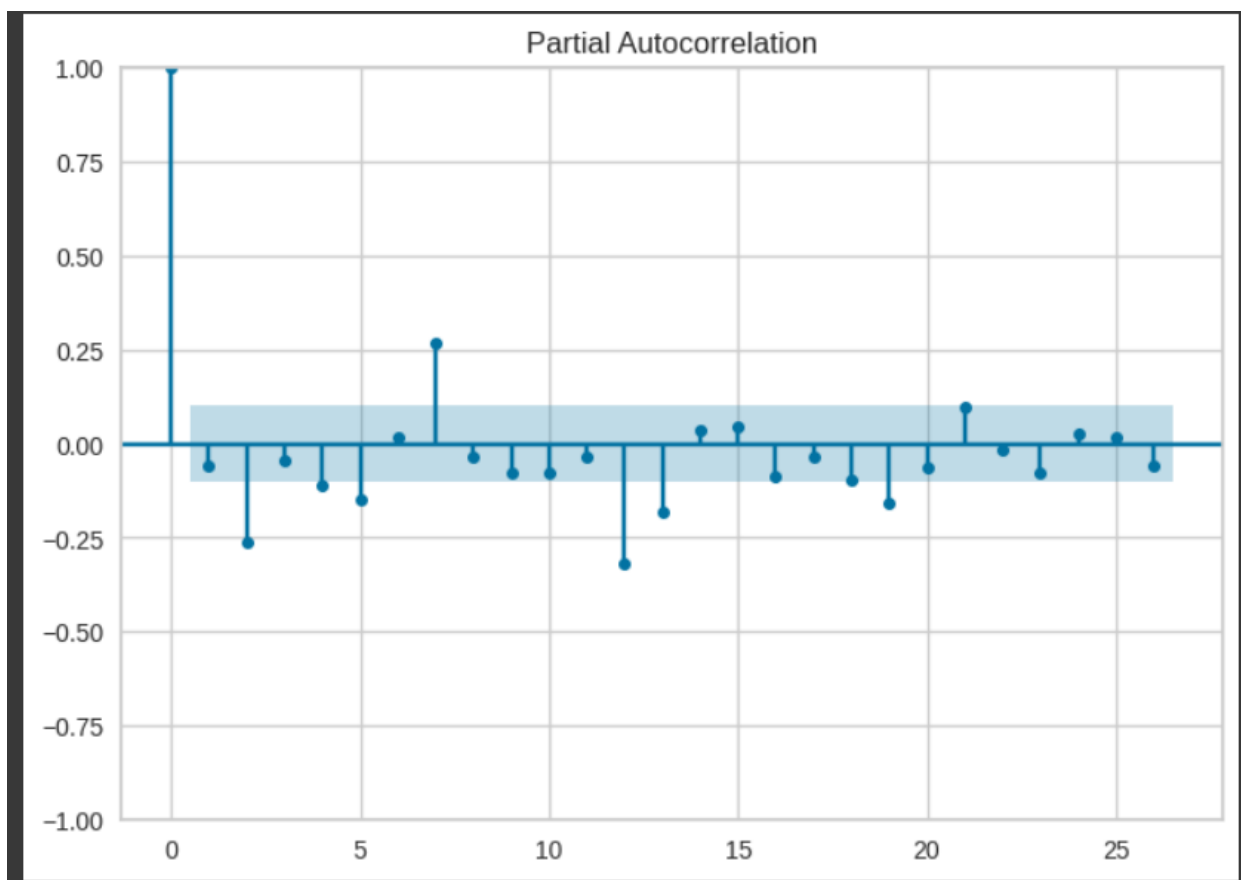
	Ngày	12	
0	01/01/2021	964.4	
1	01/02/2021	1019.7	
2	01/03/2021	988.4	
3	01/04/2021	1010.8	
4	01/05/2021	1061.5	
5	01/06/2021	1061.5	
6	01/07/2021	1061.5	
7	01/08/2021	1061.5	
8	01/09/2021	1064.1	
9	01/10/2021	966.7	

Figure 3.22: 600x600 (256x1024)





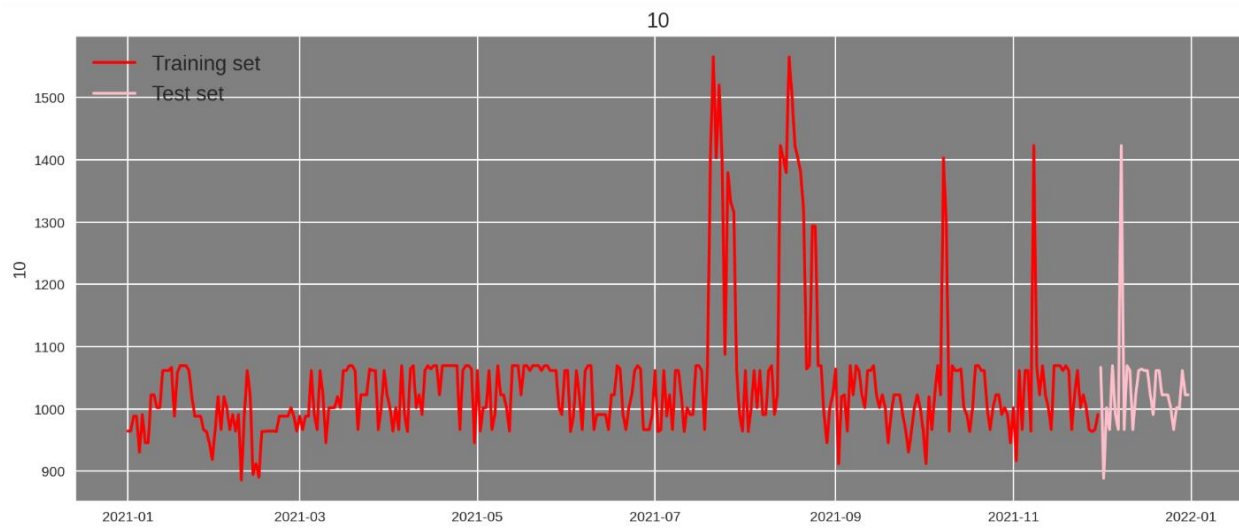
```
warnings.warn("Maximum Likelihood optimization failed to "
```

SARIMAX Results

```
=====
Dep. Variable:          y      No. Observations:          365
Model:                ARIMA(2, 0, 2)  Log Likelihood          563.223
Date:                Tue, 07 May 2024  AIC                  -1114.446
Time:                04:44:20    BIC                  -1091.047
Sample:              0      HQIC                  -1105.147
                             - 365
Covariance Type:      opg
=====
```

	coef	std err	z	P> z	[0.025	0.975]
const	-4.829e-06	0.001	-0.006	0.995	-0.002	0.002
ar.L1	-0.1328	0.059	-2.233	0.026	-0.249	-0.016
ar.L2	0.5723	0.054	10.593	0.000	0.466	0.678
ma.L1	-0.0099	0.040	-0.249	0.804	-0.087	0.068
ma.L2	-0.8488	0.043	-19.543	0.000	-0.934	-0.764
sigma2	0.0027	0.000	25.500	0.000	0.002	0.003

```
=====
Ljung-Box (L1) (Q):          0.26  Jarque-Bera (JB):          665.01
Prob(Q):                   0.61  Prob(JB):              0.00
Heteroskedasticity (H):      1.72  Skew:                  0.50
Prob(H) (two-sided):         0.00  Kurtosis:              9.54
=====
```



4. link git