In [1]:
```
#Import libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import norm
```

In [2]:
```
ttps://drive.google.com/uc?export=download&id=1flc3xQQRt3qatXNGRsTfnV6BPYs0rhCu"

ataset_path)
```

Out[2]: (912000, 4)

In [3]:
```
# print data
pd.set_option('display.max_columns',None)
pd.set_option('display.max_rows',None)
df.head(6)
```

Out[3]:

	Time	Signal1	Signal2	Label
0	2017-08-14 11:37:49.791	33668.091913	0.856726	A
1	2017-08-14 11:37:49.801	32564.903040	0.856705	A
2	2017-08-14 11:37:49.811	31454.043305	0.856683	A
3	2017-08-14 11:37:49.821	30335.387166	0.856659	A
4	2017-08-14 11:37:49.831	29207.142938	0.856634	A
5	2017-08-14 11:37:49.841	28066.378446	0.856607	A

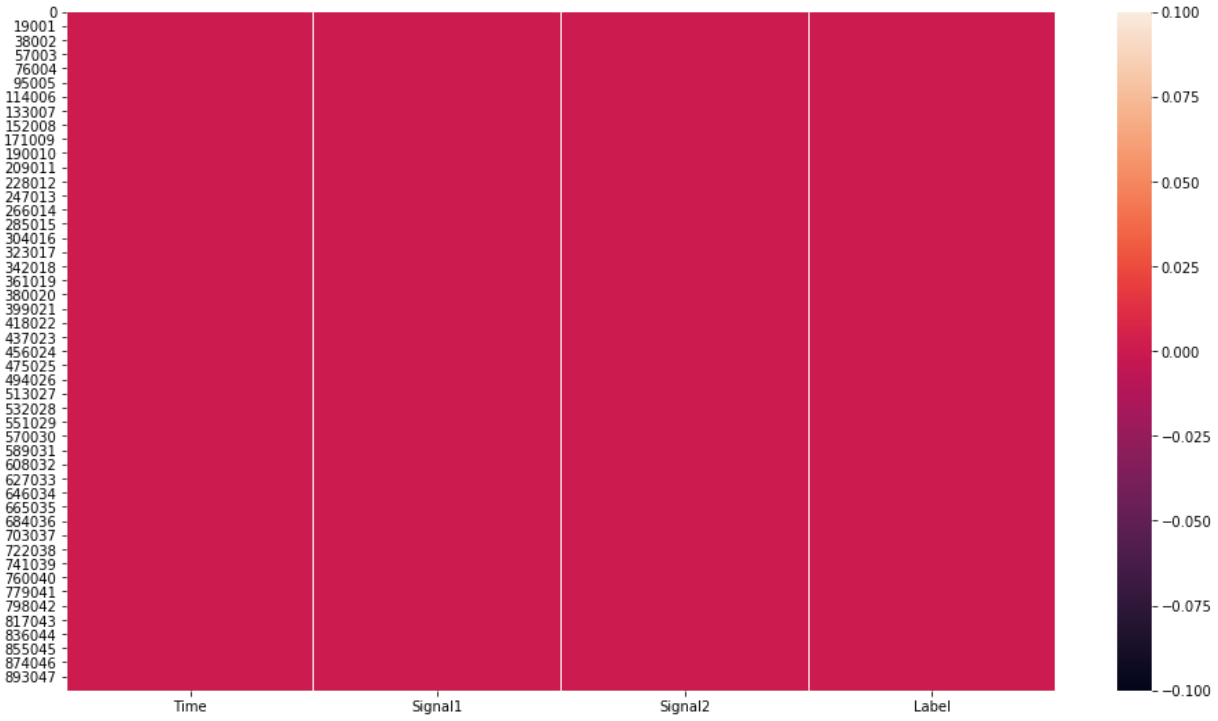
In [4]:
```
df.info() # information of data type
df.isnull().sum()  # check missing data
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 912000 entries, 0 to 911999
Data columns (total 4 columns):
 #   Column   Non-Null Count   Dtype  
---  ------   --------------   -----  
 0   Time     912000 non-null  object 
 1   Signal1  912000 non-null  float64
 2   Signal2  912000 non-null  float64
 3   Label    912000 non-null  object 
dtypes: float64(2), object(2)
memory usage: 27.8+ MB
```

