In [1]:
```
from scipy import stats
```

In [2]:
```
stats.norm.cdf(x=70,loc=60,scale=10)
```

Out[2]: 0.8413447460685429

In [3]:
```
stats.norm.cdf(x=80,loc=60,scale=10)
```

Out[3]: 0.9772498680518208

In [4]:
```
stats.norm.cdf(x=680,loc=711,scale=29)
```

Out[4]: 0.14254260383881612

In [5]:
```
1 - stats.norm.cdf(x=750,loc=711,scale=29)
```

Out[5]: 0.08934045974656879

In [6]:
```
stats.norm.cdf(x=740,loc=711,scale=29)
```

Out[6]: 0.8413447460685429

In [7]:
```
0.8413447460685429 - 0.14254260383881612
```

Out[7]: 0.6988021422297268

In [8]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [9]:
```
import warnings
warnings.filterwarnings("ignore")
```

In [10]:
```
ble_df=pd.read_csv(r"C:\Users\arnak\Downloads\BEML (1).csv")
```

In [11]:
```
glo_df=pd.read_csv(r"C:\Users\arnak\Downloads\GLAXO (1).csv")
```

In [12]:
```
ble_df
```

Out[12]:

	Date	Open	High	Low	Last	Close	Total Trade Quantity	Turnover (Lacs)
0	2010-01-04	1121.00	1151.00	1121.00	1134.00	1135.60	101651.0	1157.18
1	2010-01-05	1146.80	1149.00	1128.75	1135.00	1134.60	59504.0	676.47
2	2010-01-06	1140.00	1164.25	1130.05	1137.00	1139.60	128908.0	1482.84
3	2010-01-07	1142.00	1159.40	1119.20	1141.00	1144.15	117871.0	1352.98
4	2010-01-08	1156.00	1172.00	1140.00	1141.20	1144.05	170063.0	1971.42
...	...	...	...	...	...	...	...	...
1734	2016-12-26	965.00	965.05	935.00	950.10	950.25	398696.0	3783.63
1735	2016-12-27	960.70	989.00	952.35	974.00	975.70	808561.0	7885.14
1736	2016-12-28	980.75	985.00	970.15	977.00	974.40	367041.0	3592.49
1737	2016-12-29	977.10	997.95	974.55	985.15	986.05	555233.0	5489.14
1738	2016-12-30	986.00	1006.95	985.90	1004.00	1000.60	460675.0	4606.48

1739 rows × 8 columns

In [13]:
```
glo_df
```

Out[13]:

	Date	Open	High	Low	Last	Close	Total Trade Quantity	Turnover (Lacs)
0	2010-01-04	1613.00	1629.10	1602.00	1629.0	1625.65	9365.0	151.74
1	2010-01-05	1639.95	1639.95	1611.05	1620.0	1616.80	38148.0	622.58
2	2010-01-06	1618.00	1644.00	1617.00	1639.0	1638.50	36519.0	595.09
3	2010-01-07	1645.00	1654.00	1636.00	1648.0	1648.70	12809.0	211.00
4	2010-01-08	1650.00	1650.00	1626.55	1640.0	1639.80	28035.0	459.11
...	...	...	...	...	...	...	...	...
1734	2016-12-26	2703.00	2740.00	2677.00	2715.0	2723.50	3953.0	107.15
1735	2016-12-27	2722.95	2725.00	2683.00	2692.0	2701.75	10600.0	286.10
1736	2016-12-28	2701.75	2718.00	2690.00	2698.0	2702.15	6050.0	163.44
1737	2016-12-29	2702.05	2739.00	2691.95	2710.0	2727.90	7649.0	207.87
1738	2016-12-30	2730.00	2740.45	2705.00	2730.0	2729.80	6513.0	177.65

1739 rows × 8 columns

In [14]:
```
ble_df=ble_df[['Date','Close']]
```

In [15]:
```
glo_df=glo_df[['Date','Close']]
```

In [16]:
```
ble_df
```

Out[16]:

	Date	Close
0	2010-01-04	1135.60
1	2010-01-05	1134.60
2	2010-01-06	1139.60
3	2010-01-07	1144.15
4	2010-01-08	1144.05
...	...	...
1734	2016-12-26	950.25
1735	2016-12-27	975.70
1736	2016-12-28	974.40
1737	2016-12-29	986.05
1738	2016-12-30	1000.60

