In [14]:
```
import os
import pandas as pd
from neuralforecast.losses.numpy import mae, mse
```

In [15]:
```
folder_path = 'Exchange_rate'
exchange_result = []

for filename in os.listdir(folder_path):
    if filename.endswith('.csv'):
        file_path = os.path.join(folder_path, filename)
        df = pd.read_csv(file_path, low_memory=False, index_col = 0)
        exchange_result.append(df)
```

In [16]:
```
Exchange_results = exchange_result[0]
```

In [17]:
```
Exchange_results['AutoNHITS'] = exchange_result[1]['AutoNHITS']
Exchange_results['Informer'] = exchange_result[2]['Informer']
Exchange_results['NBEATS'] = exchange_result[3]['NBEATS']
Exchange_results['NHITS'] = exchange_result[4]['NHITS']
```

In [18]:
```
models = ['AutoNHITS', 'NBEATS','Autoformer','Informer']
metrics = ['MSE', 'MAE']

results = {}

for model in models:
    for metric in metrics:
        y_true = Exchange_results['y']
        y_pred = Exchange_results[model]

        if metric == 'MSE':
            result = mse(y_true, y_pred)
        elif metric == 'MAE':
            result = mae(y_true, y_pred)

        key = f'{model}_{metric}'
        results[key] = result

exchange_rate = pd.DataFrame.from_dict(results, orient='index', columns=['Ex
exchange_rate = exchange_rate.rename(columns={"AutoNHITS_MSE": "NHITS_MSE",
exchange_rate
```

Out[18]:

	NHITS_MSE	NHITS_MAE	NBEATS_MSE	NBEATS_MAE	Autoform
Exchange_rate	0.086735	0.208902	0.088692	0.207726	(

In [19]:
```
folder_path = 'Ettm2'
ettm2_result = []

for filename in os.listdir(folder_path):
    if filename.endswith('.csv'):
        file_path = os.path.join(folder_path, filename)
```

```
        df = pd.read_csv(file_path, low_memory=False, index_col = 0)
        ettm2_result.append(df)
```

In [20]:
```
ettm2_results = ettm2_result[2]
ettm2_results['Autoformer'] = ettm2_result[0]['Autoformer']
ettm2_results['Informer'] = ettm2_result[1]['Informer']
```

In [21]:
```
models = ['NBEATS','Autoformer','Informer']
metrics = ['MSE', 'MAE']

results = {}

for model in models:
    for metric in metrics:
        y_true = ettm2_results['y']
        y_pred = ettm2_results[model]

        if metric == 'MSE':
            result = mse(y_true, y_pred)
        elif metric == 'MAE':
            result = mae(y_true, y_pred)

        key = f'{model}_{metric}'
        results[key] = result

Ettm2 = pd.DataFrame.from_dict(results, orient='index', columns=['Ettm2']).T
Ettm2.insert(0, 'NHITS_MSE', 0.18279484416711375)
Ettm2.insert(1, 'NHITS_MAE', 0.26096806135482414)
Ettm2
```

Out[21]:

	NHITS_MSE	NHITS_MAE	NBEATS_MSE	NBEATS_MAE	Autoformer_MSE
Ettm2	0.182795	0.260968	0.184381	0.267878	0.255369

In [22]:
```
import pandas as pd
NHITS_wea = pd.read_csv('Weather/df_NHITS.csv',index_col=0)
NBEATS_wea = pd.read_csv('Weather/df_NBEATS.csv',index_col=0)
Auto_wea = pd.read_csv('Weather/df_Autoformer.csv',index_col=0)
informer_wea = pd.read_csv('Weather/df_Informer.csv',index_col=0)
```

In [23]:
```
Weather = pd.concat([NHITS_wea,NBEATS_wea,Auto_wea,informer_wea], axis=1)
Weather
```

Out[23]:

	NHITS_MSE	NHITS_MAE	NBEATS_MSE	NBEATS_MAE	Autoformer_MS
Weather	0.165567	0.197574	0.178303	0.207848	0.23059

In [24]:
```
results = pd.concat([Ettm2,exchange_rate,Weather], axis=0)
results = results.round(3)
```

In [25]:
```
results
```

Out[25]:

	NHITS_MSE	NHITS_MAE	NBEATS_MSE	NBEATS_MAE	Autoform
Ettm2	0.183	0.261	0.184	0.268	
Exchange_rate	0.087	0.209	0.089	0.208	
Weather	0.166	0.198	0.178	0.208	

In [26]:
```
results.to_csv('final_results.csv',index=0)
```

In [ ]:
```

```