Erwin Antepuesto

PROGRAMMING ASSIGNMENT

Data: https://archive.ics.uci.edu (https://archive.ics.uci.edu)

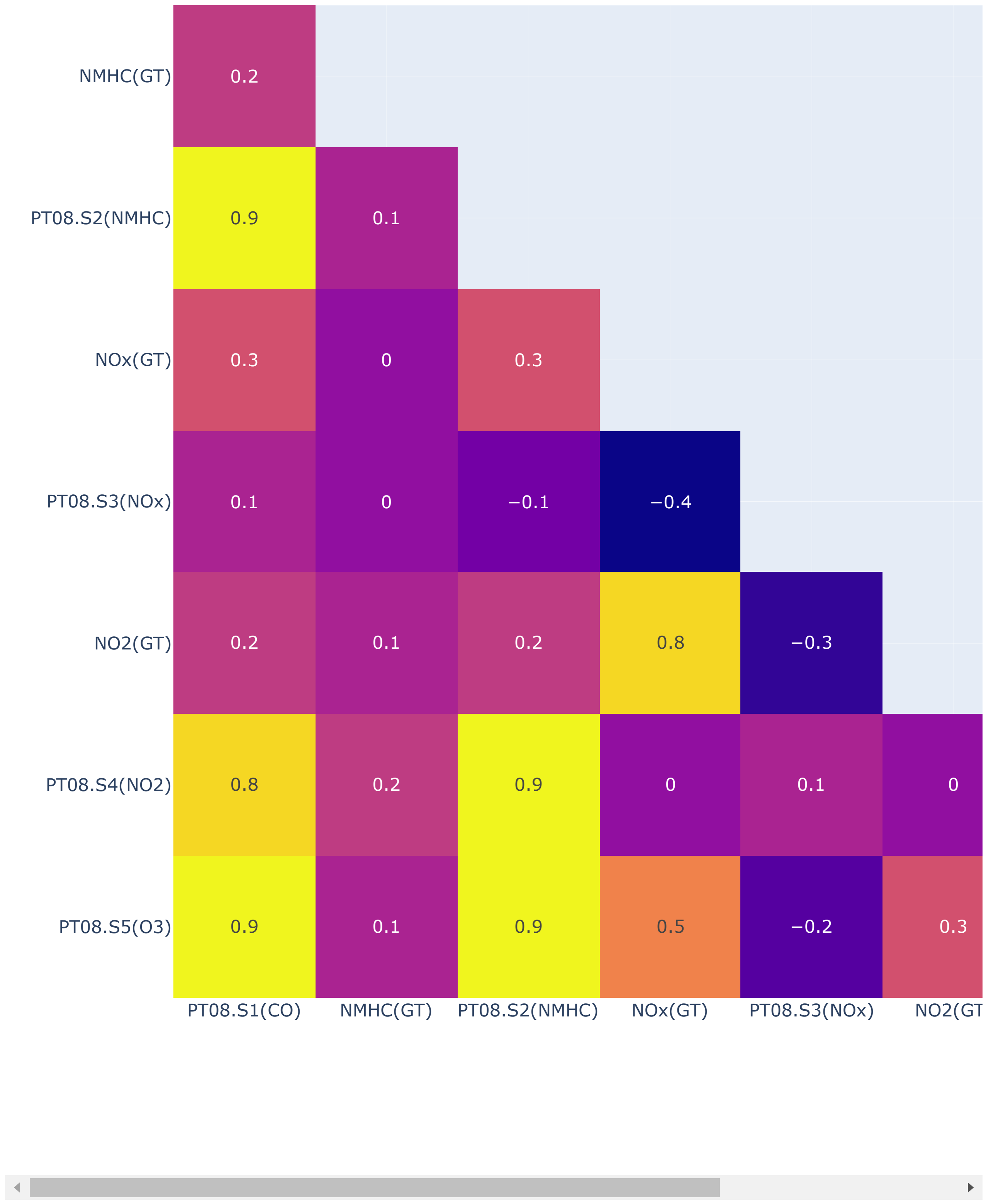
Instructions: Choose a dataset of your liking and perform the following:

1. Create a Correlation Plot
2. Check the distribution of each column and determine which probability distiribution it fits.
3. Create a summary statistics.
4. Perform a hypothesis test (Code from scratch).

1.Create a Correlation Plot

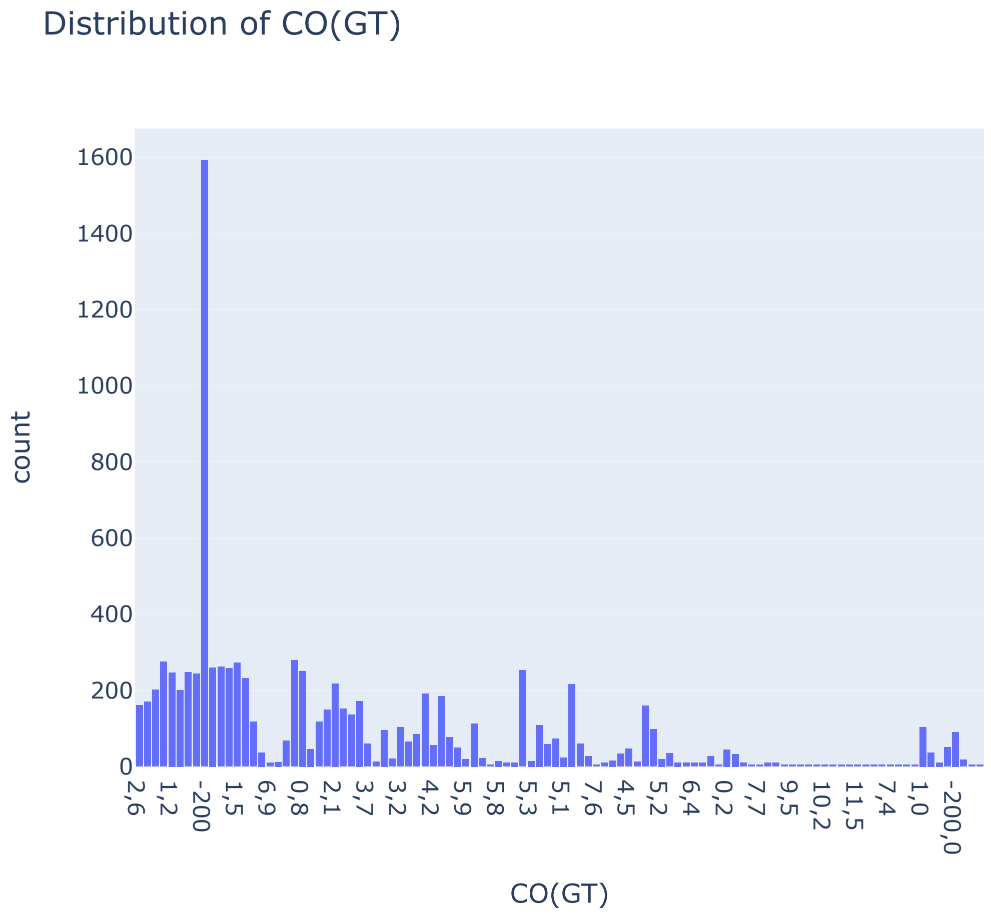
In [20]:

```
import pandas as pd
import numpy as np
import plotly.express as px

# Load the Air Quality dataset
df = pd.read_csv('AirQualityUCI.csv', delimiter=';')

# Correlation with explicit numeric_only parameter
df_corr = df.corr(numeric_only=True).round(1)

# Mask to matrix
mask = np.zeros_like(df_corr, dtype=bool)
mask[np.triu_indices_from(mask)] = True

# Visualization with keyword arguments in dropna
df_corr_viz = df_corr.mask(mask).dropna(how='all', axis=0).dropna(how='all', axis=1)
fig = px.imshow(df_corr_viz, text_auto=True)
fig.update_layout(height=900, width=900)
fig.show()
```

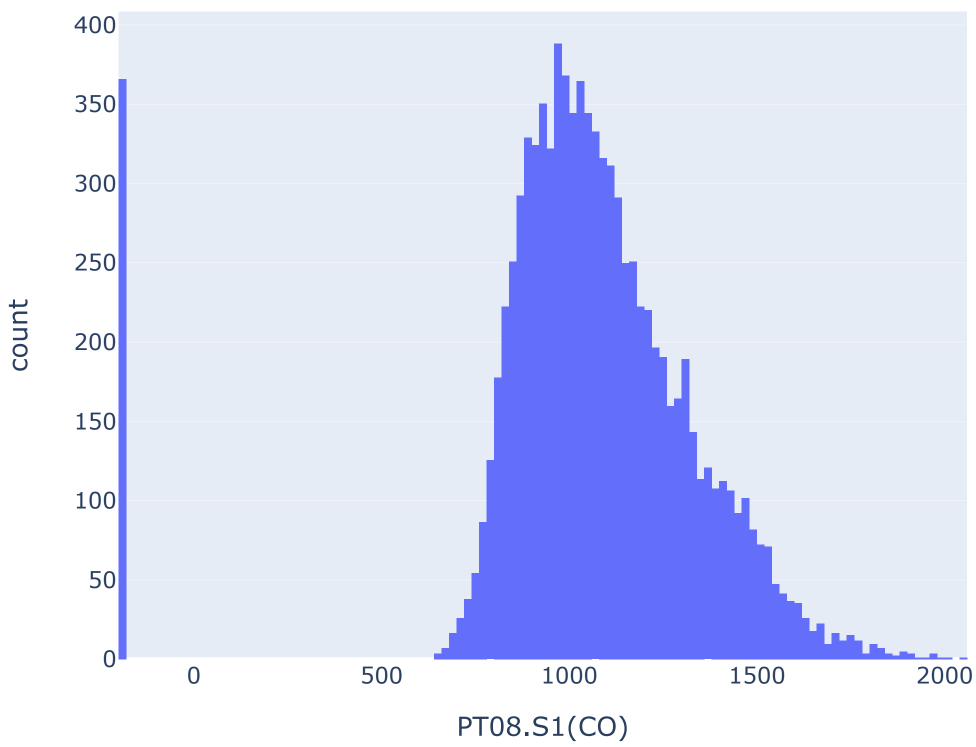

NMHC(GT)
PT08.S2(NMHC)
NOx(GT)
PT08.S3(NOx)
NO2(GT)
PT08.S4(NO2)
PT08.S5(O3)
PT08.S1(CO)
NMHC(GT)
PT08.S2(NMHC)
NOx(GT)
PT08.S3(NOx)
NO2(GT
0.2
0.9
0.1
0.3
0
0.3
0.1
0
−0.1
−0.4
0.2
0.1
0.2
0.8
−0.3
0.8
0.2
0.9
0
0.1
0
0.9
0.1
0.9
0.5
−0.2
0.3

2.Check the distribution of each column and determine which probability distiribution it fits.

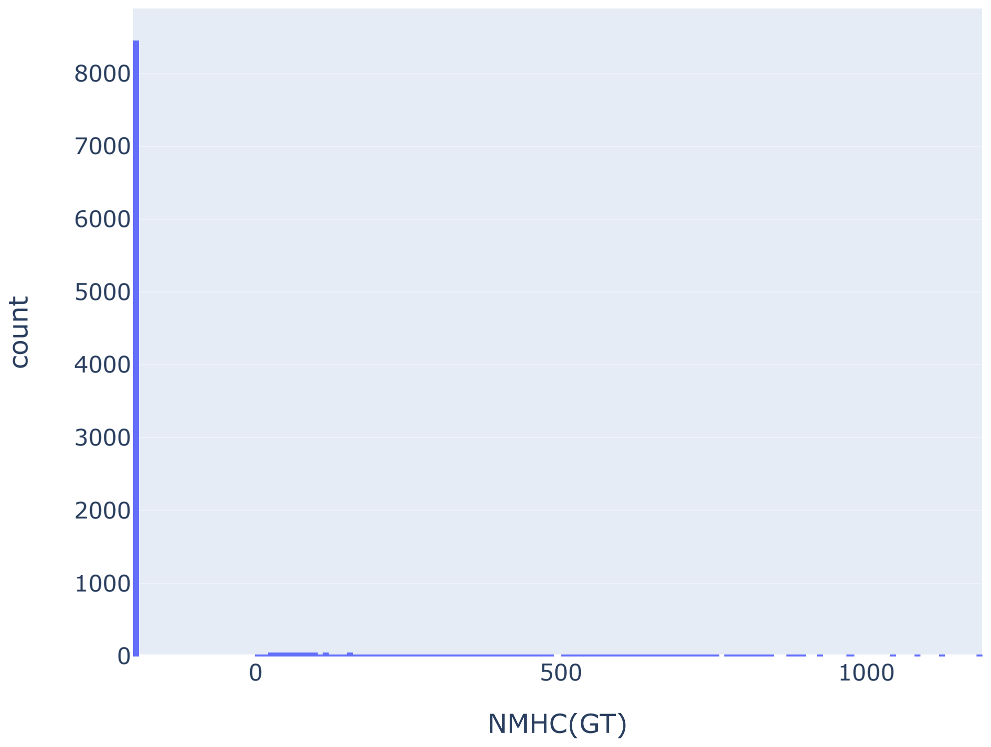
In [49]:
```
import pandas as pd
import plotly.express as px

# Load the dataset
data = pd.read_csv('AirQualityUCI.csv', delimiter=';')

# Remove the 'Date' and 'Time' columns
data = data.drop(['Date', 'Time'], axis=1)

# Visualize the distribution of each column using Plotly Express
fig = px.histogram(data, x='CO(GT)', title=f'Distribution of CO(GT)')
fig.show()

# Visualize the distribution of each column using Plotly Express
fig = px.histogram(data, x='PT08.S1(CO)', title=f'Distribution of PT08.S1(CO)')
fig.show()

# Visualize the distribution of each column using Plotly Express
fig = px.histogram(data, x='NMHC(GT)', title=f'Distribution of NMHC(GT)')
fig.show()
```



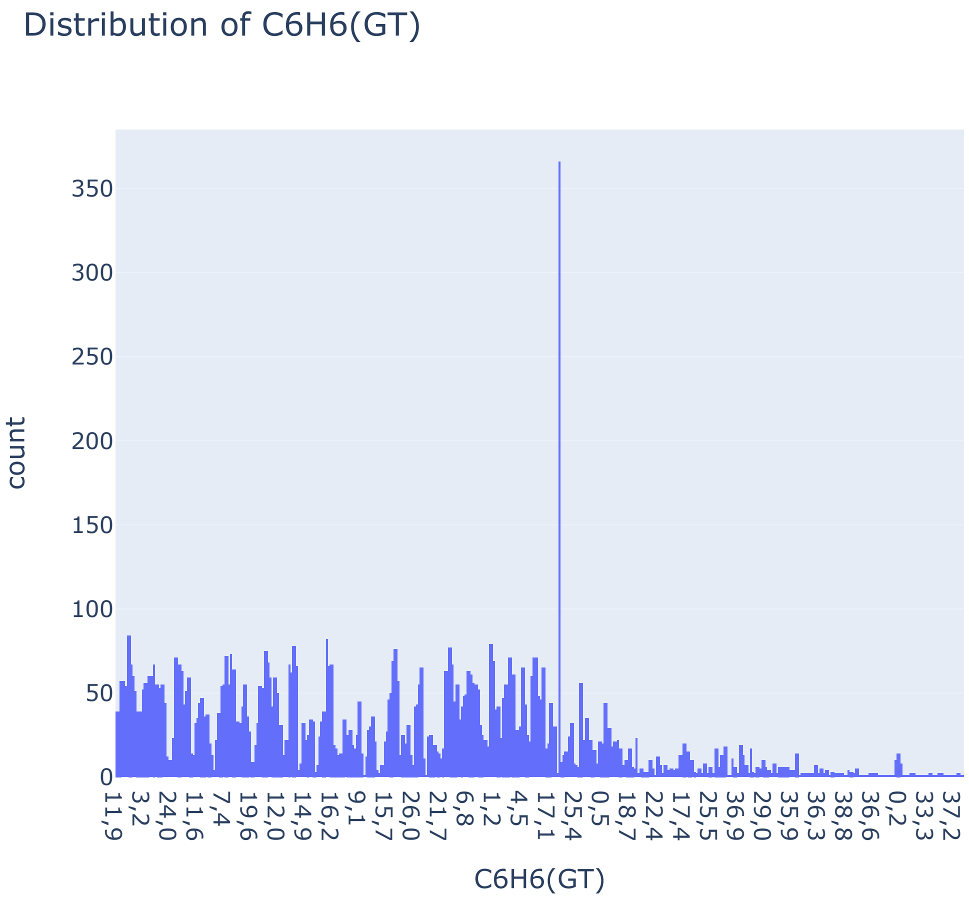
Distribution of PT08.S1(CO)

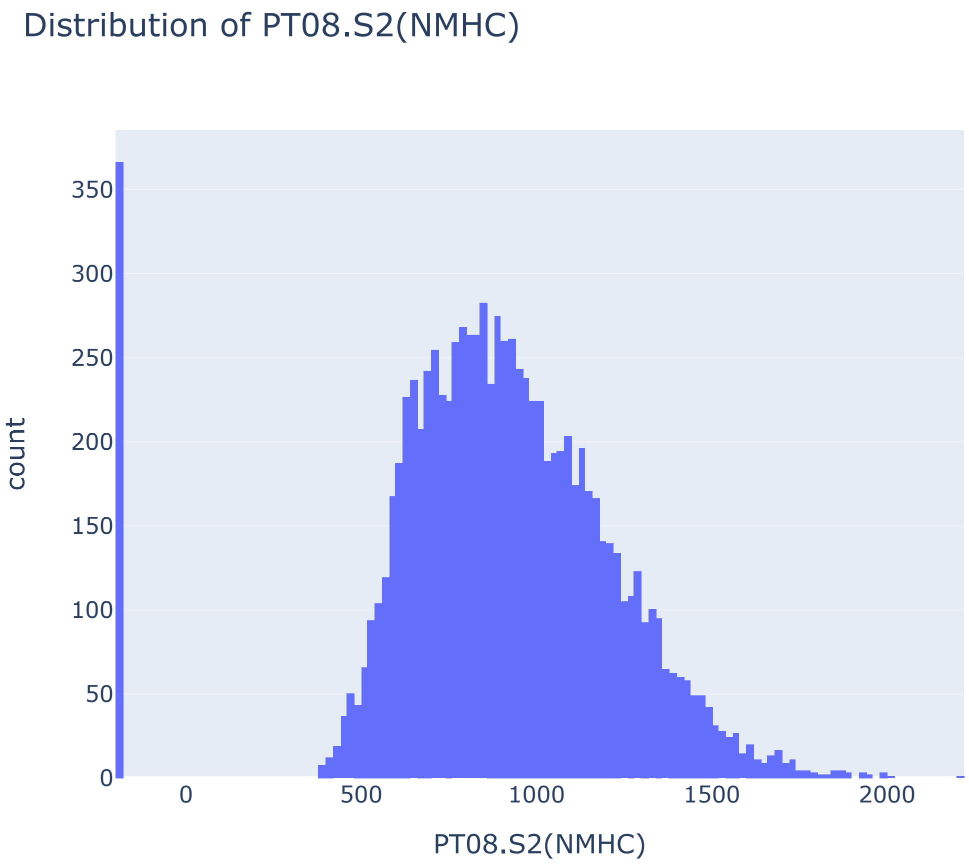


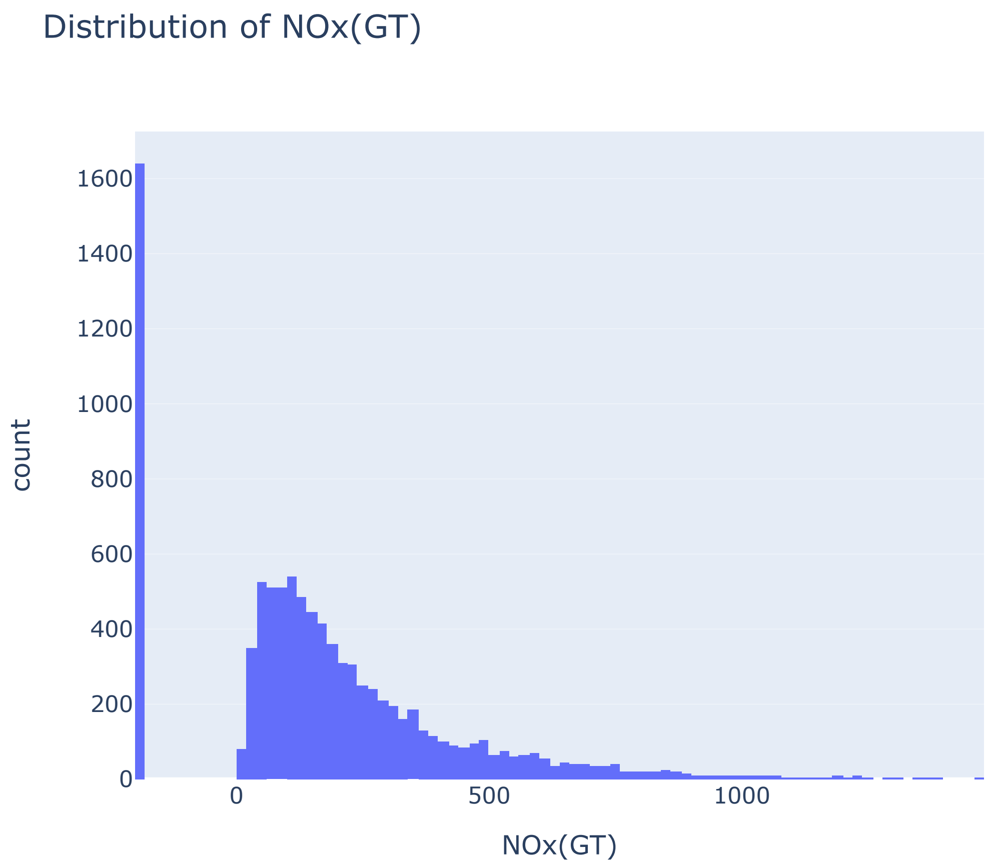
Distribution of NMHC(GT)



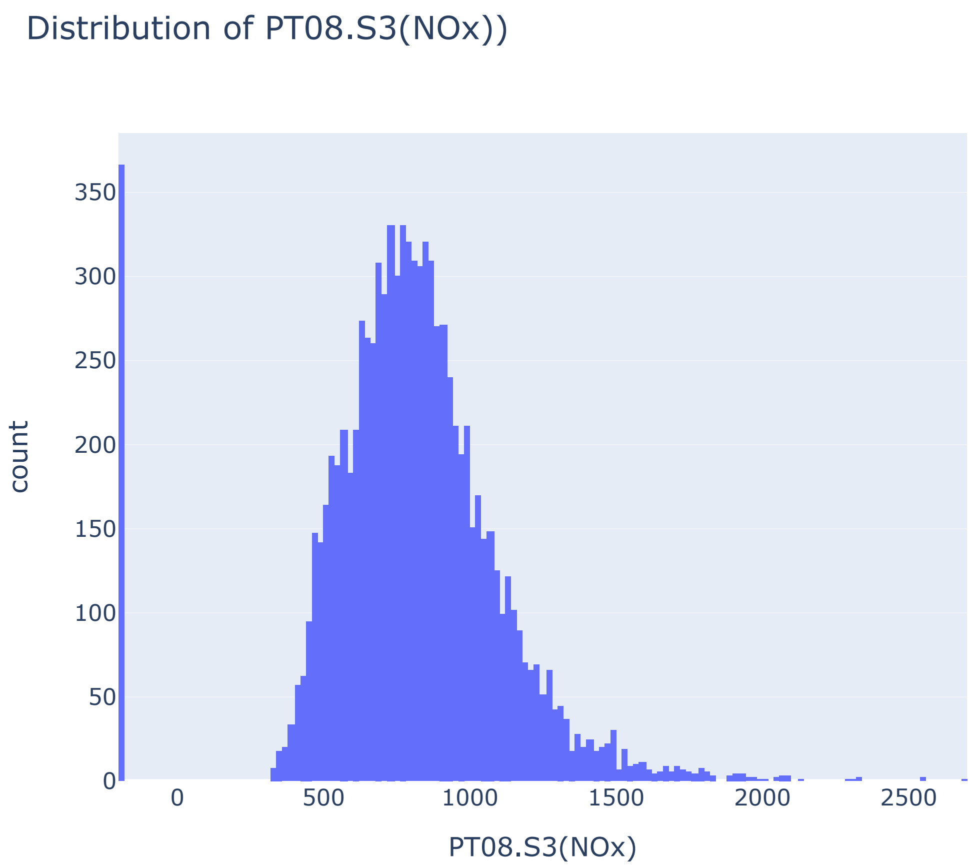
```
In [50]: fig = px.histogram(data, x='C6H6(GT)', title=f'Distribution of C6H6(GT)')
         fig.show()

         fig = px.histogram(data, x='PT08.S2(NMHC)', title=f'Distribution of PT08.S2(NMHC)')
         fig.show()

         fig = px.histogram(data, x='NOx(GT)', title=f'Distribution of NOx(GT)')
         fig.show()
```

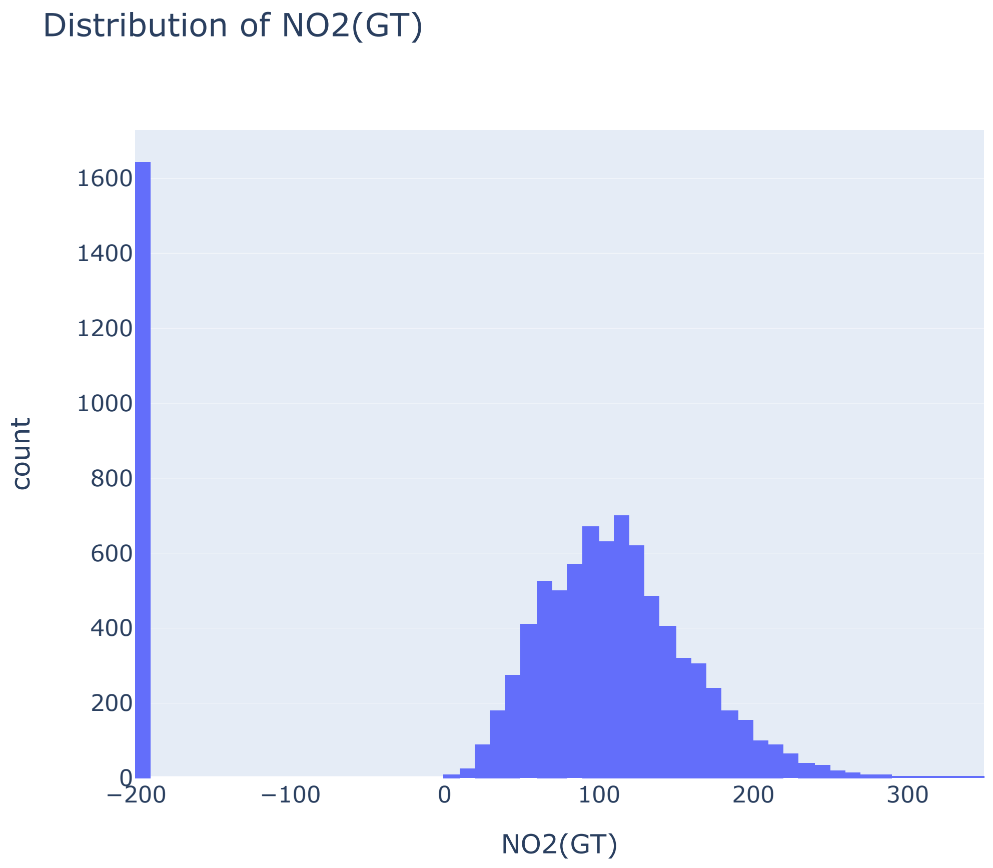


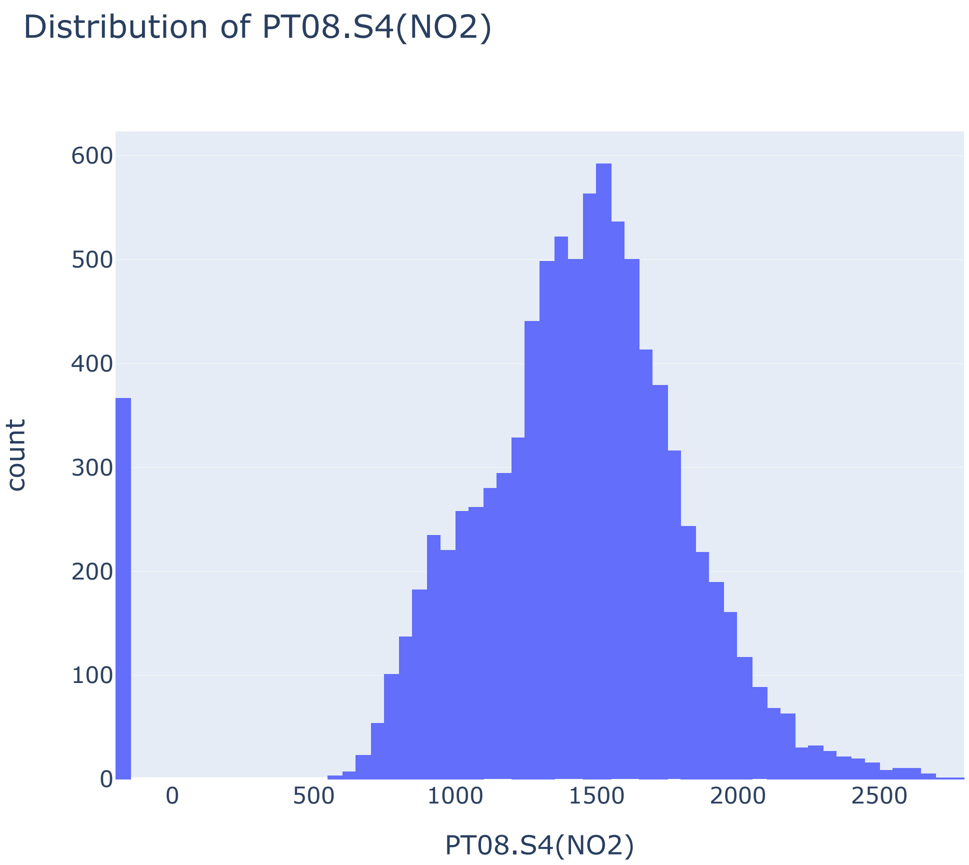



Distribution of NOx(GT)
count
1600
1400
1200
1000
800
600
400
200
0
0
500
1000
NOx(GT)

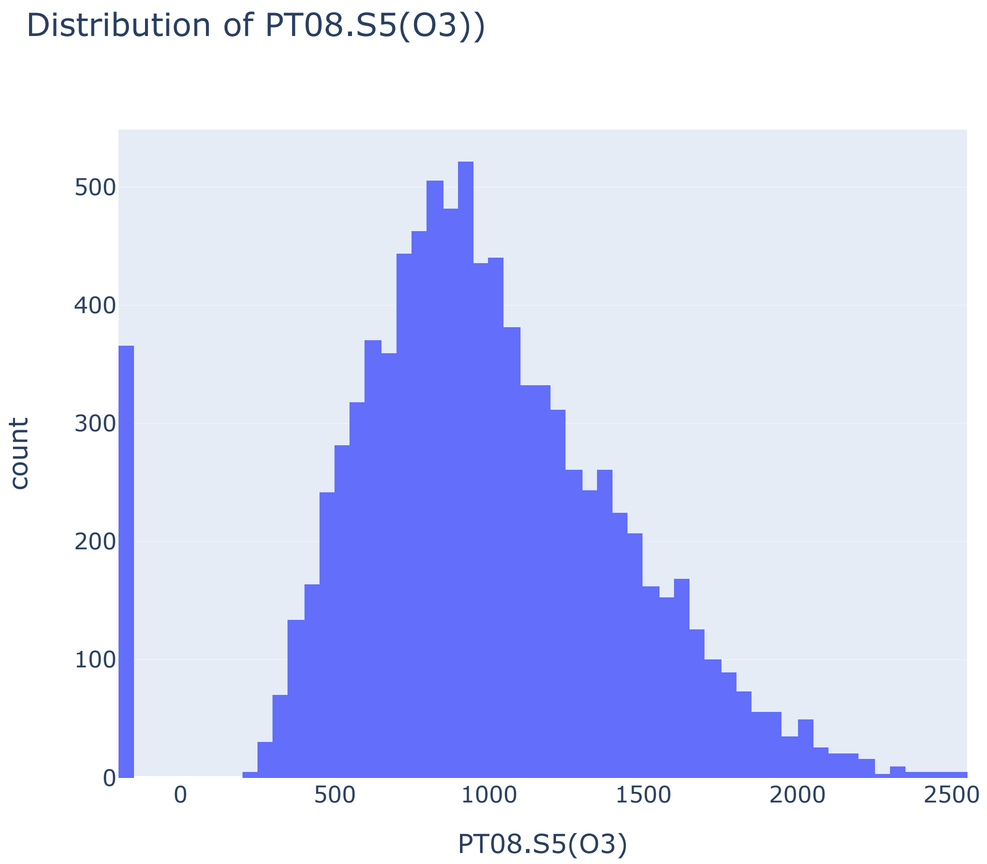
In [51]:
```
fig = px.histogram(data, x='PT08.S3(NOx)', title=f'Distribution of PT08.S3(NOx))')
fig.show()

fig = px.histogram(data, x='NO2(GT)', title=f'Distribution of NO2(GT)')
fig.show()

fig = px.histogram(data, x='PT08.S4(NO2)', title=f'Distribution of PT08.S4(NO2)')
fig.show()
```

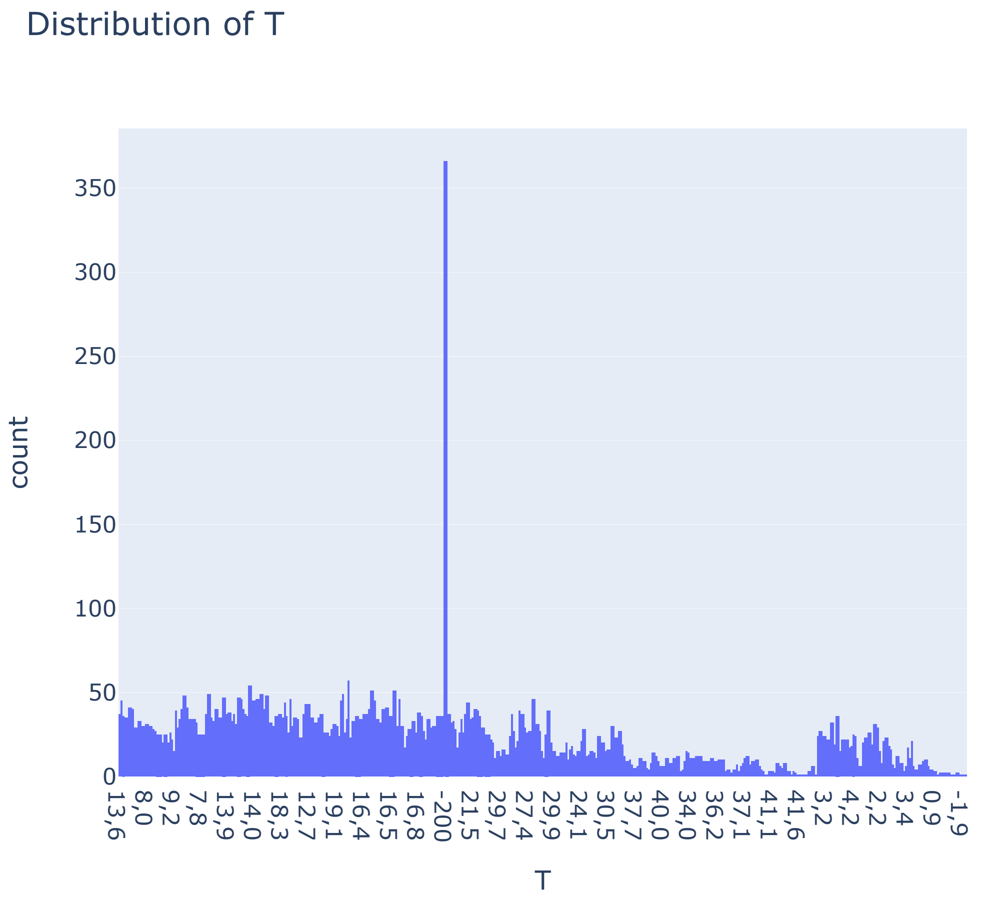




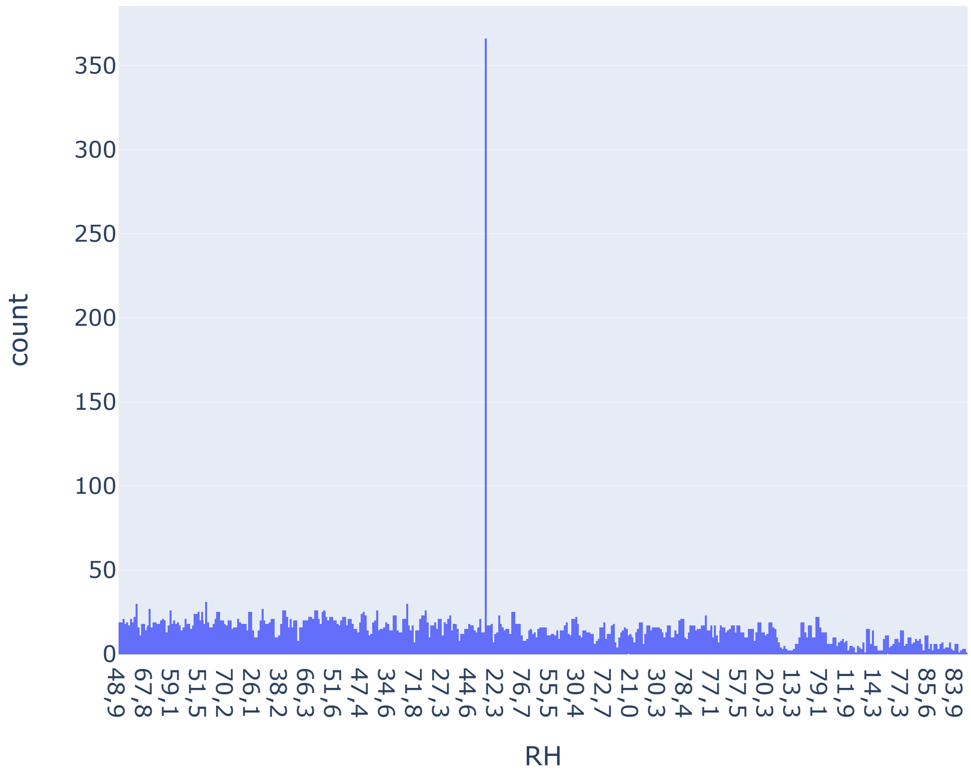

Distribution of PT08.S4(NO2)
count
0
100
200
300
400
500
600
0
500
1000
1500
2000
2500
PT08.S4(NO2)

```
In [53]: fig = px.histogram(data, x='PT08.S5(O3)', title=f'Distribution of PT08.S5(O3))')
         fig.show()

         fig = px.histogram(data, x='T', title=f'Distribution of T')
         fig.show()

         fig = px.histogram(data, x='RH', title=f'Distribution of RH')
         fig.show()
```



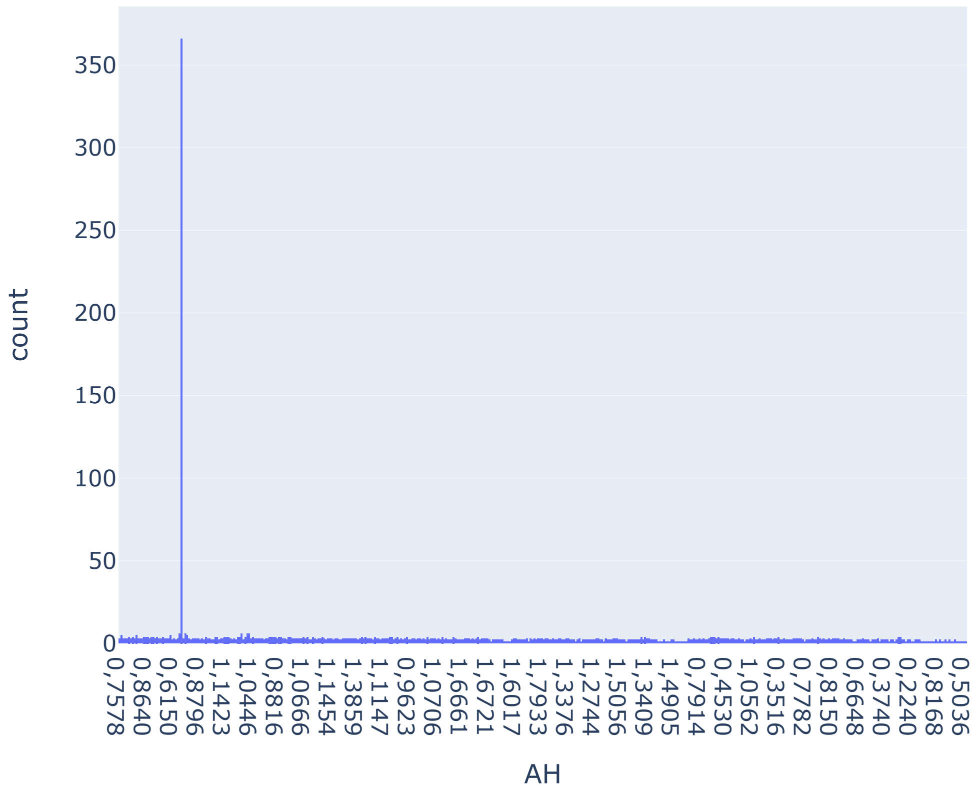


Distribution of RH



In [54]:
```
fig = px.histogram(data, x='AH', title=f'Distribution of AH')
fig.show()
```

Distribution of AH



```
In [28]: # Generate summary statistics
         summary_stats = data.describe()
         print(summary_stats)
```

```
       PT08.S1(CO)     NMHC(GT)  PT08.S2(NMHC)      NOx(GT)  PT08.S3(NOx)  \
count  9357.000000  9357.000000    9357.000000  9357.000000   9357.000000   
mean   1048.990061  -159.090093     894.595276   168.616971    794.990168   
std     329.832710   139.789093     342.333252   257.433866    321.993552   
min    -200.000000  -200.000000    -200.000000  -200.000000   -200.000000   
25%     921.000000  -200.000000     711.000000    50.000000    637.000000   
50%    1053.000000  -200.000000     895.000000   141.000000    794.000000   
75%    1221.000000  -200.000000    1105.000000   284.000000    960.000000   
max    2040.000000  1189.000000    2214.000000  1479.000000   2683.000000   

           NO2(GT)  PT08.S4(NO2)  PT08.S5(O3)  Unnamed: 15  Unnamed: 16  
count  9357.000000   9357.000000  9357.000000          0.0          0.0  
mean     58.148873   1391.479641   975.072032          NaN          NaN  
std     126.940455    467.210125   456.938184          NaN          NaN  
min    -200.000000   -200.000000  -200.000000          NaN          NaN  
25%      53.000000   1185.000000   700.000000          NaN          NaN  
50%      96.000000   1446.000000   942.000000          NaN          NaN  
75%     133.000000   1662.000000  1255.000000          NaN          NaN  
max     340.000000   2775.000000  2523.000000          NaN          NaN  
```

```
In [ ]:
```