Out[4]:
```
Time       0
Signal1    0
Signal2    0
Label      0
dtype: int64
```

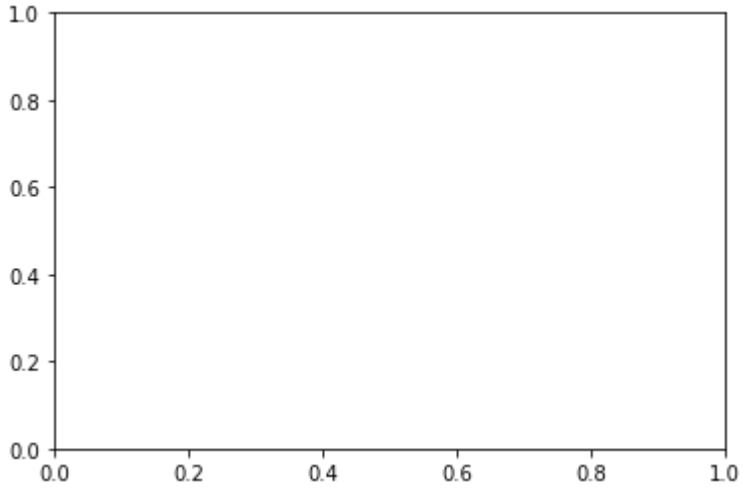
In [5]:
```
# create heatmap chart s for checking data is clear or not
plt.figure(figsize=(16,9))  # set size of figure
sns.heatmap(df.isnull())
```

