In [2]:
```
#pip install matplotlib numpy pandas
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [6]:
```
FILEPATH="./jeu_de_donnée/effects-of-covid-19-on-trade-at-15-december-2021-provisional.csv"
HORIZON=np.random.randint(1,100)
FEATURE="Cumulative"
SEP=","
TITLE="prix cumulé des couts de transport dans une direction"
```

In [7]:
```
df=pd.read_csv(FILEPATH,sep=SEP)
```

In [8]:
```
df.head()
```

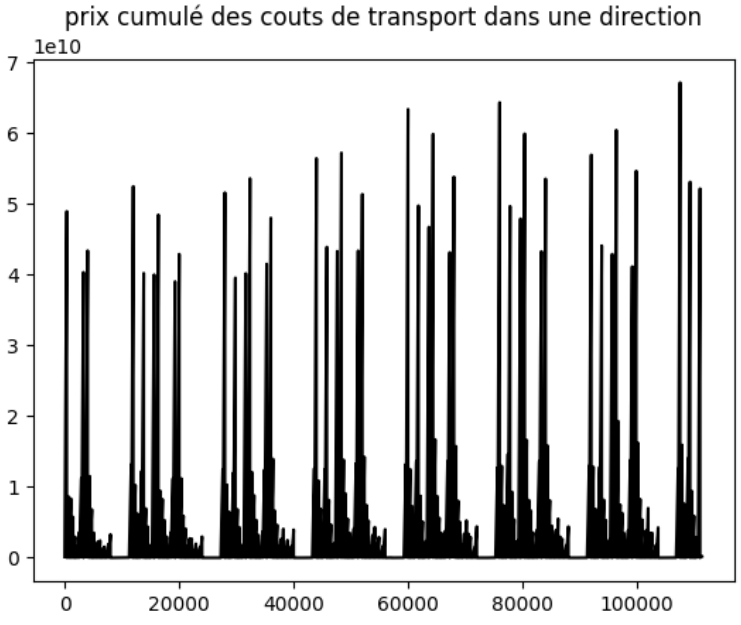
Out[8]:

	Direction	Year	Date	Weekday	Country	Commodity	Transport_Mode	Measure	Value	Cumulative
0	Exports	2015	01/01/2015	Thursday	All	All	All	$	104000000	104000000
1	Exports	2015	02/01/2015	Friday	All	All	All	$	96000000	200000000
2	Exports	2015	03/01/2015	Saturday	All	All	All	$	61000000	262000000
3	Exports	2015	04/01/2015	Sunday	All	All	All	$	74000000	336000000
4	Exports	2015	05/01/2015	Monday	All	All	All	$	105000000	442000000

In [9]:
```
all_alpha=[0.001,0.005, 0.01, 0.05, 0.1, 0.3, 0.5, 0.9]
```

In [14]:
```
plt.plot(df[FEATURE], c="black", ls='-')
plt.title(TITLE)
```

Out[14]:
```
Text(0.5, 1.0, 'prix cumulé des couts de transport dans une direction')
```



In [15]:
```
def predict_simple_expo_lissage(data, alpha, taille, horizon=1):
    results=[]
    for i in range(taille-horizon):
        if i==0:
            results.append((1-alpha)*data[i])
        else:
            tmp=(1-alpha)*data[i]+alpha*results[i-1]
            results.append(tmp)
    return results
```

In [16]:
```
alpha=all_alpha[-1]
res=predict_simple_expo_lissage(df[FEATURE], alpha, len(df[FEATURE]))
```

In [17]:
```
all_results={}
for alpha in all_alpha:
    all_results[f"alpha_{alpha}"]=predict_simple_expo_lissage(df[FEATURE], alpha, len(df[FEATURE]), HORIZON)
```

In [18]:
```
def sum_square_error(real, predic):
    result=real-predic
    result=result**2
    return np.sum(result)
```

In [19]:
```
all_error={}
for key in all_results.keys():
    real=np.array(df[FEATURE])
    predic=np.array(all_results[key])
    all_error[key]=sum_square_error(real[:real.shape[0]-HORIZON],predic)
```

In [21]:
```
plt.figure(figsize=[35, 25])
for i in range(len(all_alpha)):
    plt.subplot(2, 4, i+1)
    plt.plot(df[FEATURE] , label='Real Data', c='gray', lw=2)
    m1=np.max(df[FEATURE])+10
    m2=np.min(df[FEATURE])-10
    x=len(df)-HORIZON
    y=np.linspace(m2,m1,100)
    x=np.ones(y.shape)*x
    plt.plot(x,y, c="red", lw=2.7, ls="-")
    last=all_results[f"alpha_{all_alpha[i]}"][-1]
    tmp=all_results[f"alpha_{all_alpha[i]}"]
    for j in range(HORIZON):
        tmp.append(last)
    plt.plot(tmp, label='Prediction', lw=2.5 , ls=':', c='blue')
    alpha=f"alpha_{all_alpha[i]}"
    plt.title(f"alpha = {all_alpha[i]} : error = {all_error[alpha]:.3f}")
    plt.legend()
```