In [74]:
```
#Import Modules
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.ensemble import IsolationForest
import seaborn as sns
import re
```

In [98]:
```
import sys
import os
import pandas as pd
from glob import glob
import warnings
```

In [64]:
```
# getting excel files from Directory Desktop
path = "/home/nbertrand/Payroll_Data/PayrollByPeriod/"

# read all the files with extension .xlsx i.e. excel
file_list = []
filenames = glob(path + "/*.xlsx")
for file in filenames:
    file_list.append(file)
file_list
```

Out[64]:
```
['/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2022.xls
x',
 '/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2020.xls
x',
 '/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2021.xls
x',
 '/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2019.xls
x']
```

In [4]:
```
a = [None]*len(file_list)
a
```

Out[4]:
```
[None, None, None, None]
```

In [65]:
```
dicts1 = {}
keys1 = range(len(file_list))
values1 = file_list
for i in keys1:
    dicts1[i] = values1[i]
```

In [66]:
```
dicts1
```

Out[66]:
```
{0: '/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2022.
xlsx',
 1: '/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2020.
xlsx',
 2: '/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2021.
xlsx',
 3: '/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2019.
xlsx'}
```

In [63]:
```
keys1
```

Out[63]:
```
range(0, 4)
```

In [6]:
```
dicts1[0]
```

Out[6]:
```
'/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2022.xls
x'
```

In [42]:
```
dicts1[0]
```

Out[42]:
```
'/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2022.xls
x'
```

In [7]:
```
dicts1[1]
```

Out[7]:
```
'/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2020.xls
x'
```

In [8]:
```
dicts1[2]
```

Out[8]:
```
'/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2021.xls
x'
```

In [9]:
```
dicts1[3]
```

Out[9]:
```
'/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2019.xls
x'
```

In [10]:
```
#dict(list(enumerate(values1)))
```

In [11]:
```
for key in keys1:
  print(key)
```

```
0
1
2
3
```

In [12]:
```
for val in values1:
    print(val)
```

```
/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2022.xlsx
/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2020.xlsx
/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2021.xlsx
/home/nbertrand/Payroll_Data/PayrollByPeriod/Payroll_byPayPeriod_FY2019.xlsx
```

In [72]:
```
random_state = np.random.RandomState(42)
model = IsolationForest(n_estimators=1000, max_samples='auto', contamination=f
loat(0.05),random_state=random_state)
```

In [90]:
```
dataFrame_list = []
df_scores_list = []
df_anomaly_list = []
j = [None] * len(dicts1)

required_columns = [3,4,5,6,7,8,10]
for j in dicts1:
    df = pd.read_excel( dicts1[j] , usecols=required_columns)
    dataFrame_list.append(df)
```