Out[5]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x27b1e37e508>
```

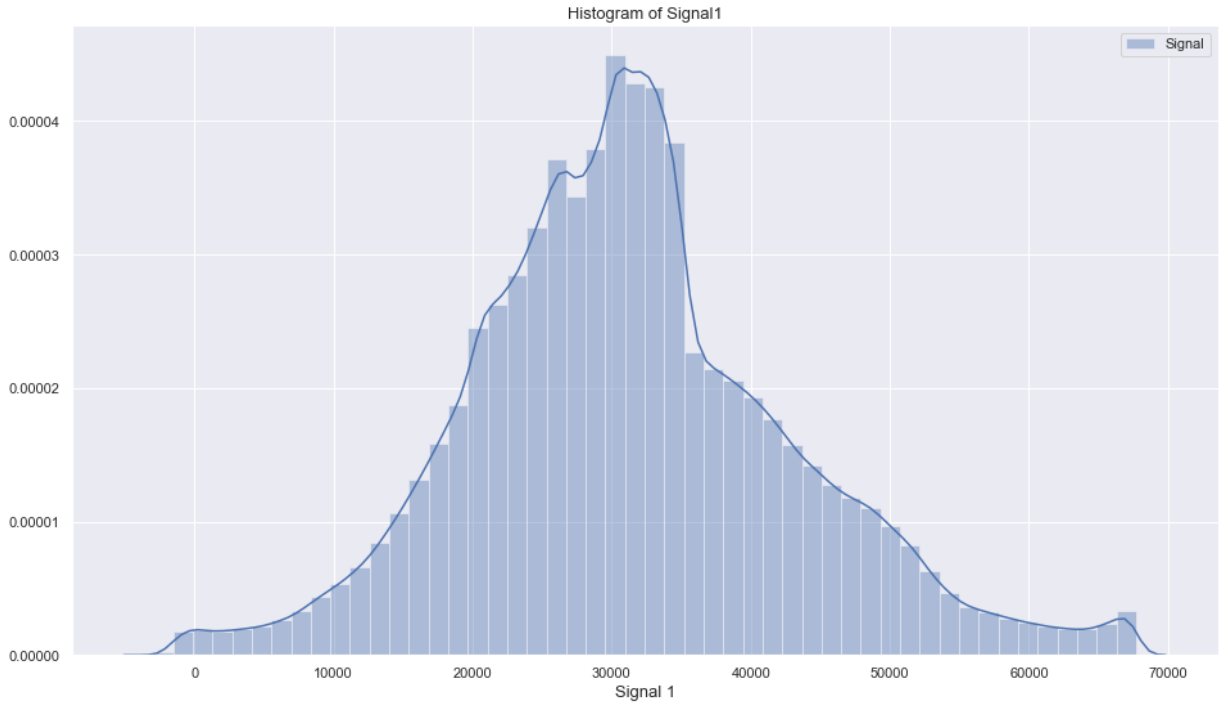


In [6]:
```
plt.plot(x= "Signal1", y = "Label", data = df)
```

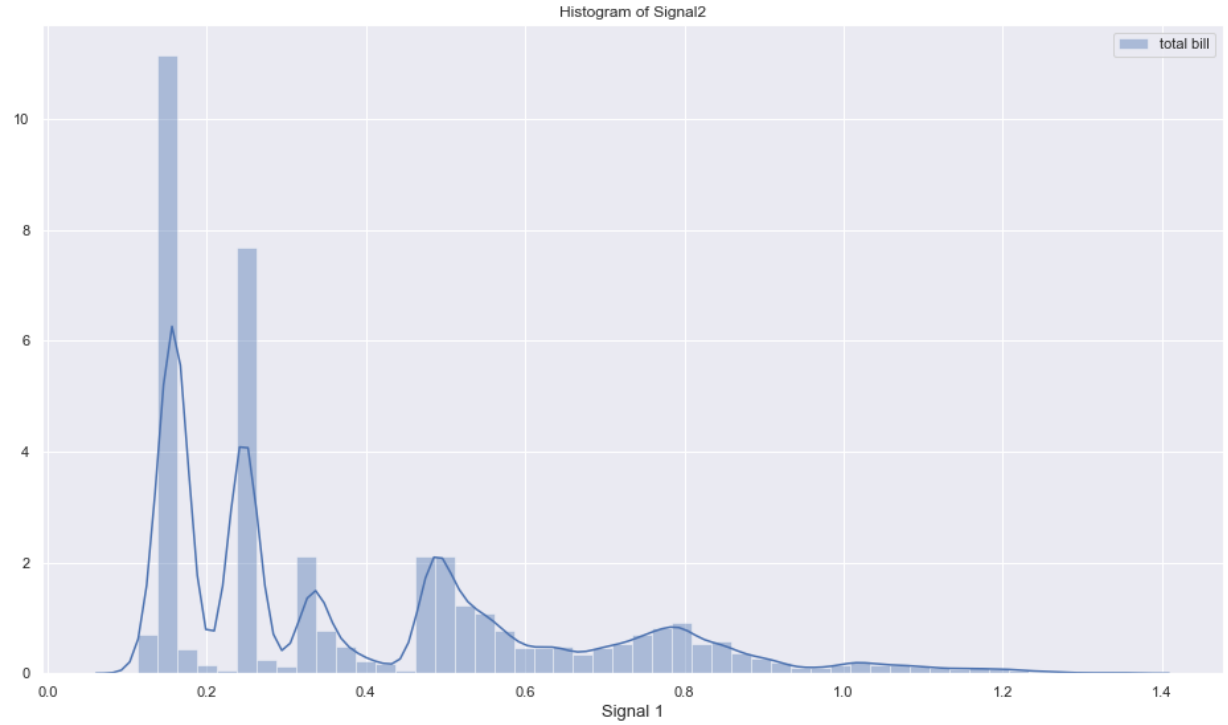
```
    173         if not args:

TypeError: plot got an unexpected keyword argument 'x'
```



In [7]:
```
# Data analysis using dffrent type of method
#using distplot analysis Signal 1
plt.figure(figsize=(16,9))
sns.set()
sns.distplot(df["Signal1"], label = "Signal")
plt.title("Histogram of Signal1", fontsize =13)
plt.xlabel("Signal 1", fontsize =13)
plt.legend()  # for label
plt.show()
```

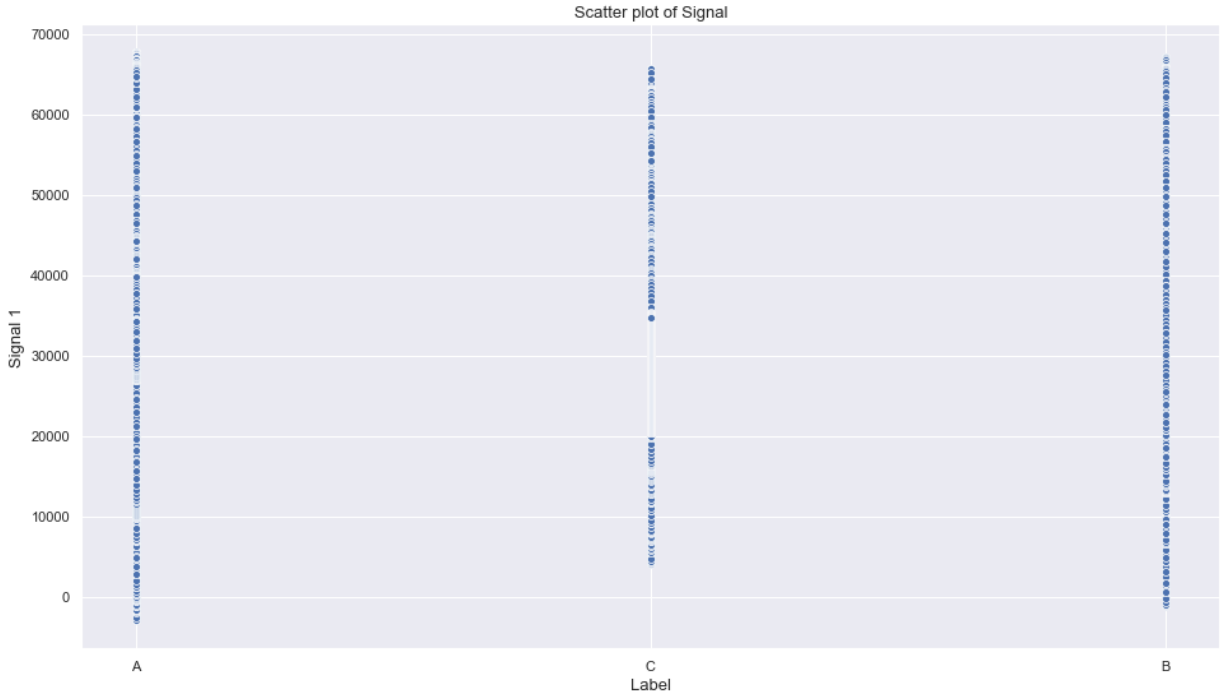


In [8]:
```
#using distplot analysis Signal 2
plt.figure(figsize=(16,9))
sns.set()
sns.distplot(df["Signal2"], label = "total bill")
plt.title("Histogram of Signal2")
plt.xlabel("Signal 1", fontsize =13)
plt.legend()  # for label
plt.show()
```



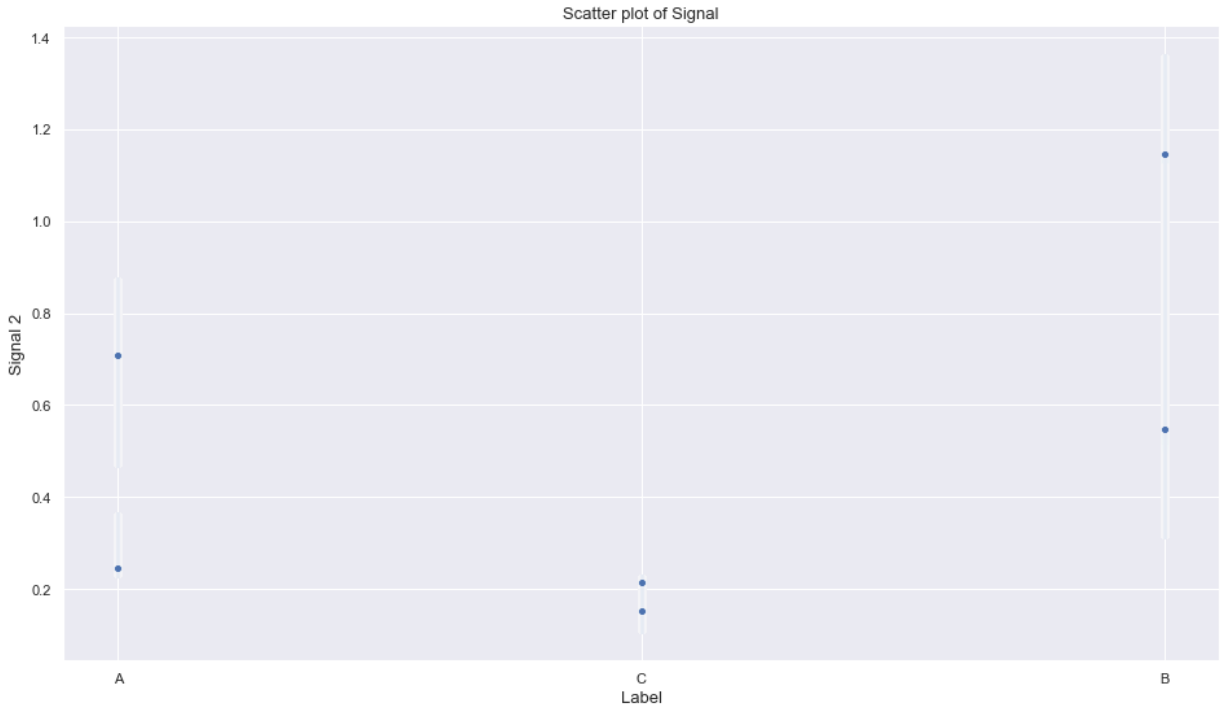
In [9]:
```
# analysis Signal 1 vs Label of data
plt.figure(figsize=(16,9))