1739 rows × 2 columns

In [17]:
```
glo_df
```

Out[17]:

	Date	Close
0	2010-01-04	1625.65
1	2010-01-05	1616.80
2	2010-01-06	1638.50
3	2010-01-07	1648.70
4	2010-01-08	1639.80
...	...	...
1734	2016-12-26	2723.50
1735	2016-12-27	2701.75
1736	2016-12-28	2702.15
1737	2016-12-29	2727.90
1738	2016-12-30	2729.80

1739 rows × 2 columns

In [18]:
```
ble_df.set_index('Date', inplace=True)
```

In [19]:
```
ble_df
```

Out[19]:

Date	Close
2010-01-04	1135.60
2010-01-05	1134.60
2010-01-06	1139.60
2010-01-07	1144.15
2010-01-08	1144.05
...	...
2016-12-26	950.25
2016-12-27	975.70
2016-12-28	974.40
2016-12-29	986.05
2016-12-30	1000.60

1739 rows × 1 columns

In [20]:
```
glo_df.set_index('Date', inplace=True)
```

In [21]:
```
glo_df
```

Out[21]:

Date	Close
2010-01-04	1625.65
2010-01-05	1616.80
2010-01-06	1638.50
2010-01-07	1648.70
2010-01-08	1639.80
...	...
2016-12-26	2723.50
2016-12-27	2701.75
2016-12-28	2702.15
2016-12-29	2727.90
2016-12-30	2729.80

1739 rows × 1 columns

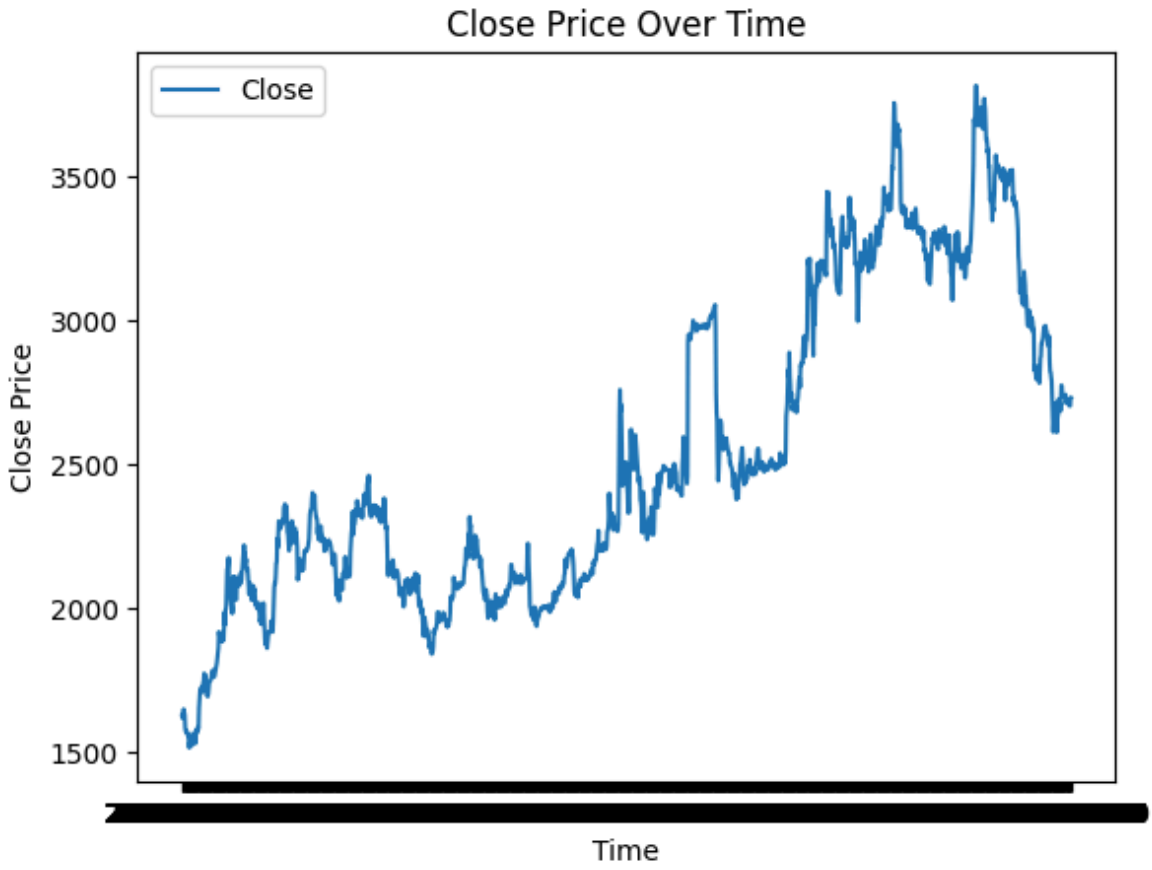
In [22]:
```
print(glo_df.dtypes)
print('*'*50)
```

```
print(ble_df.dtypes)
```

```
Close    float64
dtype: object
**********************************************
Close    float64
dtype: object
```

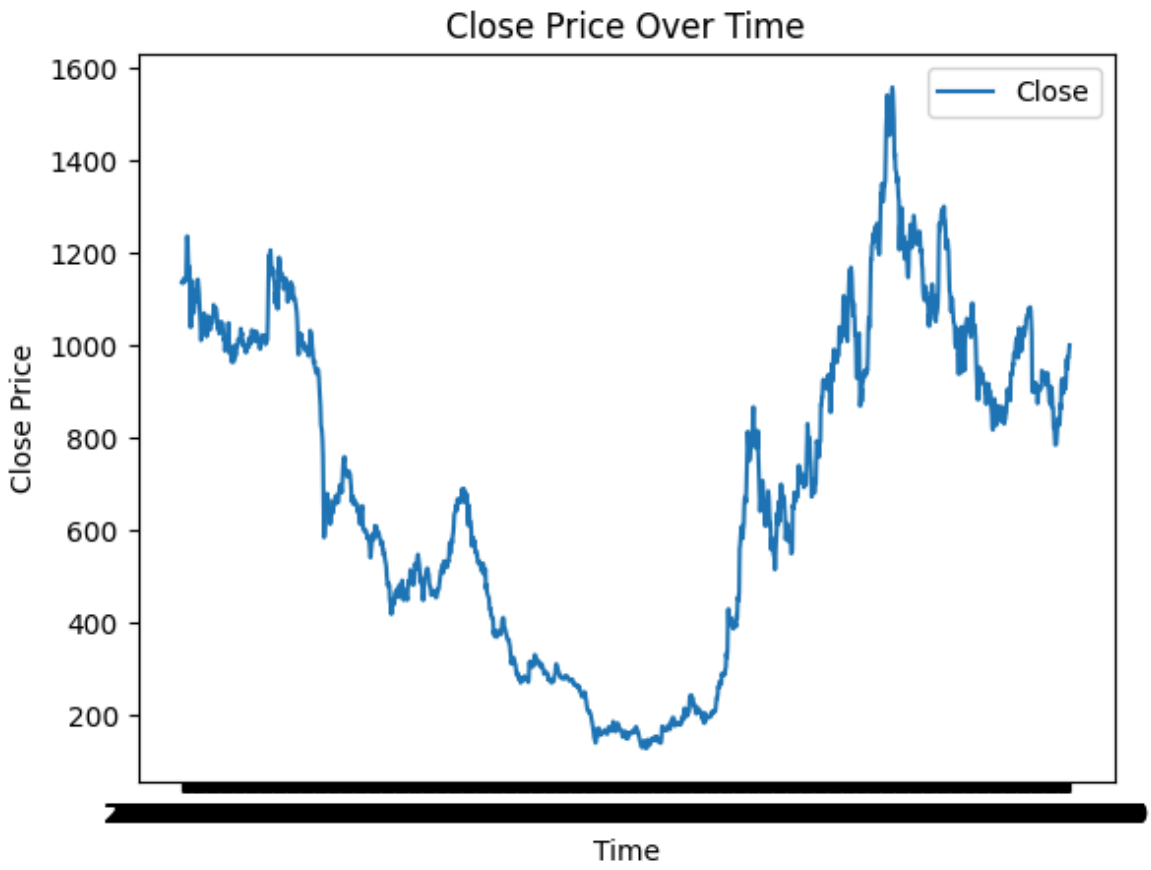
In [23]:
```
sns.lineplot(glo_df)
plt.xlabel('Time')
plt.ylabel('Close Price')
plt.title('Close Price Over Time')
```

Out[23]: Text(0.5, 1.0, 'Close Price Over Time')



In [24]:
```
sns.lineplot(ble_df)
plt.xlabel('Time')
plt.ylabel('Close Price')
plt.title('Close Price Over Time')
```

Out[24]: Text(0.5, 1.0, 'Close Price Over Time')



In [25]:
```
glo_df['gain'] = glo_df.Close.pct_change(periods = 1)
glo_df
```

Out[25]:

Date	Close	gain
2010-01-04	1625.65	NaN
2010-01-05	1616.80	-0.005444
2010-01-06	1638.50	0.013422
2010-01-07	1648.70	0.006225
2010-01-08	1639.80	-0.005398
...	...	...
2016-12-26	2723.50	-0.001283
2016-12-27	2701.75	-0.007986
2016-12-28	2702.15	0.000148
2016-12-29	2727.90	0.009529
2016-12-30	2729.80	0.000697

1739 rows × 2 columns

In [26]:
```
ble_df['gain'] = ble_df.Close.pct_change(periods = 1)
```

In [27]:
```
ble_df
```

Out[27]:

Date	Close	gain
2010-01-04	1135.60	NaN
2010-01-05	1134.60	-0.000881
2010-01-06	1139.60	0.004407
2010-01-07	1144.15	0.003993
2010-01-08	1144.05	-0.000087
...	...	...
2016-12-26	950.25	-0.021924
2016-12-27	975.70	0.026782
2016-12-28	974.40	-0.001332
2016-12-29	986.05	0.011956
2016-12-30	1000.60	0.014756

1739 rows × 2 columns

In [28]:
```
glo_df = glo_df.dropna()
ble_df = ble_df.dropna()
```

In [29]:
```
glo_df
```

Out[29]:

Date	Close	gain
2010-01-05	1616.80	-0.005444
2010-01-06	1638.50	0.013422
2010-01-07	1648.70	0.006225
2010-01-08	1639.80	-0.005398
2010-01-11	1629.45	-0.006312
...	...	...
2016-12-26	2723.50	-0.001283
2016-12-27	2701.75	-0.007986
2016-12-28	2702.15	0.000148
2016-12-29	2727.90	0.009529
2016-12-30	2729.80	0.000697

1738 rows × 2 columns

In [30]:
```
ble_df
```

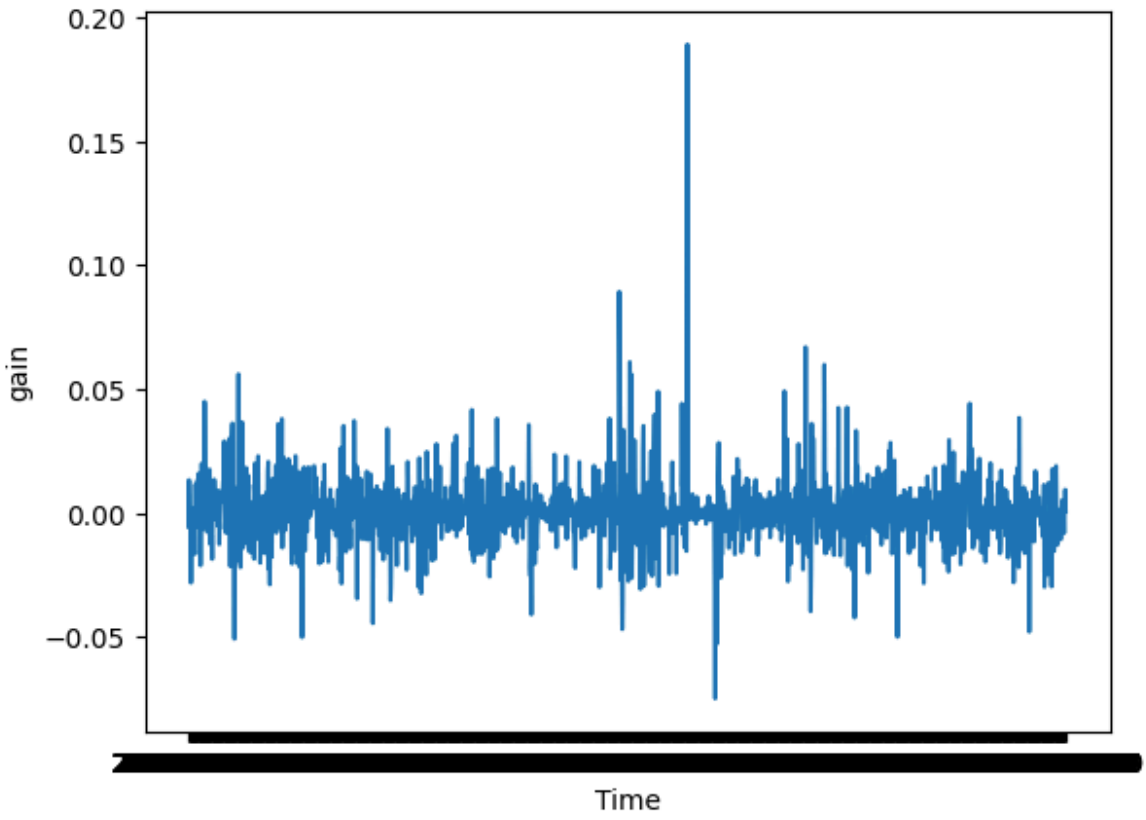
Out[30]:

Date	Close	gain
2010-01-05	1134.60	-0.000881
2010-01-06	1139.60	0.004407
2010-01-07	1144.15	0.003993
2010-01-08	1144.05	-0.000087
2010-01-11	1137.00	-0.006162
...	...	...
2016-12-26	950.25	-0.021924
2016-12-27	975.70	0.026782
2016-12-28	974.40	-0.001332
2016-12-29	986.05	0.011956
2016-12-30	1000.60	0.014756

1738 rows × 2 columns

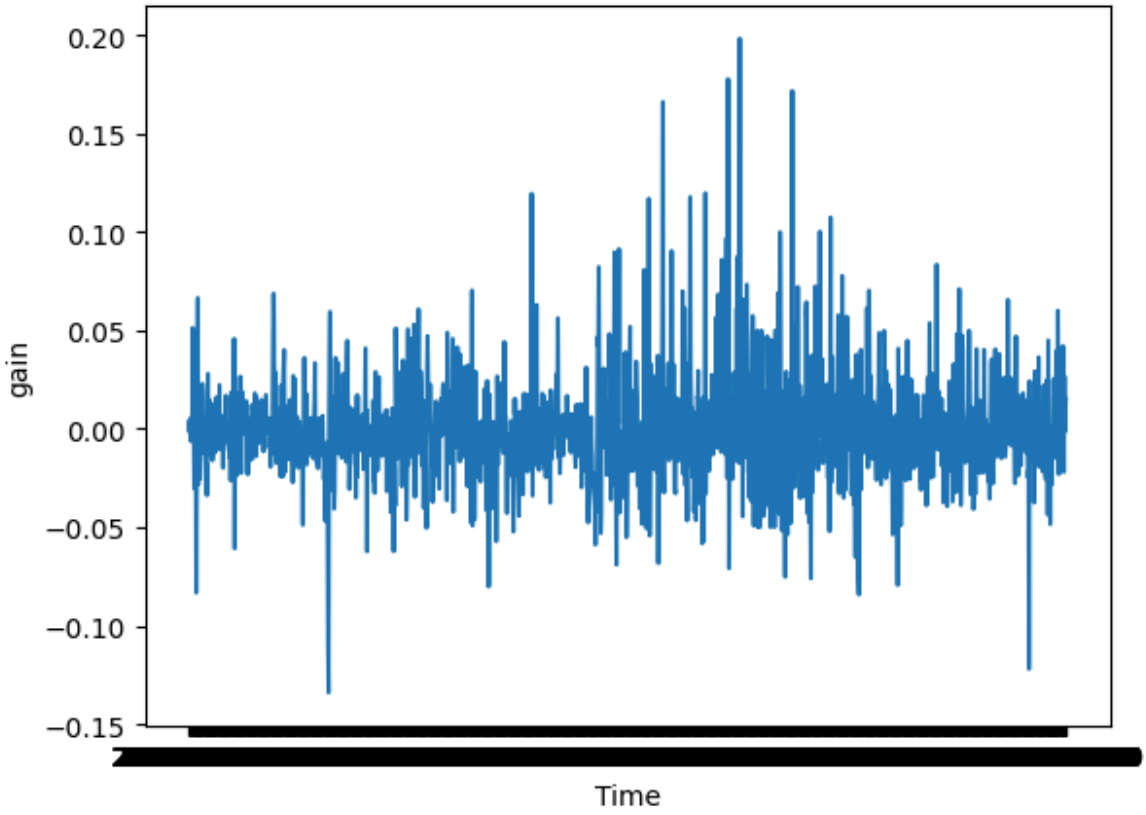
In [31]:
```
sns.lineplot(glo_df.gain)
plt.xlabel('Time')
plt.ylabel('gain')
```

Out[31]:
```
Text(0, 0.5, 'gain')
```



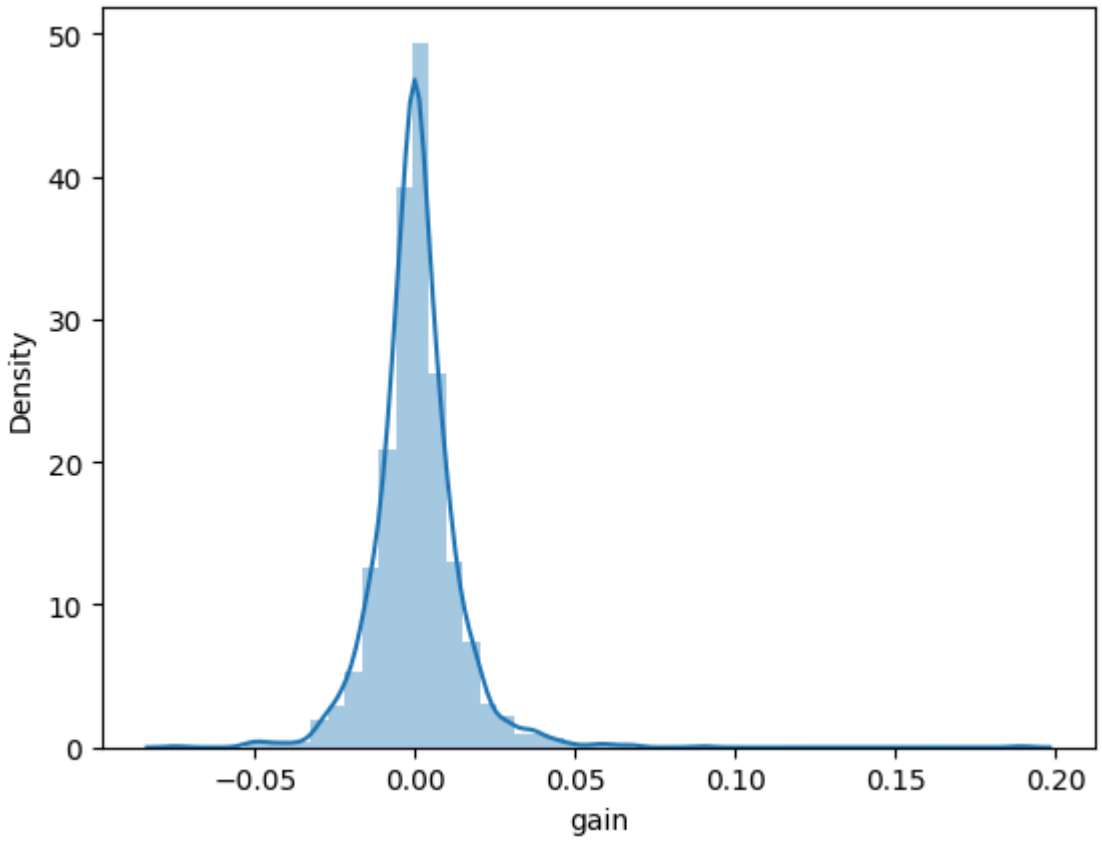
In [32]:
```
sns.lineplot(ble_df.gain)
plt.xlabel('Time')
plt.ylabel('gain')
```

Out[32]:
```
Text(0, 0.5, 'gain')
```



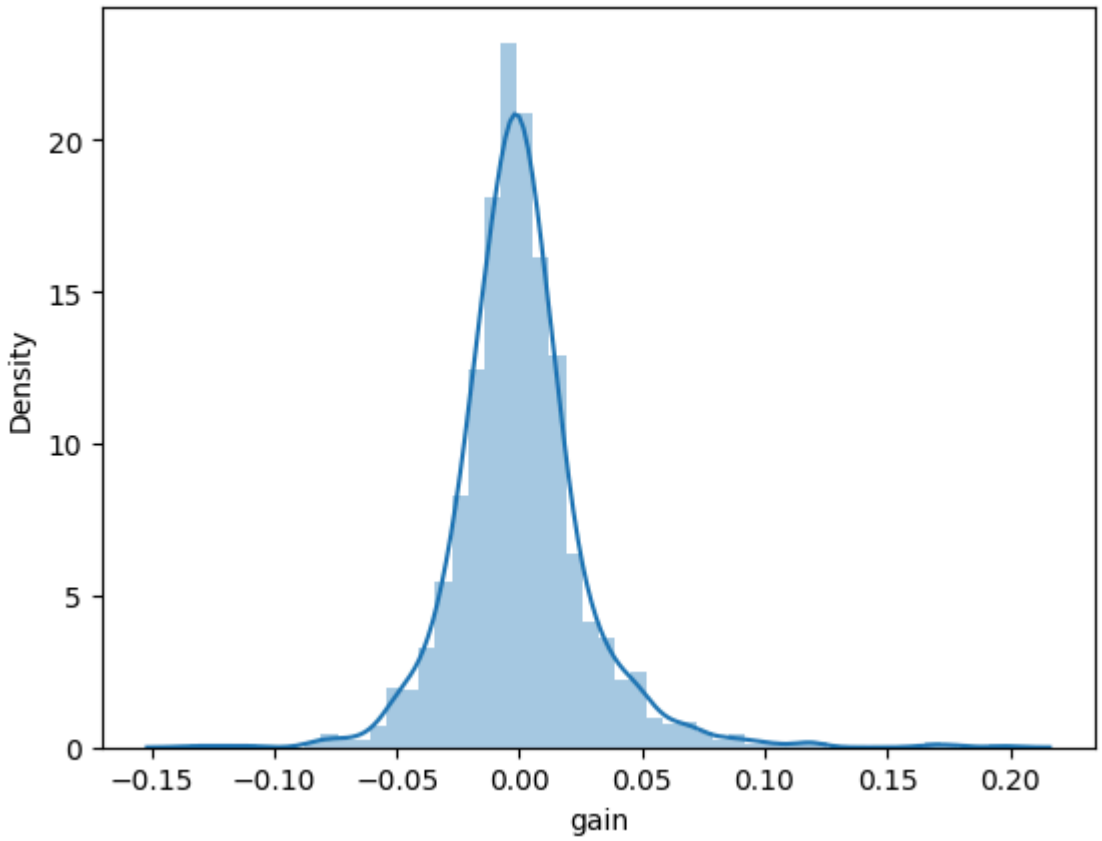
In [33]:
```
sns.distplot(glo_df.gain)
```

Out[33]:
```
<Axes: xlabel='gain', ylabel='Density'>
```



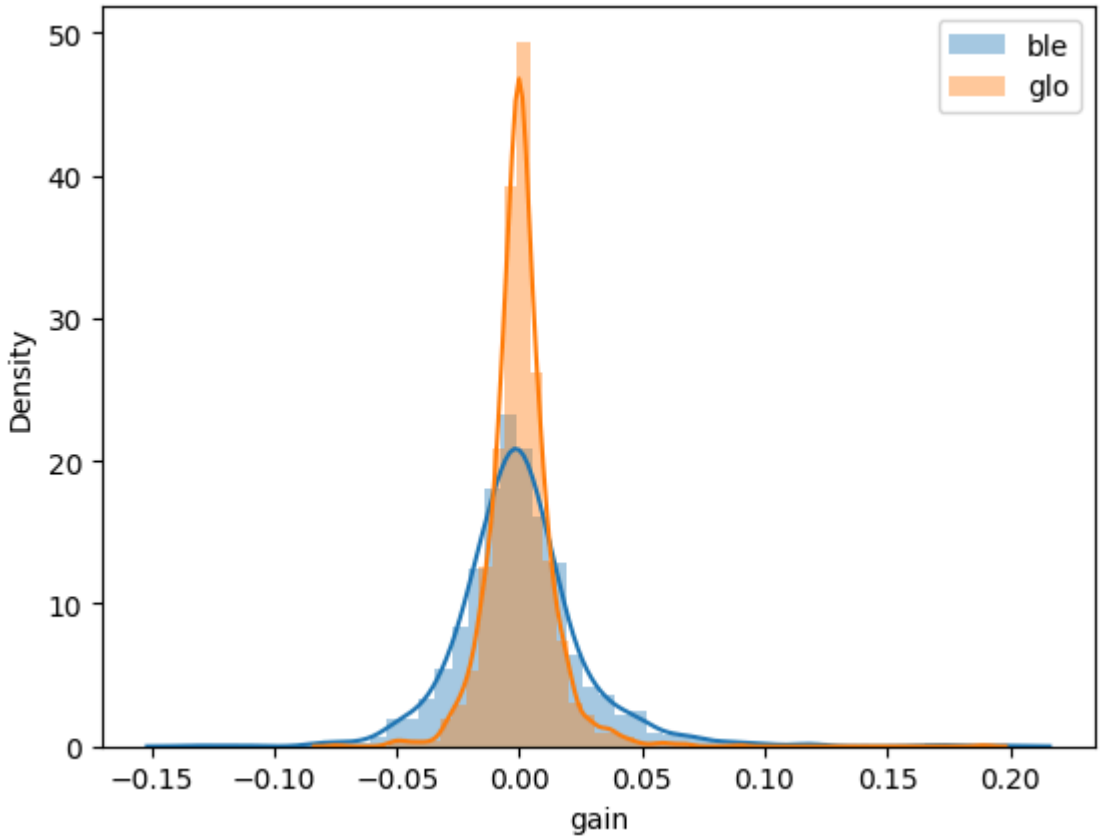
In [34]:
```
sns.distplot(ble_df.gain)
```

Out[34]: <Axes: xlabel='gain', ylabel='Density'>



In [35]:
```
sns.distplot(ble_df.gain,label='ble')
sns.distplot(glo_df.gain,label='glo')
plt.legend()
```

Out[35]: <matplotlib.legend.Legend at 0x1b60c349410>



In [36]:
```
glo_df.describe()
```

Out[36]:

	Close	gain
count	1738.000000	1738.000000
mean	2533.153596	0.000386
std	540.441532	0.013361
min	1514.300000	-0.074719
25%	2096.987500	-0.005850
50%	2365.700000	-0.000021
75%	3010.562500	0.005826
max	3814.750000	0.189196

In [37]:
```
ble_df.describe()
```

Out[37]:

	Close	gain
count	1738.000000	1738.000000
mean	698.183688	0.000271
std	357.378754	0.026431
min	129.150000	-0.133940
25%	370.650000	-0.013736
50%	682.100000	-0.001541
75%	1010.350000	0.011985
max	1558.500000	0.198329

norm.cdf() and norm.ppf() function difference

Compute 2% loss or gain for Ble

In [38]:
```
stats.norm.cdf(-0.02,0.000271,0.026431)
```

Out[38]:
```
0.22155849527795074
```

In [39]:
```
1- stats.norm.cdf(0.02,0.000271,0.026431)
```

Out[39]:
```
0.22770260982009338
```

In [52]:
```
stats.norm.ppf(0.22155849527795074,0.000271,0.026431)
```

Out[52]:
```
-0.020000000000000004
```

Compute 2% loss or gain for glo

In [40]:
```
stats.norm.cdf(-0.02,0.000386,0.013361)
```

Out[40]:
```
0.06353183355777392
```

In [41]:
```
1- stats.norm.cdf(0.02,0.000386,0.013361)
```

Out[41]:
```
0.07105158286739244
```

In [50]:
```
stats.norm.ppf(0.06353183355777392,0.000386,0.013361)
```

Out[50]:
```
-0.019999999999999997
```

In [ ]:
```

```