In [91]:
```
dataFrame_list
```

Out[91]:
```
[        EmployeeID  ObjectClass              Org ApprCode        BPAC  \
0           130692         1101  4040100000000000      101  7002000000
1           130692         1101  4040100000000000      101  7001000000
2           130692         1202  4040100000000000      101  7001000000
3           130692         1202  4040100000000000      101  7002000000
4           130692         1203  4040100000000000      101  7001000000
...            ...          ...               ...      ...         ...
266094      905820         1275  1030300000000000      201  7002000000
266095      901935         1105   100000000000000      201         NaN
266096      902485         1183  5140000000000000      201          10
266097      902485         1213  5140000000000000      201          10
266098      902485         1269  5140000000000000      201          10

       PayPlan  Gross_Pay
0           CT    5096.64
1           CT    3397.76
2           CT       6.72
3           CT      10.08
4           CT     293.20
...        ...        ...
266094      CT      43.35
266095      CT   -2488.80
266096      CT   50710.40
266097      CT     735.30
266098      CT    1457.15

[266099 rows x 7 columns],
        EmployeeID  ObjectClass              Org ApprCode        BPAC  \
0           130692         1101  4004001000000000      901  7001000000
1           130692         1101  4004001000000000      901  7002000000
2           130692         1101  4004001000000000      901  7003000000
3           130692         1101  4004001000000000      901  6809000000
4           130692         1202  4004001000000000      901  6809000000
...            ...          ...               ...      ...         ...
378083      902859         1274  2000000000000000      001         NaN
378084      902859         1275  2000000000000000      001         NaN
378085      902859         4350  2000000000000000      001         NaN
378086      903065         1195  9003001000000000      001         NaN
378087      903472         1205   102001020000000      001         NaN

       PayPlan  Gross_Pay
0           CT     910.26
1           CT     278.14
2           CT     455.13
3           CT    6447.67
4           CT      12.80
...        ...        ...
378083      CT       0.11
378084      CT       0.47
378085      CT       0.61
378086      CT    -175.00
378087      CT    -131.21

[378088 rows x 7 columns],
        EmployeeID  ObjectClass              Org ApprCode        BPAC  \
```

```
0           130692         1101  4004001000000000      001  7002000000
1           130692         1202  4004001000000000      001  7002000000
2           130692         1203  4004001000000000      001  7002000000
3           130692         1213  4004001000000000      001  7002000000
4           130692         1260  4004001000000000      001  7002000000
...            ...          ...               ...      ...         ...
373795      904723         1205  1060300000000000      101         NaN
373796      905025         1183  2040400000000000      101          01
373797      905025         1213  2040400000000000      101          01
373798      905025         1269  2040400000000000      101          01
373799      905531         1195  2070600000000000      201         NaN

       PayPlan  Gross_Pay
0           CT    8327.20
1           CT      16.50
2           CT     708.23
3           CT     115.03
4           CT      70.57
...        ...        ...
373795      CT    -129.36
373796      CT   10470.20
373797      CT     151.82
373798      CT     649.15
373799      CT   -1000.00

[373800 rows x 7 columns],
        EmployeeID  ObjectClass               Org ApprCode        BPAC  \
0           130692         1101  4004001000000000      801  6811000000
1           130692         1101  4004001000000000      801  7001000000
2           130692         1101  4004001000000000      801  7002000000
3           130692         1101  4004001000000000      801  7003000000
4           130692         1101  4004001000000000      801  6807000000
...            ...          ...               ...      ...         ...
374514      904422         4350  6002000000000000      901         NaN
374515      904430         4350  6002000000000000      901         NaN
374516      904501         4350   100000000000000      901         NaN
374517      904629         4350   500000000000000      901         NaN
374518      904724         4350   101000000000000      901         NaN

       PayPlan  Gross_Pay
0           CT     786.56
1           CT      98.32
2           CT    3637.84
3           CT     196.64
4           CT    3146.24
...        ...        ...
374514      CT       0.57
374515      CT       0.75
374516      CT       0.54
374517      CT       0.40
374518      CT       0.24

[374519 rows x 7 columns]]
```

In [99]:
```
warnings.filterwarnings('ignore')
```

In [239]:
```
df_scores_list = []
df_anomaly_list = []
df_frame_list = []
for frame in dataFrame_list:
    frame = frame.replace('',np.nan,regex=True)
    np.asarray(frame[["Gross_Pay"]],dtype=float)
    model.fit(frame[['Gross_Pay']])
    df_frame_list.append(frame)

    frame['scores']= model.decision_function(frame[['Gross_Pay']])
    df_scores_list.append(frame['scores'])

    frame['anomaly_check']= model.predict(frame[['Gross_Pay']])
    df_anomaly_list.append(frame['anomaly_check'])
```

In [250]:
```
df_frame_list[0]
```

Out[250]:

	EmployeeID	ObjectClass	Org	ApprCode	BPAC	PayPlan	Gross_Pa
0	130692	1101	4040100000000000	101	7002000000	CT	5096.6
1	130692	1101	4040100000000000	101	7001000000	CT	3397.7
2	130692	1202	4040100000000000	101	7001000000	CT	6.7
3	130692	1202	4040100000000000	101	7002000000	CT	10.0
4	130692	1203	4040100000000000	101	7001000000	CT	293.2
...	...	...	...	...	...	...	.
266094	905820	1275	1030300000000000	201	7002000000	CT	43.3
266095	901935	1105	100000000000000	201	NaN	CT	-2488.8
266096	902485	1183	5140000000000000	201	10	CT	50710.4
266097	902485	1213	5140000000000000	201	10	CT	735.3
266098	902485	1269	5140000000000000	201	10	CT	1457.1

266099 rows × 9 columns

In [273]:
```
#No loop allowed here:

full_data_frame0 = df_frame_list[0]
anomalies0 = full_data_frame0.loc[full_data_frame0['anomaly_check']==-1]

full_data_frame1 = df_frame_list[1]
anomalies1 = full_data_frame1.loc[full_data_frame1['anomaly_check']==-1]

full_data_frame2 = df_frame_list[2]
anomalies2 = full_data_frame2.loc[full_data_frame2['anomaly_check']==-1]

full_data_frame3 = df_frame_list[3]
anomalies3 = full_data_frame3.loc[full_data_frame3['anomaly_check']==-1]
```

In [276]:
```
anomaly_index0=list(anomalies0.index)
anomaly_index1=list(anomalies1.index)
anomaly_index2=list(anomalies2.index)
anomaly_index3=list(anomalies3.index)
```

In [264]:
```
anomaly_length0 = len(anomalies0)
anomaly_length1 = len(anomalies1)
anomaly_length2 = len(anomalies2)
anomaly_length3 = len(anomalies3)
```

In [266]:
```
print(anomaly_length0)
print(anomaly_length1)
print(anomaly_length2)
print(anomaly_length3)
```

```
13263
18898
18666
18723
```

In [268]:
```
anomalies0.sort_values(by='scores',ascending=True)
```

Out[268]:

	EmployeeID	ObjectClass	Org	ApprCode	BPAC	PayPlan	Gross_Pa
372015	903382	1183	7000000000000000	901	10	CT	101264.8
371730	903348	1101	2003004000000000	901	6702000000	CT	73548.6
339881	904238	1183	1000000000000000	901	10	CT	63349.6
365399	901490	1183	3000000000000000	901	10	CT	63408.8
364533	901399	1101	1003002010000000	901	6403000000	CT	64788.7
...	...	...	...	...	...	...	.
269640	300182	1101	2007000000000000	901	6702000000	CT	2484.3
261354	901419	1101	3005002030000000	901	7002000000	CT	2488.9
215649	901419	1101	3005002030000000	901	6805000000	CT	2488.9
56639	904397	1101	1006004000000000	901	6100000000	CT	2499.2
56640	904397	1101	1006004000000000	901	6501000000	CT	2499.2

18723 rows × 9 columns

In [269]:
```
anomalies1.sort_values(by='scores',ascending=True)
```

Out[269]:

	EmployeeID	ObjectClass	Org	ApprCode	BPAC	PayPlan	Gross_Pa
133061	300408	1183	4005000000000000	001	10	CT	47137.9
204329	300136	1101	5005002010000000	001	7000000000	CT	39509.0
201256	903094	1101	7001000000000000	001	6600000000	CT	39537.9
133000	300114	1183	1003002010000000	001	10	CT	38838.2
133003	300116	1183	8005002040000000	001	10	CT	41313.6
...	...	...	...	...	...	...	.
289452	300158	1101	2007001000000000	001	6700000000	CT	2565.3
357093	902852	1101	1003001020000000	001	6403000000	CT	2565.3
338490	380875	1101	8005005010000000	001	6000000000	CT	2565.3
265281	900129	1131	2007003000000000	001	7002000000	CT	2565.3
230332	300167	1101	2007002000000000	001	6703000000	CT	2565.3

18898 rows × 9 columns

In [270]:
```
anomalies2.sort_values(by='scores',ascending=True)
```

Out[270]:

	EmployeeID	ObjectClass	Org	ApprCode	BPAC	PayPlan	Gross_Pa
169243	904942	1101	3008002000000000	001	6401200000	CT	101353.2
250504	900198	1101	2080102000000000	101	6401000000	CT	98014.2
191256	901751	1101	5110000000000000	101	6400000000	CT	68506.1
132692	901744	1183	2080100000000000	101	1000000000	CT	50429.8
170227	904825	1183	100000000000000	101	1000000000	CT	66157.1
...	...	...	...	...	...	...	.
264719	901494	1101	2070400000000000	101	6703000000	CT	3149.5
323237	903820	1101	2030200000000000	101	6702000000	CT	3153.9
113394	901376	1101	8020200000000000	101	7002000000	CT	3148.7
96831	377623	1101	8050202000000000	101	6403200000	CT	3156.5
22419	900201	1101	2008001010000000	101	6401000000	CT	3154.8

18666 rows × 9 columns

In [271]:
```
anomalies3.sort_values(by='scores',ascending=True)
```

Out[271]:

	EmployeeID	ObjectClass	Org	ApprCode	BPAC	PayPlan	Gross_Pa
372015	903382	1183	7000000000000000	901	10	CT	101264.8
371730	903348	1101	2003004000000000	901	6702000000	CT	73548.6
339881	904238	1183	1000000000000000	901	10	CT	63349.6
365399	901490	1183	3000000000000000	901	10	CT	63408.8
364533	901399	1101	1003002010000000	901	6403000000	CT	64788.7
...	...	...	...	...	...	...	.
269640	300182	1101	2007000000000000	901	6702000000	CT	2484.3
261354	901419	1101	3005002030000000	901	7002000000	CT	2488.9
215649	901419	1101	3005002030000000	901	6805000000	CT	2488.9
56639	904397	1101	1006004000000000	901	6100000000	CT	2499.2
56640	904397	1101	1006004000000000	901	6501000000	CT	2499.2

18723 rows × 9 columns

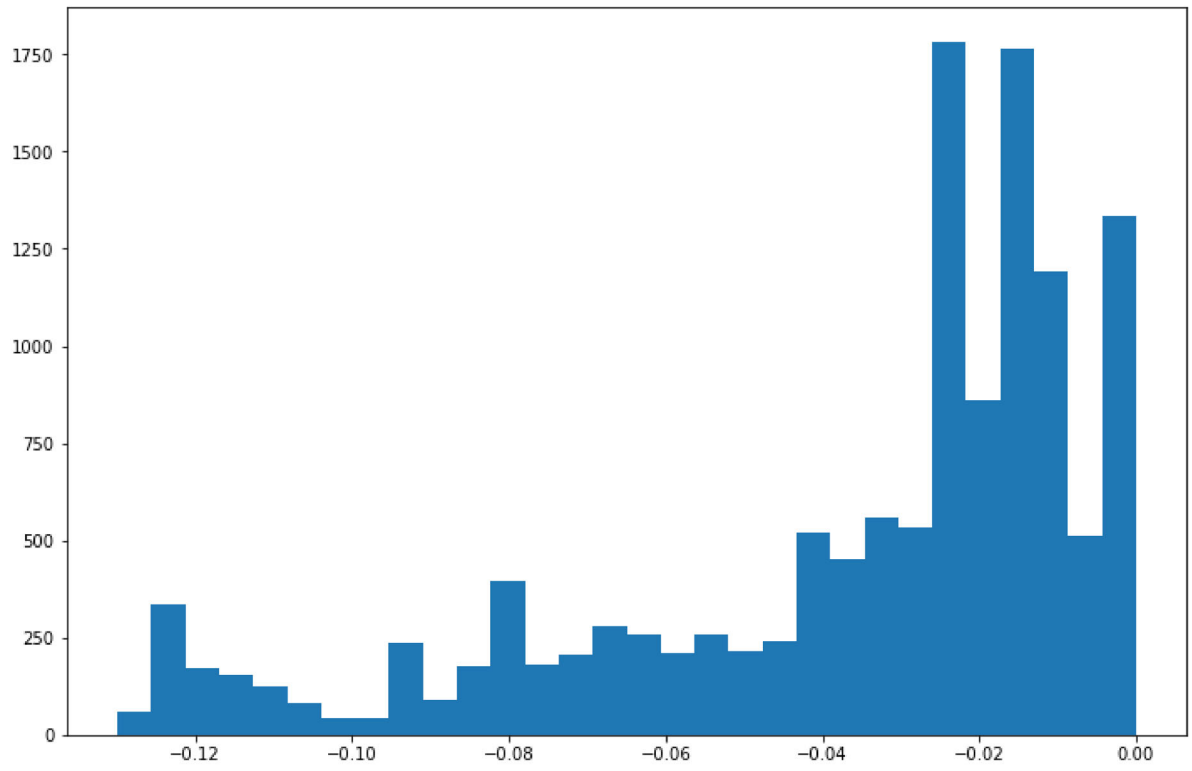
In [274]:
```
a=len(full_data_frame0)
b=len(full_data_frame1)
c=len(full_data_frame2)
d=len(full_data_frame3)
```

In [278]:
```
contamination_check0 = (len(anomaly_index0)/len(full_data_frame0))
contamination_check1 = (len(anomaly_index1)/len(full_data_frame1))
contamination_check2 = (len(anomaly_index2)/len(full_data_frame2))
contamination_check3 = (len(anomaly_index3)/len(full_data_frame3))
print(contamination_check0)
print(contamination_check1)
print(contamination_check2)
print(contamination_check3)
```

```
0.04984235190662122
0.04998307272381033
0.04993579454253612
0.04999212323006309
```

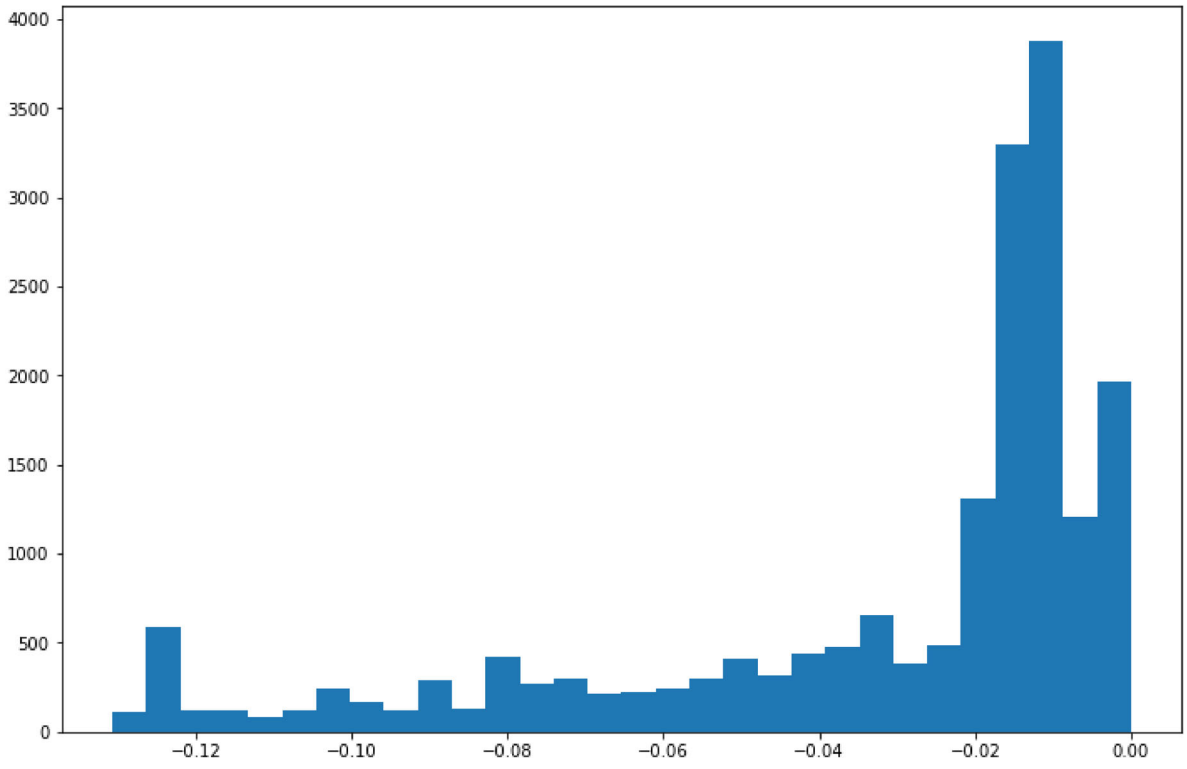
In [279]:
```
plt.figure(figsize=(12,8))
plt.hist(anomalies0['scores'], bins=30)
```

Out[279]:
```
(array([  58.,  336.,  172.,  155.,  123.,   81.,   44.,   41.,  235.,
         89.,  177.,  396.,  182.,  205.,  280.,  258.,  208.,  257.,
        215.,  241.,  520.,  453.,  558.,  535., 1783.,  859., 1764.,
       1190.,  513., 1335.]),
 array([-1.29925107e-01, -1.25594763e-01, -1.21264420e-01, -1.16934077e-01,
        -1.12603733e-01, -1.08273390e-01, -1.03943047e-01, -9.96127035e-02,
        -9.52823602e-02, -9.09520169e-02, -8.66216736e-02, -8.22913303e-02,
        -7.79609870e-02, -7.36306437e-02, -6.93003004e-02, -6.49699571e-02,
        -6.06396137e-02, -5.63092704e-02, -5.19789271e-02, -4.76485838e-02,
        -4.33182405e-02, -3.89878972e-02, -3.46575539e-02, -3.03272106e-02,
        -2.59968673e-02, -2.16665240e-02, -1.73361807e-02, -1.30058374e-02,
        -8.67549410e-03, -4.34515079e-03, -1.48074900e-05]),
 <BarContainer object of 30 artists>)
```



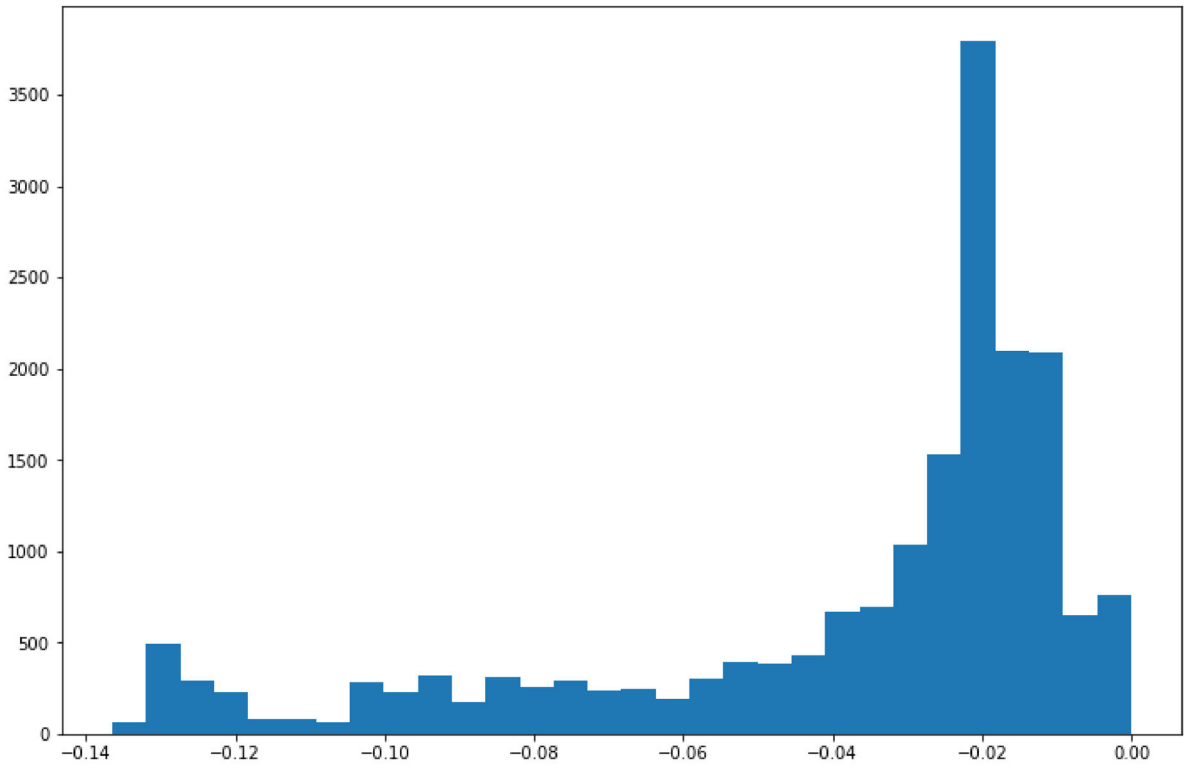
In [280]:
```
plt.figure(figsize=(12,8))
plt.hist(anomalies1['scores'], bins=30)
```

Out[280]:
```
(array([ 112.,  585.,  122.,  116.,   86.,  121.,  244.,  163.,  123.,
         290.,  131.,  419.,  267.,  299.,  216.,  223.,  245.,  297.,
         413.,  320.,  441.,  480.,  652.,  383.,  486., 1311., 3301.,
        3882., 1203., 1967.]),
 array([-1.30691117e-01, -1.26335345e-01, -1.21979574e-01, -1.17623802e-01,
        -1.13268030e-01, -1.08912259e-01, -1.04556487e-01, -1.00200715e-01,
        -9.58449436e-02, -9.14891719e-02, -8.71334002e-02, -8.27776286e-02,
        -7.84218569e-02, -7.40660852e-02, -6.97103135e-02, -6.53545419e-02,
        -6.09987702e-02, -5.66429985e-02, -5.22872268e-02, -4.79314552e-02,
        -4.35756835e-02, -3.92199118e-02, -3.48641401e-02, -3.05083685e-02,
        -2.61525968e-02, -2.17968251e-02, -1.74410534e-02, -1.30852818e-02,
        -8.72951008e-03, -4.37373840e-03, -1.79667241e-05]),
 <BarContainer object of 30 artists>)
```



In [281]:
```
plt.figure(figsize=(12,8))
plt.hist(anomalies2['scores'], bins=30)
```

Out[281]:
```
(array([  62.,  494.,  294.,  231.,   82.,   77.,   66.,  284.,  227.,
         317.,  173.,  310.,  251.,  289.,  234.,  243.,  194.,  301.,
         388.,  382.,  433.,  667.,  697., 1038., 1534., 3798., 2099.,
        2088.,  651.,  762.]),
 array([-1.36456921e-01, -1.31909864e-01, -1.27362806e-01, -1.22815749e-01,
        -1.18268692e-01, -1.13721634e-01, -1.09174577e-01, -1.04627520e-01,
        -1.00080462e-01, -9.55334049e-02, -9.09863476e-02, -8.64392902e-02,
        -8.18922329e-02, -7.73451756e-02, -7.27981182e-02, -6.82510609e-02,
        -6.37040036e-02, -5.91569462e-02, -5.46098889e-02, -5.00628316e-02,
        -4.55157742e-02, -4.09687169e-02, -3.64216596e-02, -3.18746022e-02,
        -2.73275449e-02, -2.27804876e-02, -1.82334302e-02, -1.36863729e-02,
        -9.13931558e-03, -4.59225825e-03, -4.52009165e-05]),
 <BarContainer object of 30 artists>)
```



In [282]:
```
plt.figure(figsize=(12,8))
plt.hist(anomalies3['scores'], bins=30)
```

Out[282]:
```
(array([ 159.,  139.,  160.,  155.,  184.,  345.,  208.,  176.,  214.,
         209.,  263.,  200.,  208.,  366.,  294.,  379.,  383.,  367.,
         248.,  366.,  455.,  395.,  728.,  856., 2797., 1946., 1381.,
        1561., 1619., 1962.]),
 array([-1.61289108e-01, -1.55913557e-01, -1.50538006e-01, -1.45162454e-01,
        -1.39786903e-01, -1.34411352e-01, -1.29035801e-01, -1.23660249e-01,
        -1.18284698e-01, -1.12909147e-01, -1.07533596e-01, -1.02158044e-01,
        -9.67824932e-02, -9.14069420e-02, -8.60313908e-02, -8.06558395e-02,
        -7.52802883e-02, -6.99047370e-02, -6.45291858e-02, -5.91536346e-02,
        -5.37780833e-02, -4.84025321e-02, -4.30269808e-02, -3.76514296e-02,
        -3.22758783e-02, -2.69003271e-02, -2.15247759e-02, -1.61492246e-02,
        -1.07736734e-02, -5.39812212e-03, -2.25708809e-05]),
 <BarContainer object of 30 artists>)
```