sns.scatterplot(x='Label',y = 'Signal1', data = df)
plt.title("Scatter plot of Signal", fontsize =13)
plt.xlabel("Label", fontsize =13)
plt.ylabel("Signal 1", fontsize =13)
```

Out[9]:
```
Text(0, 0.5, 'Signal 1')
```



In [10]:
```
# analysis Signal 1 vs Label of data

plt.figure(figsize=(16,9))

sns.scatterplot(x='Label',y = 'Signal2', data = df)
plt.title("Scatter plot of Signal", fontsize =13)
plt.xlabel("Label", fontsize =13)
plt.ylabel("Signal 2", fontsize =13)
```

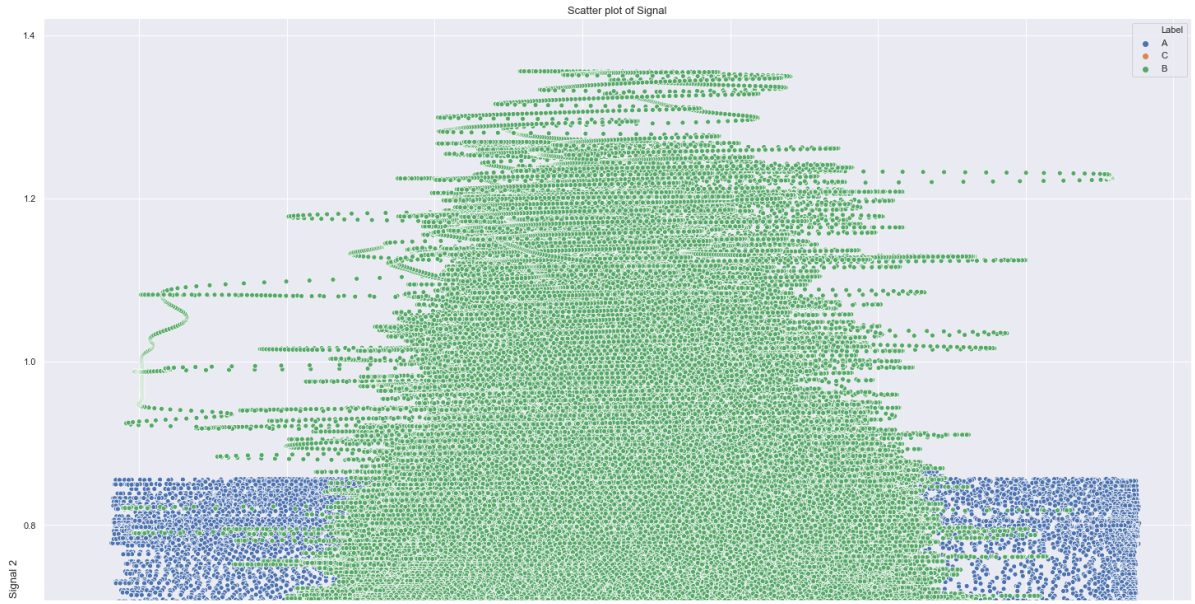
Out[10]: Text(0, 0.5, 'Signal 2')



In [11]:
```
#sns.scatterplot(x='Time',y = 'Signal2', data = df)
```

In [12]:
```
# define relation Signal_1 vs signal_2
plt.figure(figsize=(25,25))
sns.scatterplot(x='Signal1',y = 'Signal2', data = df, hue = 'Label')
plt.title("Scatter plot of Signal", fontsize =13)
plt.xlabel("Signal 1", fontsize =13)
plt.ylabel("Signal 2", fontsize =13)
```

Out[12]: Text(0, 0.5, 'Signal 2')



In [ ]: