

# Project 6: Predicting the Future Transaction from Large and Imbalanced Banking Dataset

## Datasets

### 1. Predicting the Future Transaction from Large and Imbalanced Banking Dataset

In this competition, Santander is seeking assistance from Kagglers to predict whether customers will make a particular transaction in the future, regardless of the transaction amount. The dataset provided for this challenge closely resembles the real data Santander has for solving this problem.

The dataset is anonymized, with each row consisting of 200 numerical values identified solely by a number.

Our approach involves analyzing the data, preprocessing it to make it suitable for modeling, training a predictive model, and using the model to make predictions on the test set. Finally, we will prepare a submission based on the predicted target values.

- Descriptive statistics:
  - Building Summary
  - Calculate Central/Dispersion measures
    - Mean, STD, ...
  - Get the distribution of the data (each column)
    - Define the distribution based on the plot
  - Analyze relationships between features
    - Correlation (Heatmap) and Pair plot

```
In [1]: # Libraries
```

```
import random
import numpy as np
import pandas as pd
from scipy import stats
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [2]: data = pd.read_csv('data_santander-customer-transaction-prediction.csv')
data
```

Out[2]:

	ID_code	target
0	test_0	0
1	test_1	0
2	test_2	0
3	test_3	0
4	test_4	0
...	...	...
199995	test_199995	0
199996	test_199996	0
199997	test_199997	0
199998	test_199998	0
199999	test_199999	0

200000 rows × 2 columns

In [3]:

```
data_training_set = pd.read_csv('train_set_santander-customer-transaction-prediction.csv')
data_training_set
```

Out[3]:

	ID_code	target	var_0	var_1	var_2	var_3	var_4	var_5	var_6	var_7
0	train_0	0	8.9255	-6.7863	11.9081	5.0930	11.4607	-9.2834	5.1187	18.6266
1	train_1	0	11.5006	-4.1473	13.8588	5.3890	12.3622	7.0433	5.6208	16.5338
2	train_2	0	8.6093	-2.7457	12.0805	7.8928	10.5825	-9.0837	6.9427	14.6155
3	train_3	0	11.0604	-2.1518	8.9522	7.1957	12.5846	-1.8361	5.8428	14.9250
4	train_4	0	9.8369	-1.4834	12.8746	6.6375	12.2772	2.4486	5.9405	19.2514
...	...	...	...	...	...	...	...	...	...	...
199995	train_199995	0	11.4880	-0.4956	8.2622	3.5142	10.3404	11.6081	5.6709	15.1516
199996	train_199996	0	4.9149	-2.4484	16.7052	6.6345	8.3096	-10.5628	5.8802	21.5940
199997	train_199997	0	11.2232	-5.0518	10.5127	5.6456	9.3410	-5.4086	4.5555	21.5571
199998	train_199998	0	9.7148	-8.6098	13.6104	5.7930	12.5173	0.5339	6.0479	17.0152
199999	train_199999	0	10.8762	-5.7105	12.1183	8.0328	11.5577	0.3488	5.2839	15.2058

200000 rows × 202 columns

In [4]:

```
data_test_set = pd.read_csv('test_set_santander-customer-transaction-prediction.csv')
data_test_set
```

Out[4]:

	ID_code	var_0	var_1	var_2	var_3	var_4	var_5	var_6	var_7	var_8
0	test_0	11.0656	7.7798	12.9536	9.4292	11.4327	-2.3805	5.8493	18.2675	2.1337
1	test_1	8.5304	1.2543	11.3047	5.1858	9.1974	-4.0117	6.0196	18.6316	-4.4131
2	test_2	5.4827	-10.3581	10.1407	7.0479	10.2628	9.8052	4.8950	20.2537	1.5233
3	test_3	8.5374	-1.3222	12.0220	6.5749	8.8458	3.1744	4.9397	20.5660	3.3755
4	test_4	11.7058	-0.1327	14.1295	7.7506	9.1035	-8.5848	6.8595	10.6048	2.9890
...	...	...	...	...	...	...	...	...	...	...
199995	test_199995	13.1678	1.0136	10.4333	6.7997	8.5974	-4.1641	4.8579	14.7625	-2.7239
199996	test_199996	9.7171	-9.1462	7.3443	9.1421	12.8936	3.0191	5.6888	18.8862	5.0915
199997	test_199997	11.6360	2.2769	11.2074	7.7649	12.6796	11.3224	5.3883	18.3794	1.6603
199998	test_199998	13.5745	-0.5134	13.6584	7.4855	11.2241	-11.3037	4.1959	16.8280	5.3208
199999	test_199999	10.4664	1.8070	10.2277	6.0654	10.0258	1.0789	4.8879	14.4892	-0.5902

200000 rows × 201 columns



## Descriptive statistics:

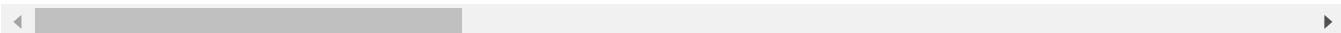
### Building Summary

In [5]: `data_training_set.describe()`

Out[5]:

	target	var_0	var_1	var_2	var_3	var_4
<b>count</b>	200000.000000	200000.000000	200000.000000	200000.000000	200000.000000	200000.000000
<b>mean</b>	0.100490	10.679914	-1.627622	10.715192	6.796529	11.078333
<b>std</b>	0.300653	3.040051	4.050044	2.640894	2.043319	1.623150
<b>min</b>	0.000000	0.408400	-15.043400	2.117100	-0.040200	5.074800
<b>25%</b>	0.000000	8.453850	-4.740025	8.722475	5.254075	9.883175
<b>50%</b>	0.000000	10.524750	-1.608050	10.580000	6.825000	11.108250
<b>75%</b>	0.000000	12.758200	1.358625	12.516700	8.324100	12.261125
<b>max</b>	1.000000	20.315000	10.376800	19.353000	13.188300	16.671400

8 rows × 201 columns

In [6]: `data_test_set.describe()`

Out[6]:

	<b>var_0</b>	<b>var_1</b>	<b>var_2</b>	<b>var_3</b>	<b>var_4</b>	<b>var_5</b>
<b>count</b>	200000.000000	200000.000000	200000.000000	200000.000000	200000.000000	200000.000000
<b>mean</b>	10.658737	-1.624244	10.707452	6.788214	11.076399	-5.050558
<b>std</b>	3.036716	4.040509	2.633888	2.052724	1.616456	7.869293
<b>min</b>	0.188700	-15.043400	2.355200	-0.022400	5.484400	-27.767000
<b>25%</b>	8.442975	-4.700125	8.735600	5.230500	9.891075	-11.201400
<b>50%</b>	10.513800	-1.590500	10.560700	6.822350	11.099750	-4.834100
<b>75%</b>	12.739600	1.343400	12.495025	8.327600	12.253400	0.942575
<b>max</b>	22.323400	9.385100	18.714100	13.142000	16.037100	17.253700

8 rows × 200 columns

## Calculate Central/Dispersion measures

- Mean, STD, ...

In [7]:

```
# Mean of the test data set
mean_test = data_test_set.mean()
mean_test
```

C:\Users\35840\AppData\Local\Temp\ipykernel\_12024\2074011161.py:2: FutureWarning:  
The default value of numeric\_only in DataFrame.mean is deprecated. In a future version, it will default to False. In addition, specifying 'numeric\_only=None' is deprecated. Select only valid columns or specify the value of numeric\_only to silence this warning.

```
mean_test = data_test_set.mean()
```

Out[7]:

```
var_0      10.658737
var_1     -1.624244
var_2      10.707452
var_3      6.788214
var_4     11.076399
...
var_195    -0.133657
var_196     2.290899
var_197     8.912428
var_198    15.869184
var_199    -3.246342
Length: 200, dtype: float64
```

In [8]:

```
# Mean of the training data set
mean_train = data_training_set.mean()
mean_train
```

C:\Users\35840\AppData\Local\Temp\ipykernel\_12024\4131632015.py:2: FutureWarning:  
The default value of numeric\_only in DataFrame.mean is deprecated. In a future version, it will default to False. In addition, specifying 'numeric\_only=None' is deprecated. Select only valid columns or specify the value of numeric\_only to silence this warning.

```
mean_train = data_training_set.mean()
```

```
Out[8]: target      0.100490
         var_0      10.679914
         var_1     -1.627622
         var_2      10.715192
         var_3      6.796529
         ...
         var_195   -0.142088
         var_196    2.303335
         var_197    8.908158
         var_198   15.870720
         var_199   -3.326537
Length: 201, dtype: float64
```

```
In [9]: # Median of the test data set
median_test = data_test_set.median()
median_test
```

```
C:\Users\35840\AppData\Local\Temp\ipykernel_12024\1283722594.py:2: FutureWarning:
The default value of numeric_only in DataFrame.median is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warning.
```

```
median_test = data_test_set.median()
```

```
Out[9]: target      10.51380
         var_0      10.56070
         var_1     -1.59050
         var_2      6.82235
         var_3      11.09975
         ...
         var_195   -0.16200
         var_196    2.40360
         var_197    8.89280
         var_198   15.94340
         var_199   -2.72595
Length: 200, dtype: float64
```

```
In [10]: # Median of the training data set
median_training = data_training_set.median()
median_training
```

```
C:\Users\35840\AppData\Local\Temp\ipykernel_12024\940557916.py:2: FutureWarning: The default value of numeric_only in DataFrame.median is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warning.
```

```
median_training = data_training_set.median()
```

```
Out[10]: target      0.00000
         var_0      10.52475
         var_1     -1.60805
         var_2      10.58000
         var_3      6.82500
         ...
         var_195   -0.17270
         var_196    2.40890
         var_197    8.88820
         var_198   15.93405
         var_199   -2.81955
Length: 201, dtype: float64
```

```
In [11]: # standard deviation of the training data set
std_training = data_training_set.std()
std_training
```

```
C:\Users\35840\AppData\Local\Temp\ipykernel_12024\2349147795.py:2: FutureWarning:
The default value of numeric_only in DataFrame.std is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warning.
```

```
    std_test = data_test_set.std()
```

```
Out[11]:
```

var_0	3.036716
var_1	4.040509
var_2	2.633888
var_3	2.052724
var_4	1.616456
	...
var_195	1.429678
var_196	5.446346
var_197	0.920904
var_198	3.008717
var_199	10.398589

Length: 200, dtype: float64

```
In [12]: # standard deviation of the test data set
std_training = data_training_set.std()
std_training
```

```
C:\Users\35840\AppData\Local\Temp\ipykernel_12024\2492939848.py:2: FutureWarning:
The default value of numeric_only in DataFrame.std is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warning.
```

```
    std_training = data_training_set.std()
```

```
Out[12]:
```

target	0.300653
var_0	3.040051
var_1	4.050044
var_2	2.640894
var_3	2.043319
	...
var_195	1.429372
var_196	5.454369
var_197	0.921625
var_198	3.010945
var_199	10.438015

Length: 201, dtype: float64

```
In [13]: # Variance of the test data set
var_test = data_test_set.var()
var_test
```

```
C:\Users\35840\AppData\Local\Temp\ipykernel_12024\4024112479.py:2: FutureWarning:
The default value of numeric_only in DataFrame.var is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warning.
```

```
    var_test = data_test_set.var()
```

```
Out[13]: var_0      9.221642
          var_1     16.325714
          var_2      6.937368
          var_3      4.213674
          var_4      2.612931
          ...
          var_195    2.043980
          var_196    29.662679
          var_197    0.848065
          var_198    9.052379
          var_199   108.130651
Length: 200, dtype: float64
```

```
In [14]: # Variance of the training data set
var_training = data_training_set.var()
var_training
```

C:\Users\35840\AppData\Local\Temp\ipykernel\_12024\975651850.py:2: FutureWarning: The default value of numeric\_only in DataFrame.var is deprecated. In a future version, it will default to False. In addition, specifying 'numeric\_only=None' is deprecated. Select only valid columns or specify the value of numeric\_only to silence this warning.

```
var_training = data_training_set.var()
```

```
Out[14]: target      0.090392
          var_0      9.241909
          var_1     16.402858
          var_2      6.974322
          var_3      4.175153
          ...
          var_195    2.043105
          var_196    29.750144
          var_197    0.849394
          var_198    9.065793
          var_199   108.952159
Length: 201, dtype: float64
```

```
In [15]: # Quartiles of the test data set

first_qurtile = data_test_set.quantile(0.25)
third_qurtile = data_test_set.quantile(0.75)

# Interquartile Range
iqr = third_qurtile - first_qurtile

print("First Qurtile: ")
print(first_qurtile)
print()
print("Third Qurtile: ")
print(third_qurtile)

print()

print("Inter-quartile range: ")
print(iqr)
```

```
C:\Users\35840\AppData\Local\Temp\ipykernel_12024\1751312427.py:3: FutureWarning:  
The default value of numeric_only in DataFrame.quantile is deprecated. In a future  
version, it will default to False. Select only valid columns or specify the value  
of numeric_only to silence this warning.  
    first_qurtile = data_test_set.quantile(0.25)  
C:\Users\35840\AppData\Local\Temp\ipykernel_12024\1751312427.py:4: FutureWarning:  
The default value of numeric_only in DataFrame.quantile is deprecated. In a future  
version, it will default to False. Select only valid columns or specify the value  
of numeric_only to silence this warning.  
    third_qurtile = data_test_set.quantile(0.75)  
First Qurtile:  
var_0      8.442975  
var_1     -4.700125  
var_2      8.735600  
var_3      5.230500  
var_4      9.891075  
...  
var_195   -1.160700  
var_196   -1.948600  
var_197    8.260075  
var_198   13.847275  
var_199   -11.124000  
Name: 0.25, Length: 200, dtype: float64  
  
Third Qurtile:  
var_0      12.739600  
var_1      1.343400  
var_2     12.495025  
var_3      8.327600  
var_4     12.253400  
...  
var_195    0.837900  
var_196    6.519800  
var_197    9.595900  
var_198   18.045200  
var_199    4.935400  
Name: 0.75, Length: 200, dtype: float64  
  
Inter-quartile range:  
var_0      4.296625  
var_1      6.043525  
var_2      3.759425  
var_3      3.097100  
var_4      2.362325  
...  
var_195    1.998600  
var_196    8.468400  
var_197    1.335825  
var_198    4.197925  
var_199   16.059400  
Length: 200, dtype: float64
```

In [16]: # Quartiles of the training data set

```
first_qurtile = data_training_set.quantile(0.25)  
third_qurtile = data_training_set.quantile(0.75)  
  
# Interquartile Range  
iqr = third_qurtile - first_qurtile  
  
print("First Qurtile: ")  
print(first_qurtile)  
print()  
print("Third Qurtile: ")
```

```
print(third_qurtile)

print()

print("Inter-quartile range: ")
print(iqr)
```

C:\Users\35840\AppData\Local\Temp\ipykernel\_12024\3738859874.py:3: FutureWarning:  
The default value of numeric\_only in DataFrame.quantile is deprecated. In a future  
version, it will default to False. Select only valid columns or specify the value  
of numeric\_only to silence this warning.

first\_qurtile = data\_training\_set.quantile(0.25)  
C:\Users\35840\AppData\Local\Temp\ipykernel\_12024\3738859874.py:4: FutureWarning:  
The default value of numeric\_only in DataFrame.quantile is deprecated. In a future  
version, it will default to False. Select only valid columns or specify the value  
of numeric\_only to silence this warning.

third\_qurtile = data\_training\_set.quantile(0.75)

First Qurtile:

target	0.000000
var_0	8.453850
var_1	-4.740025
var_2	8.722475
var_3	5.254075
...	
var_195	-1.170700
var_196	-1.946925
var_197	8.252800
var_198	13.829700
var_199	-11.208475

Name: 0.25, Length: 201, dtype: float64

Third Qurtile:

target	0.000000
var_0	12.758200
var_1	1.358625
var_2	12.516700
var_3	8.324100
...	
var_195	0.829600
var_196	6.556725
var_197	9.593300
var_198	18.064725
var_199	4.836800

Name: 0.75, Length: 201, dtype: float64

Inter-quartile range:

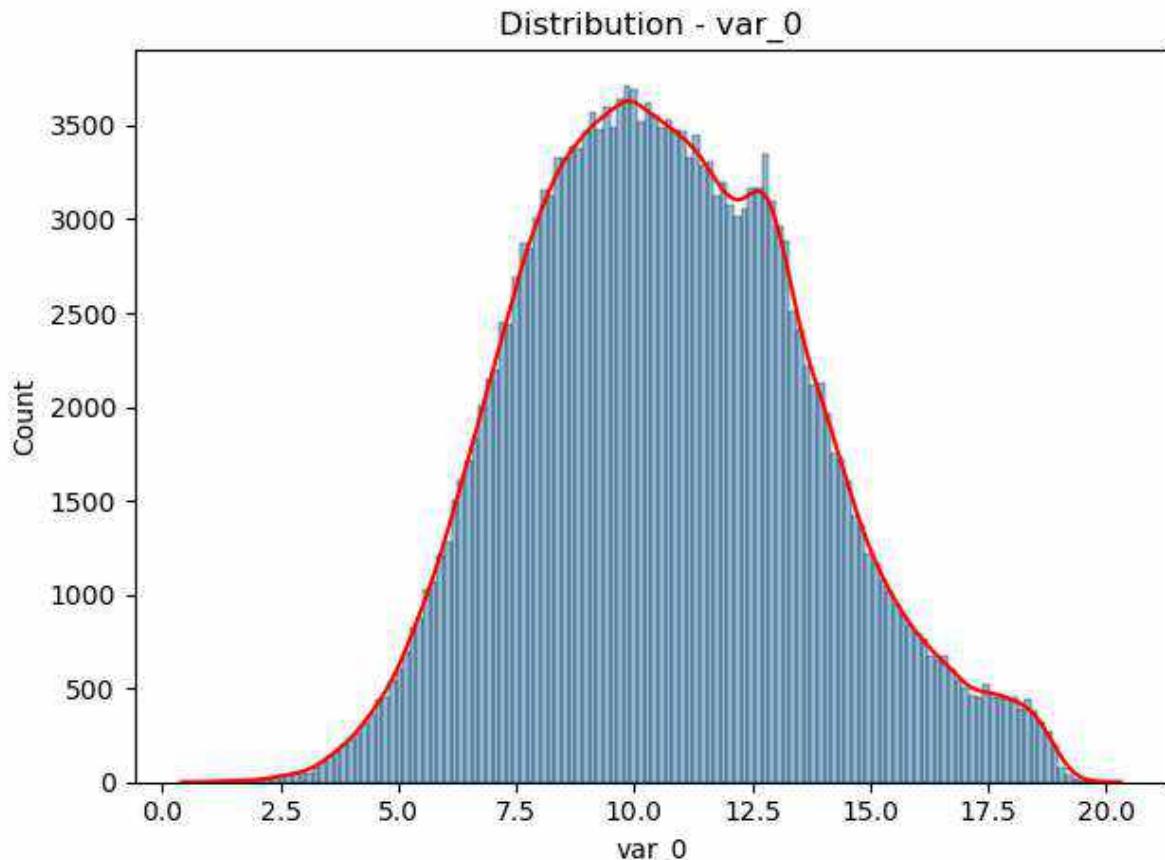
target	0.000000
var_0	4.304350
var_1	6.098650
var_2	3.794225
var_3	3.070025
...	
var_195	2.000300
var_196	8.503650
var_197	1.340500
var_198	4.235025
var_199	16.045275

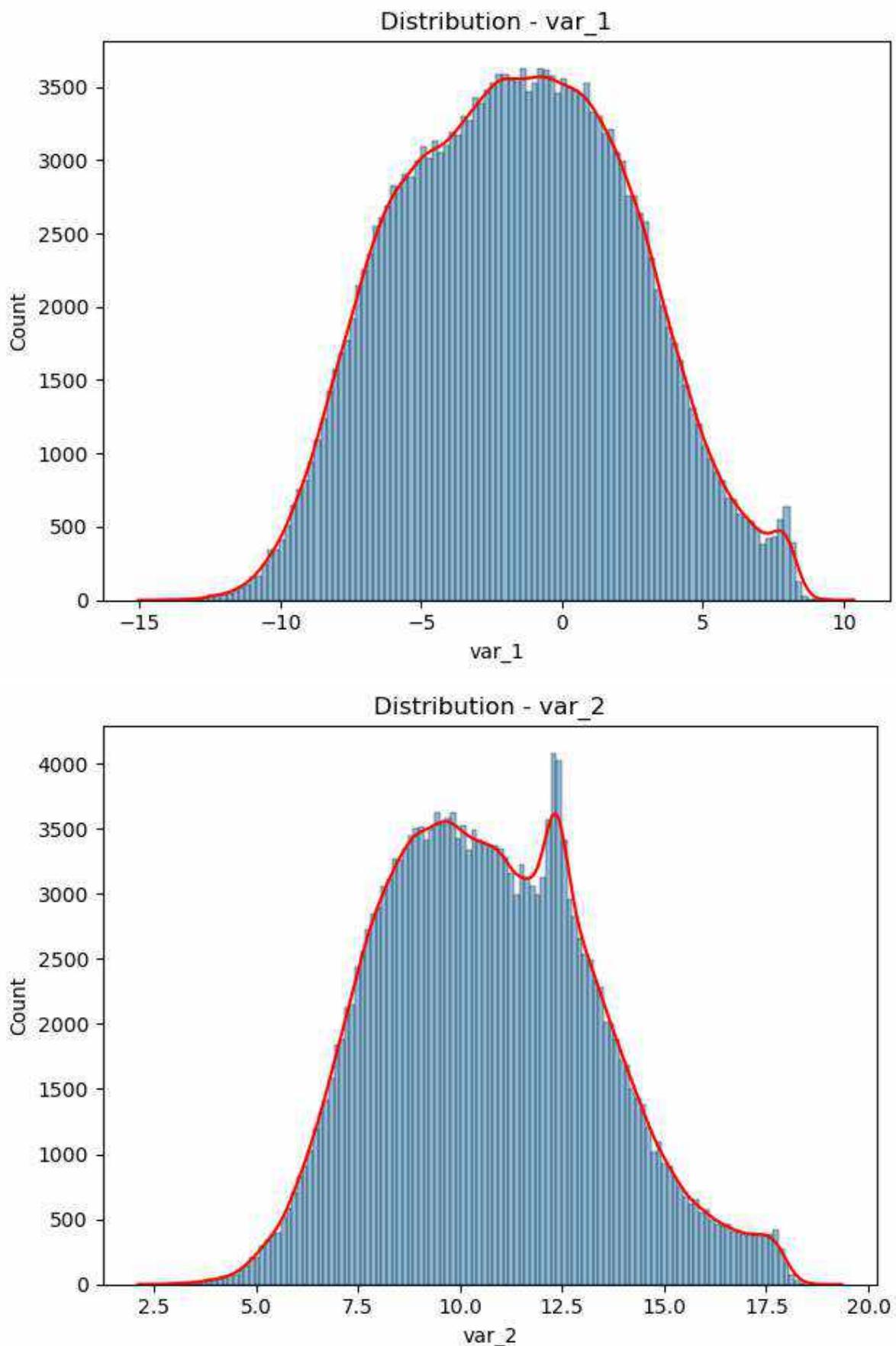
Length: 201, dtype: float64

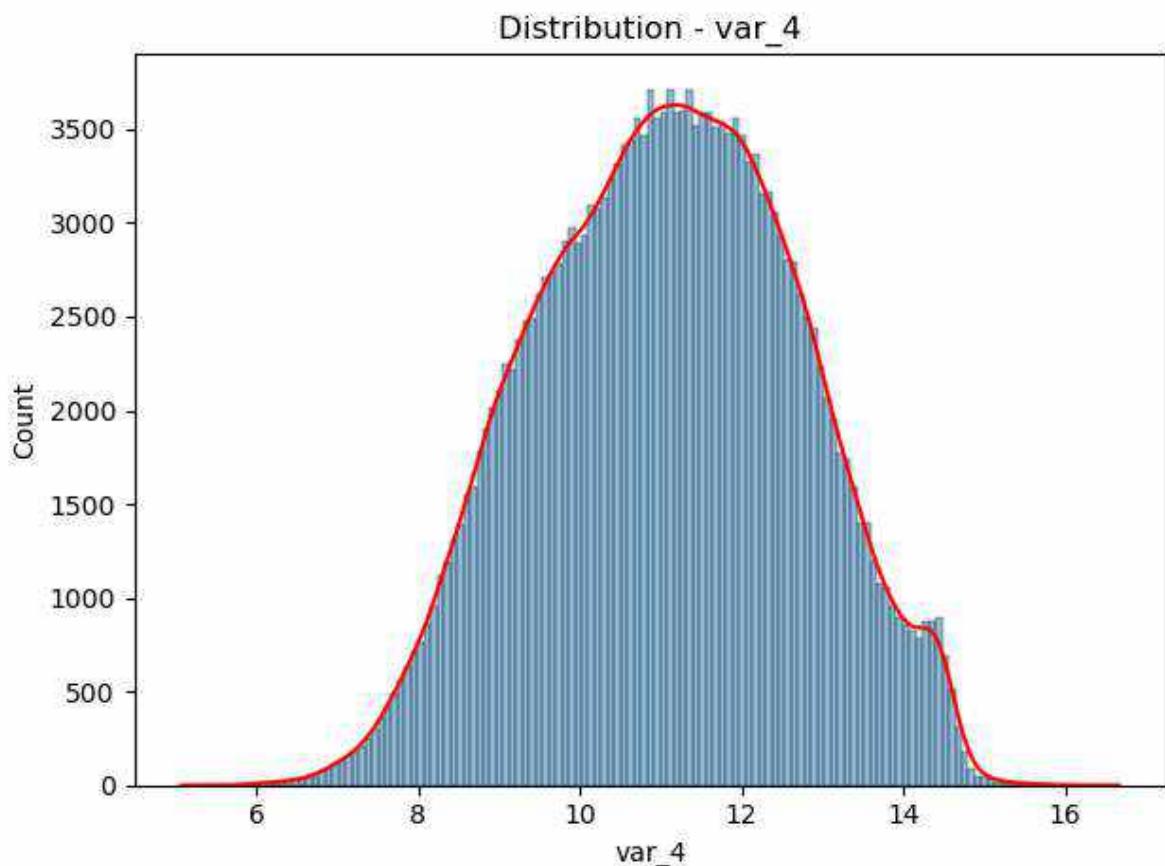
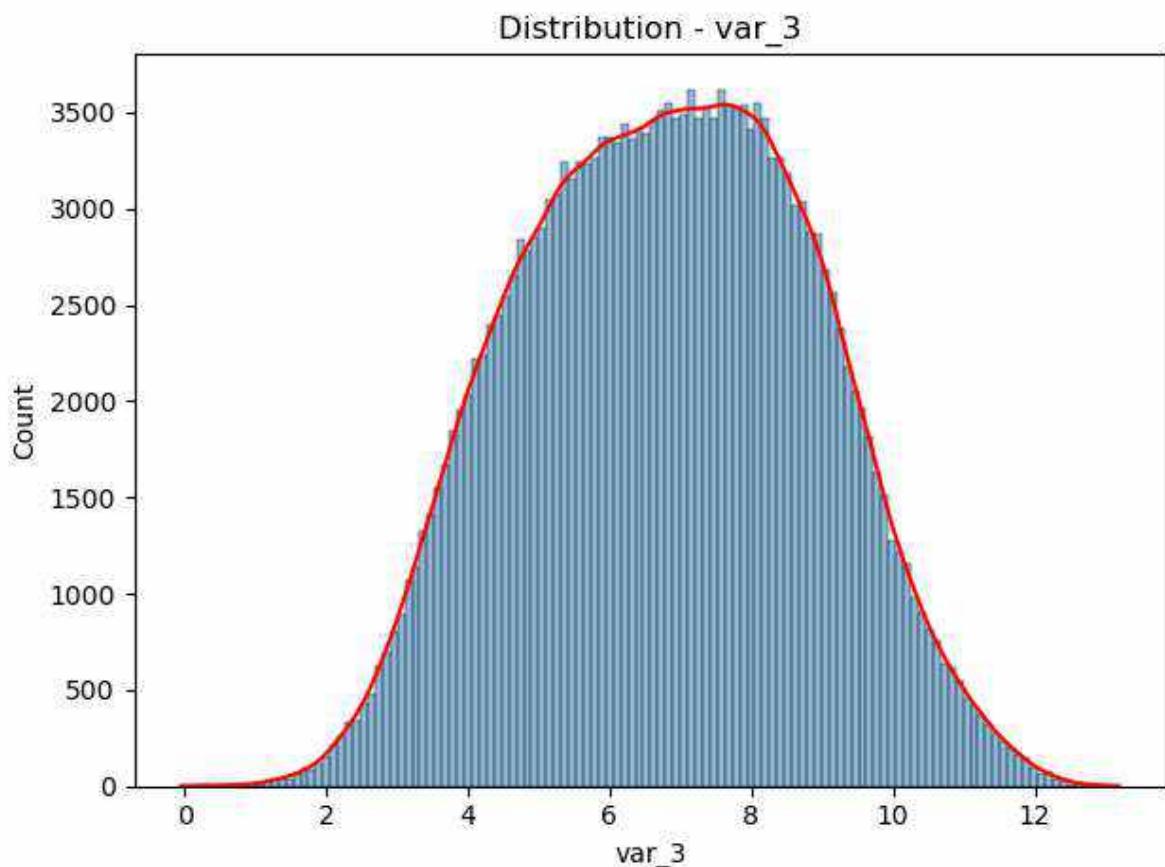
## Get the distribution of the data (each column)

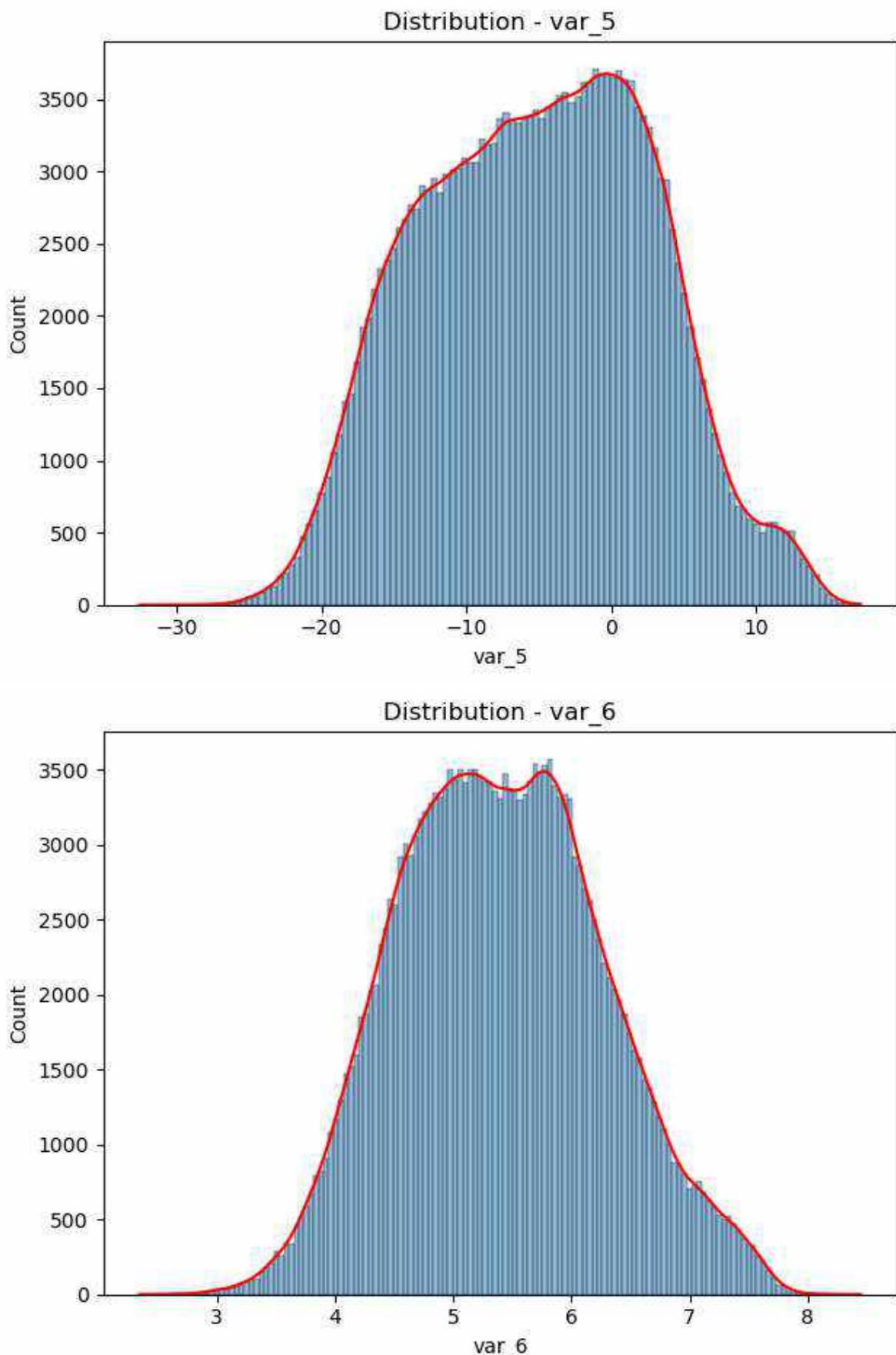
- Define the distribution based on the plot

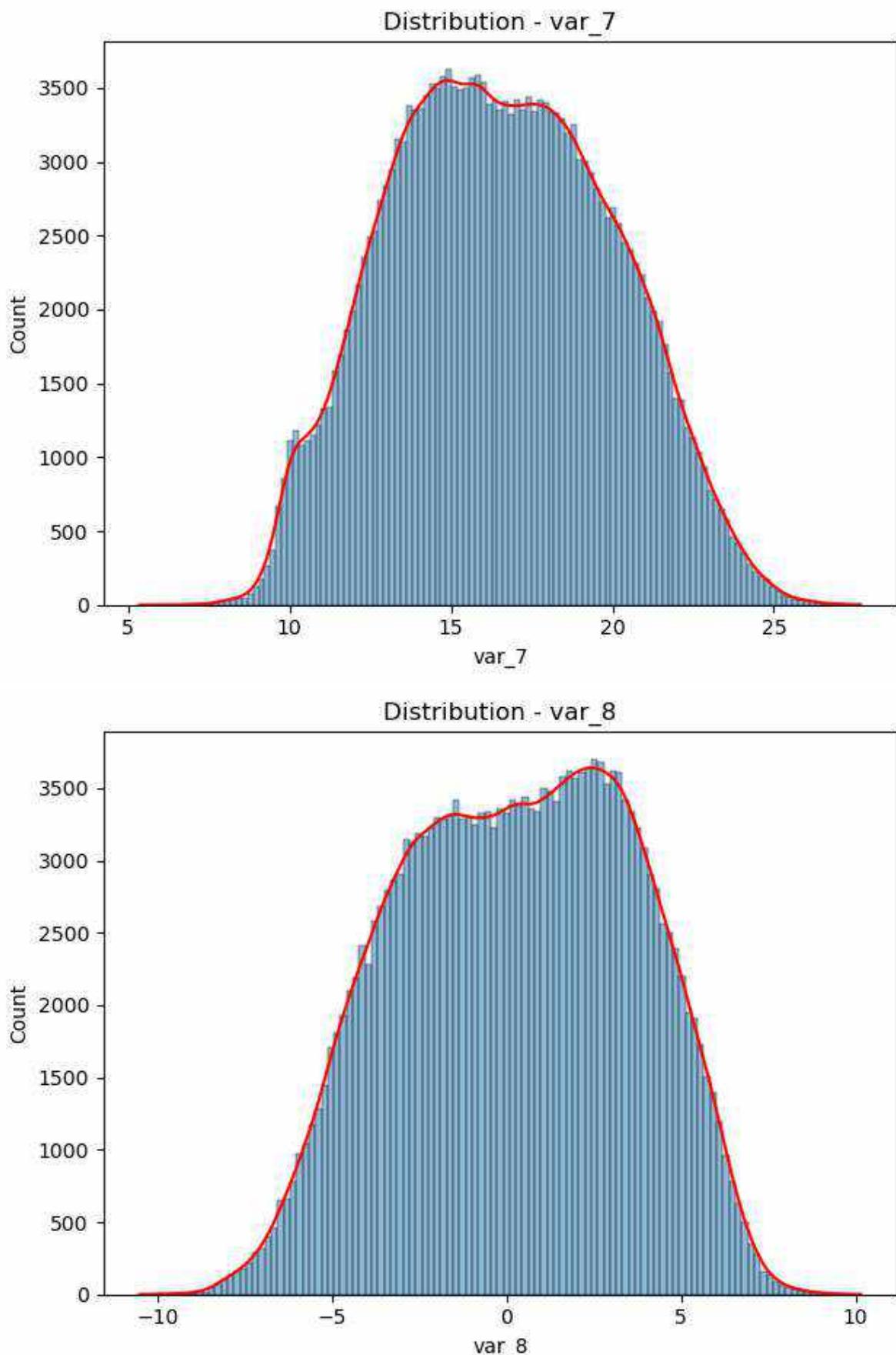
```
In [17]: for column in data_training_set.columns:  
    if column not in ['ID_code', 'target']:  
        ax = sns.histplot(data_training_set[column], kde=True)  
        ax.lines[0].set_color('red')  
        plt.title(f'Distribution - {column}')  
        plt.xlabel(column)  
        plt.ylabel('Count')  
        plt.tight_layout()  
        plt.show()
```

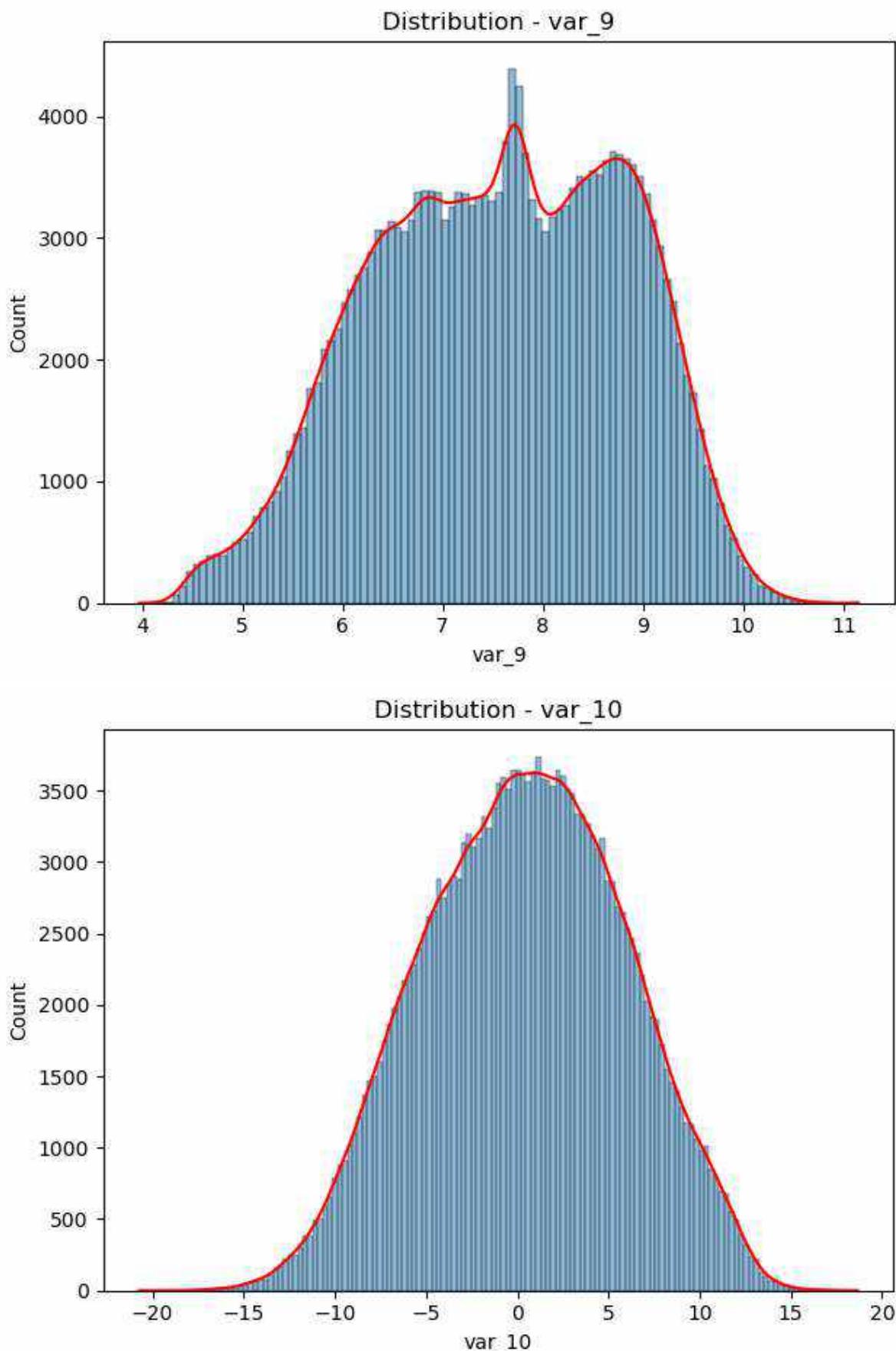


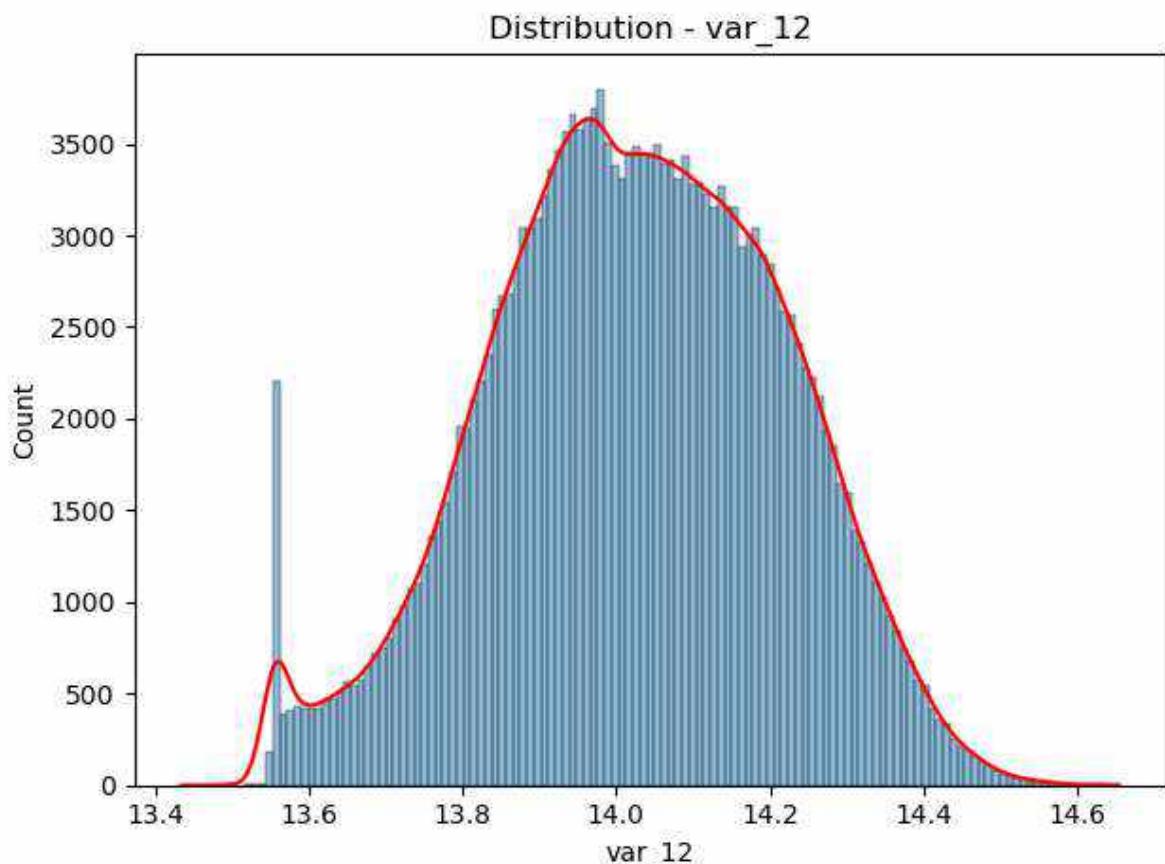
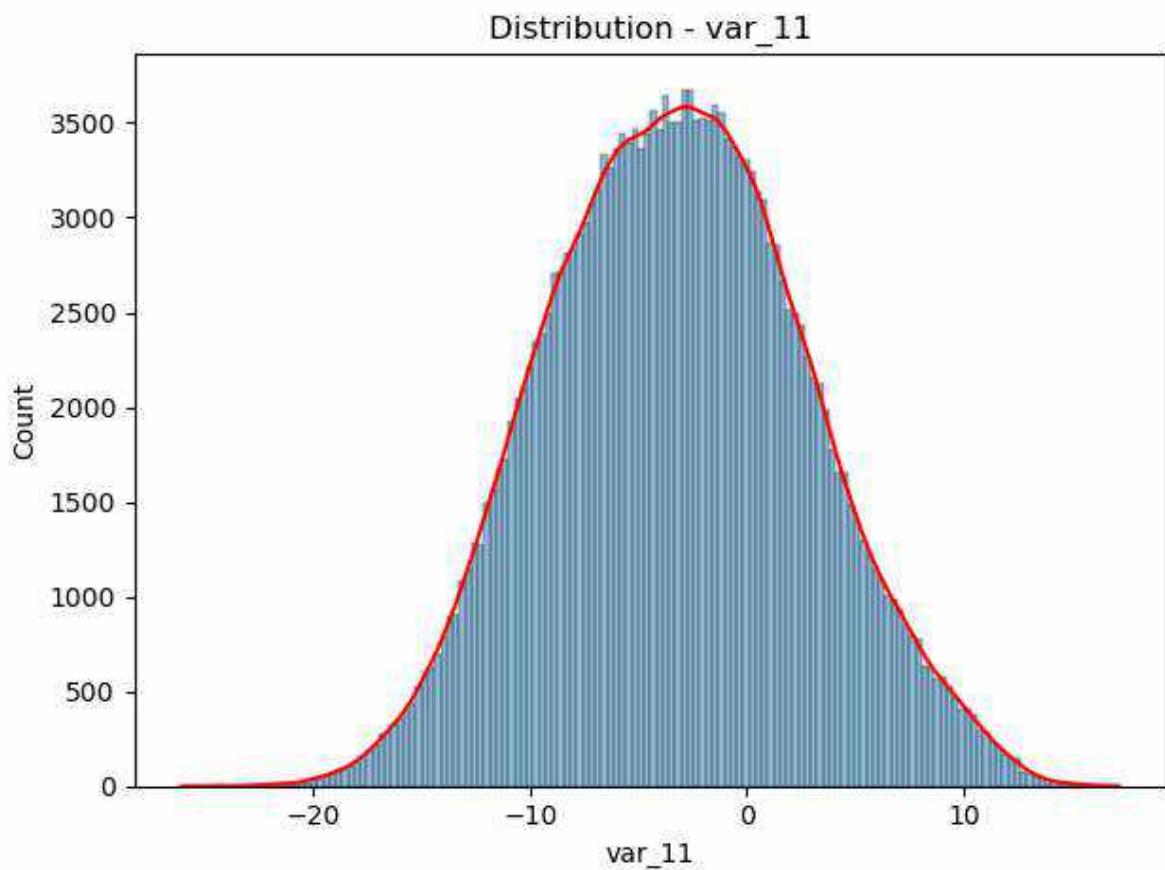


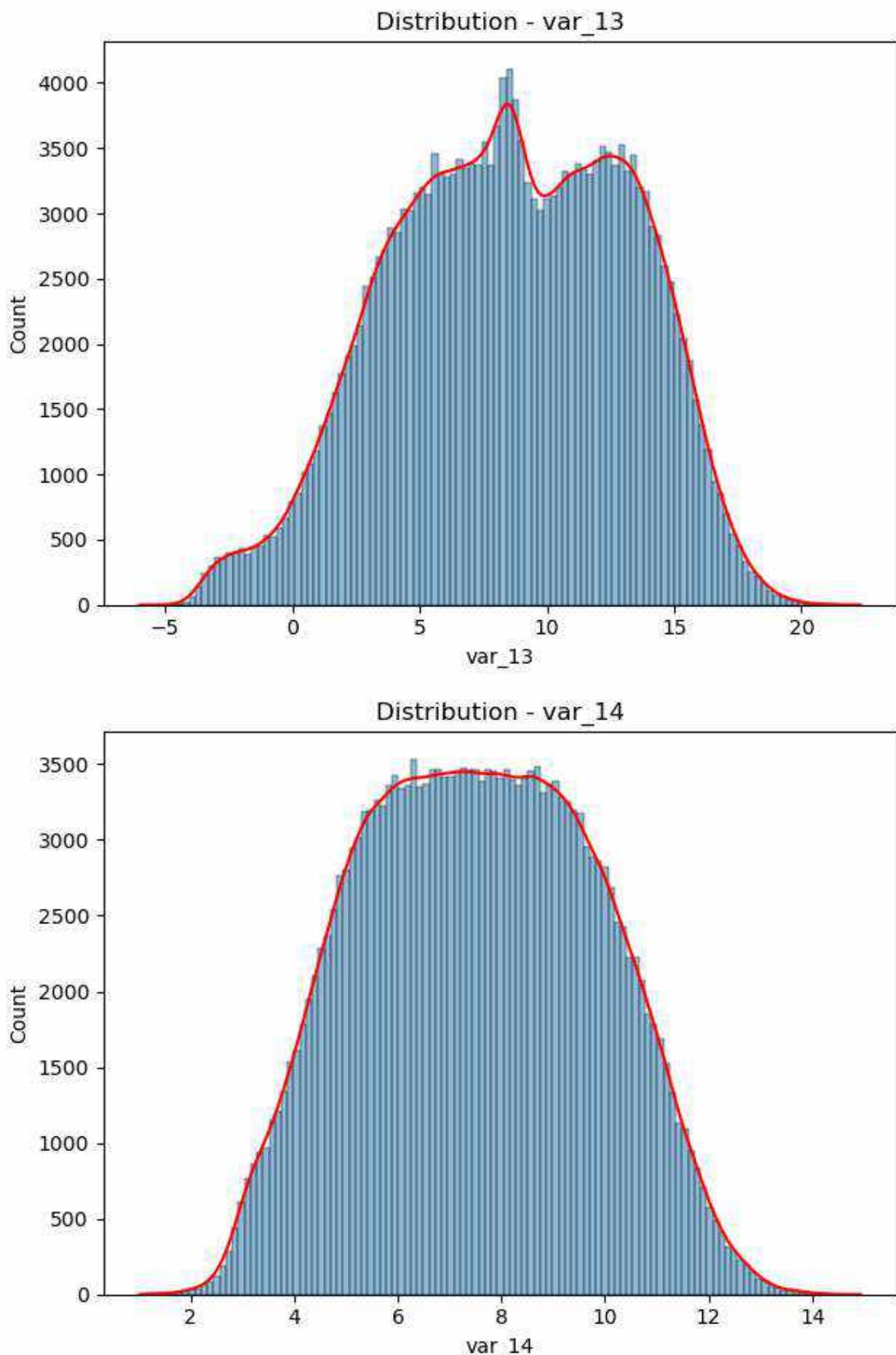


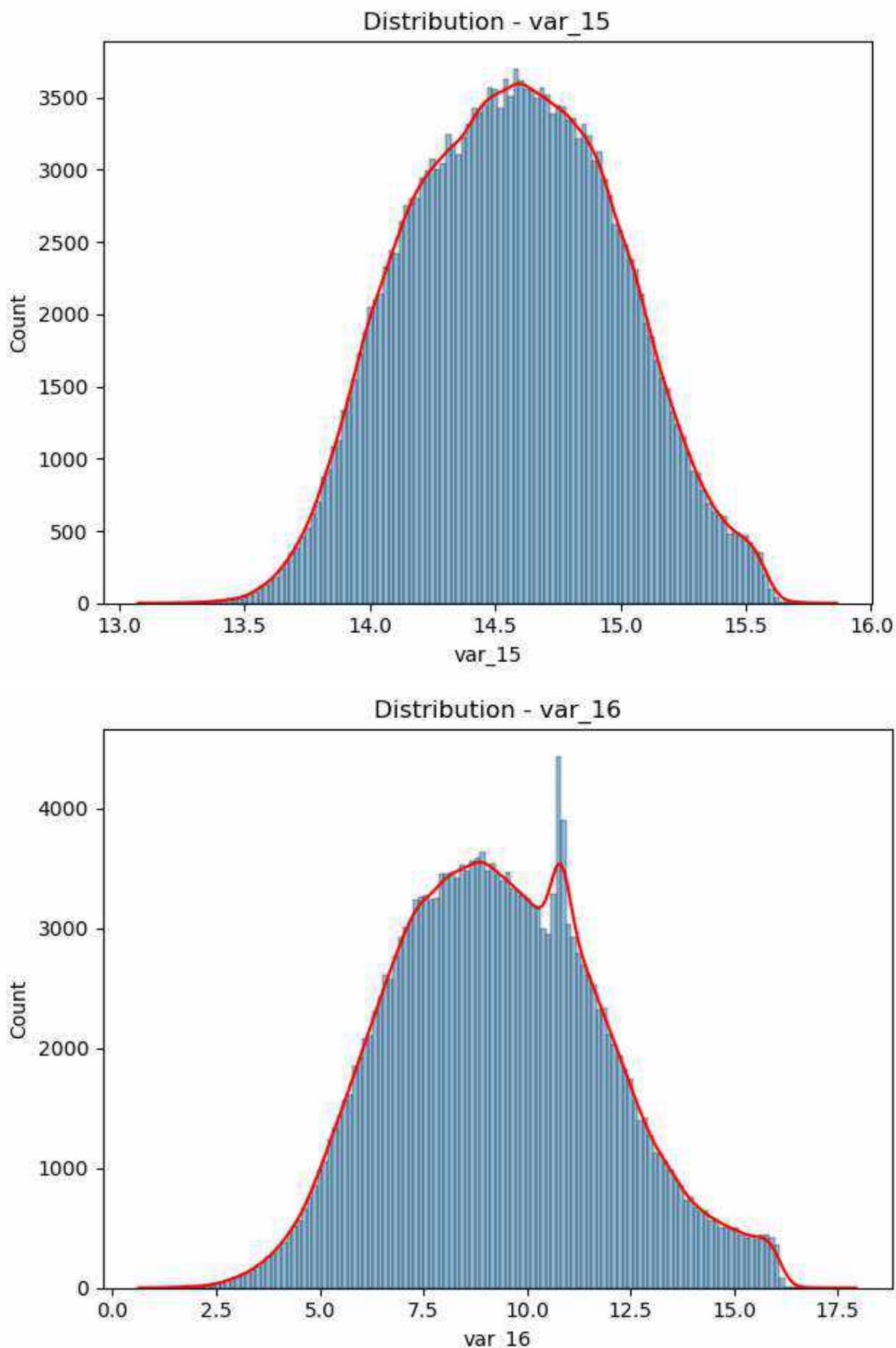


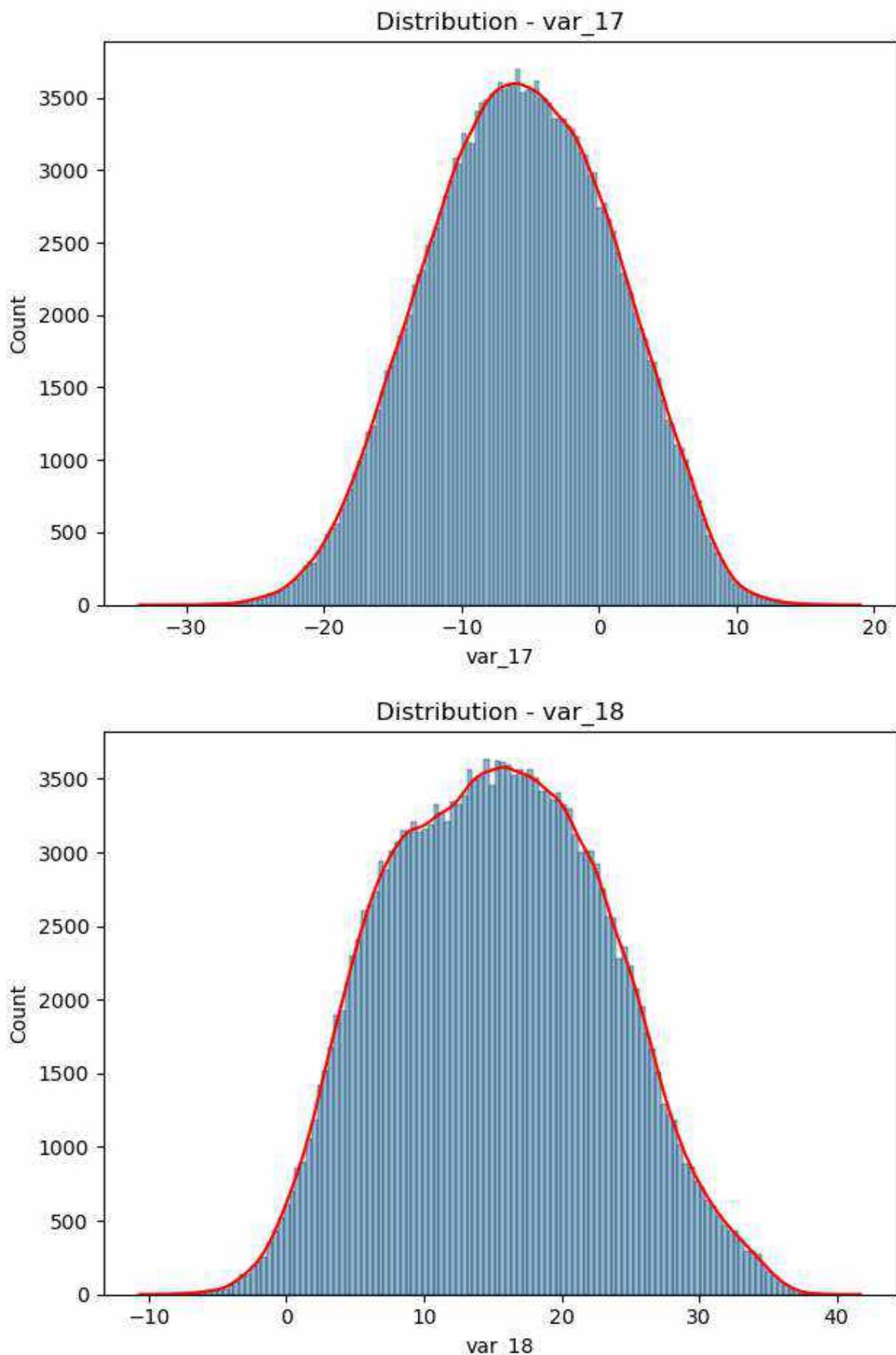


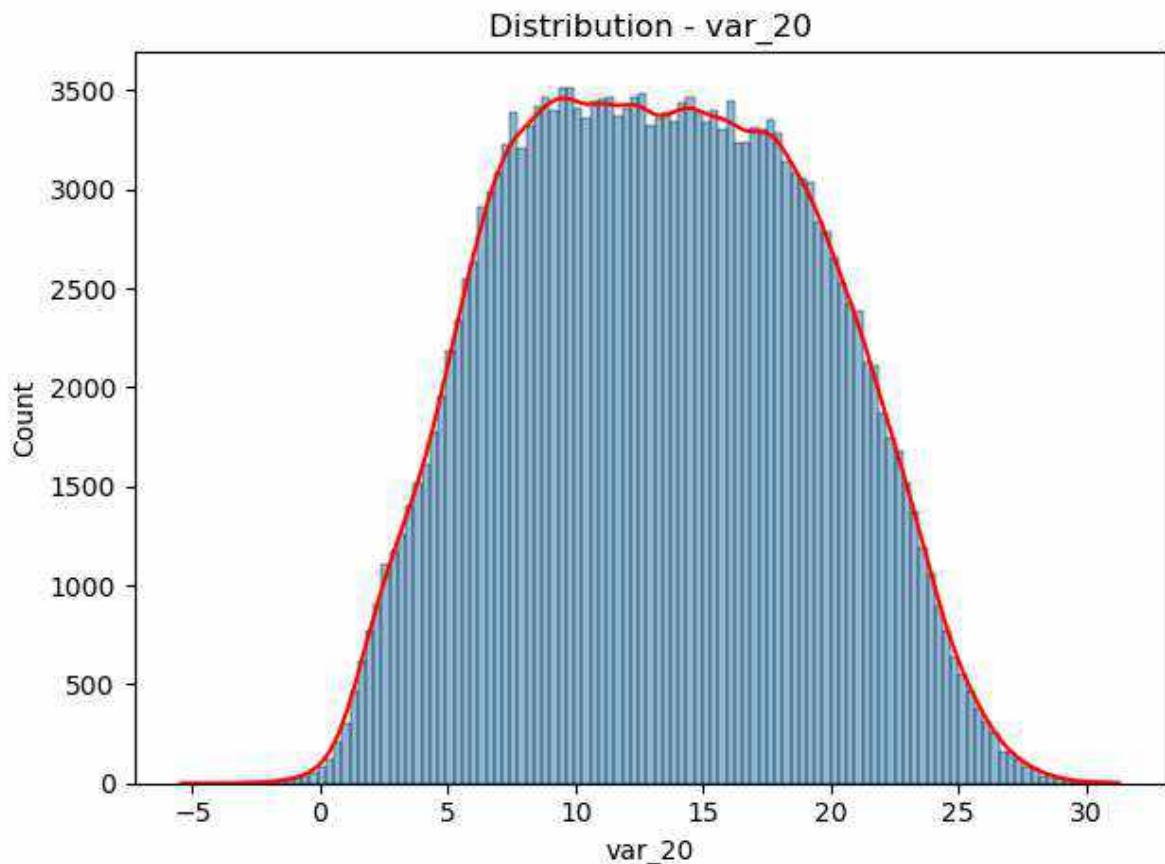
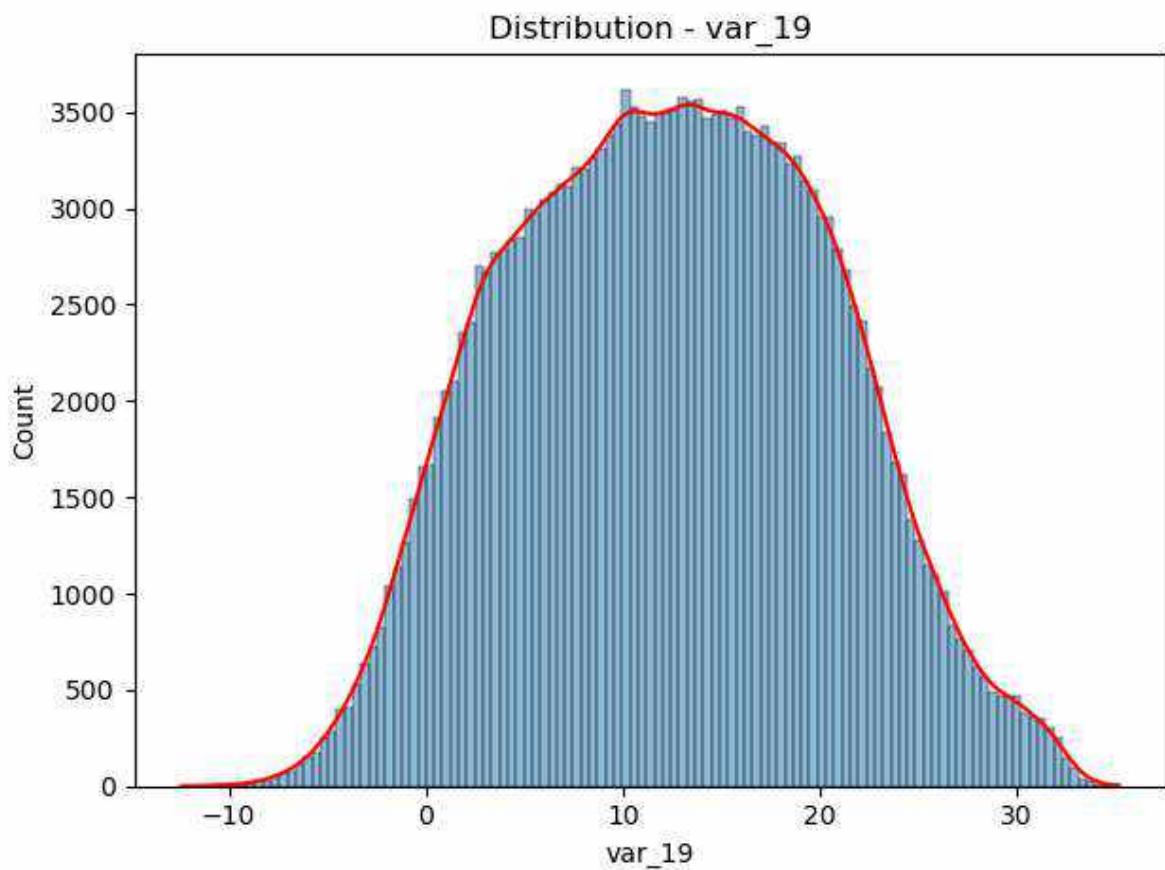


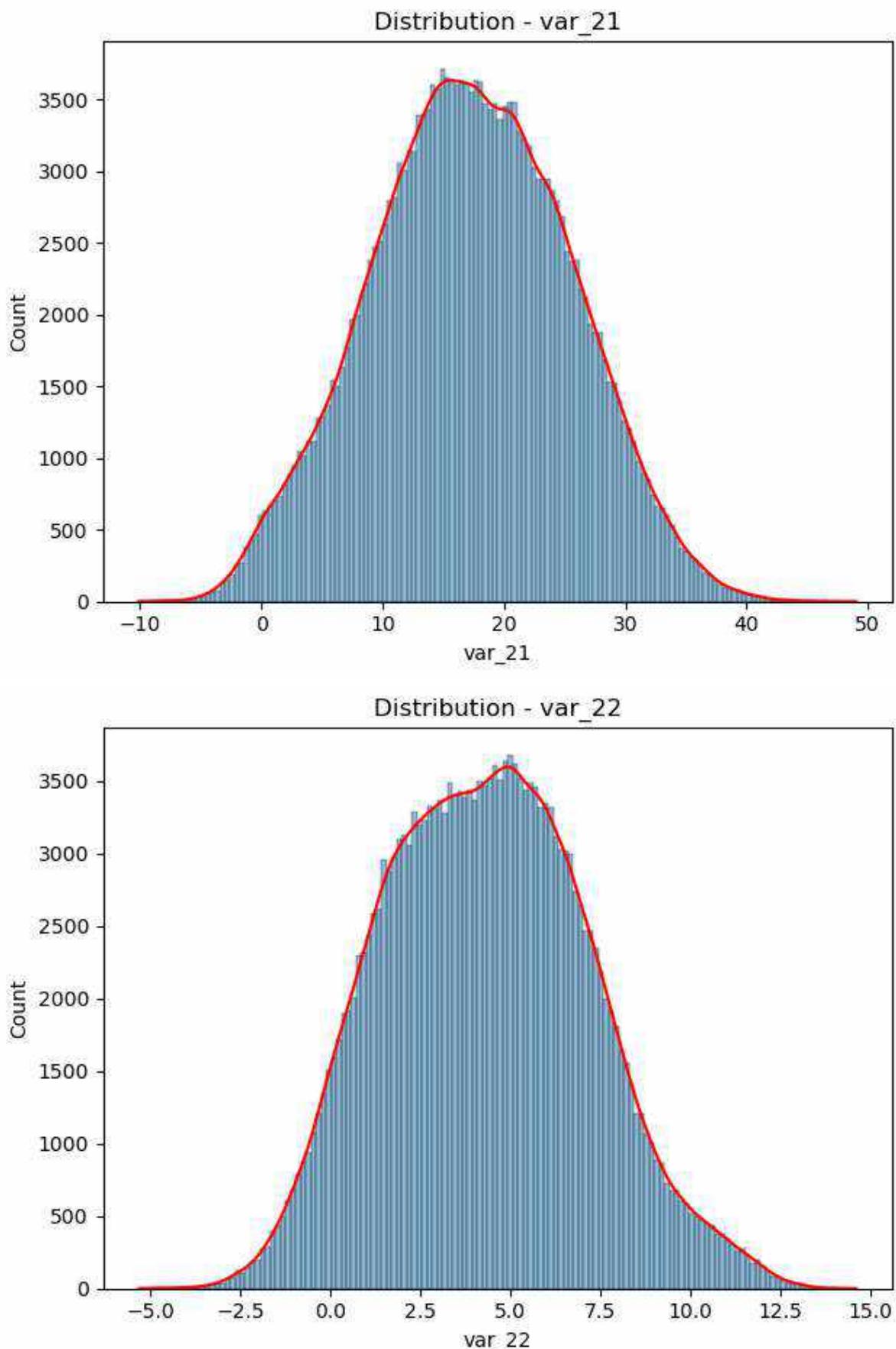


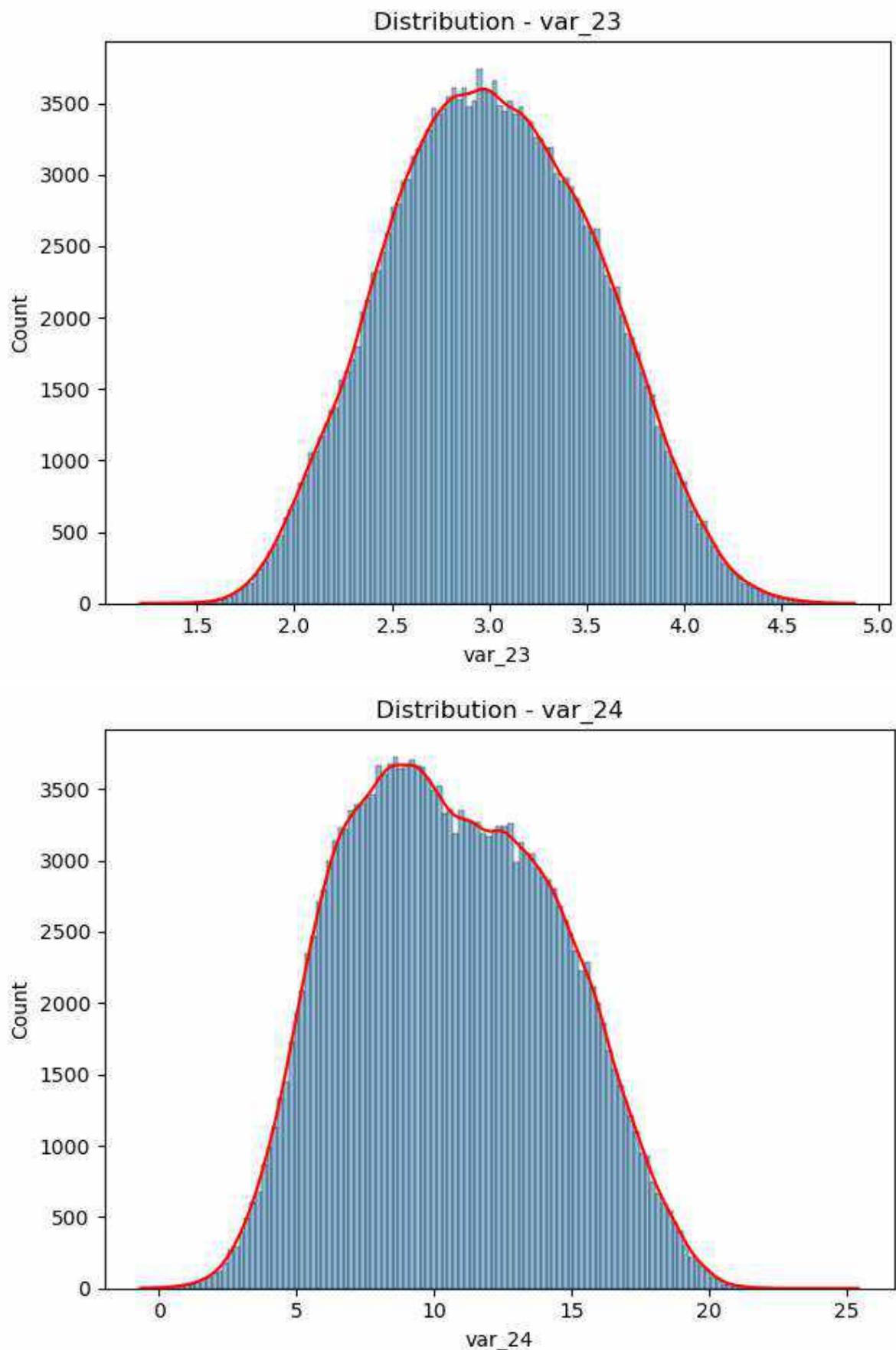


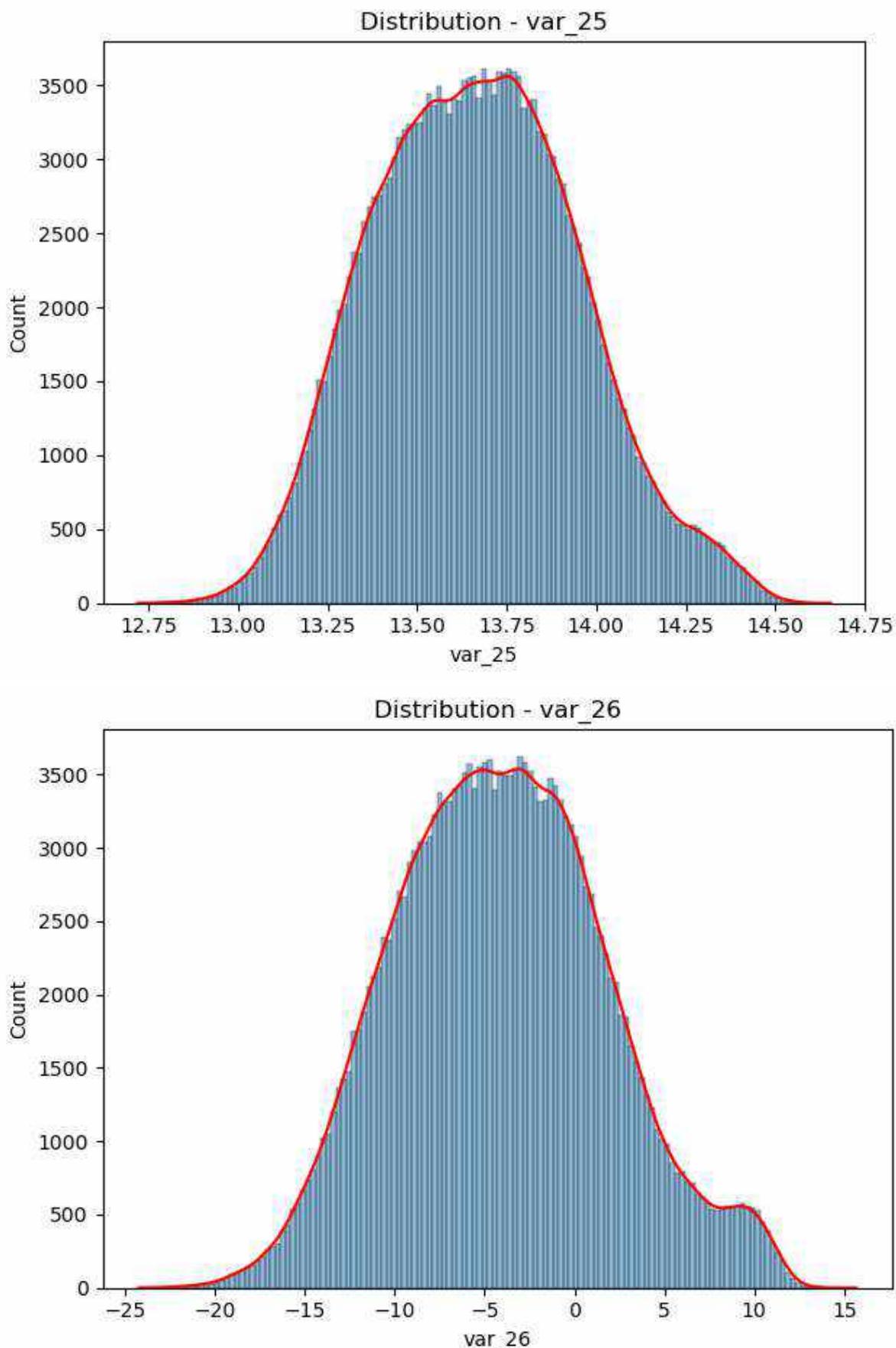


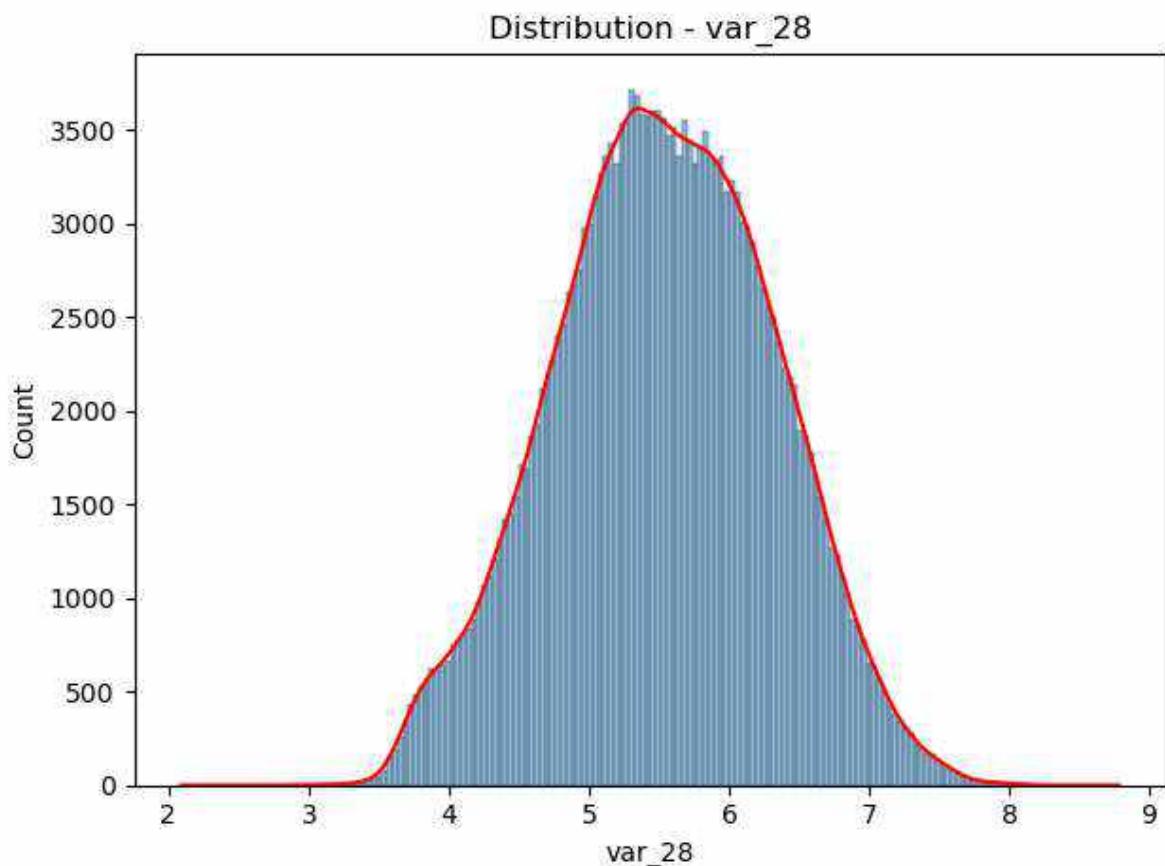
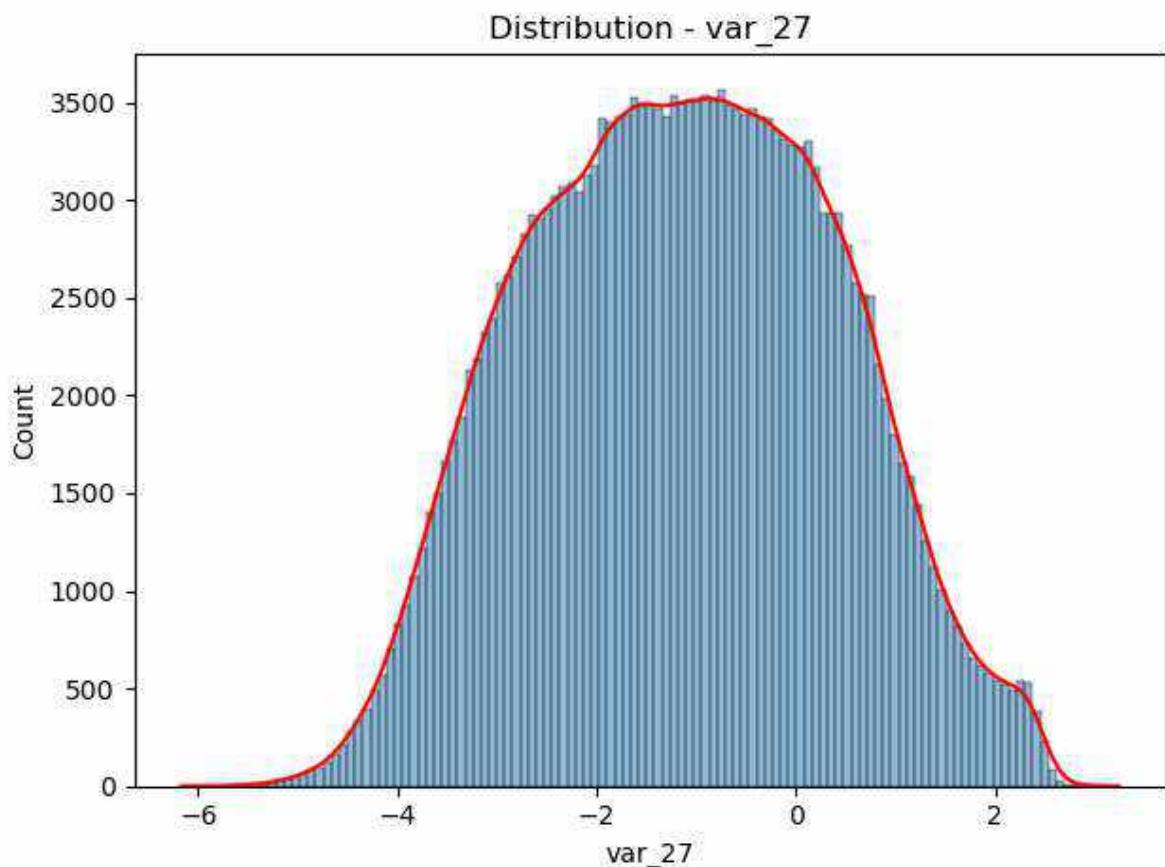


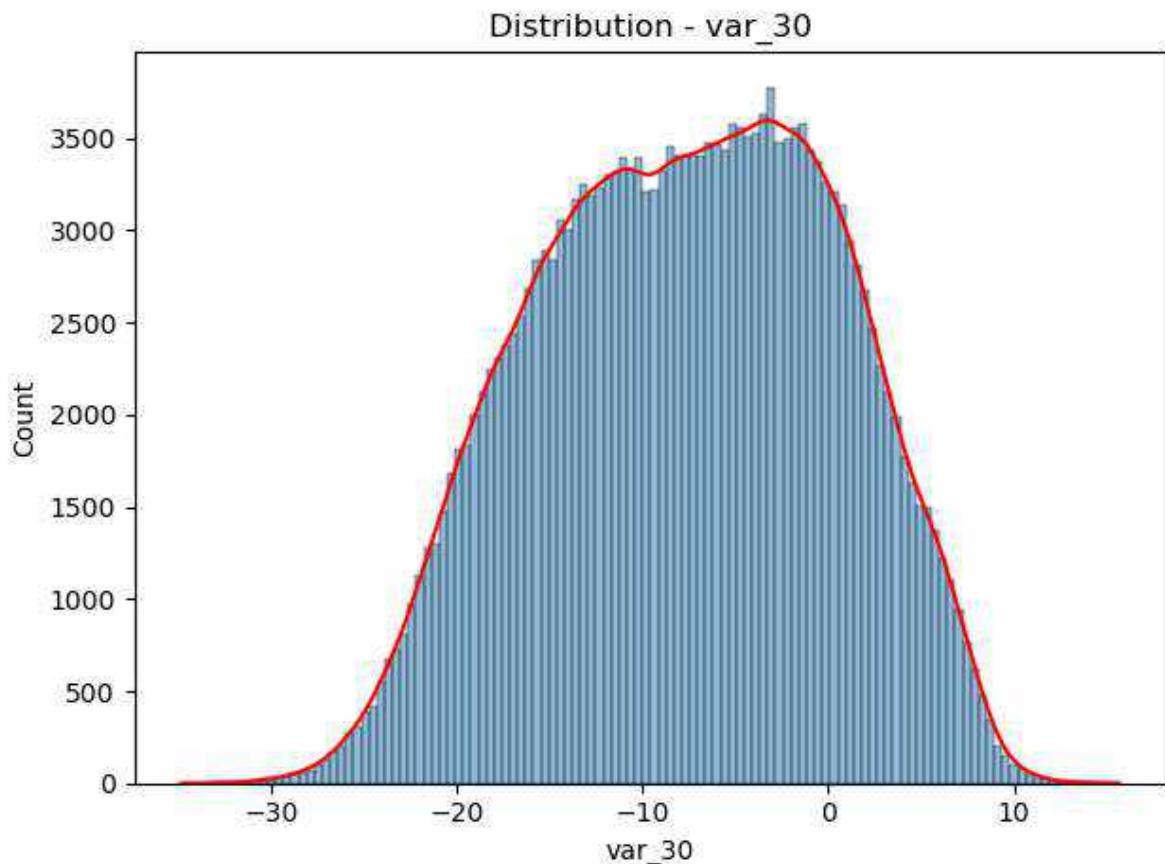
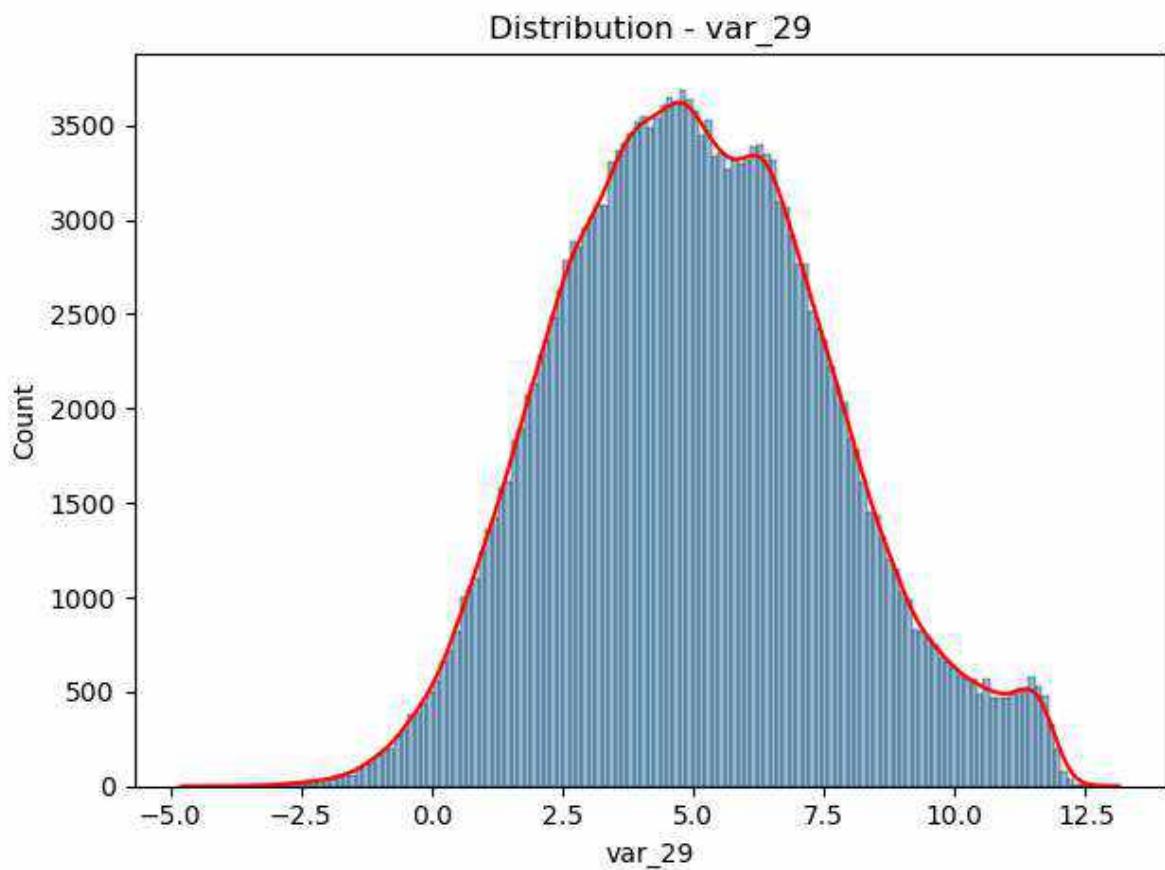


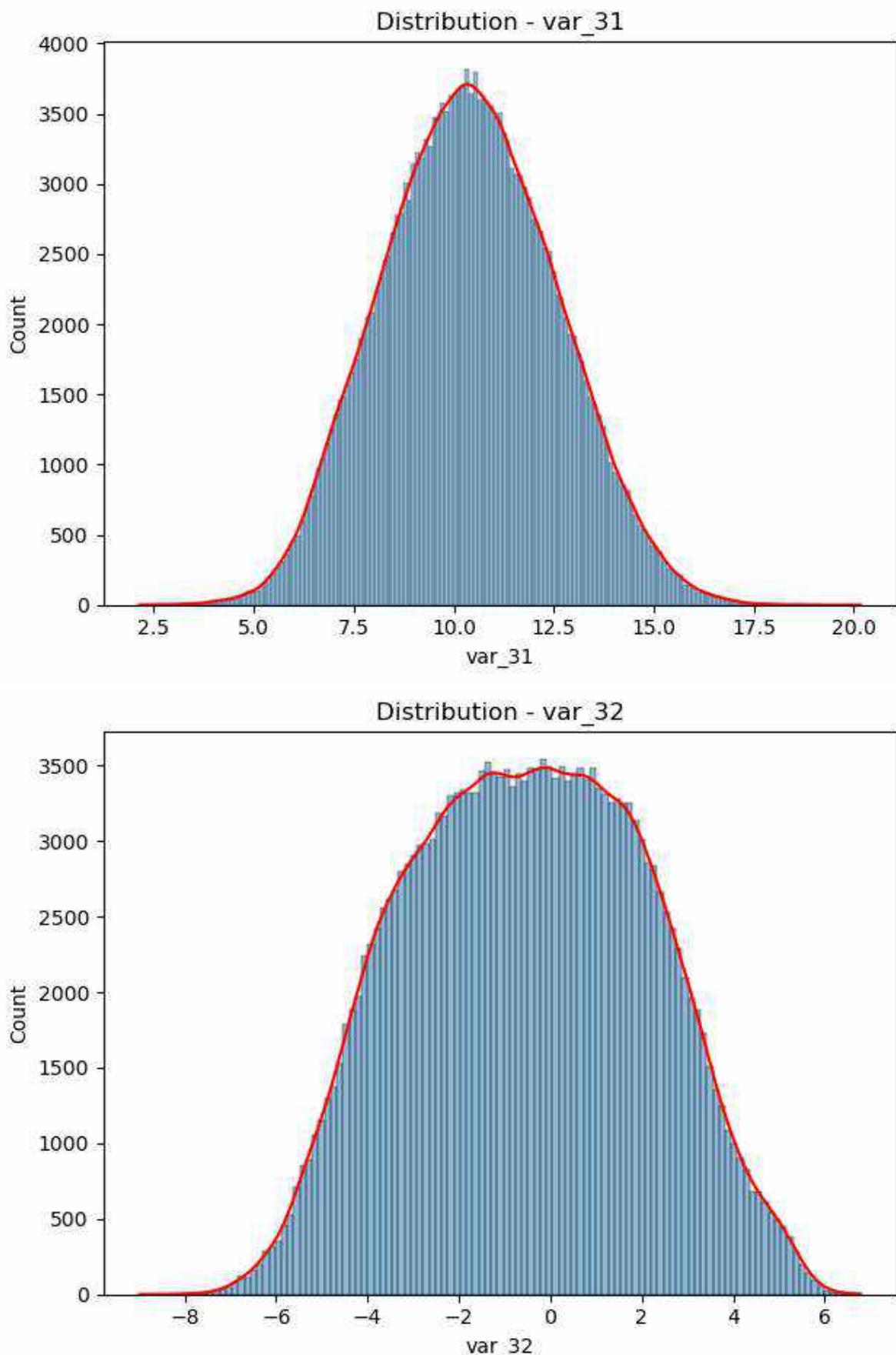


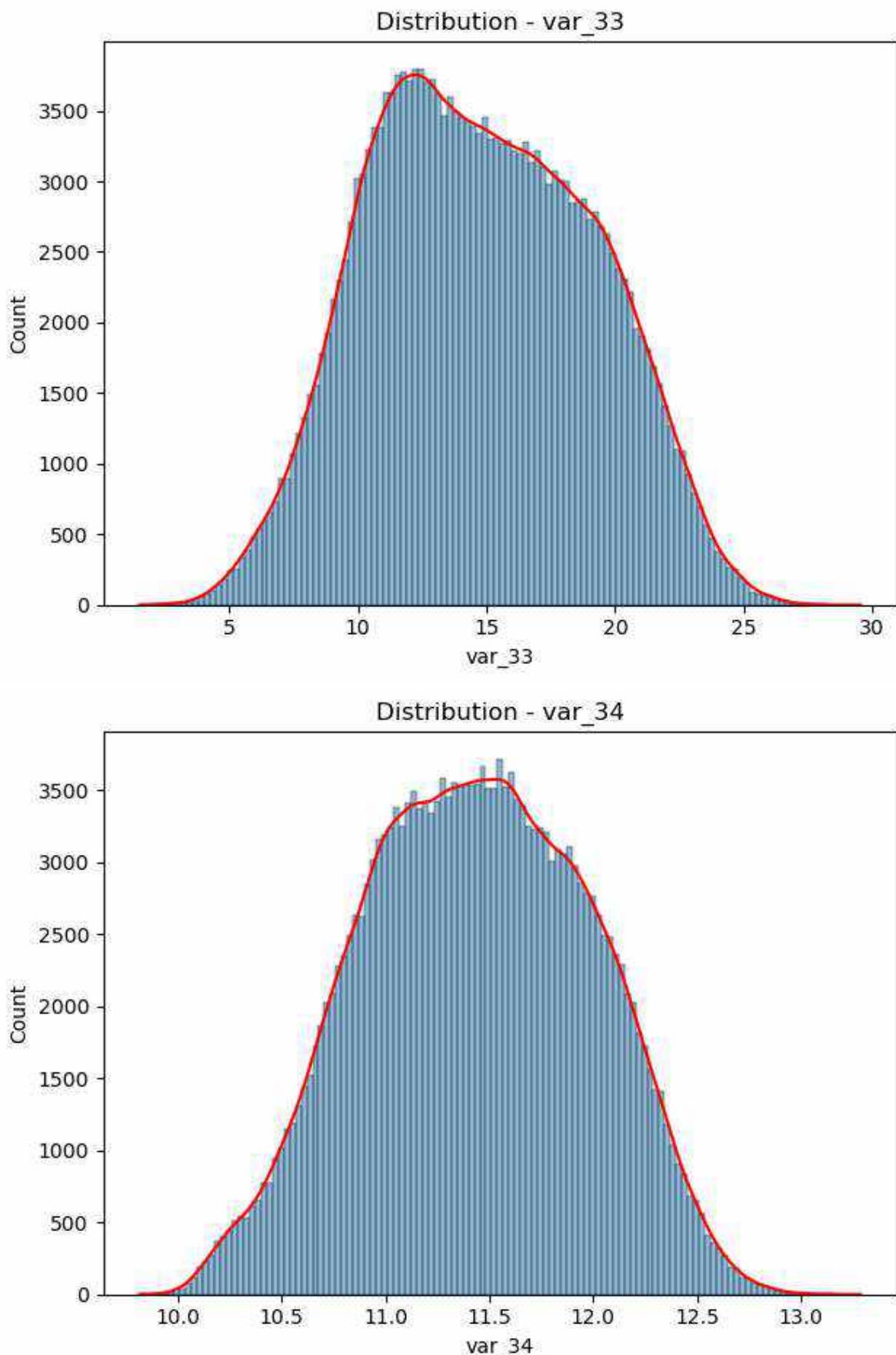


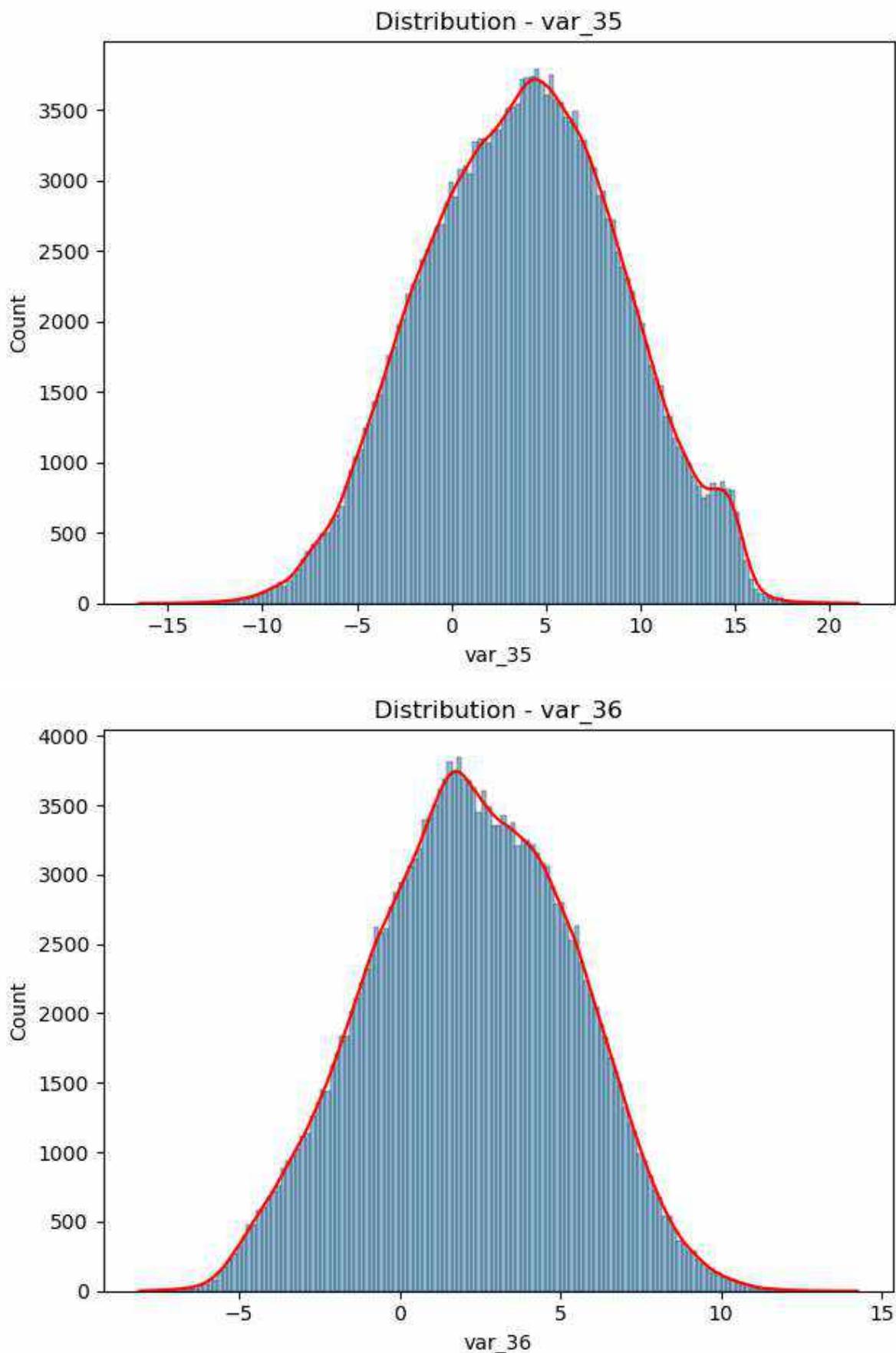


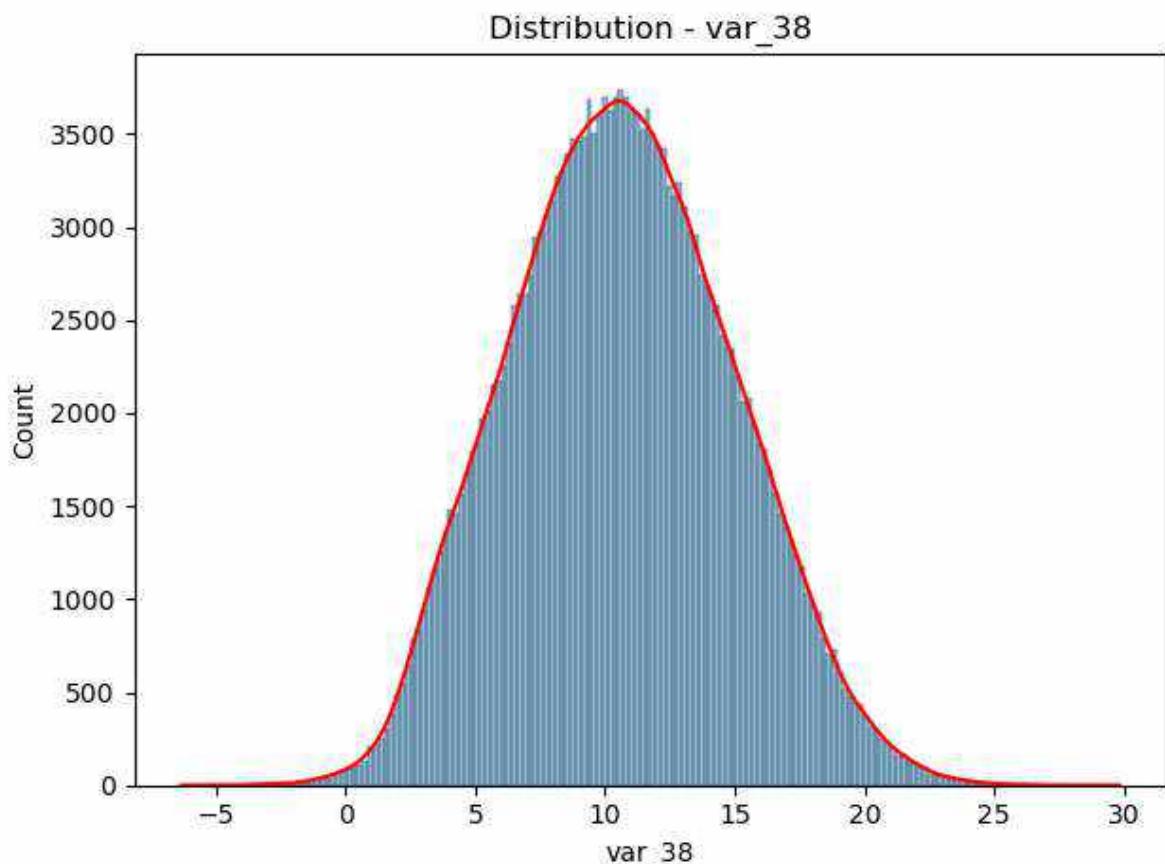
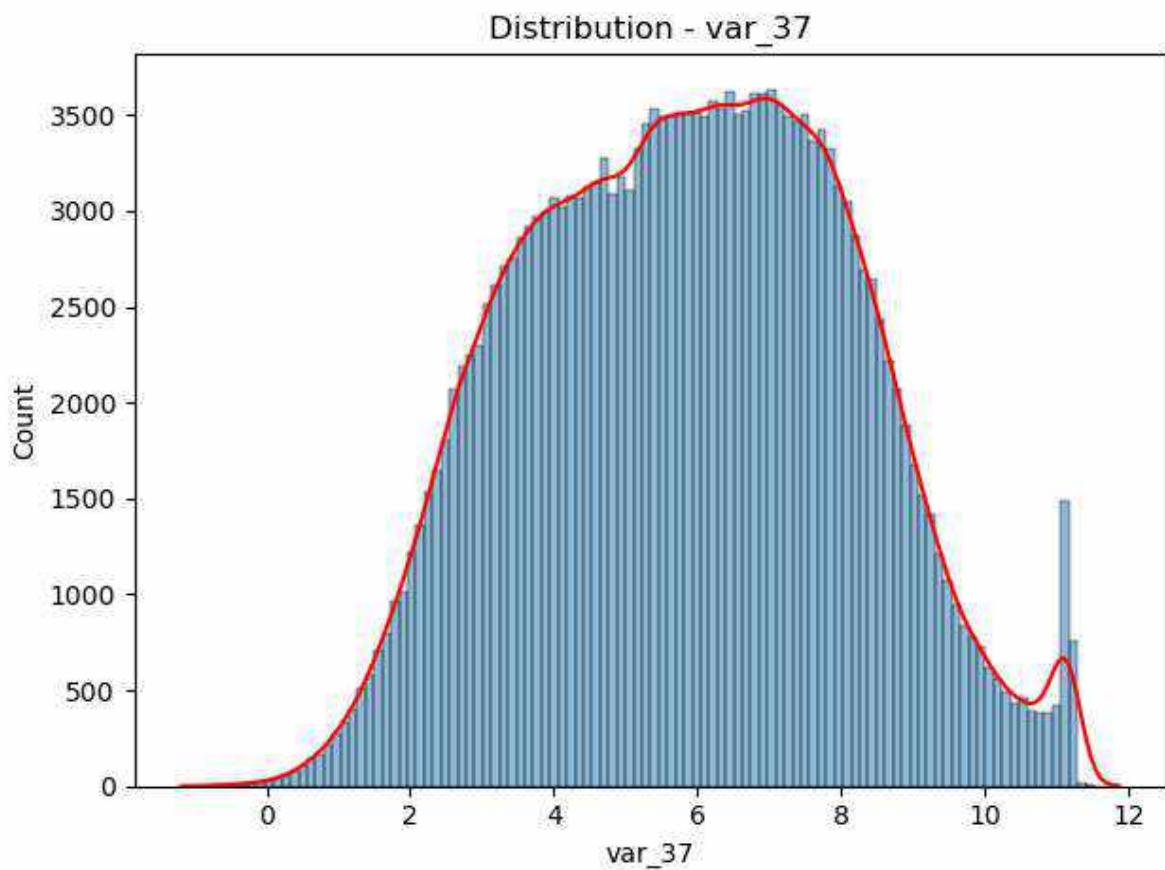


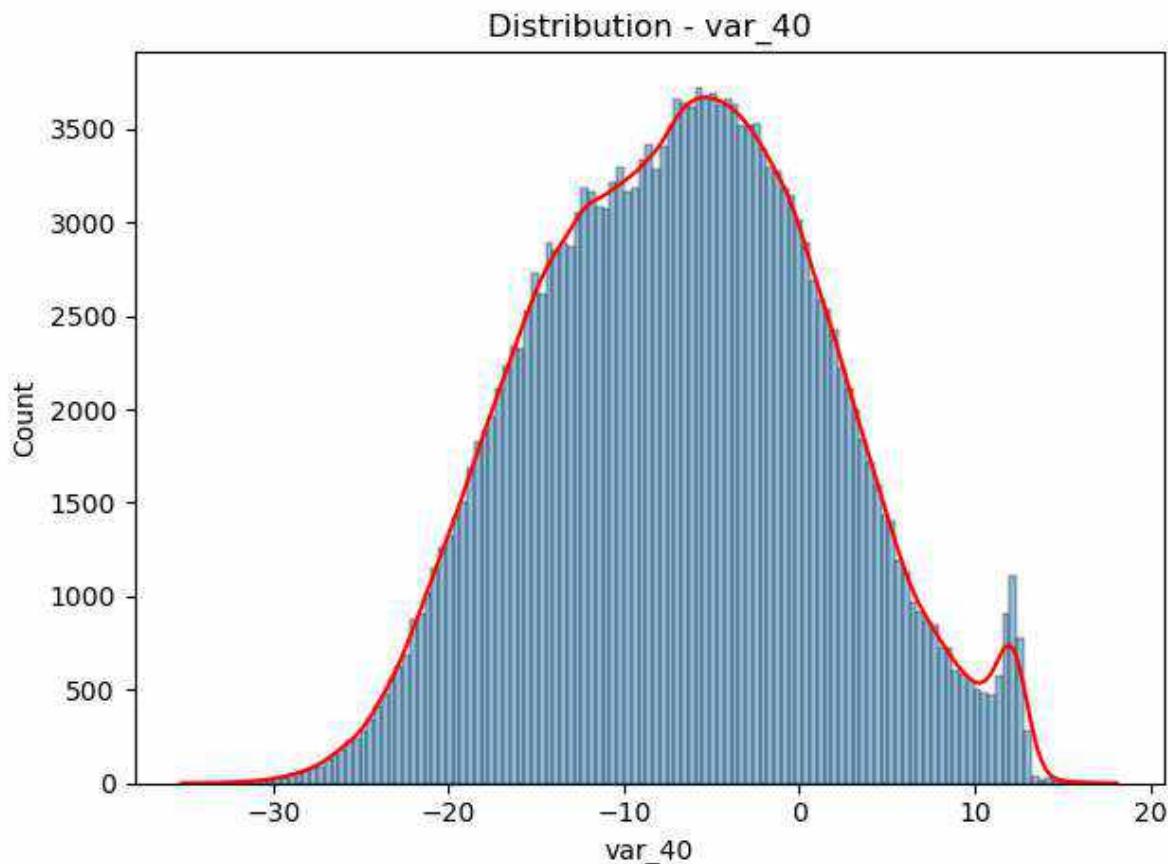
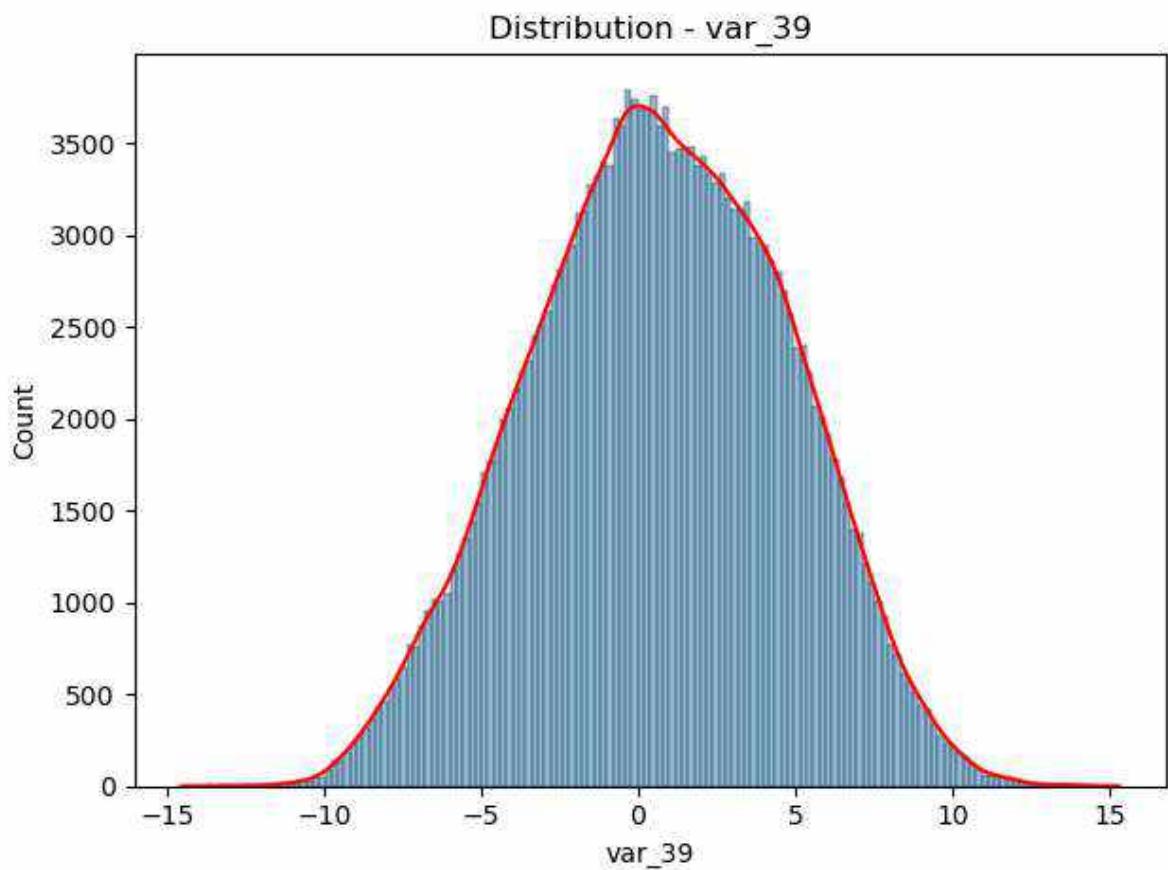


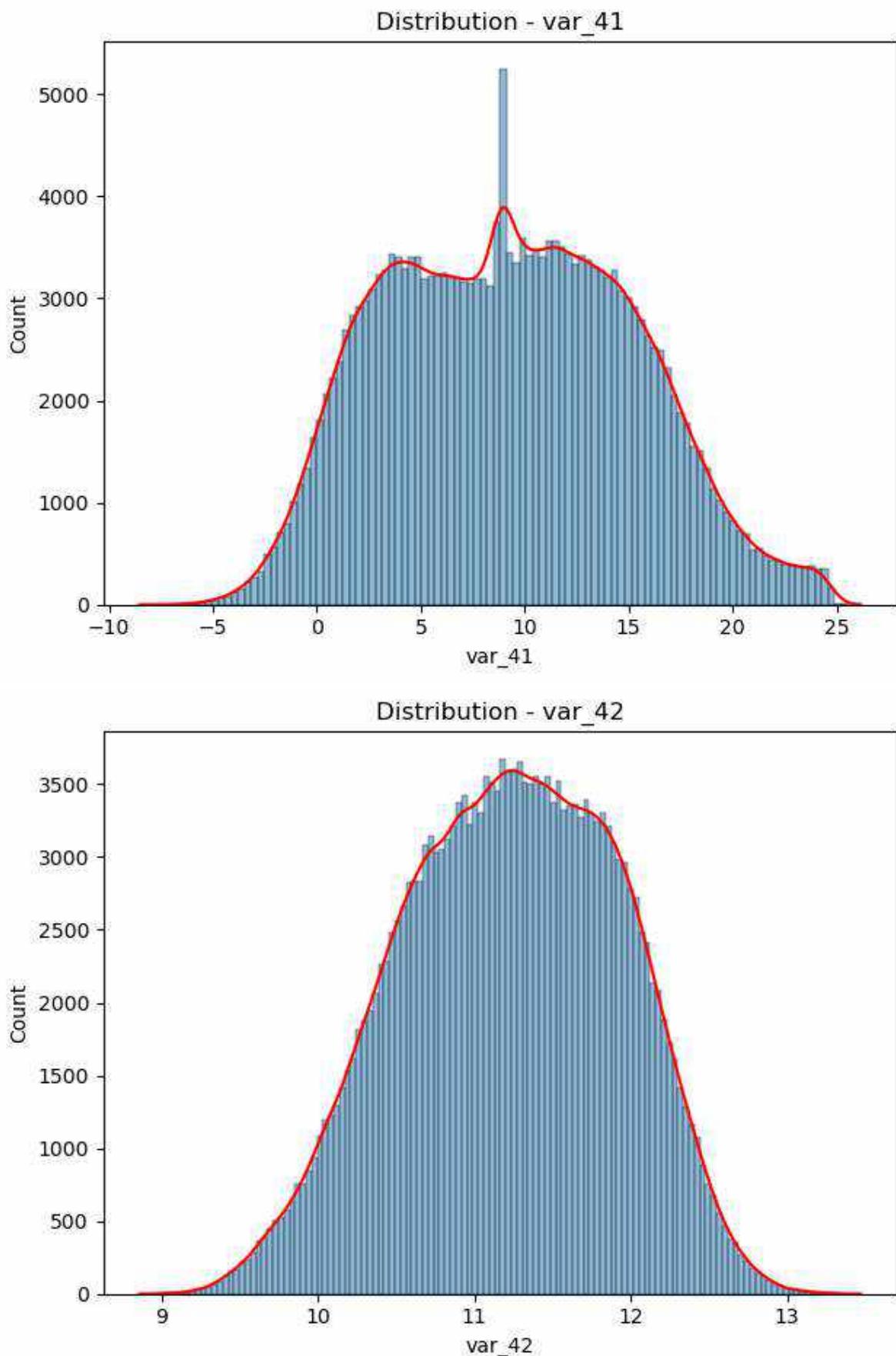


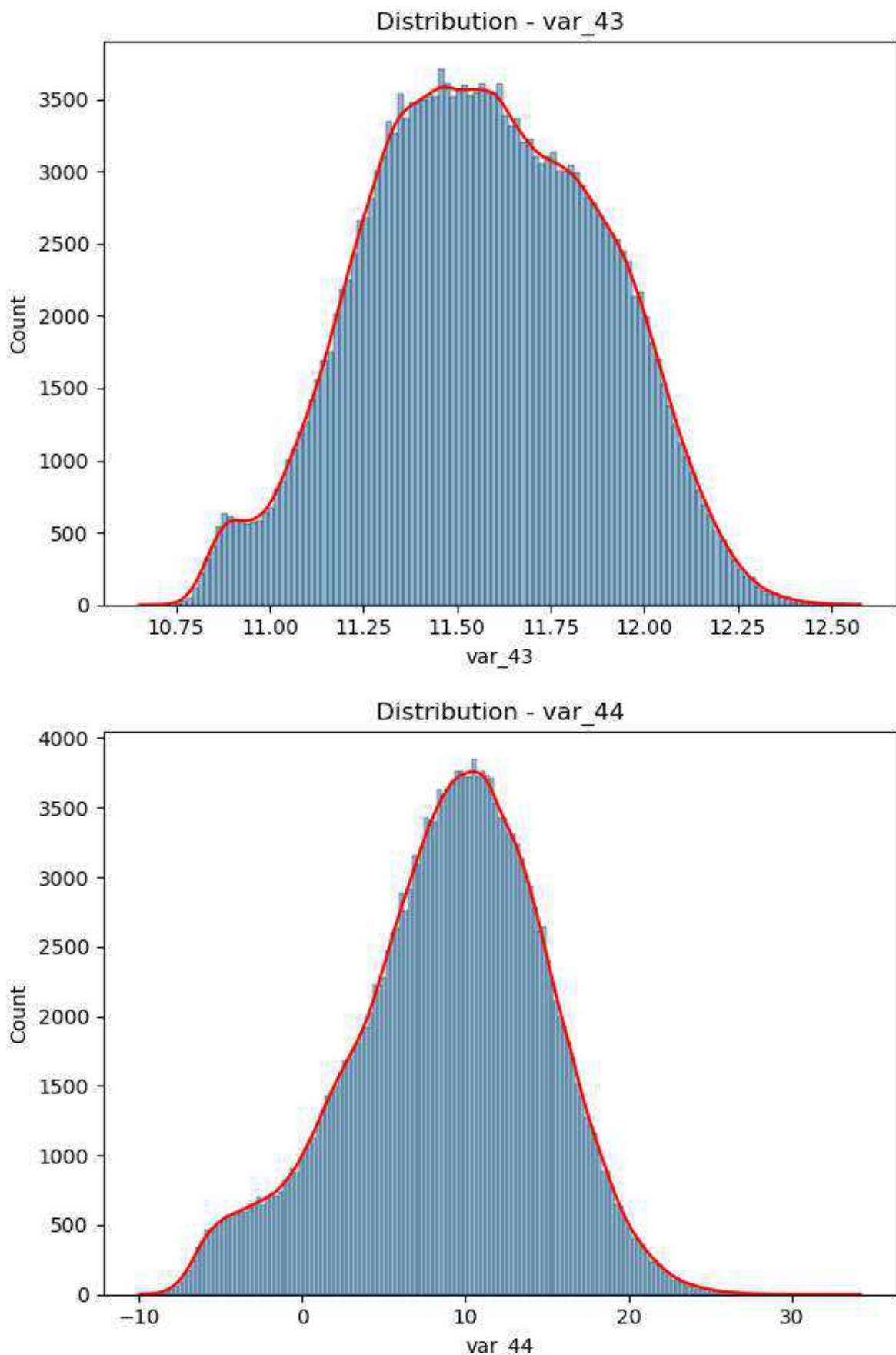


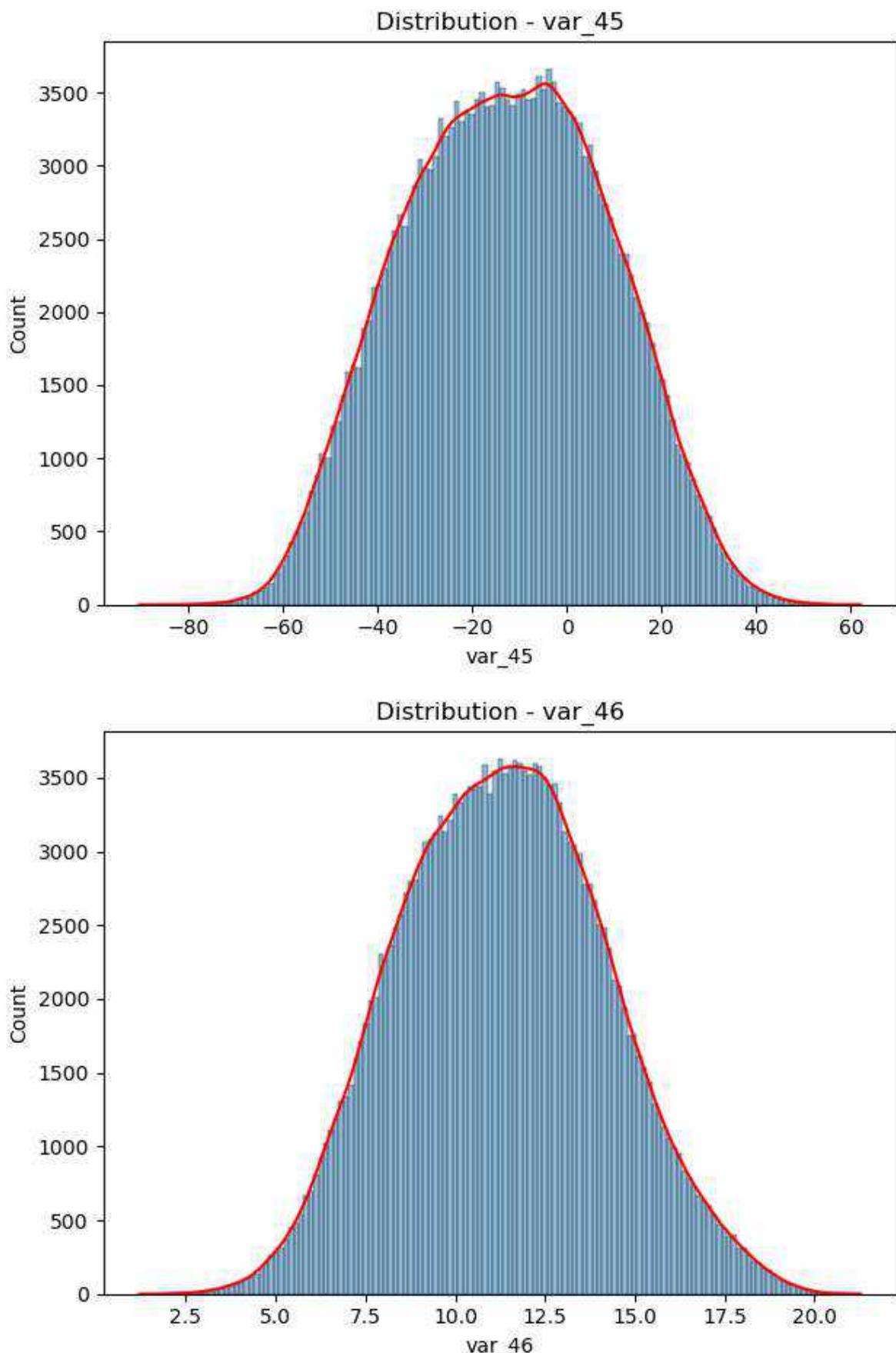


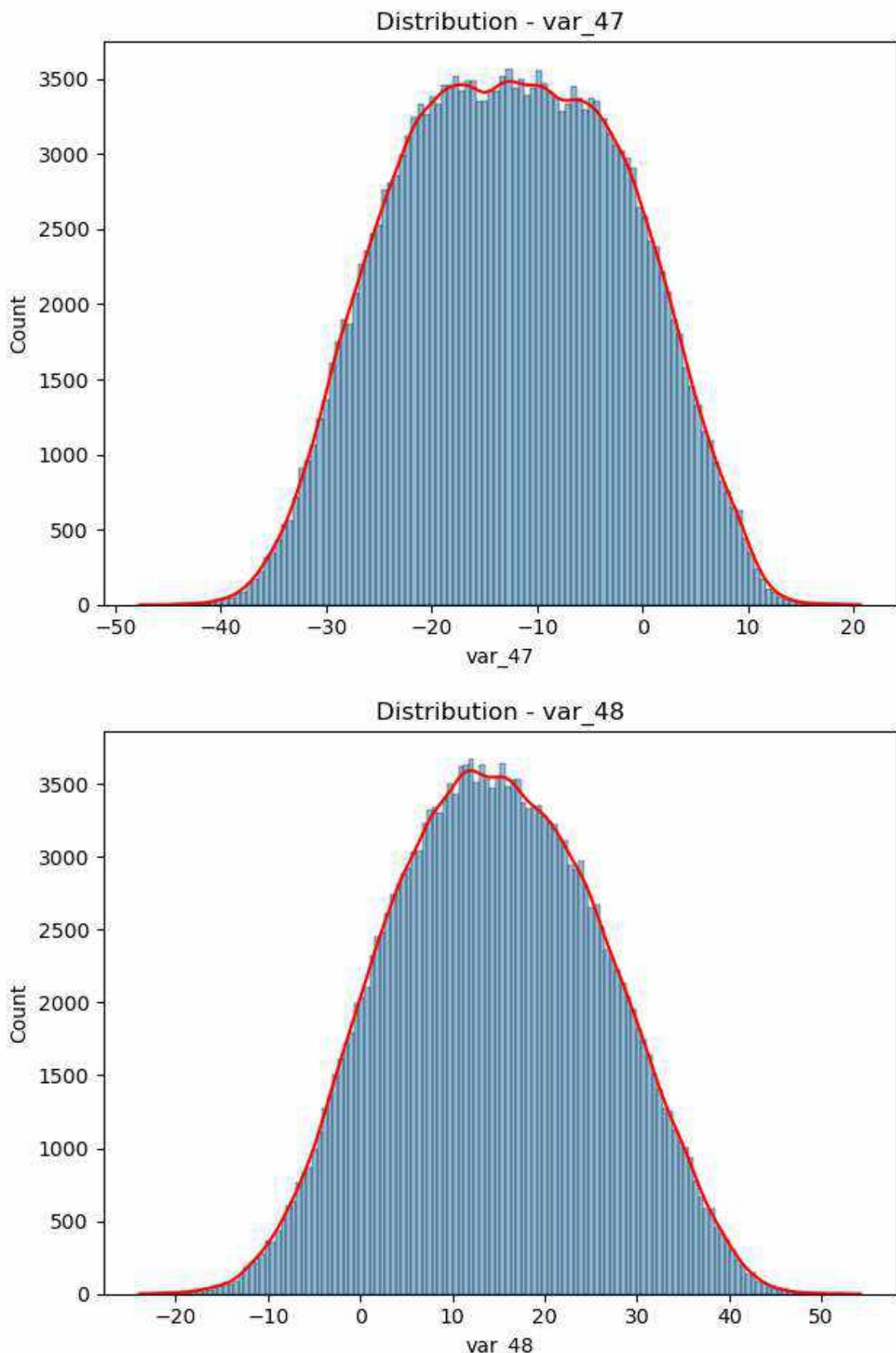


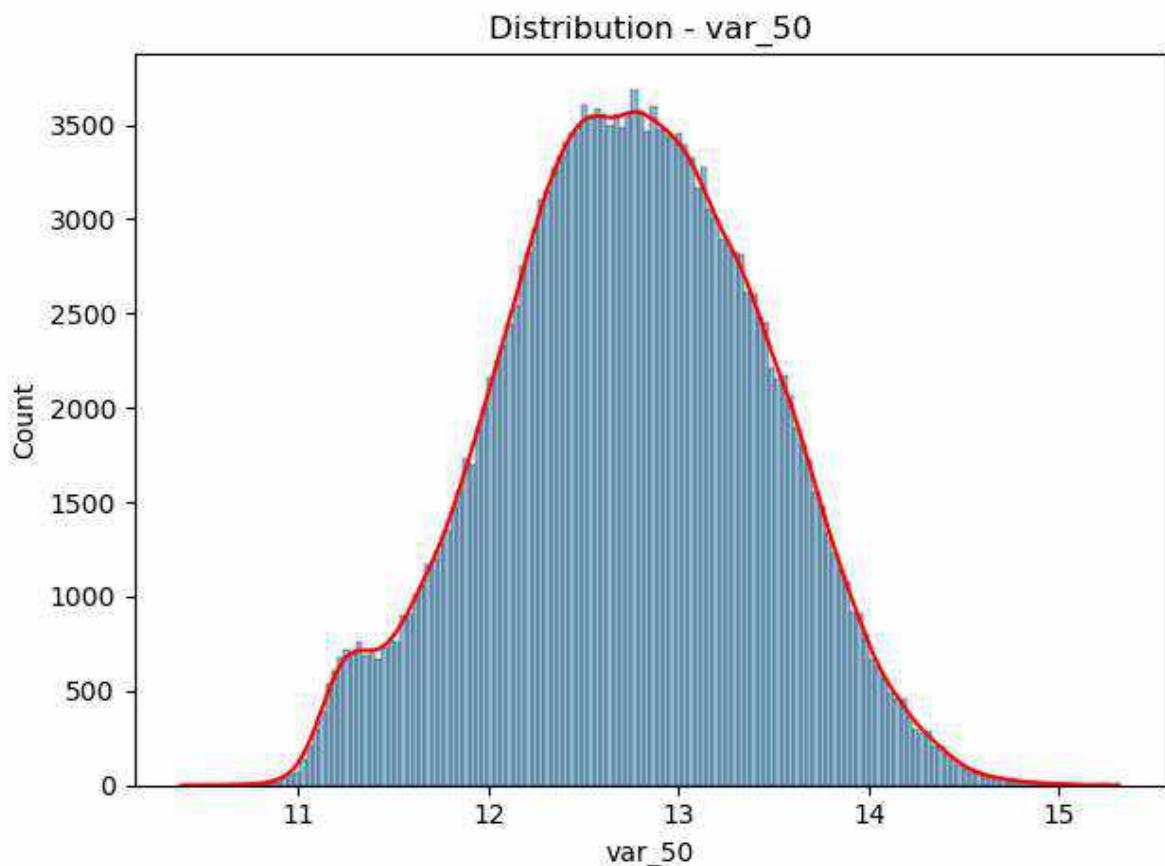
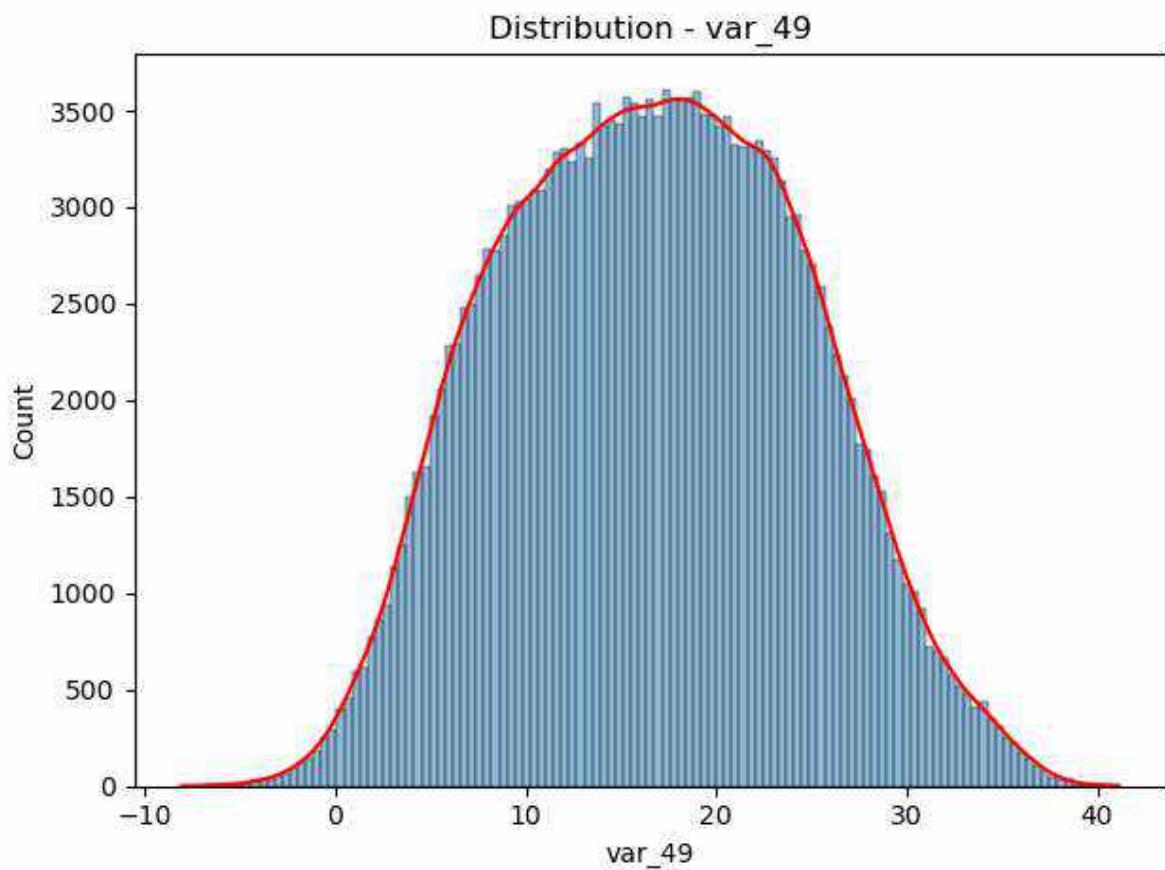


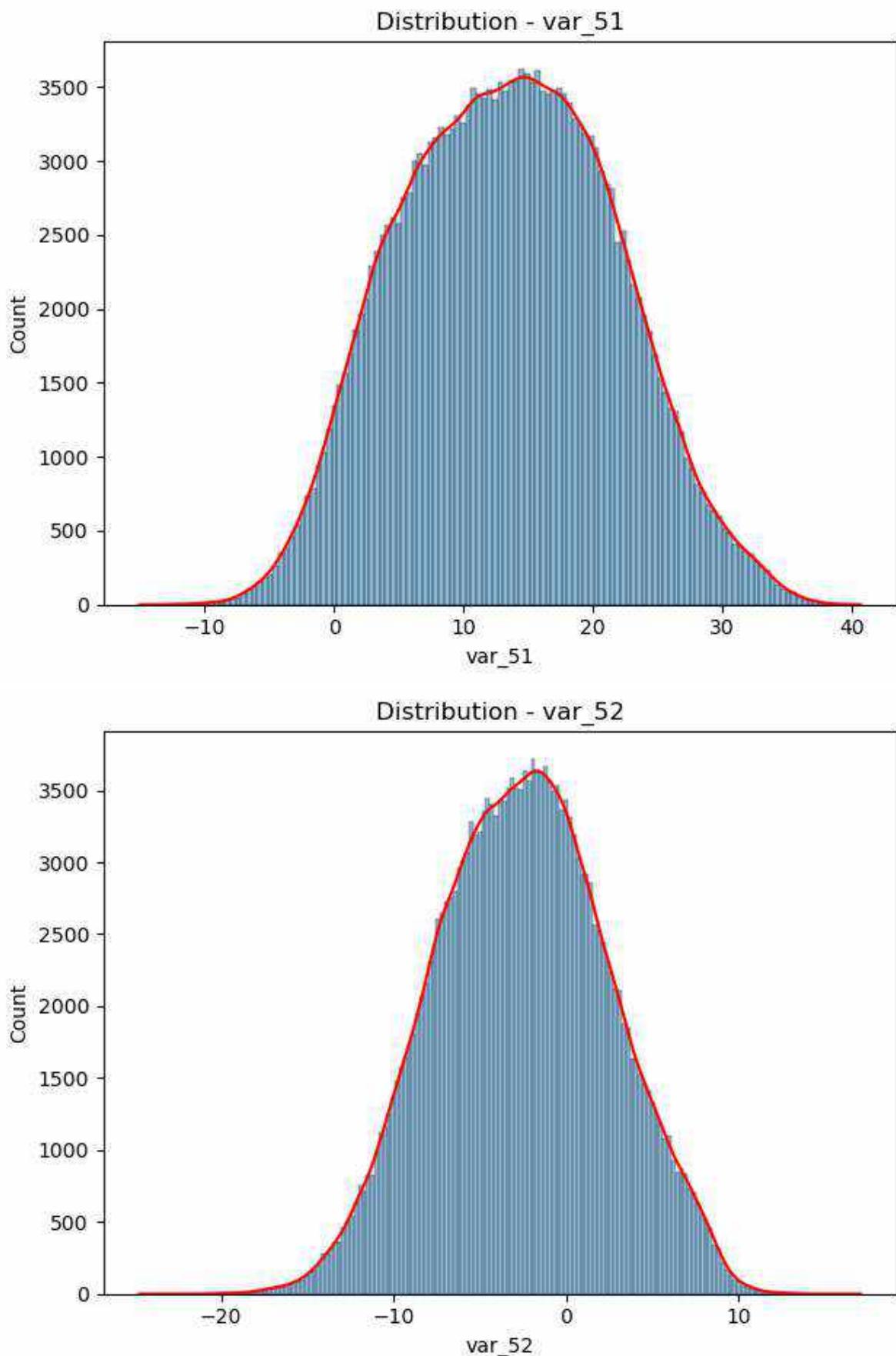


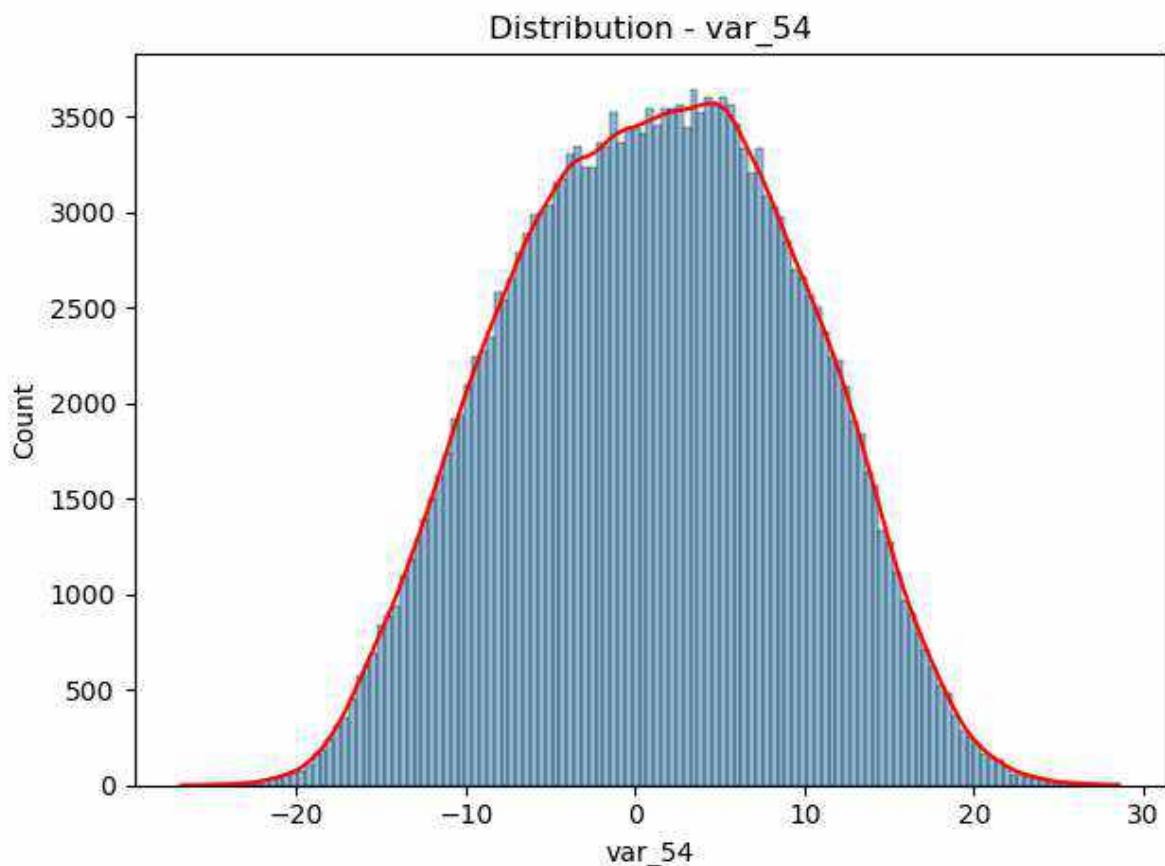
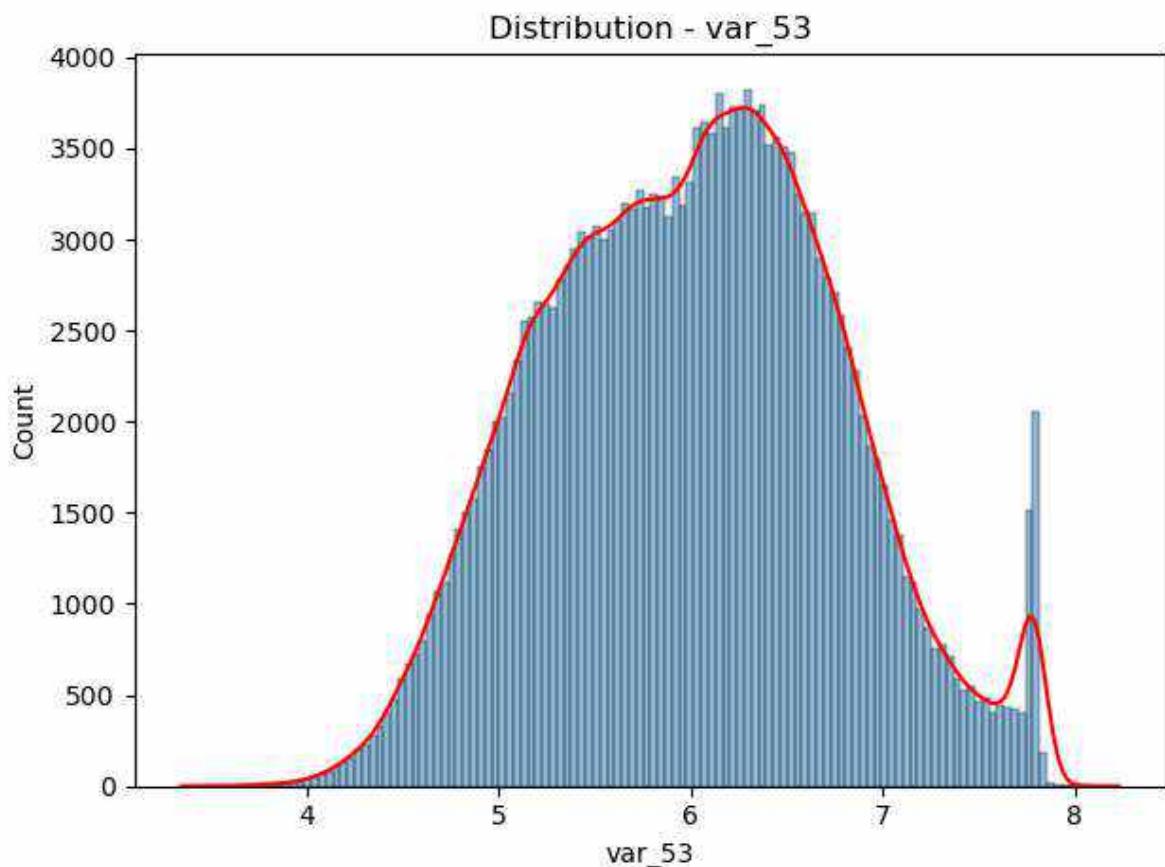


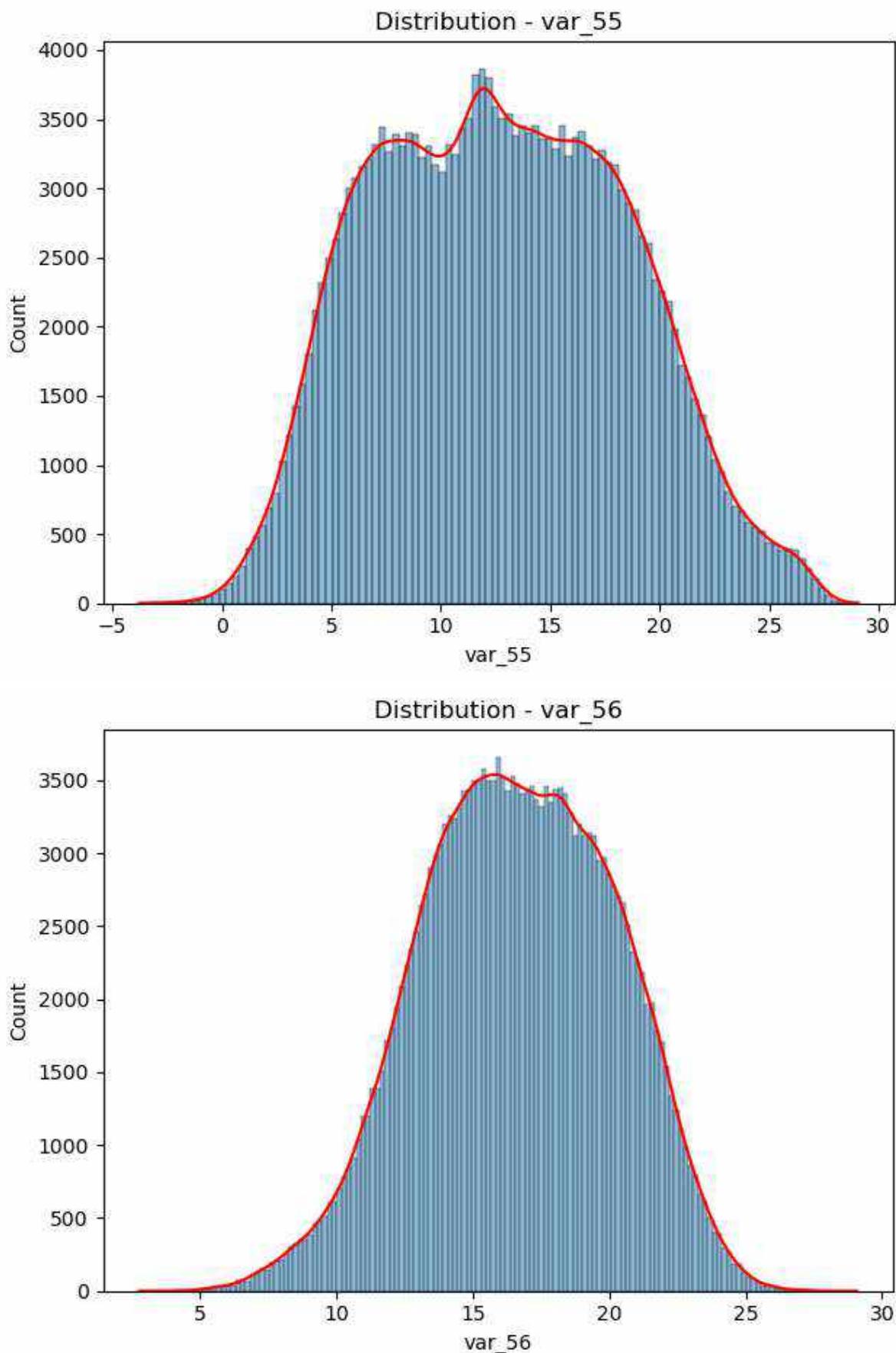


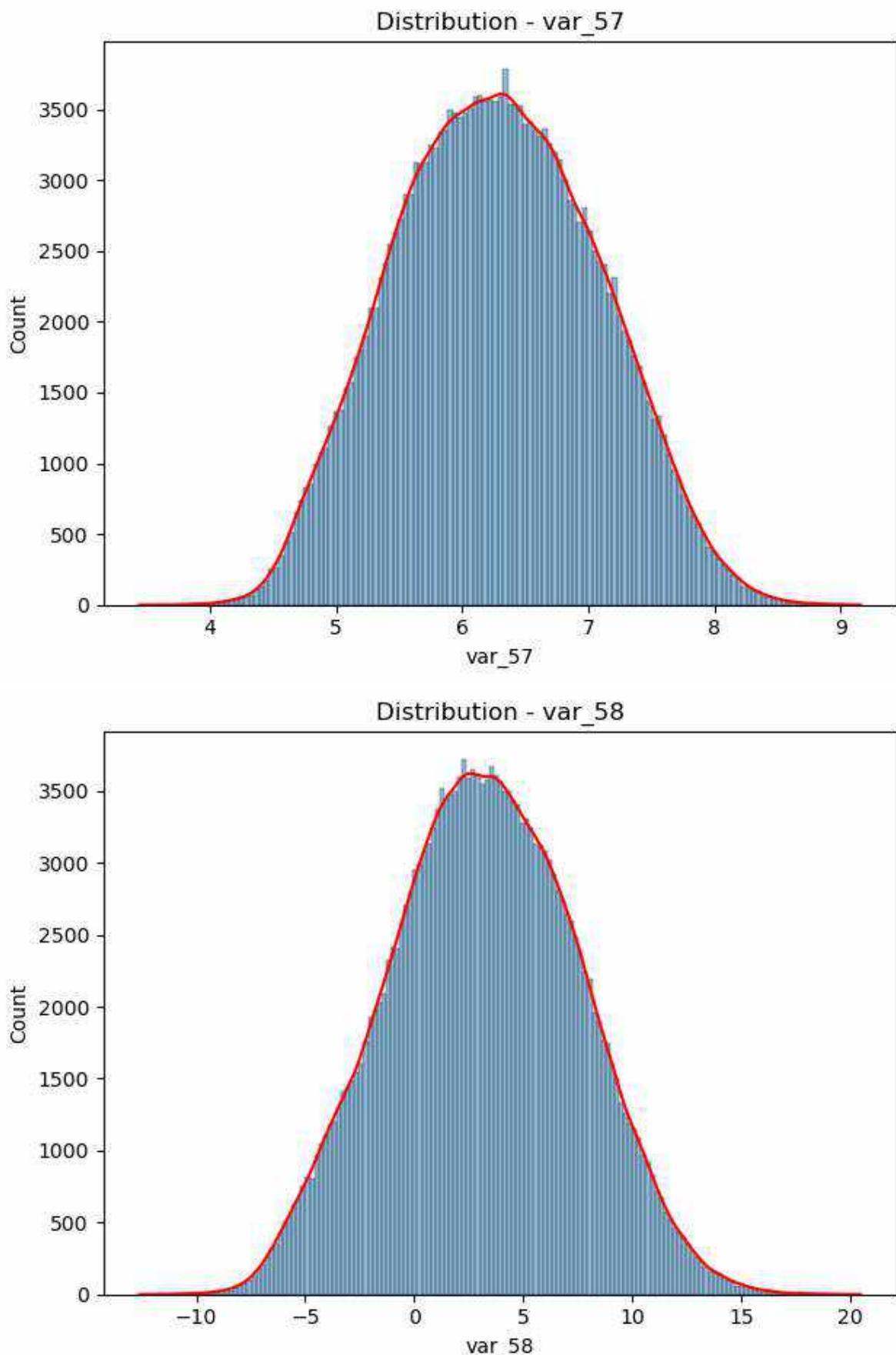


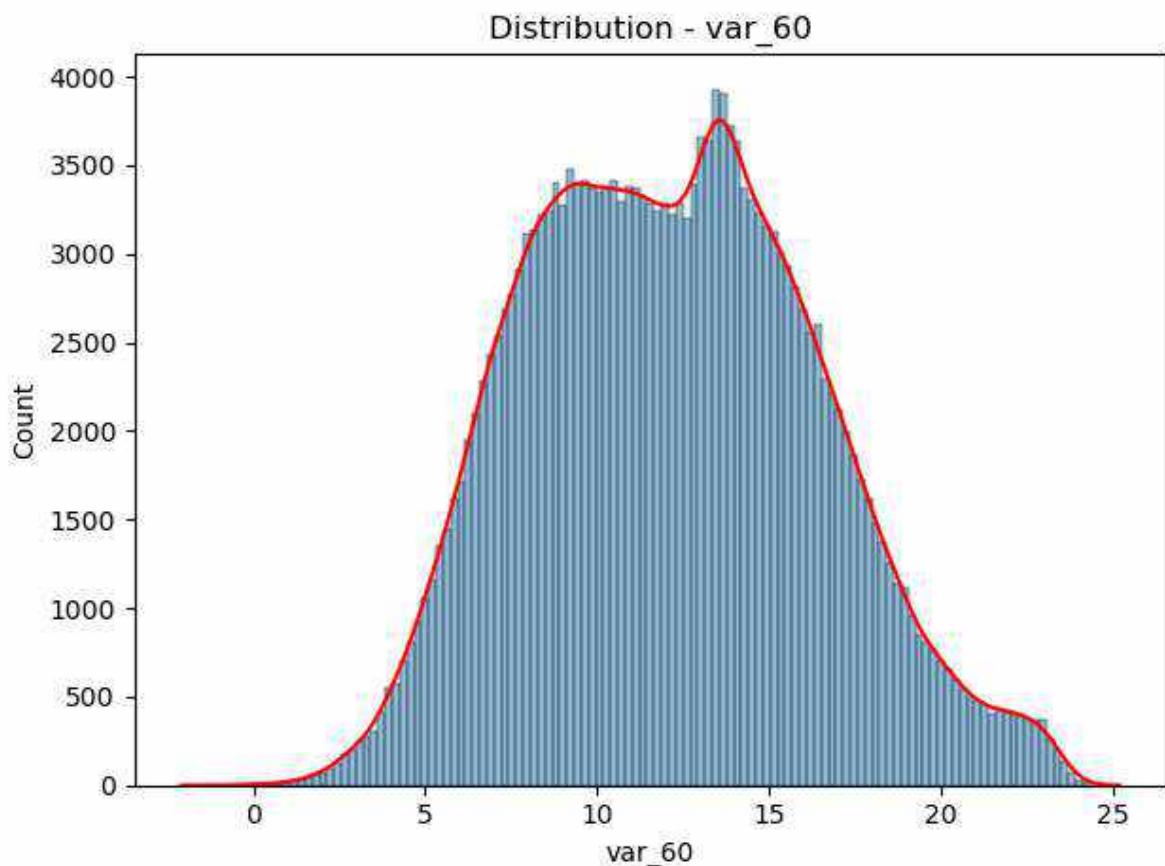
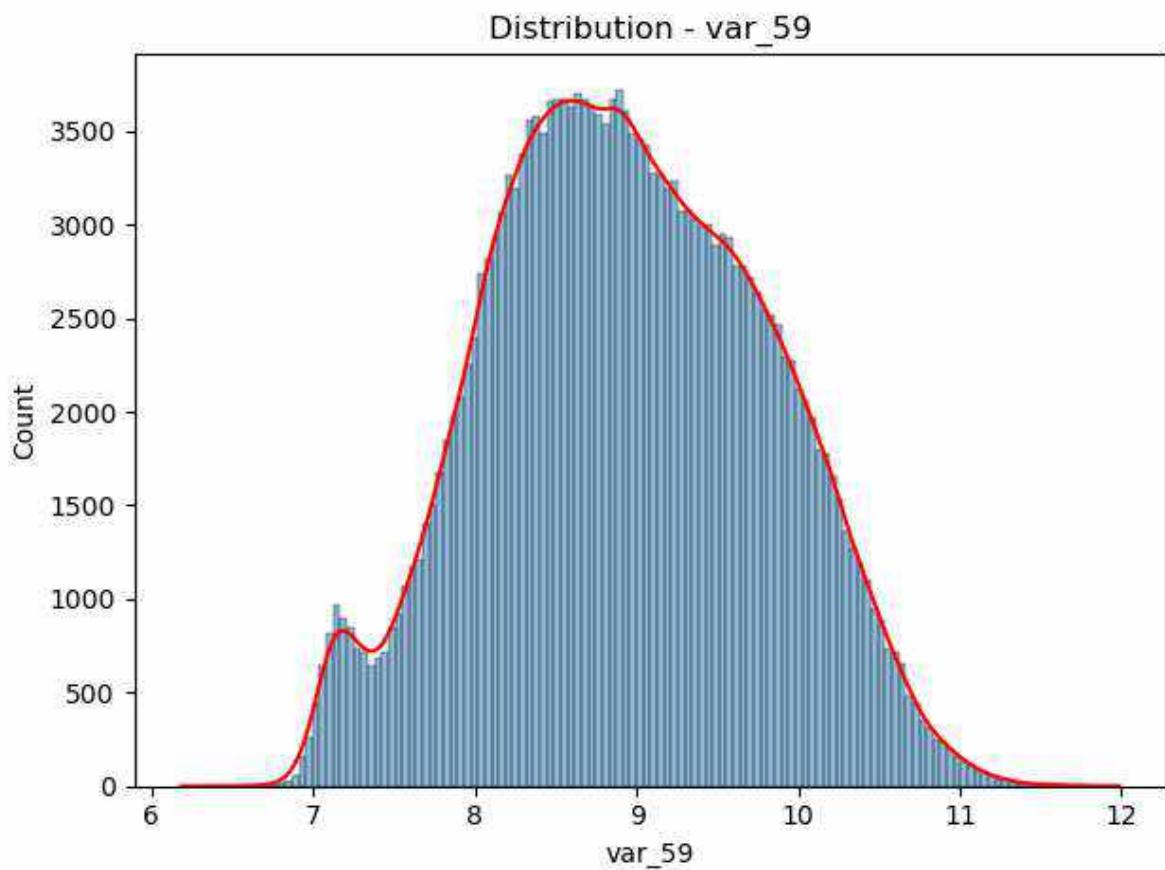


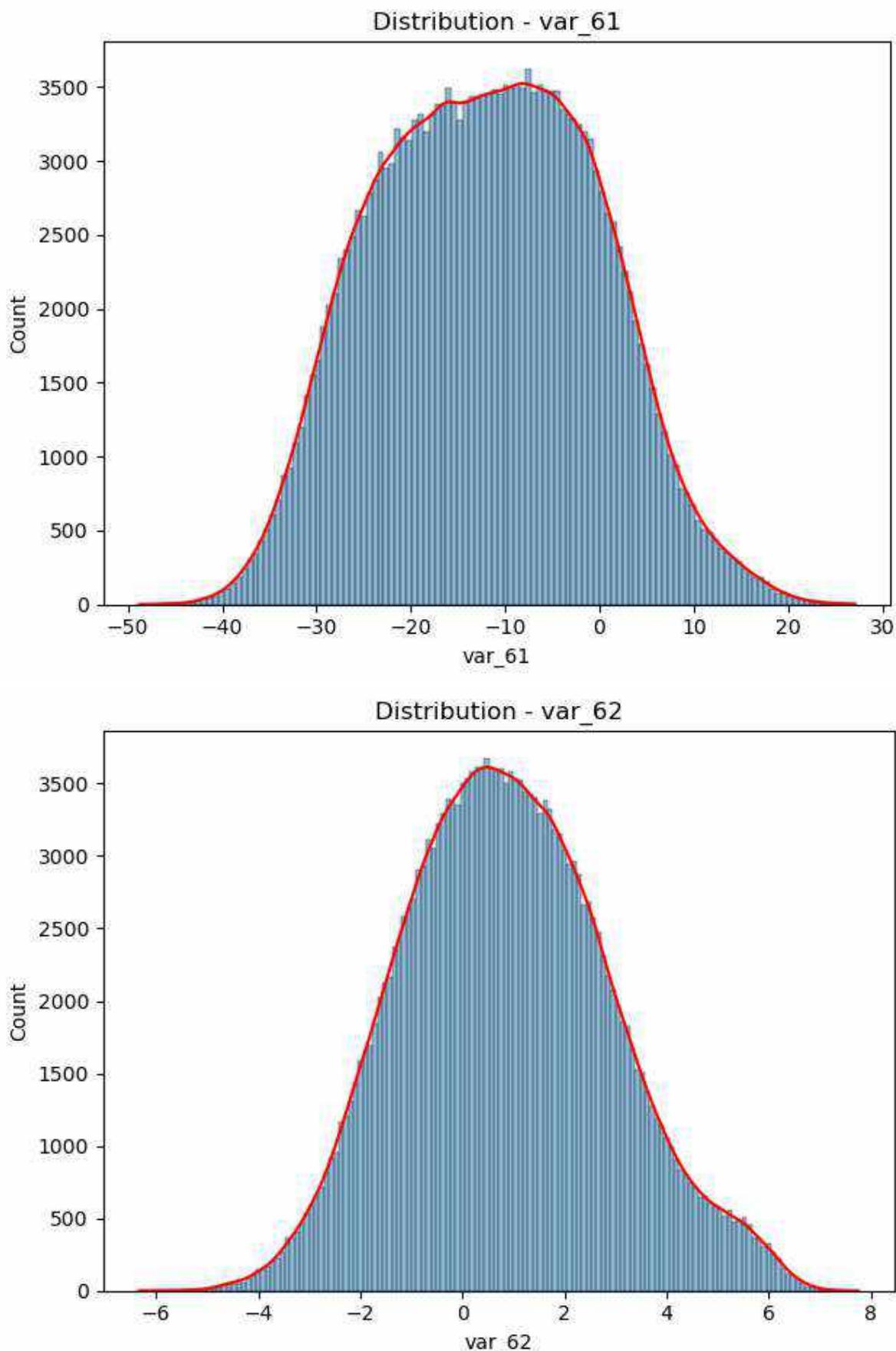


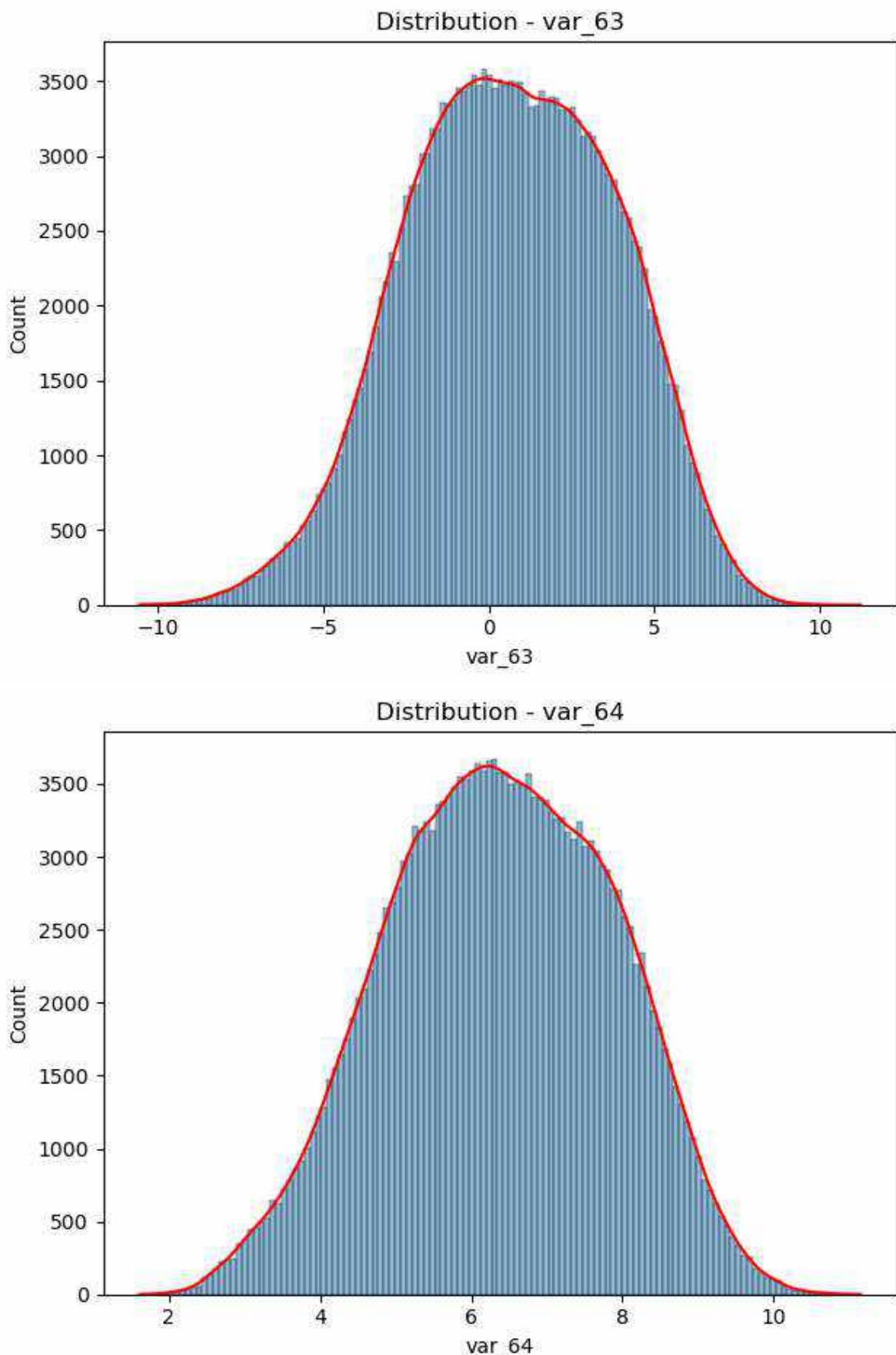


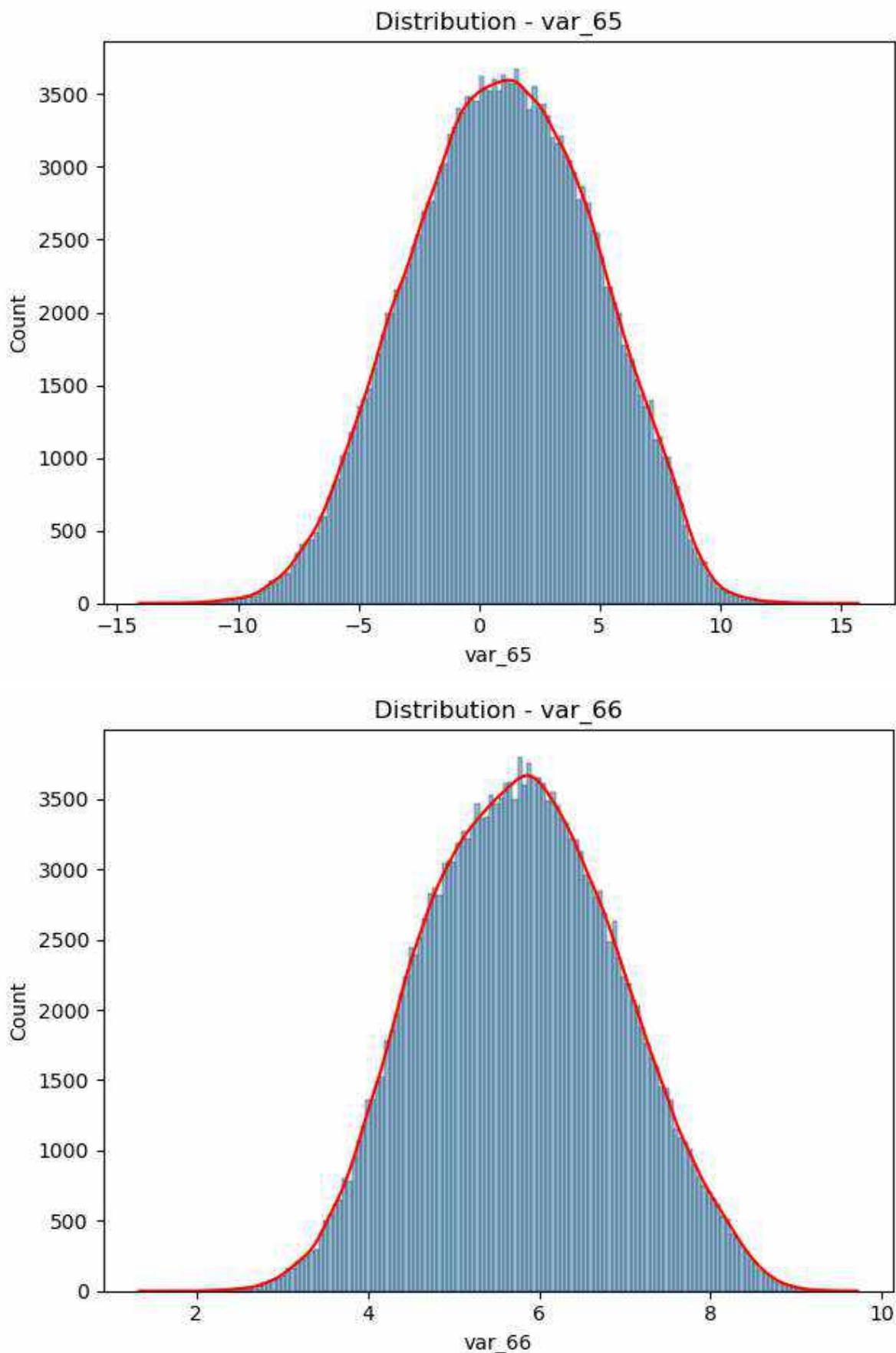


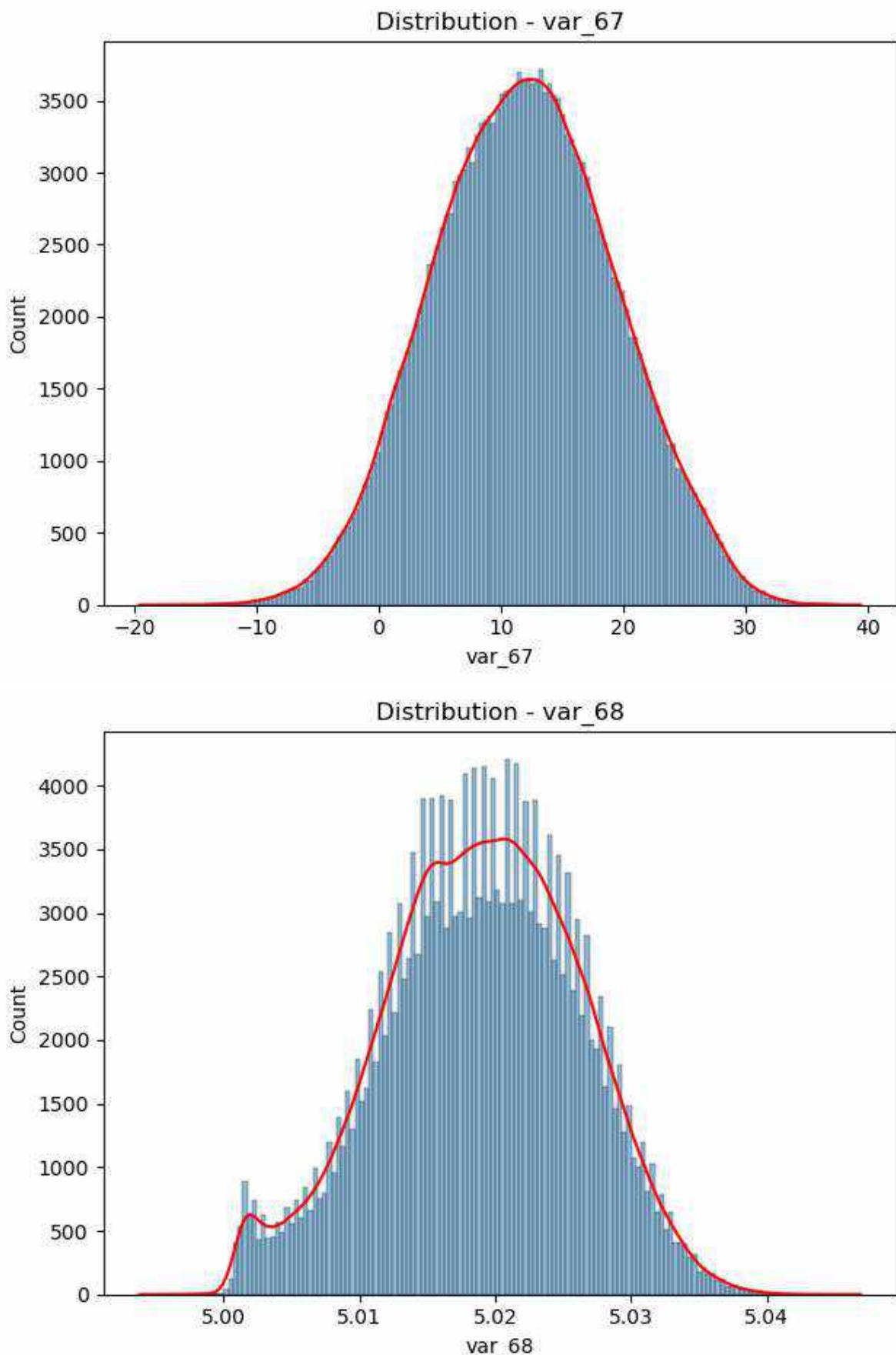


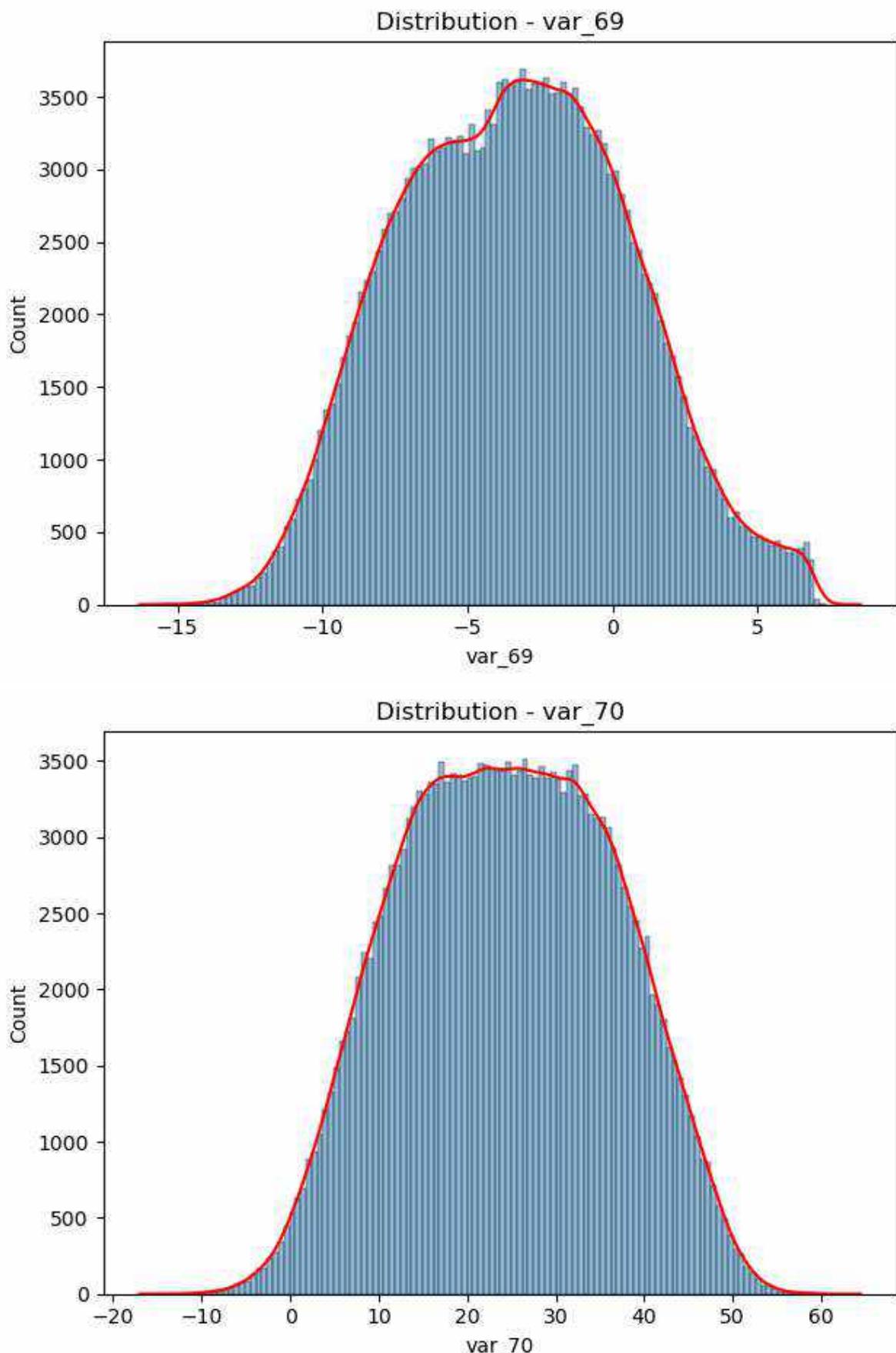


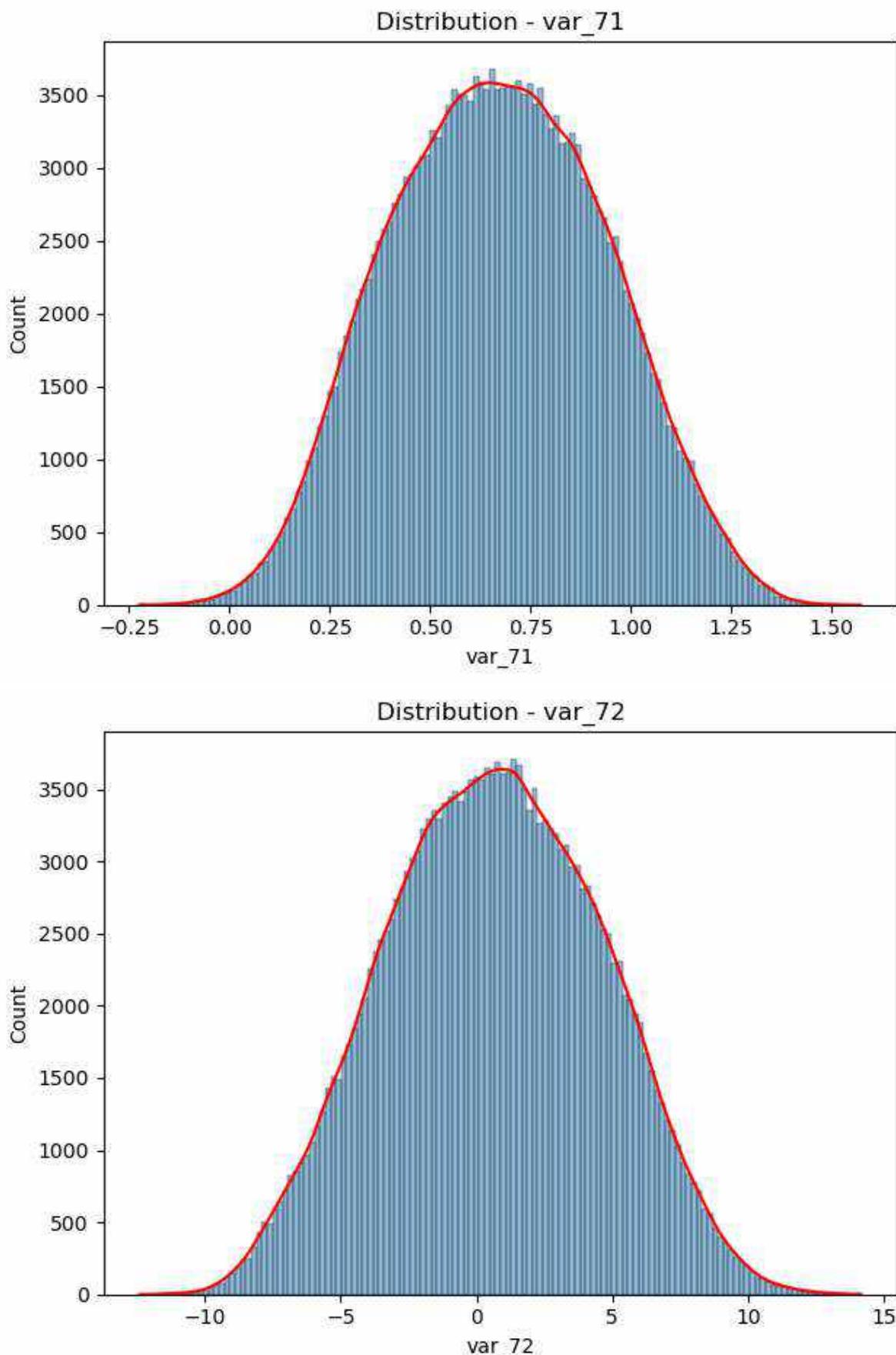




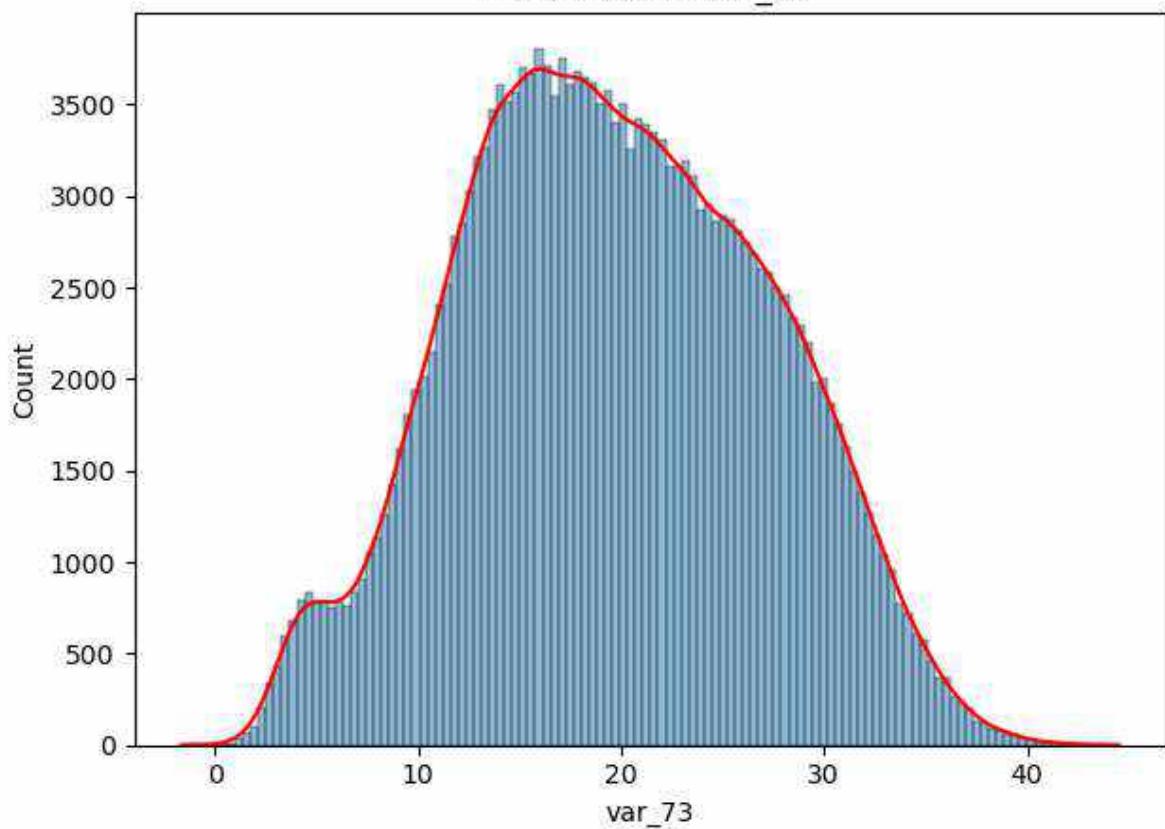




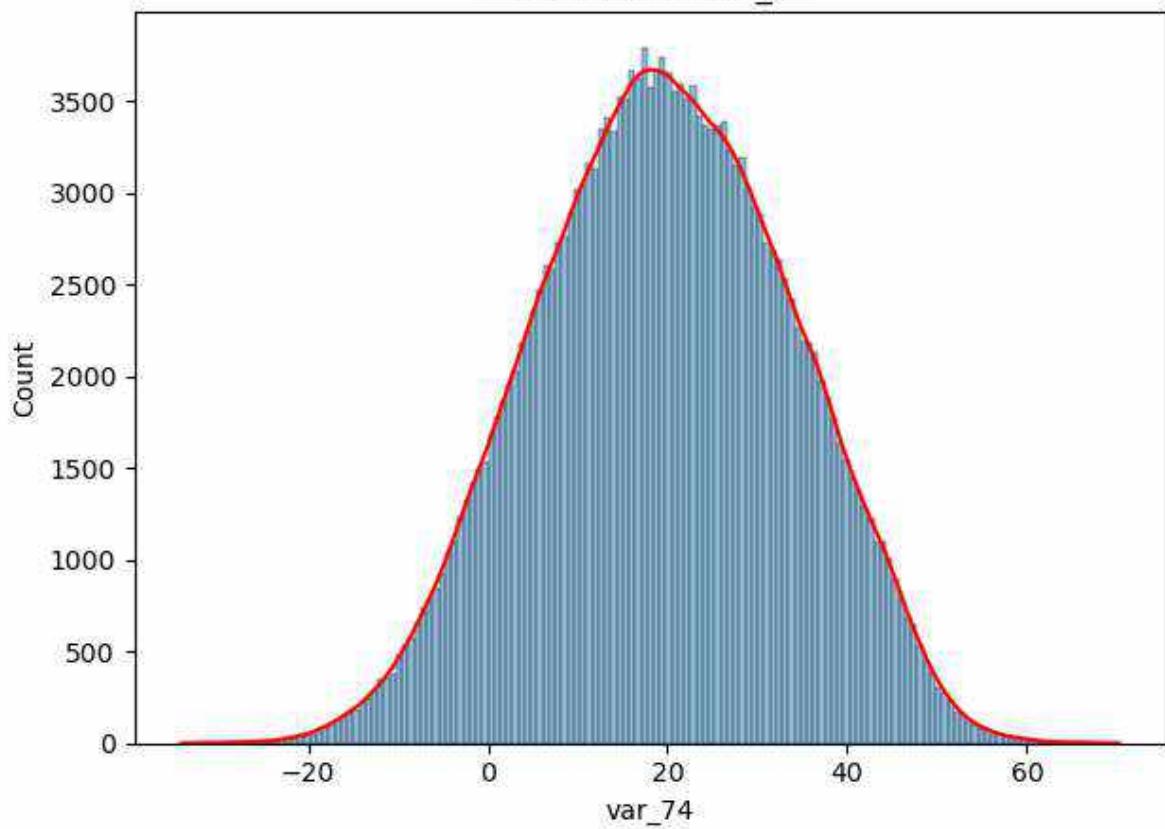


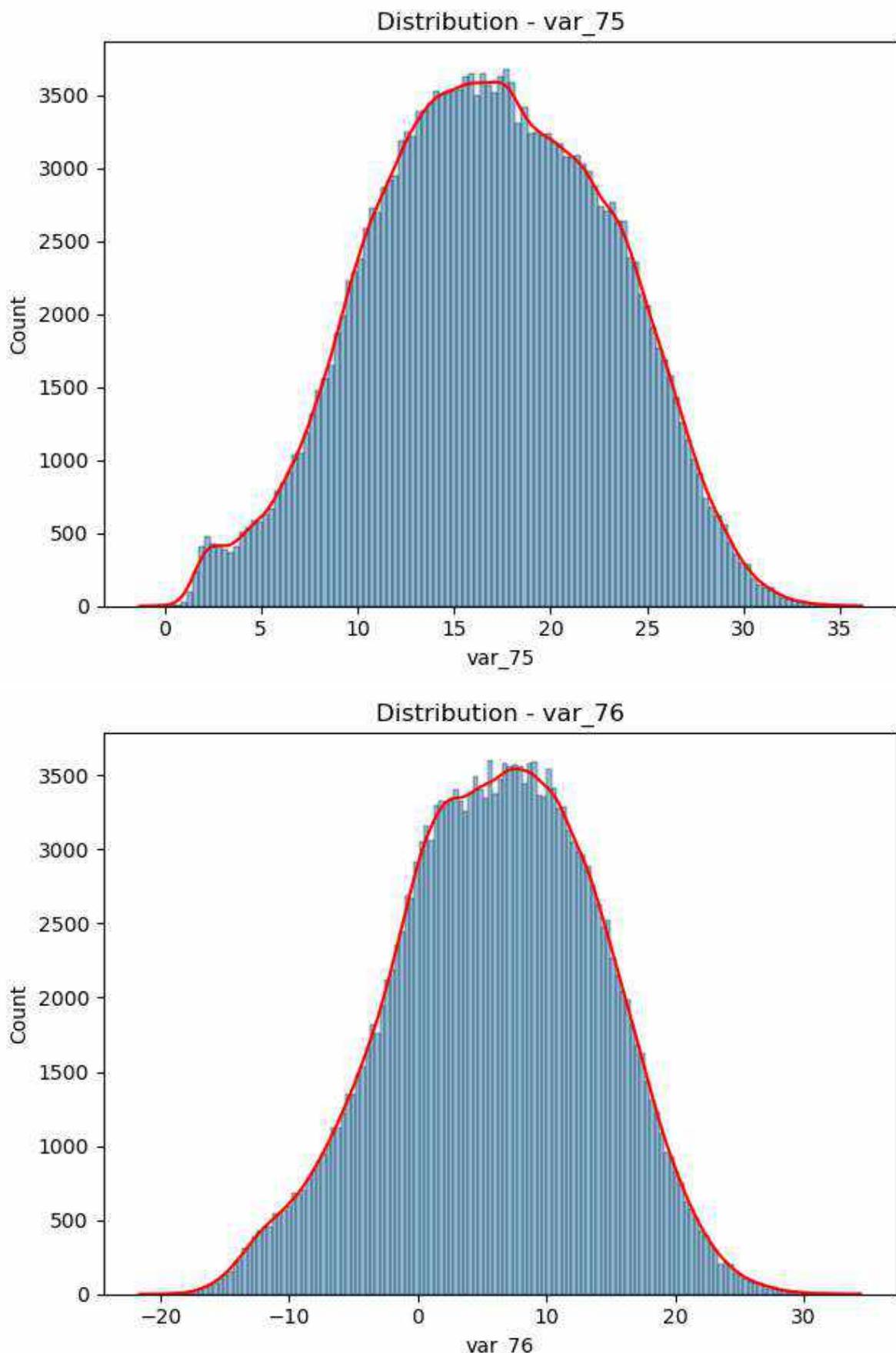


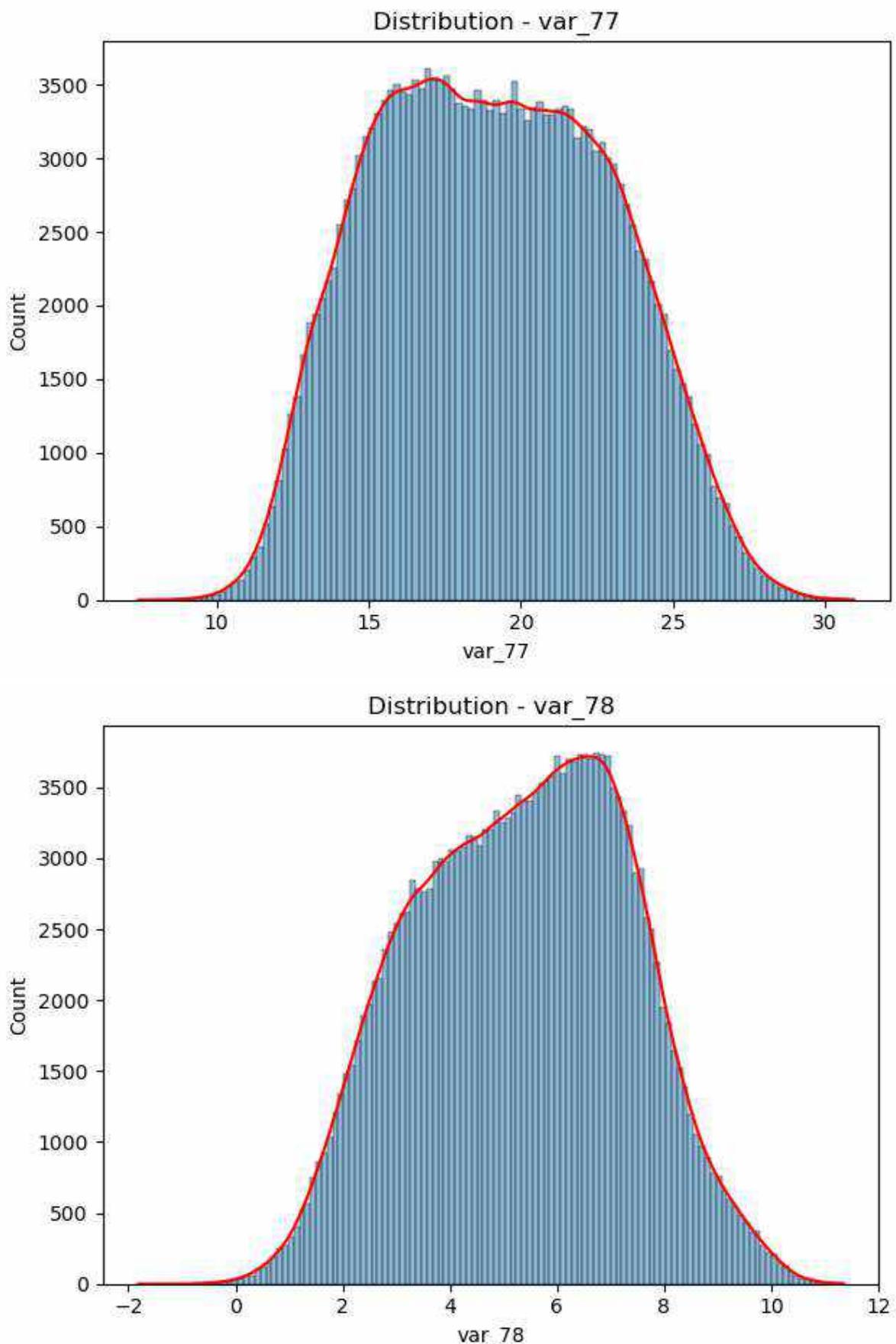
Distribution - var\_73

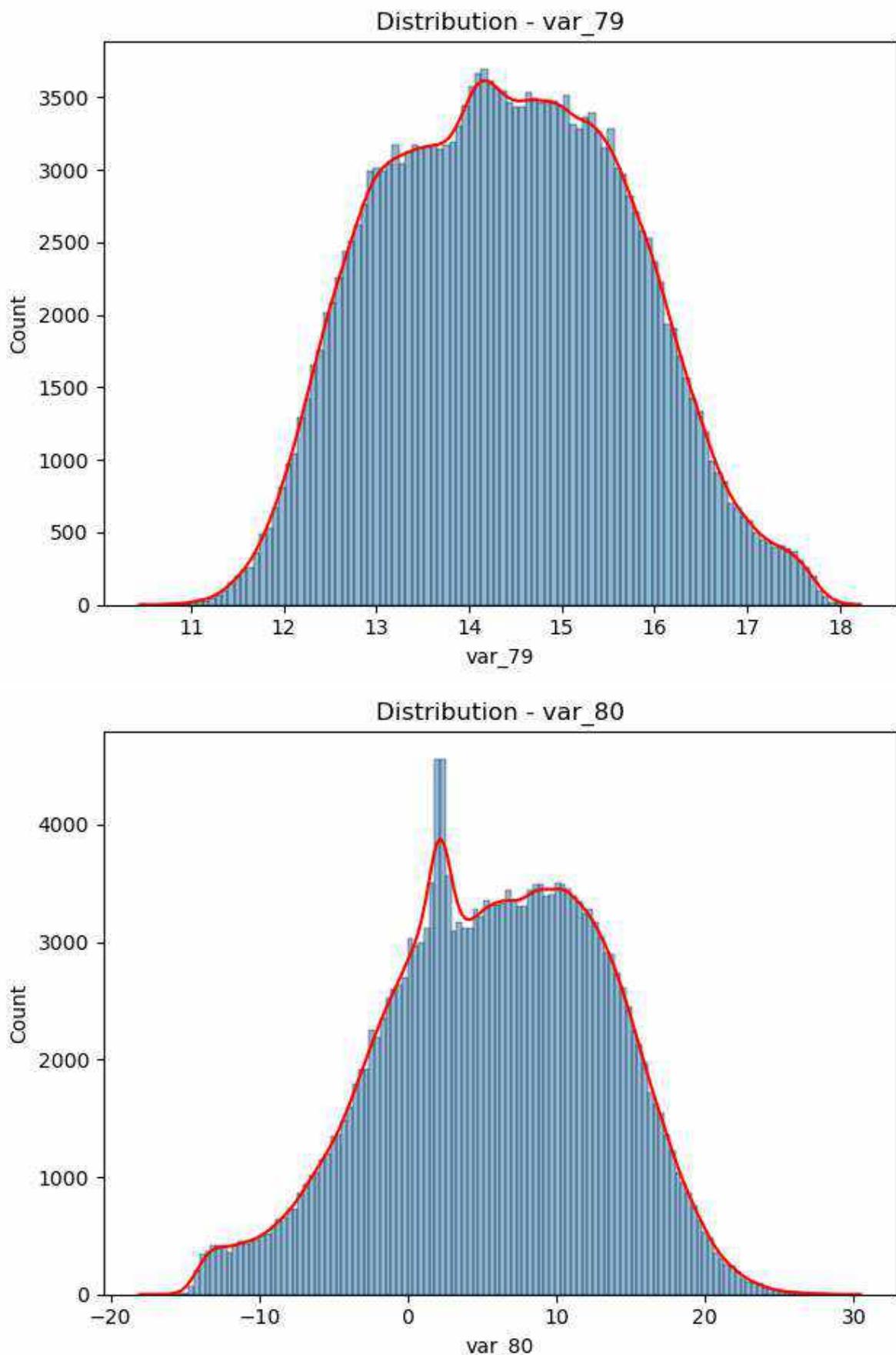


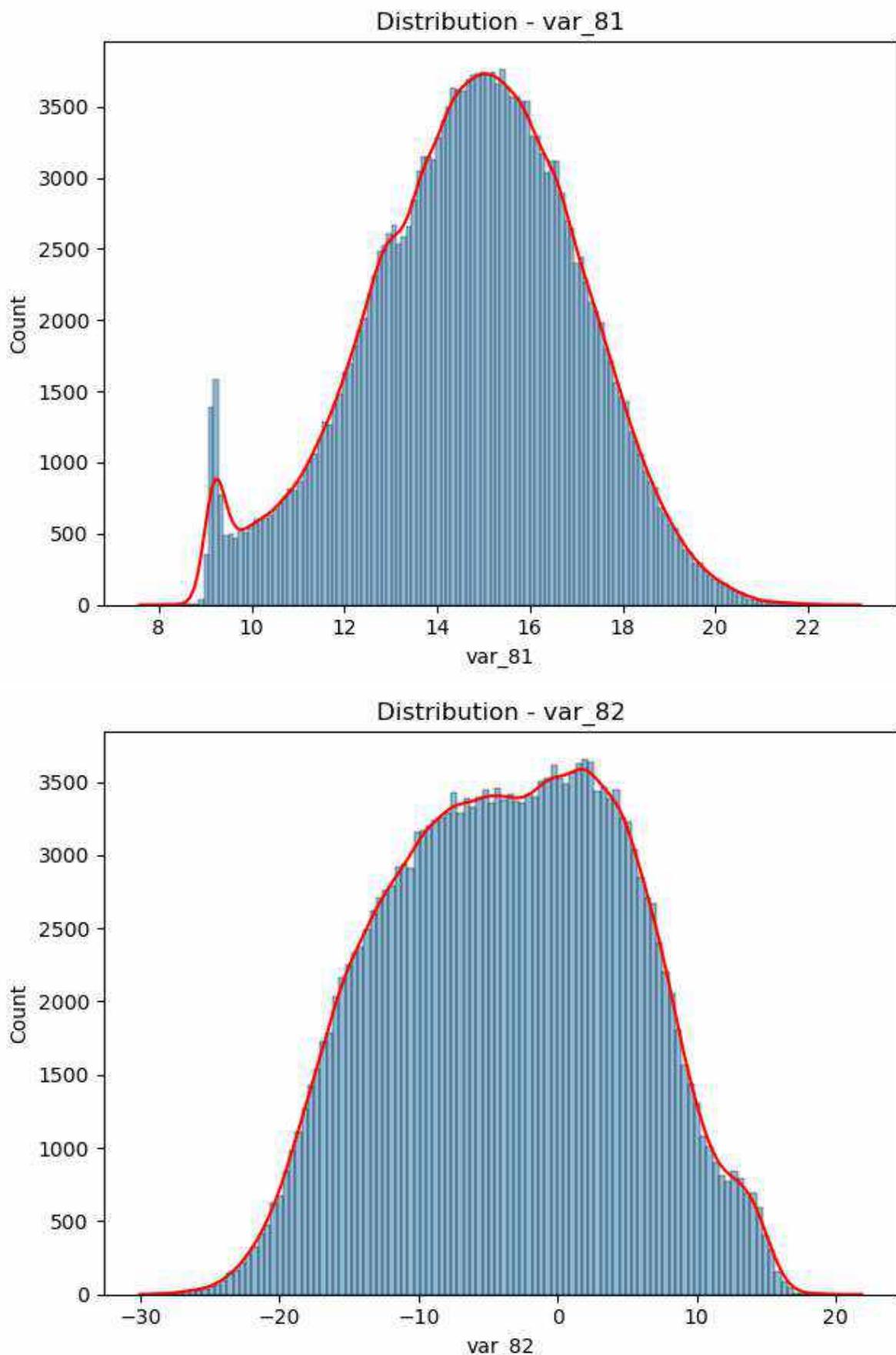
Distribution - var\_74

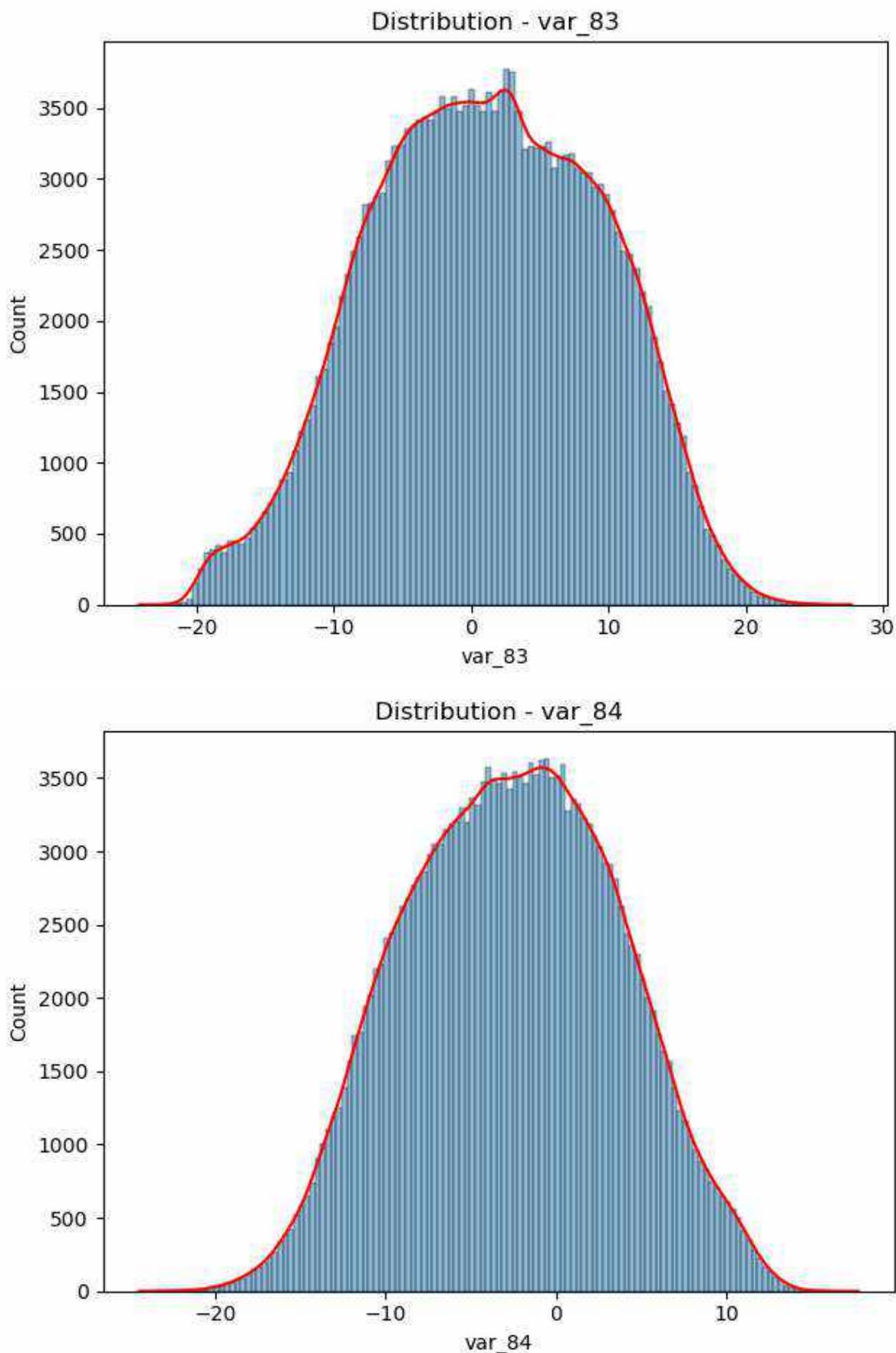


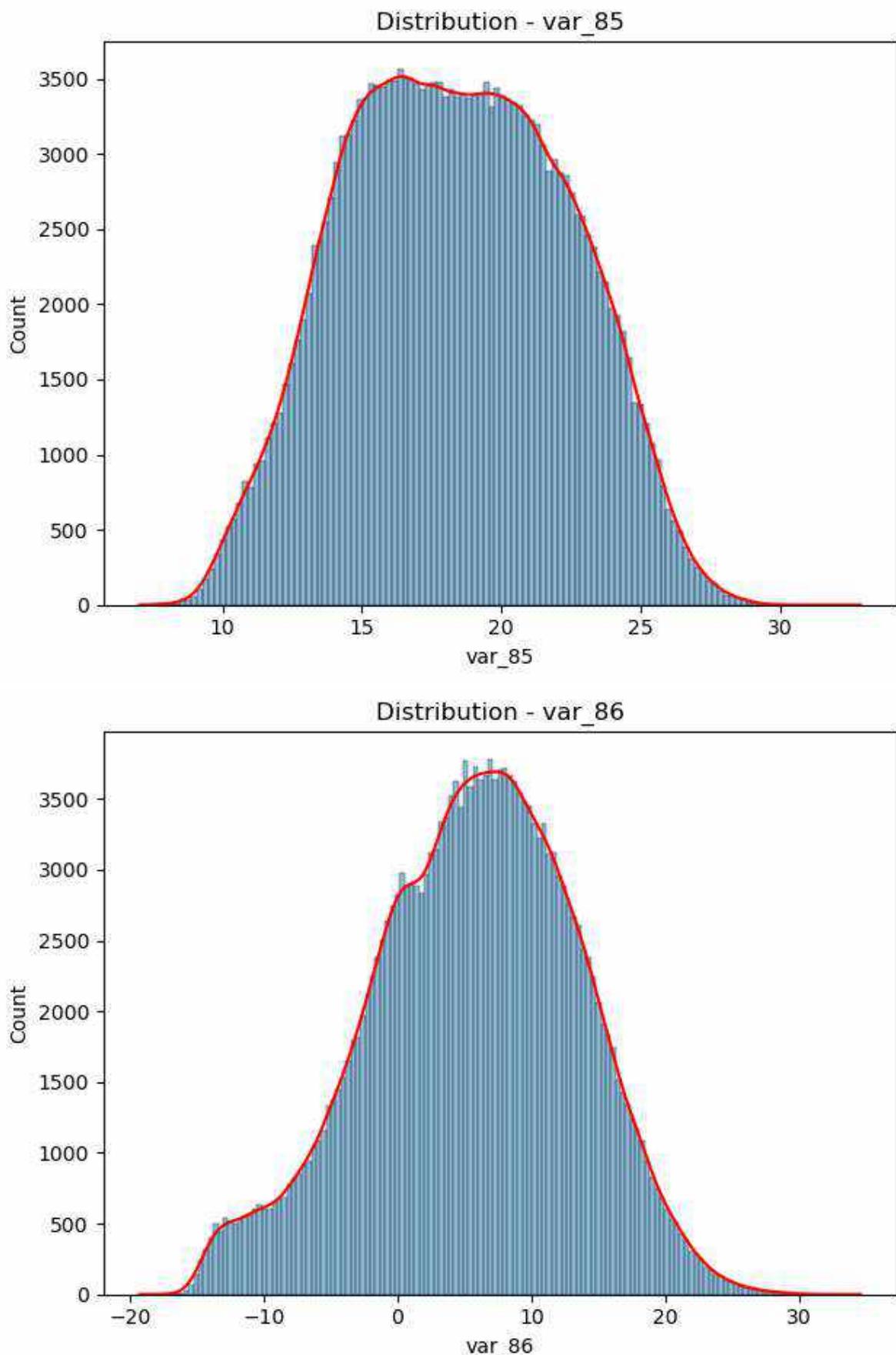


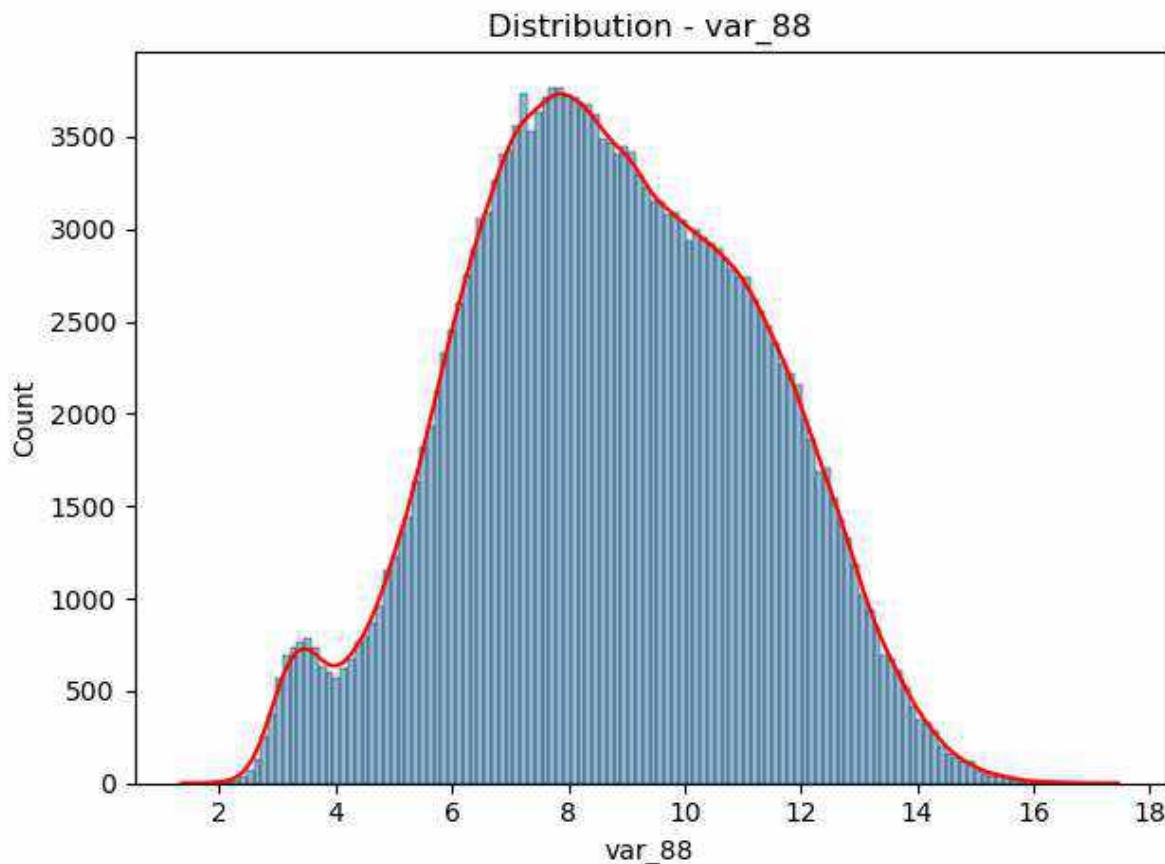
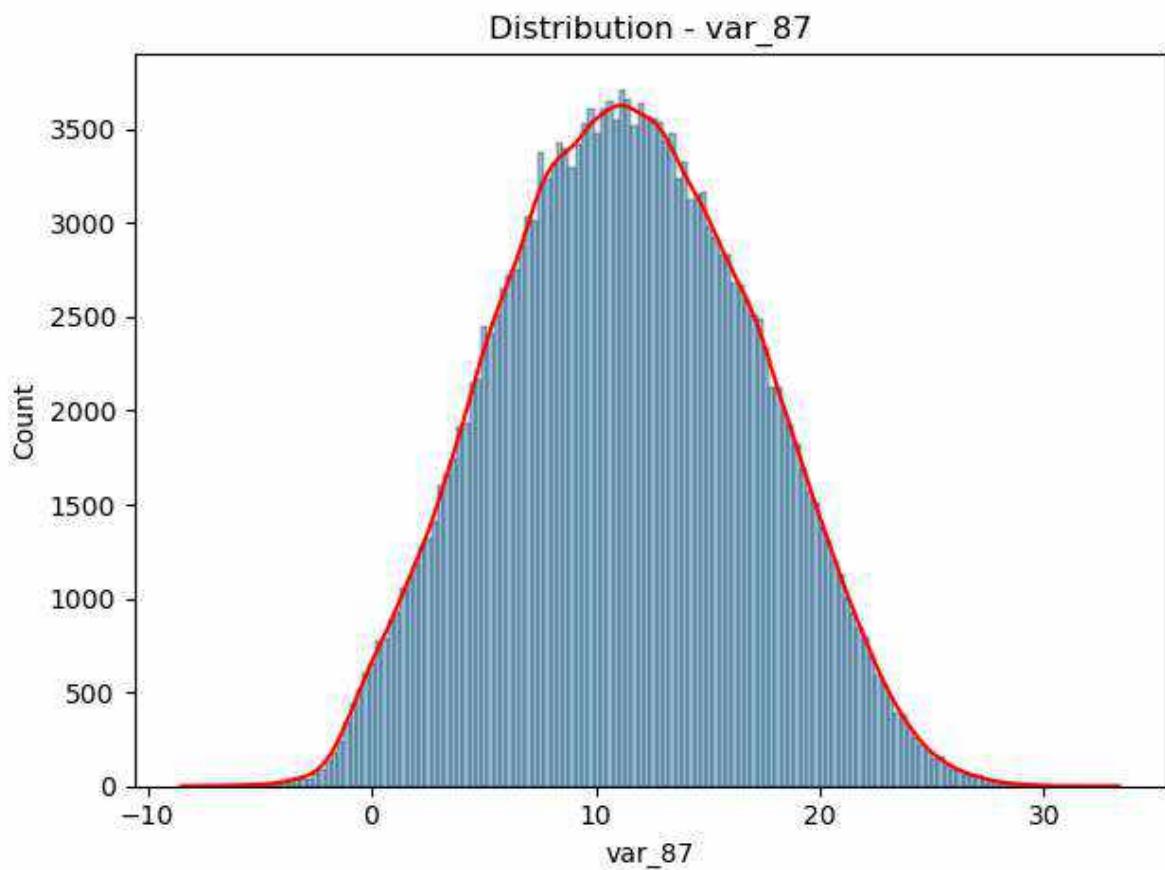


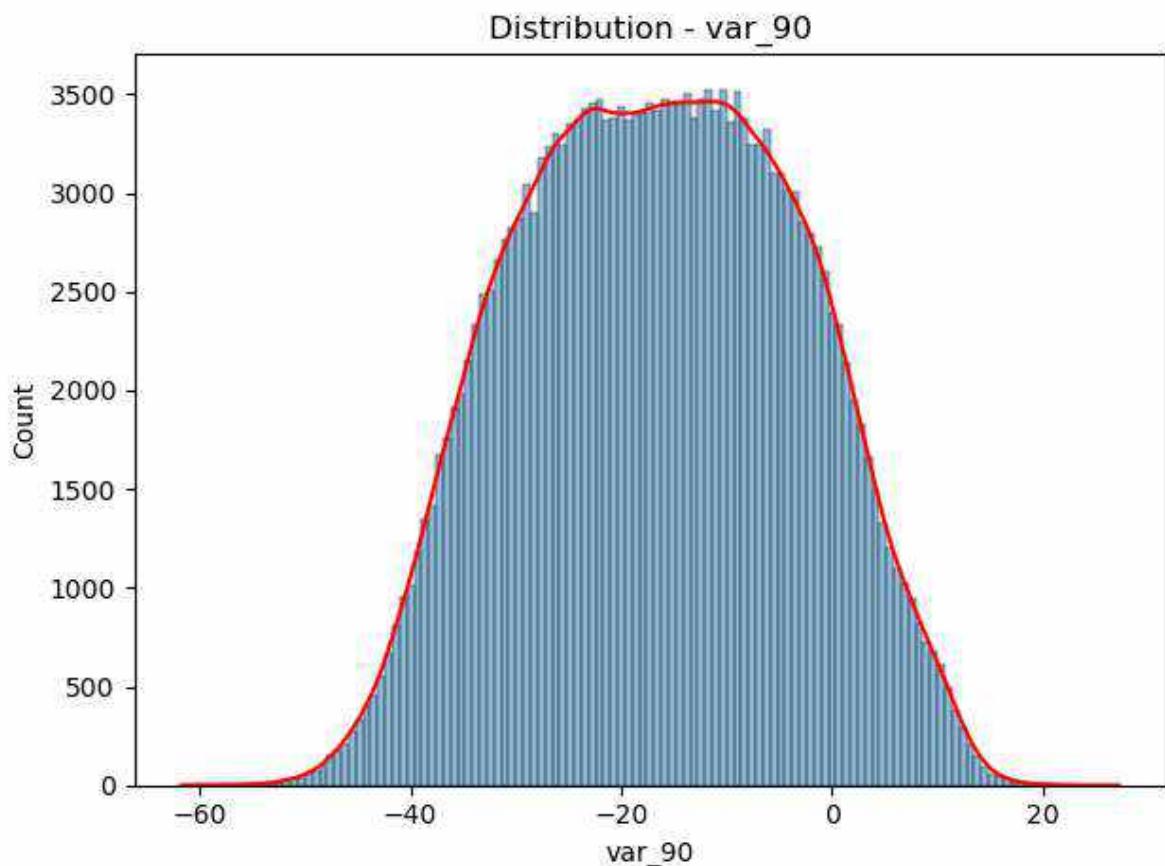
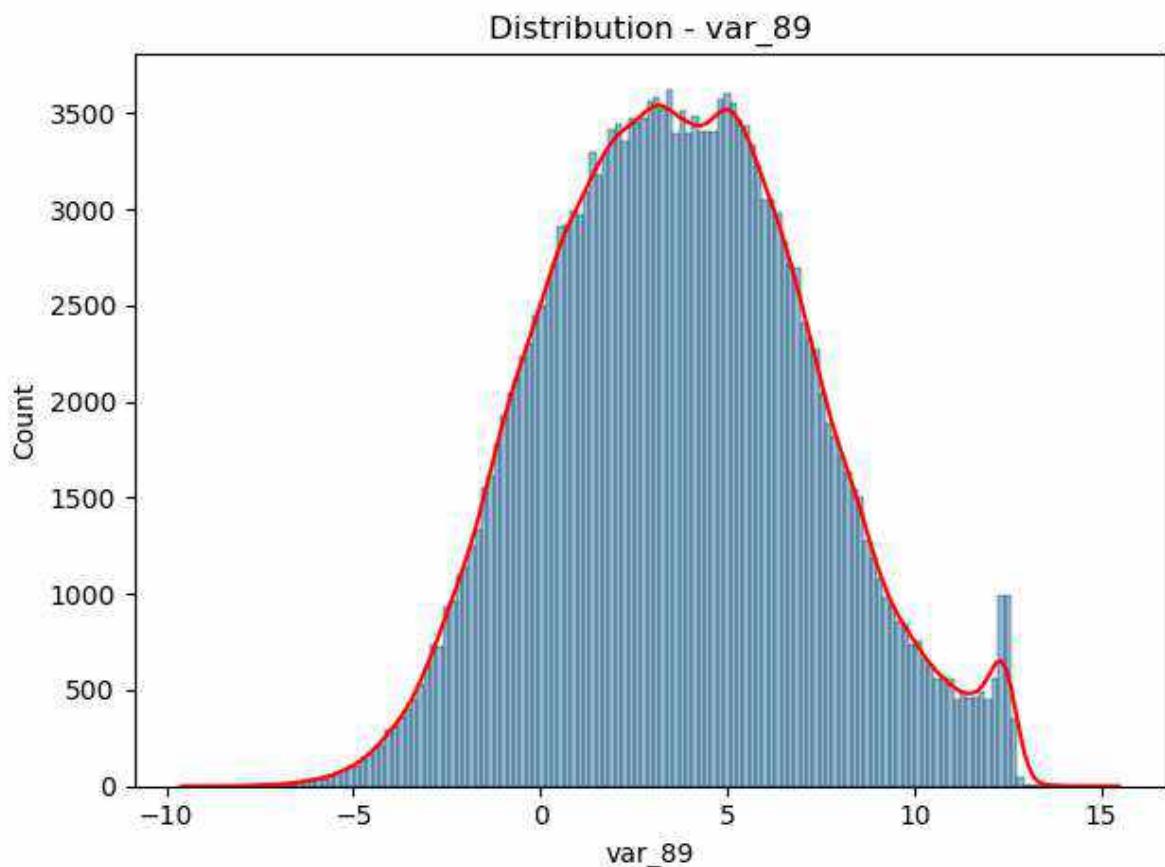


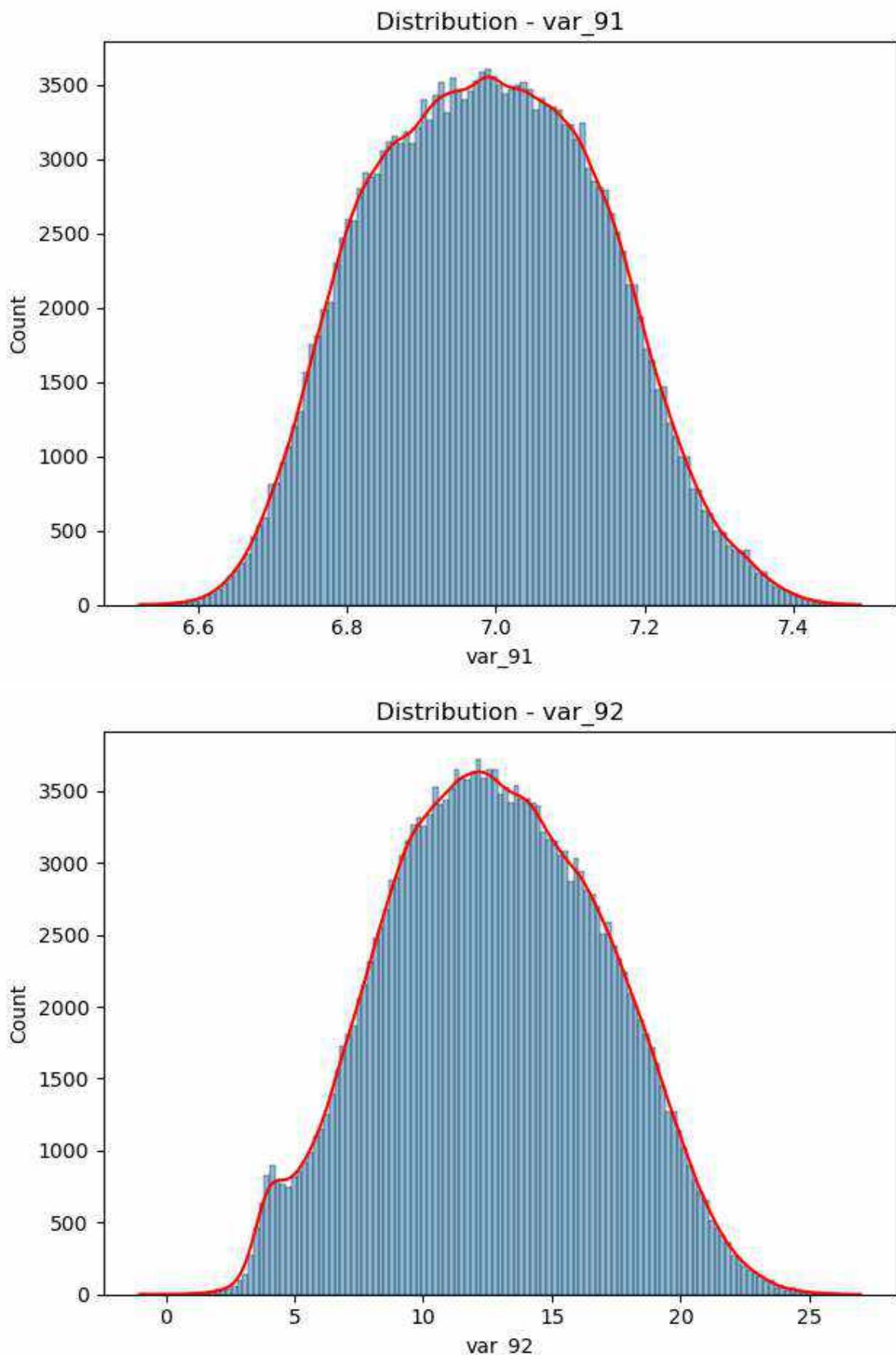


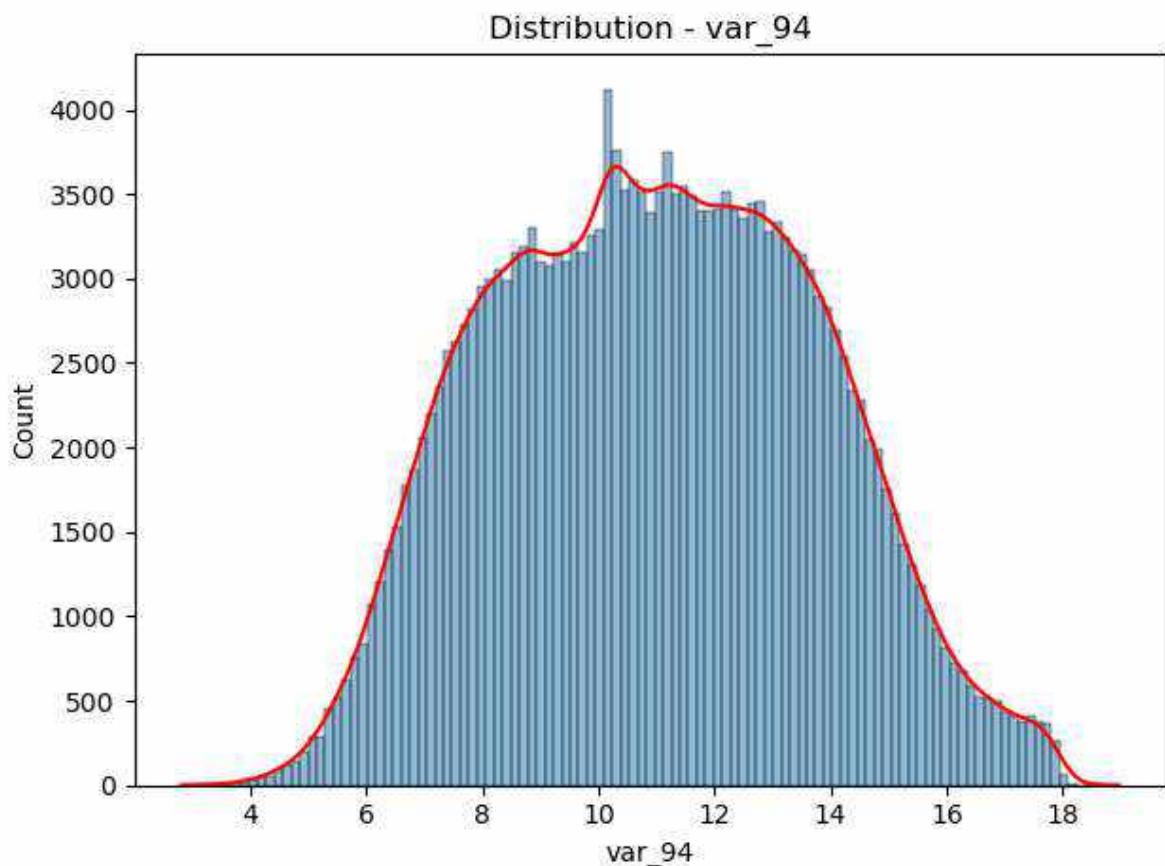
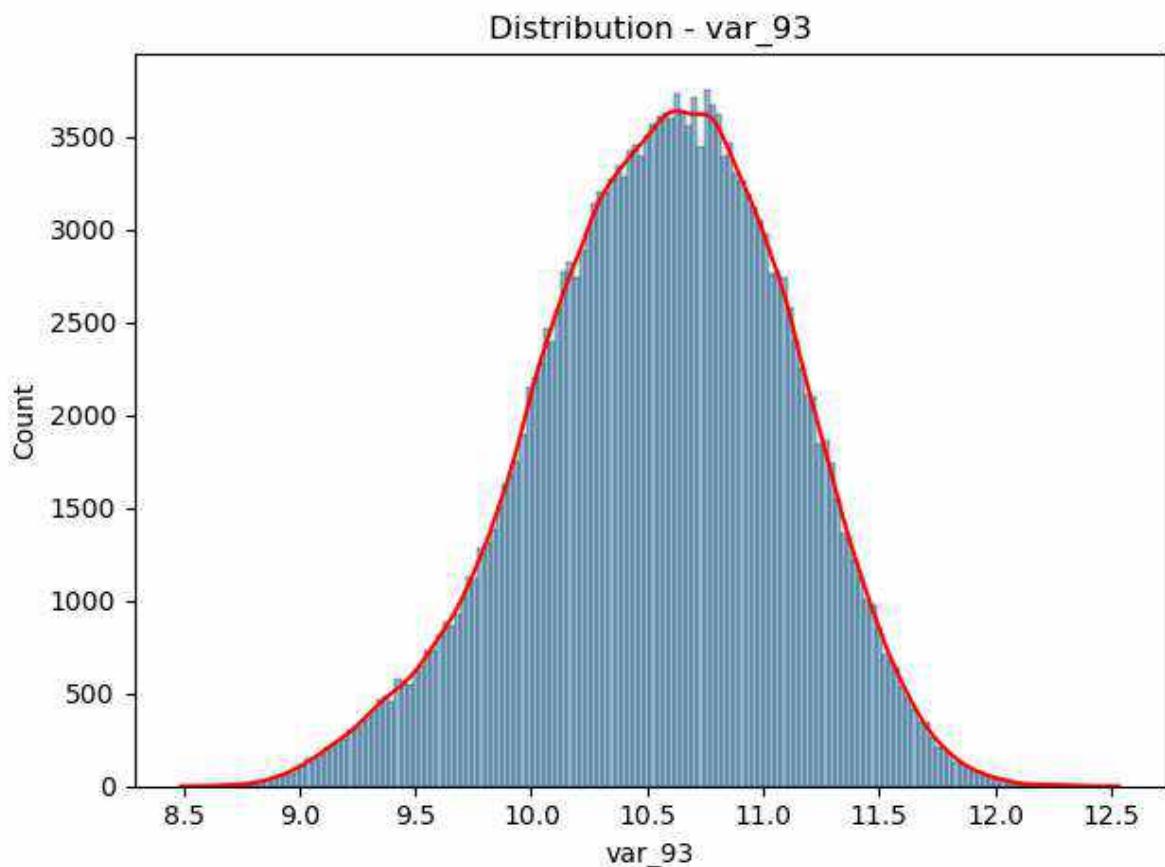


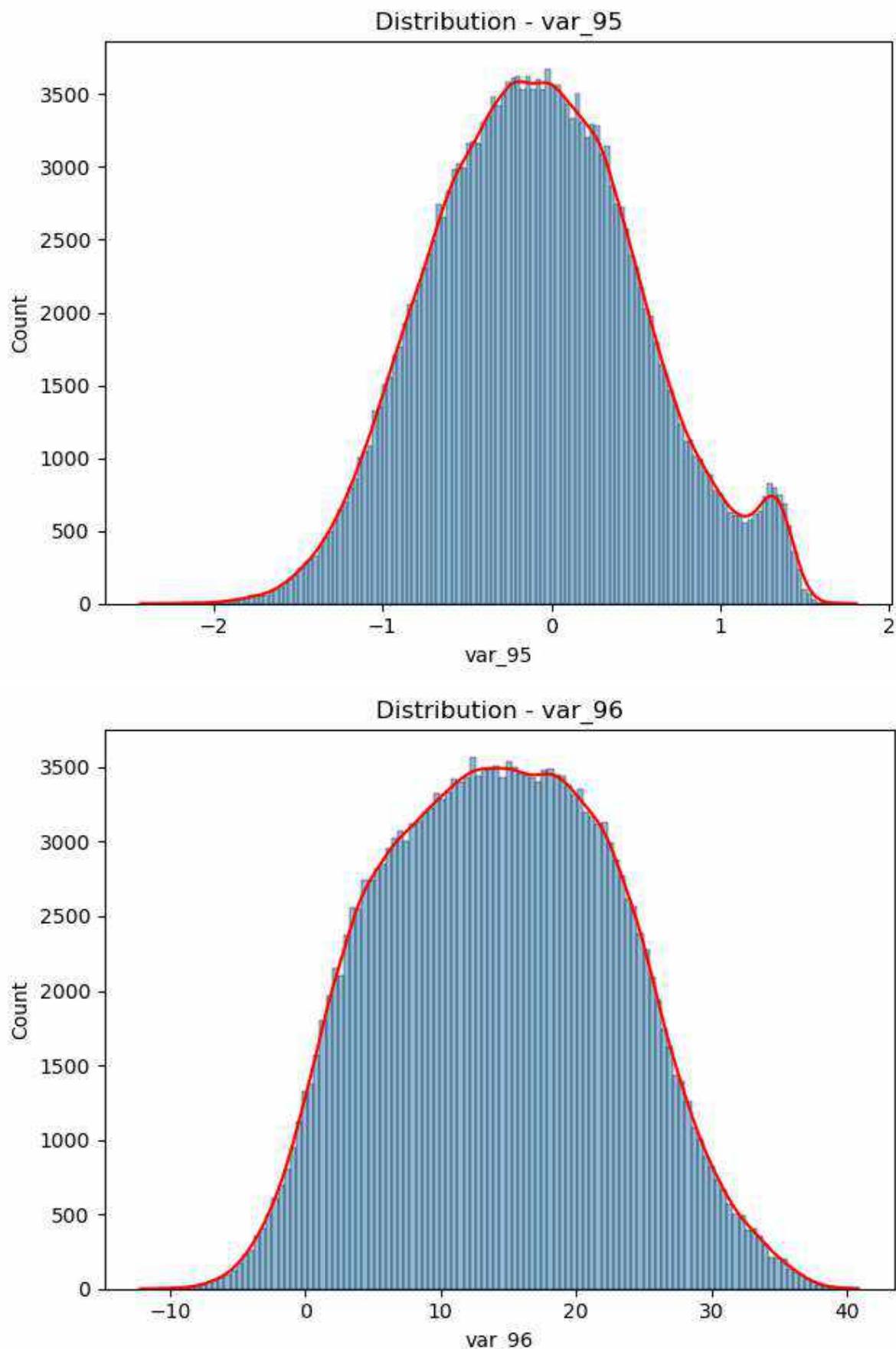


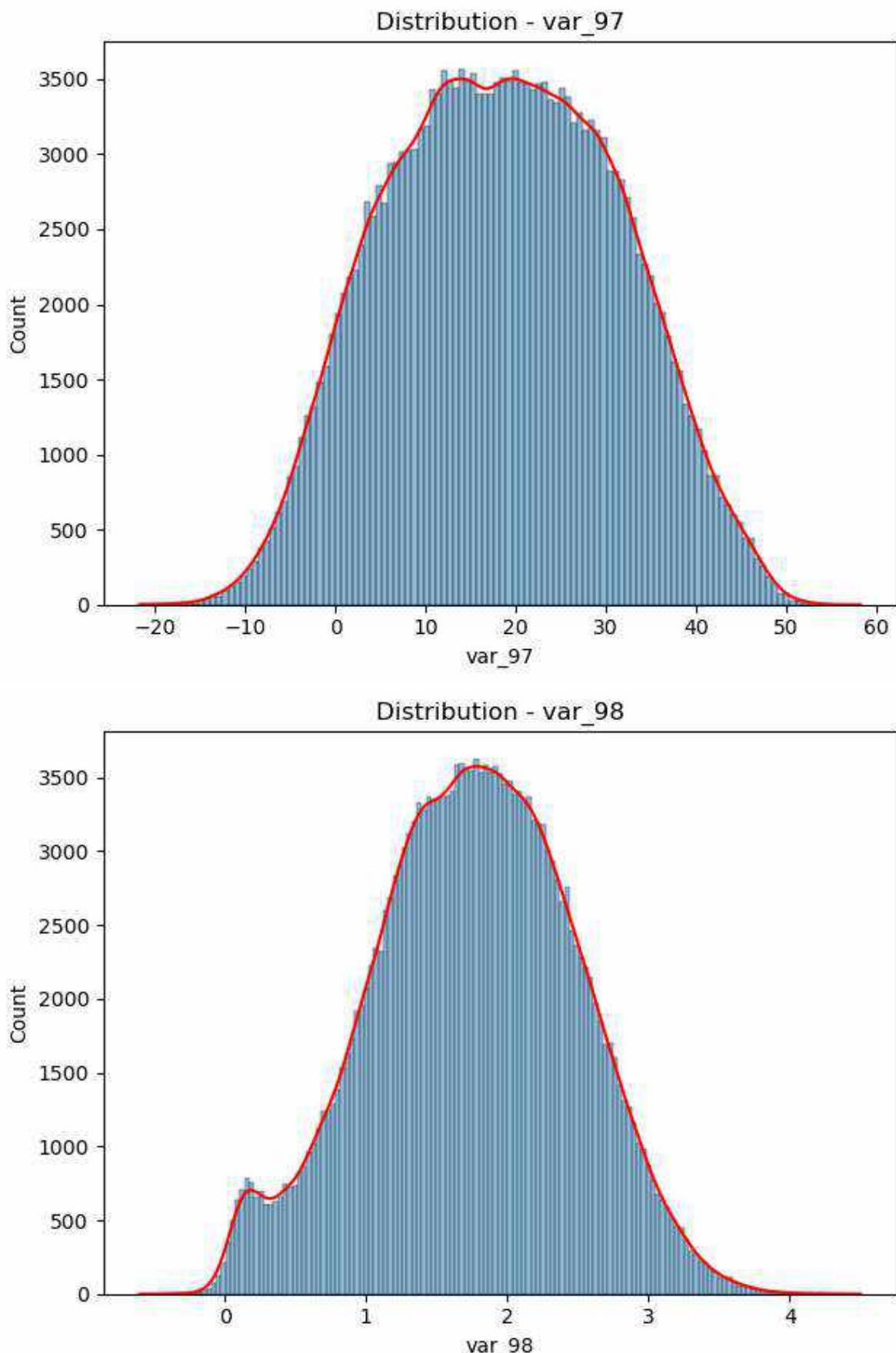


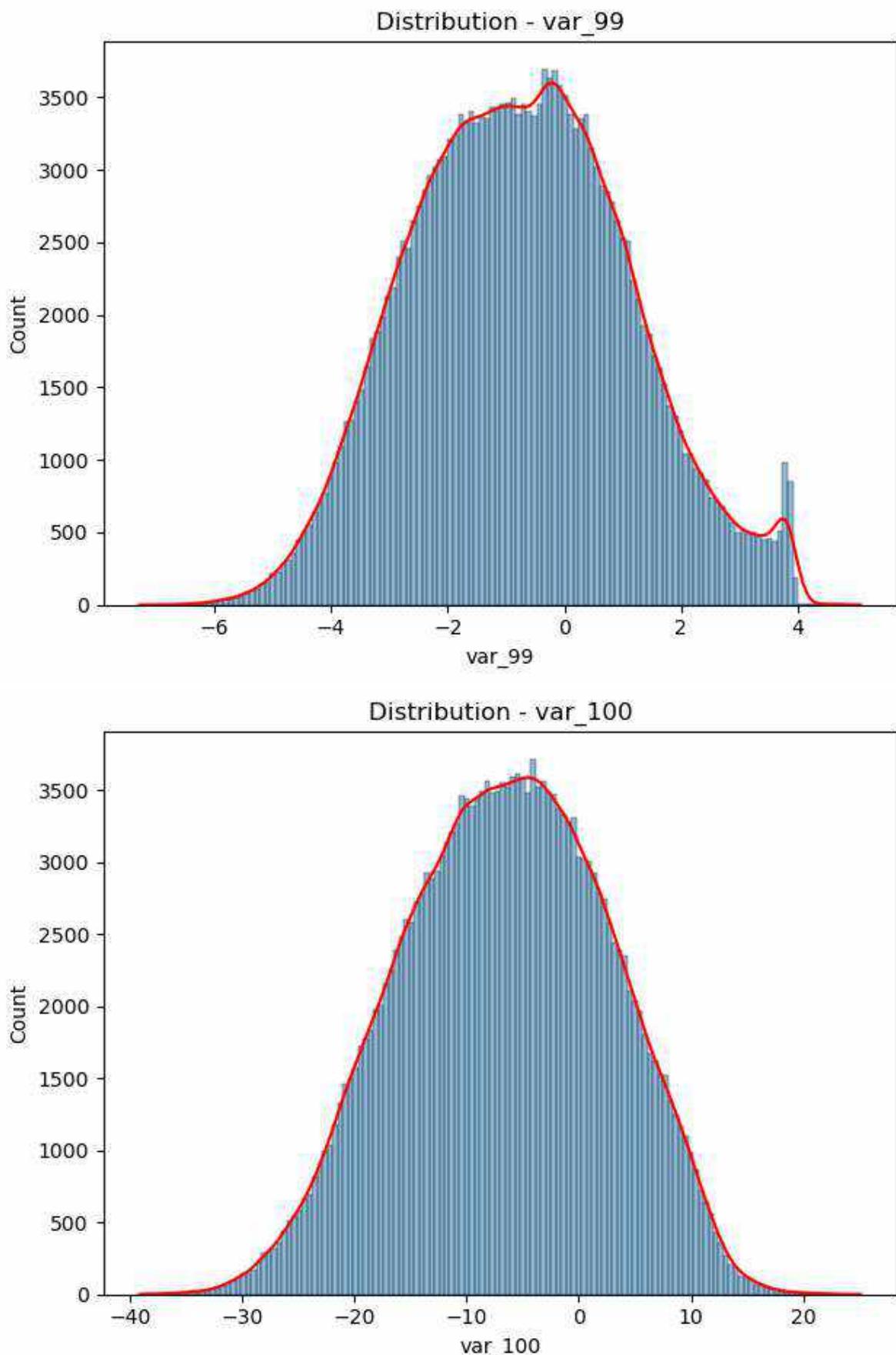


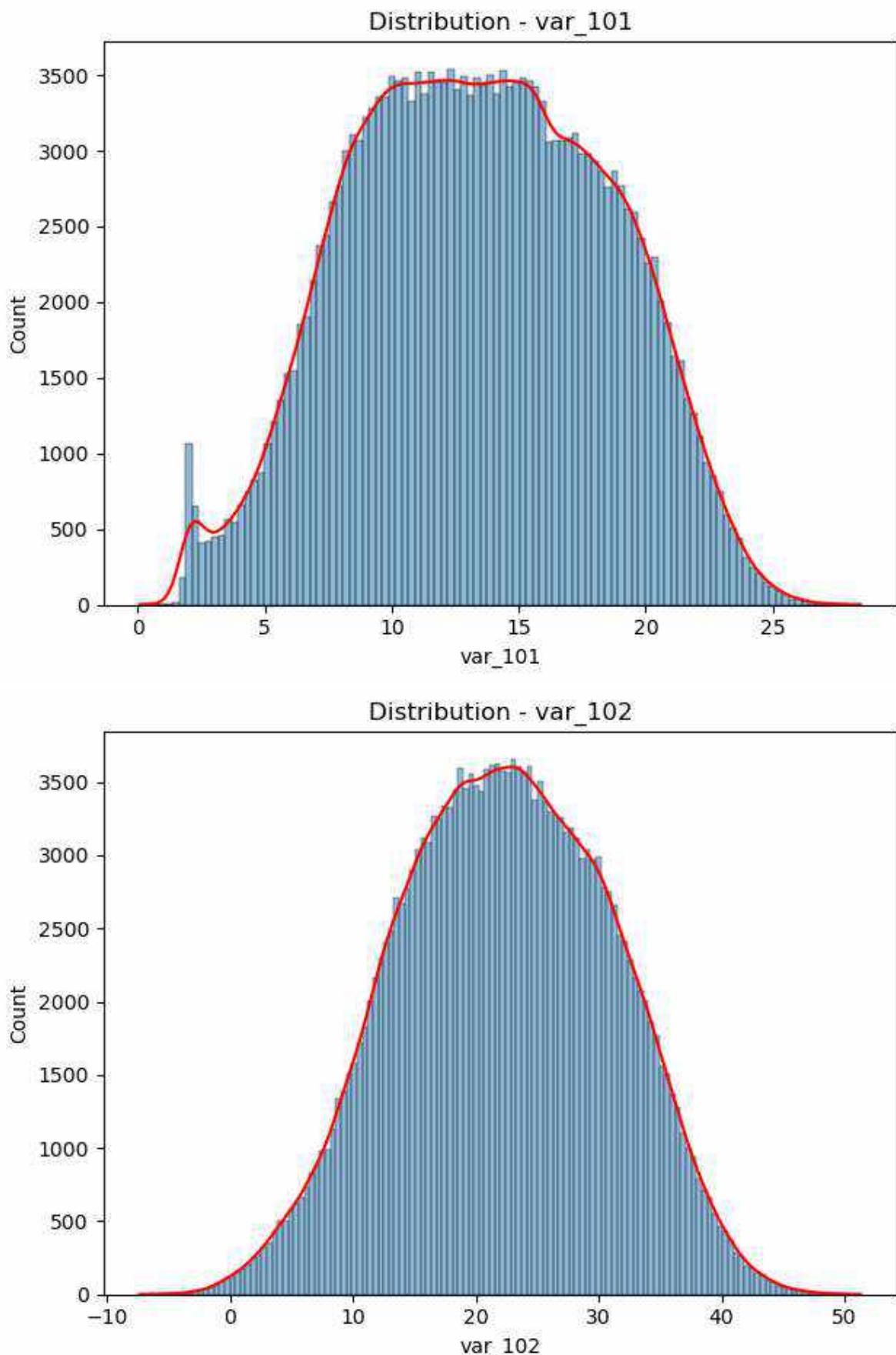


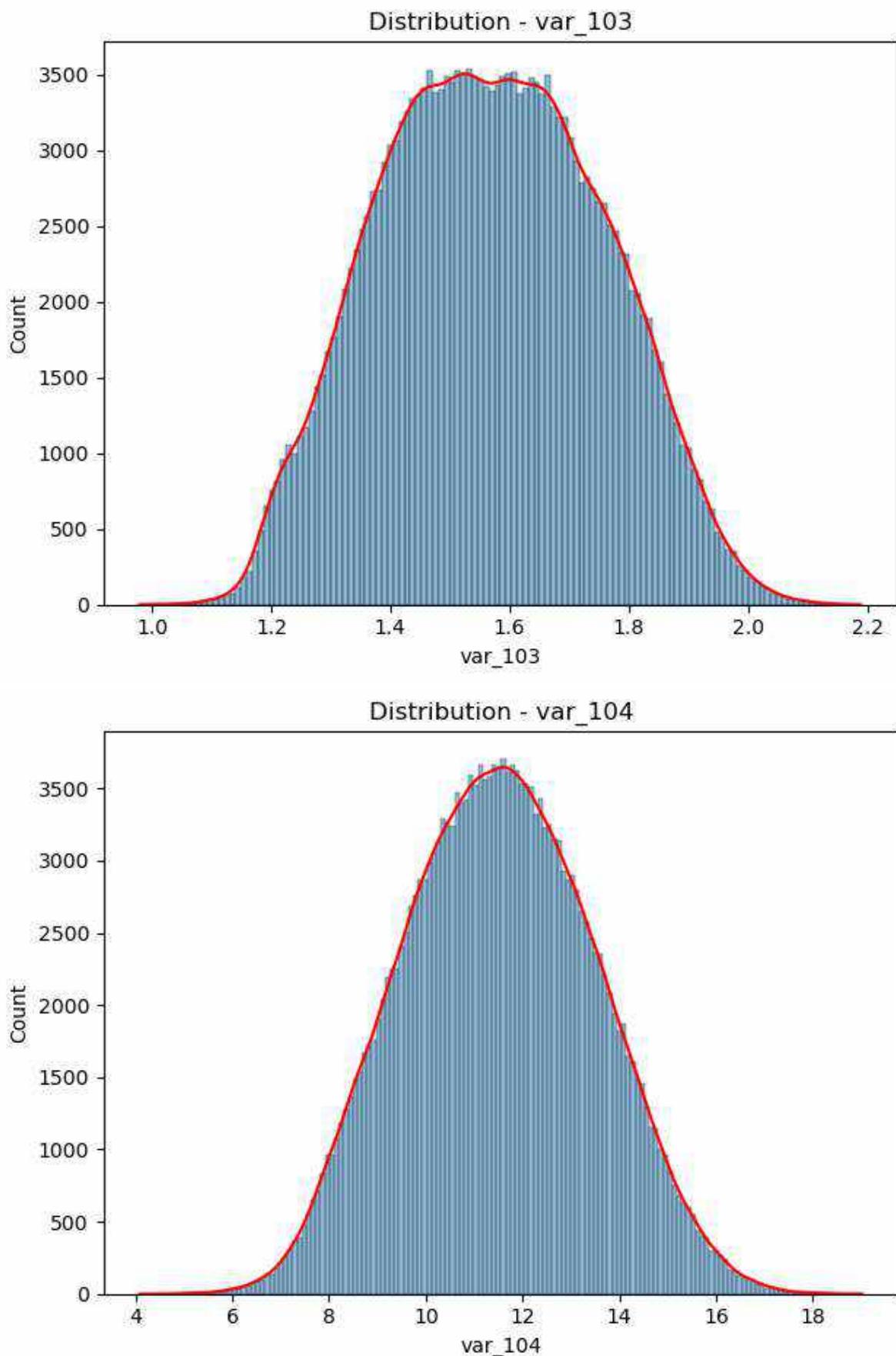


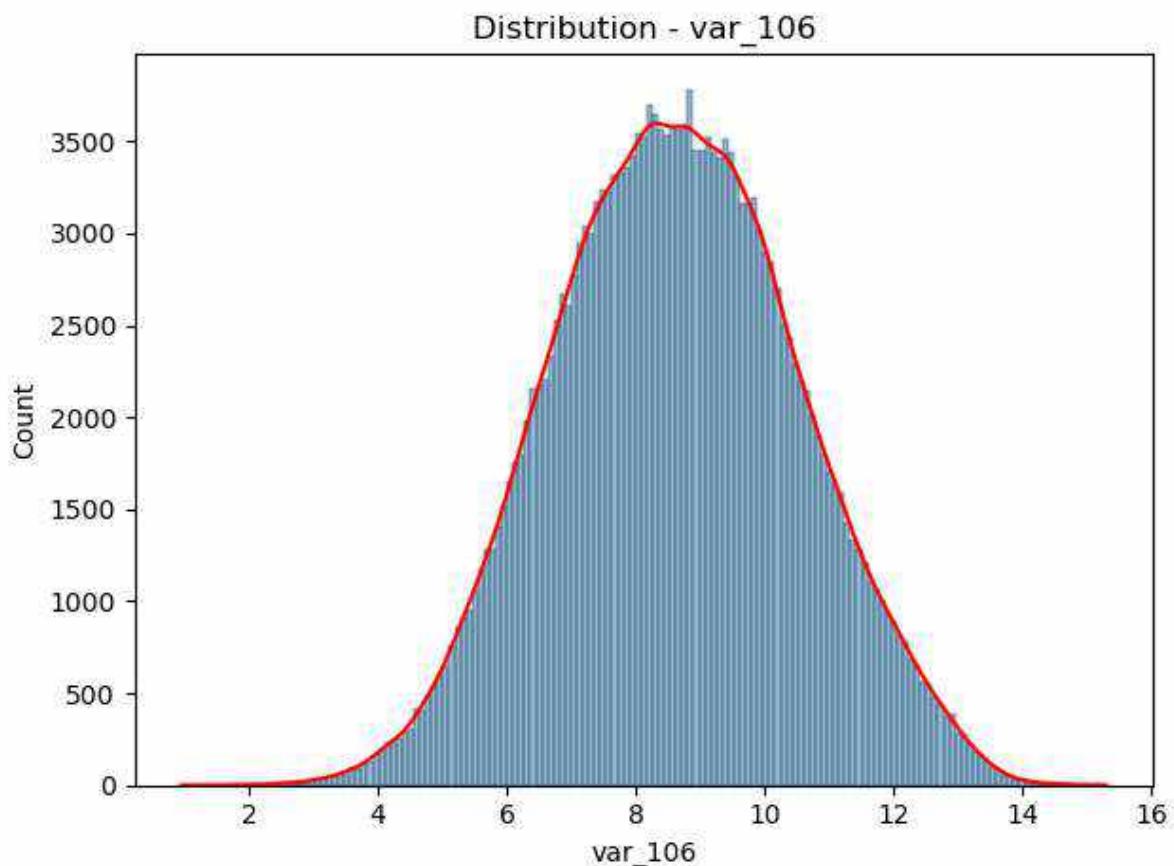
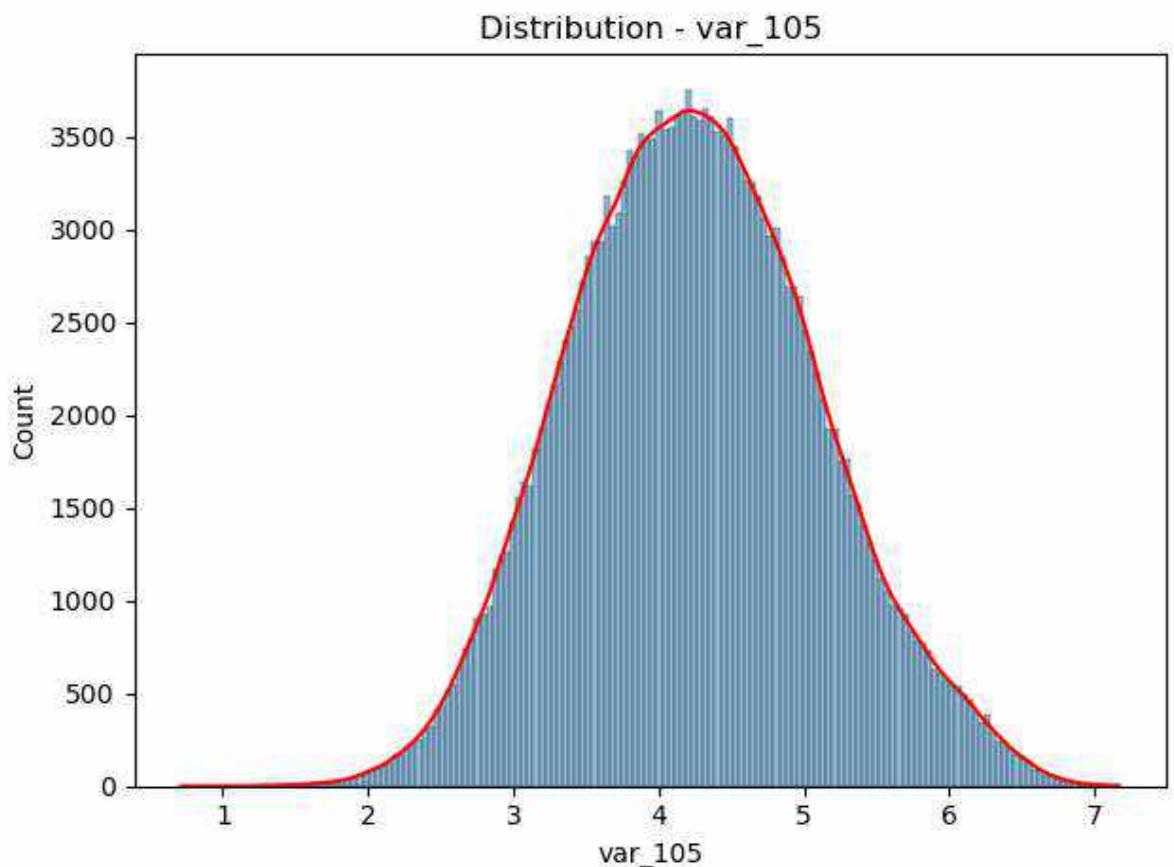


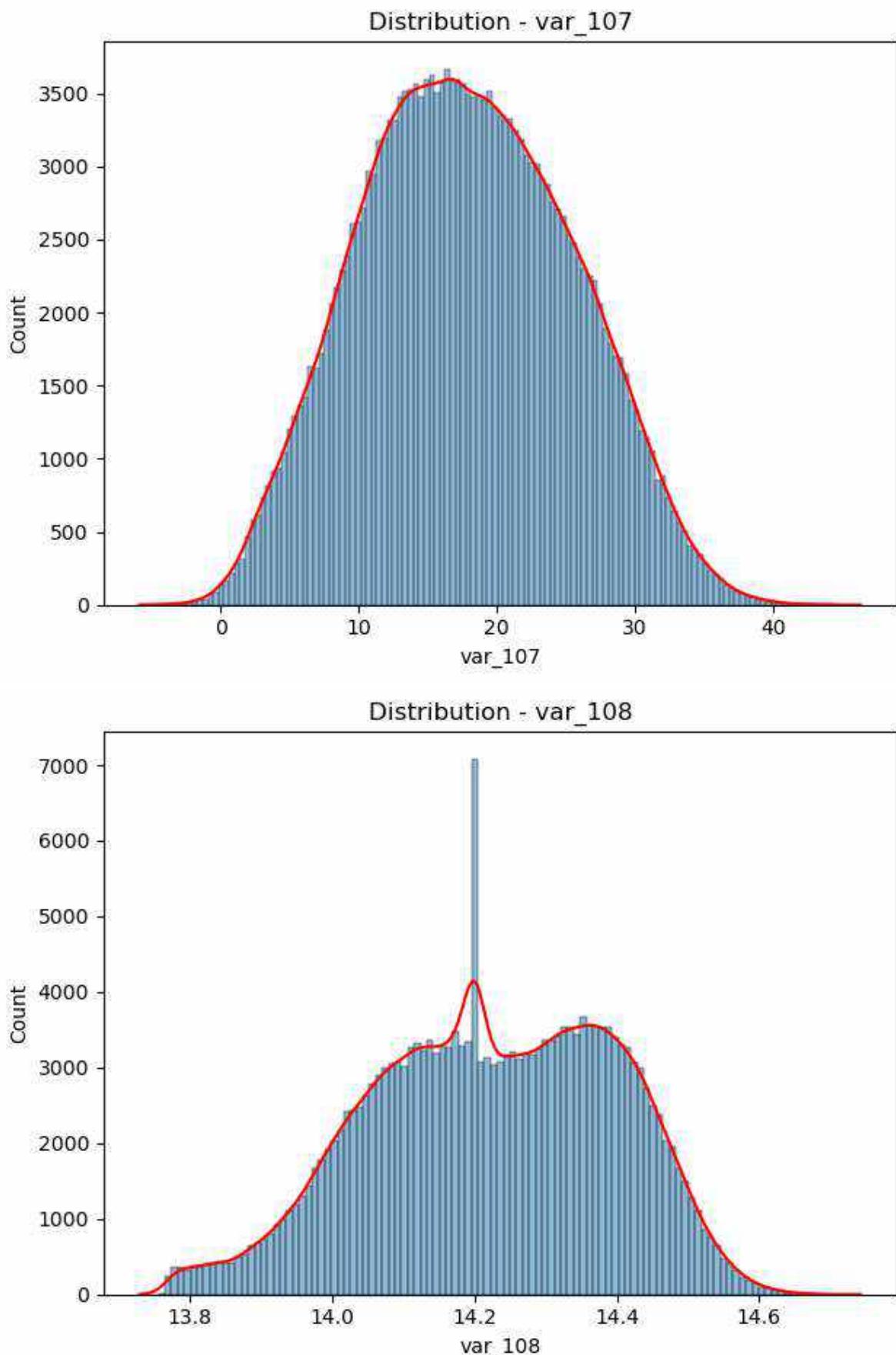


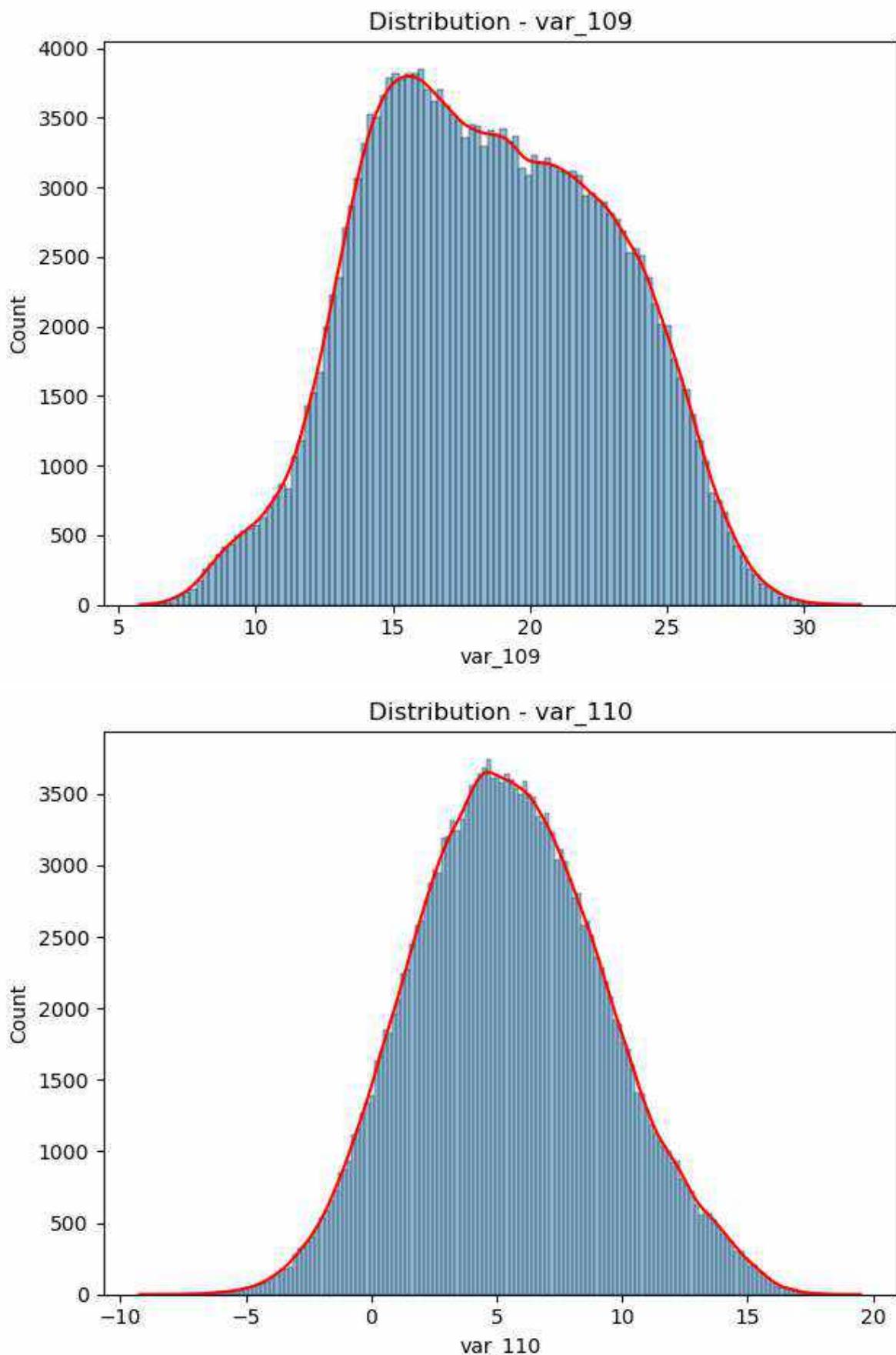


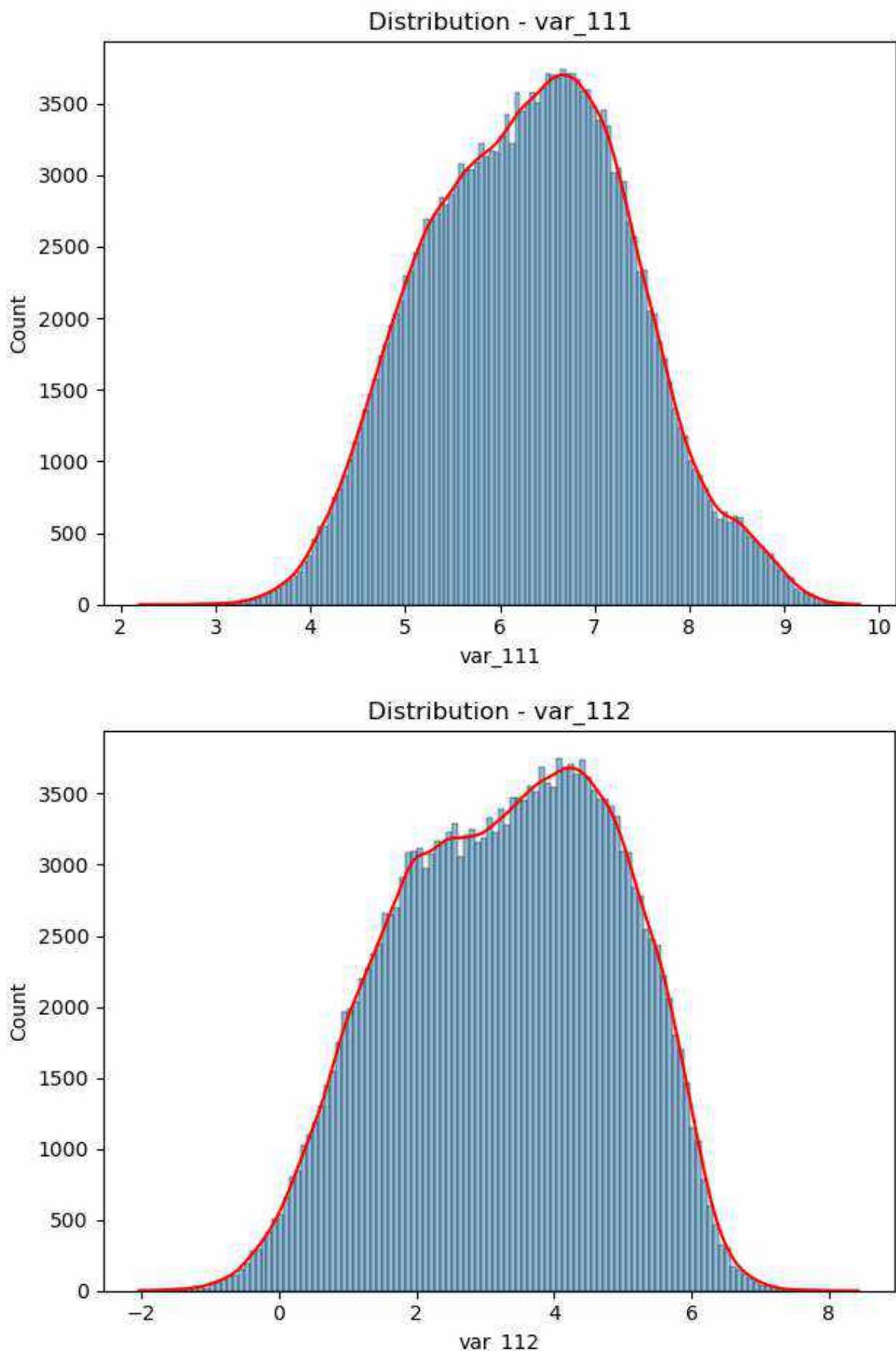


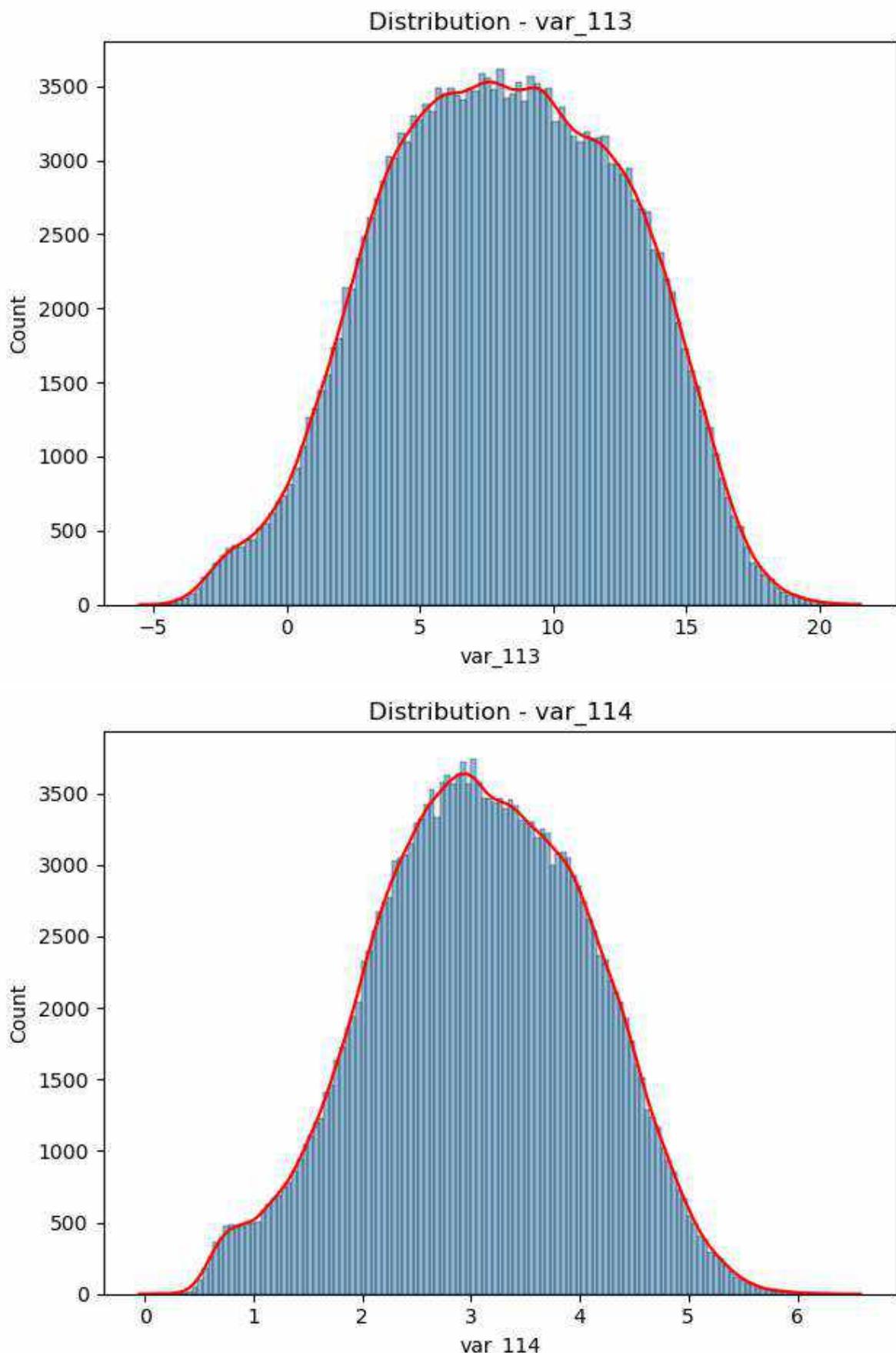


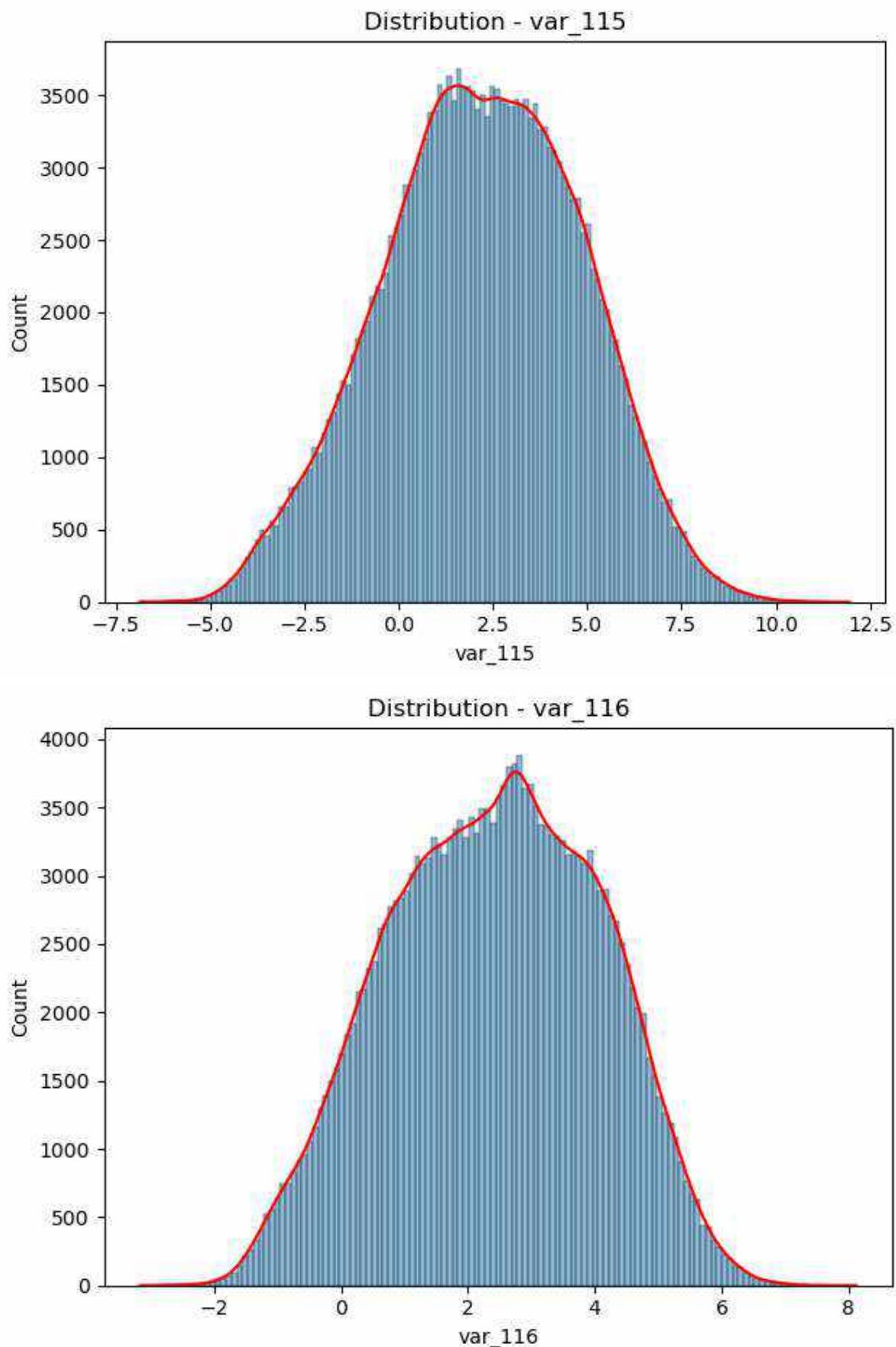


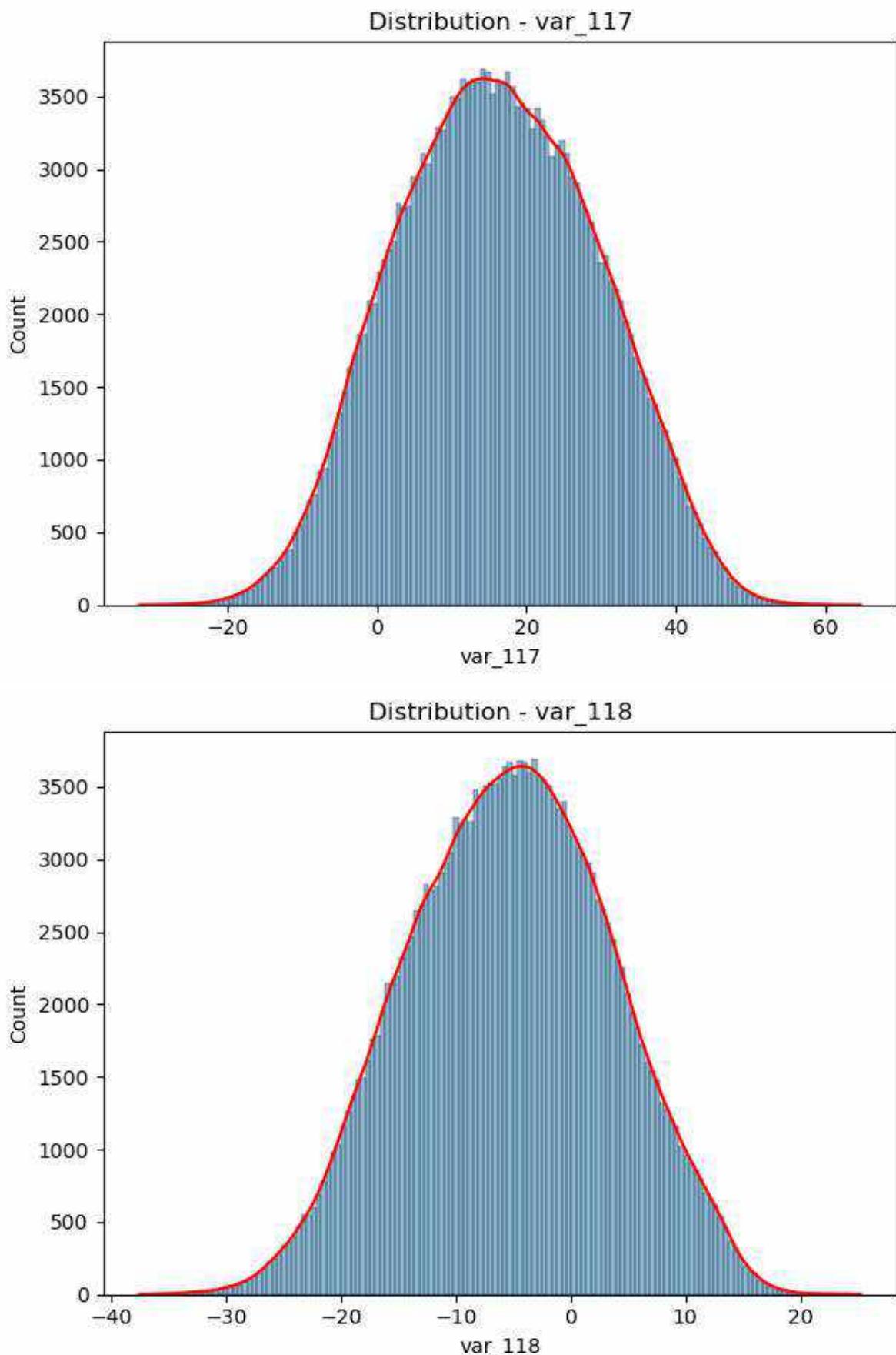


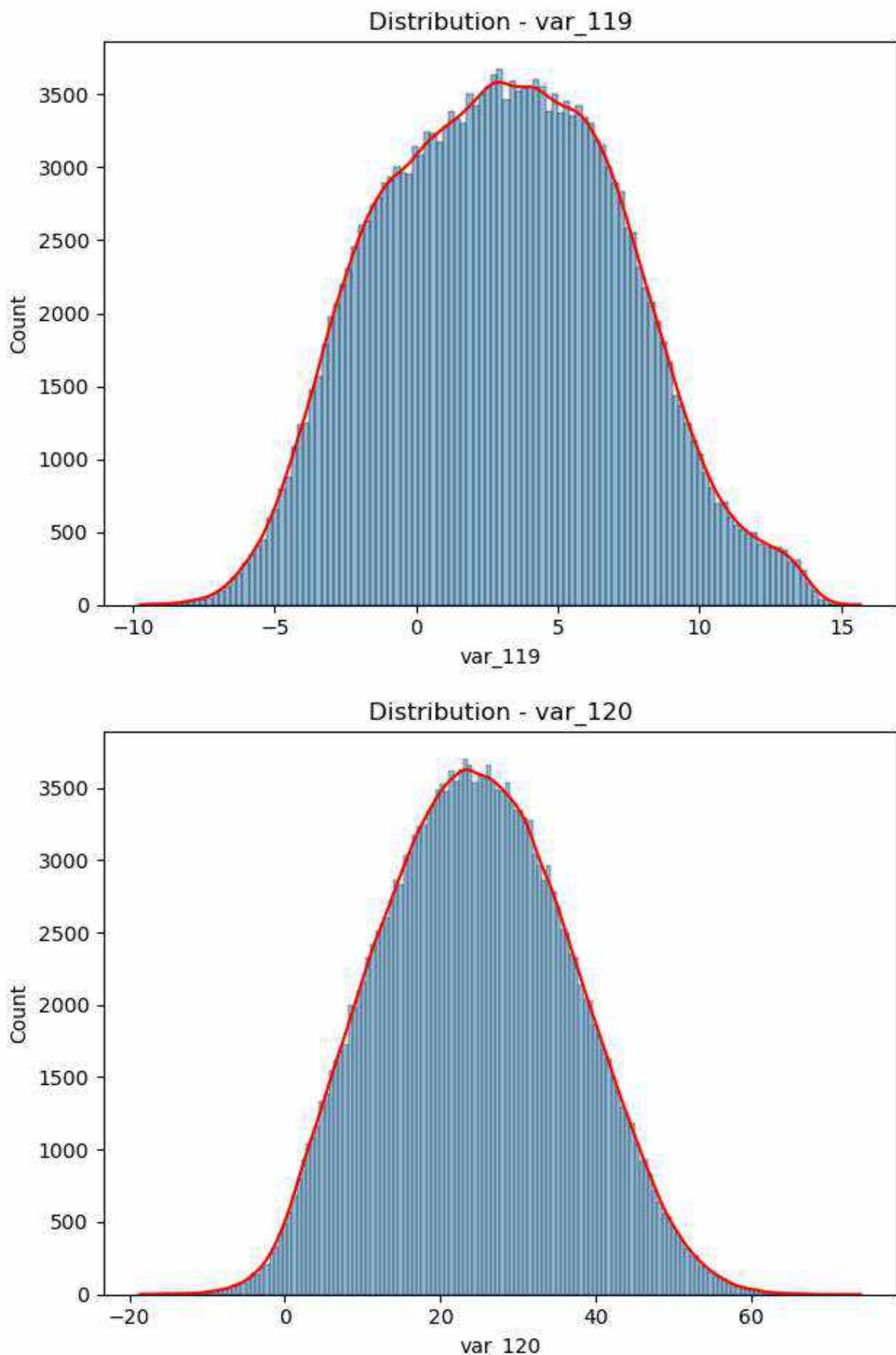


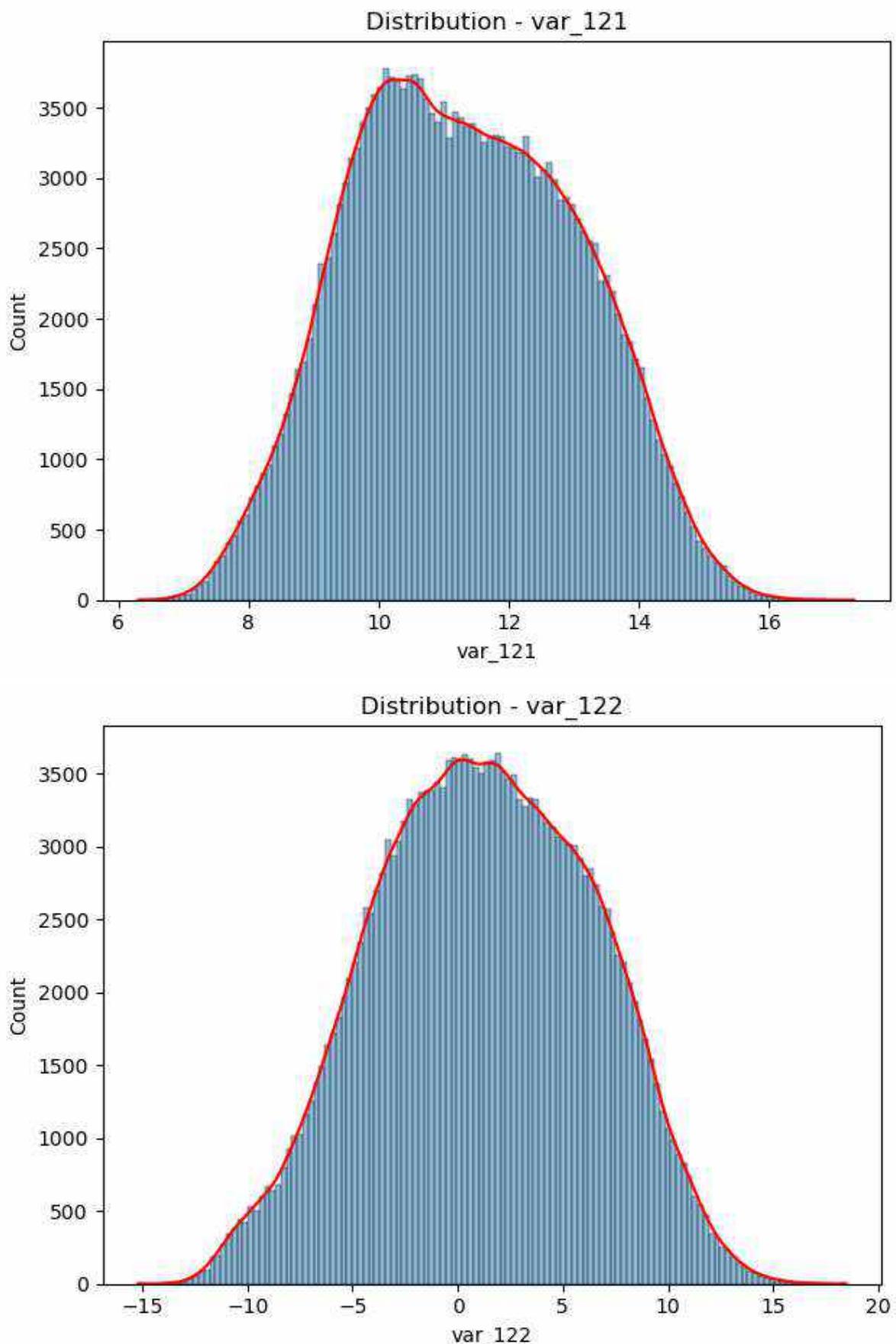


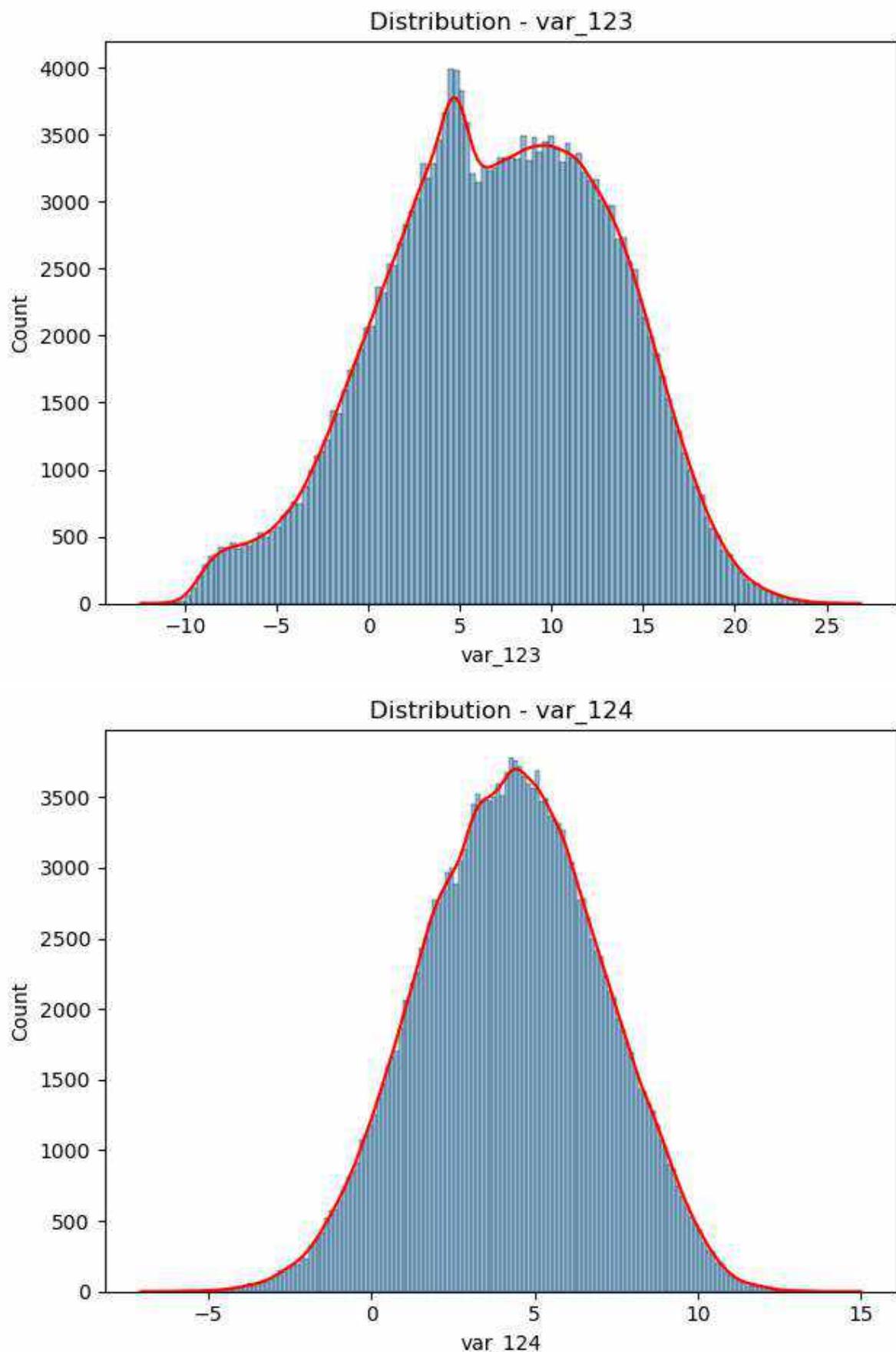


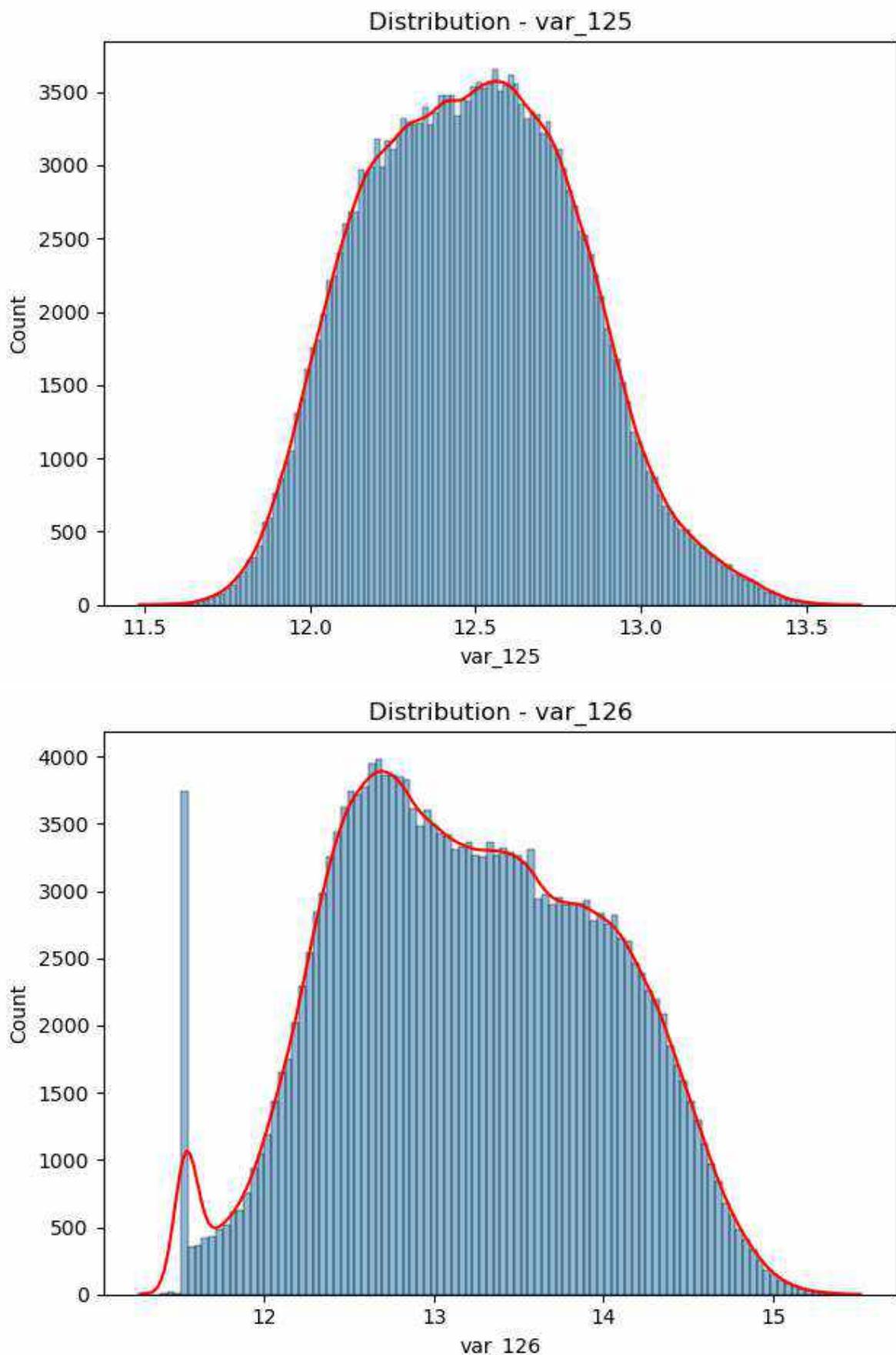


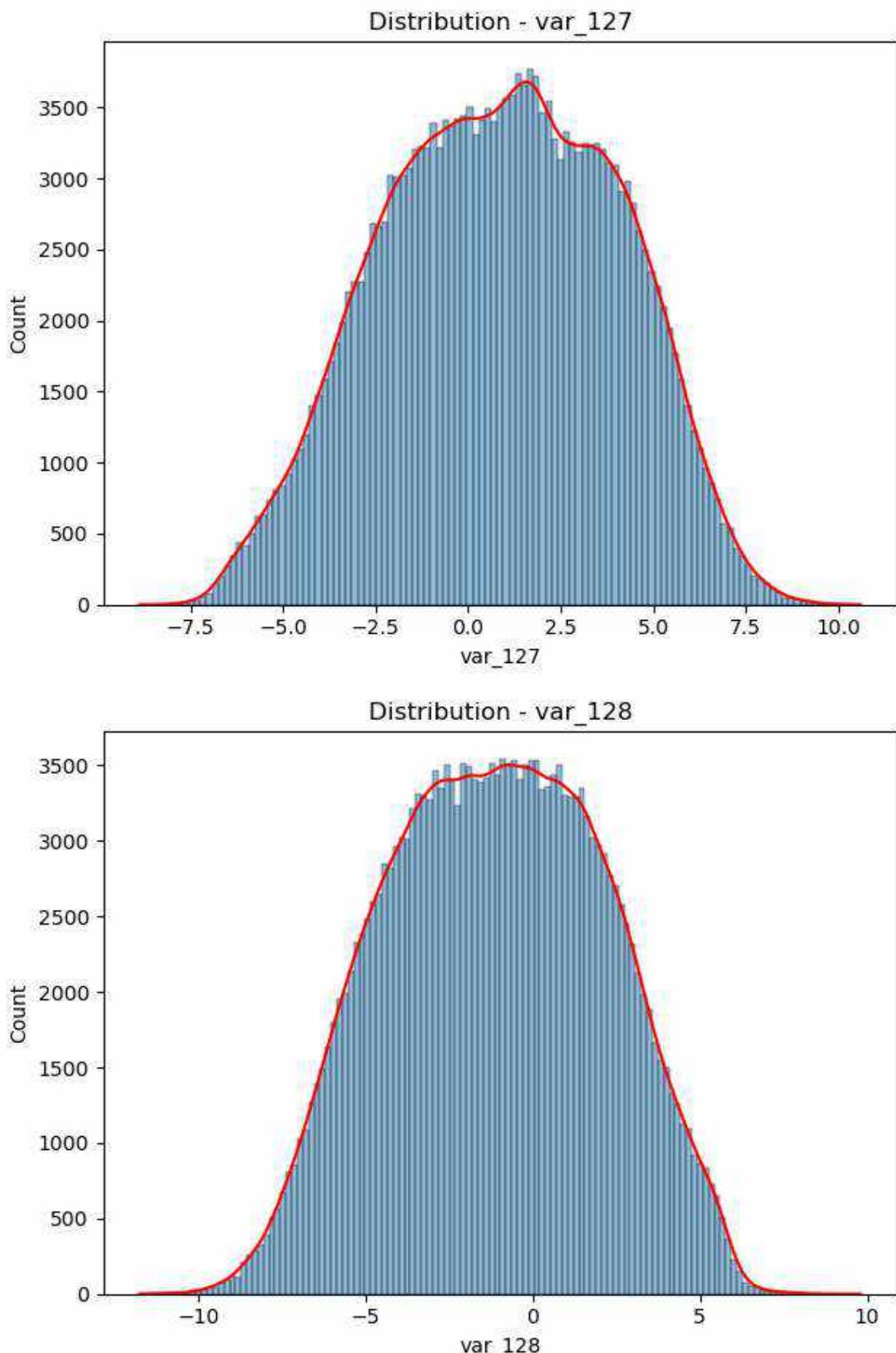


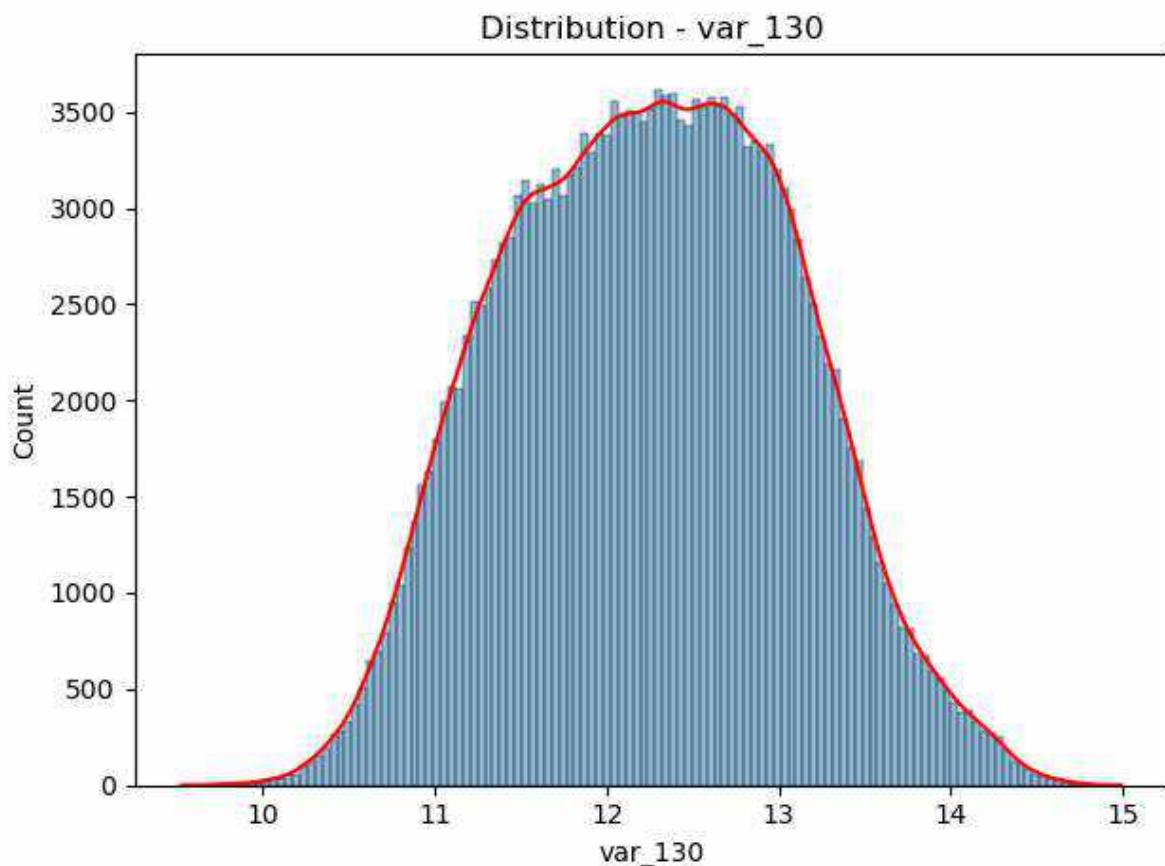
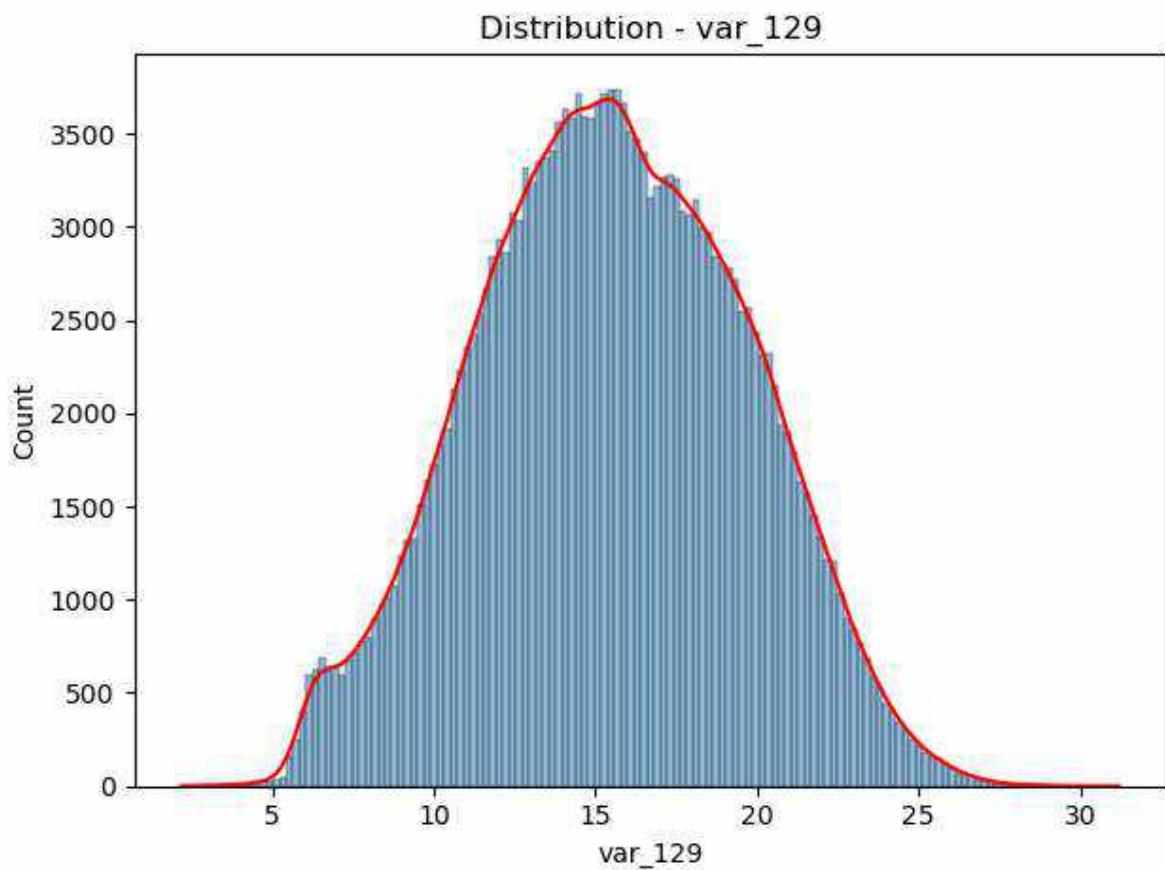


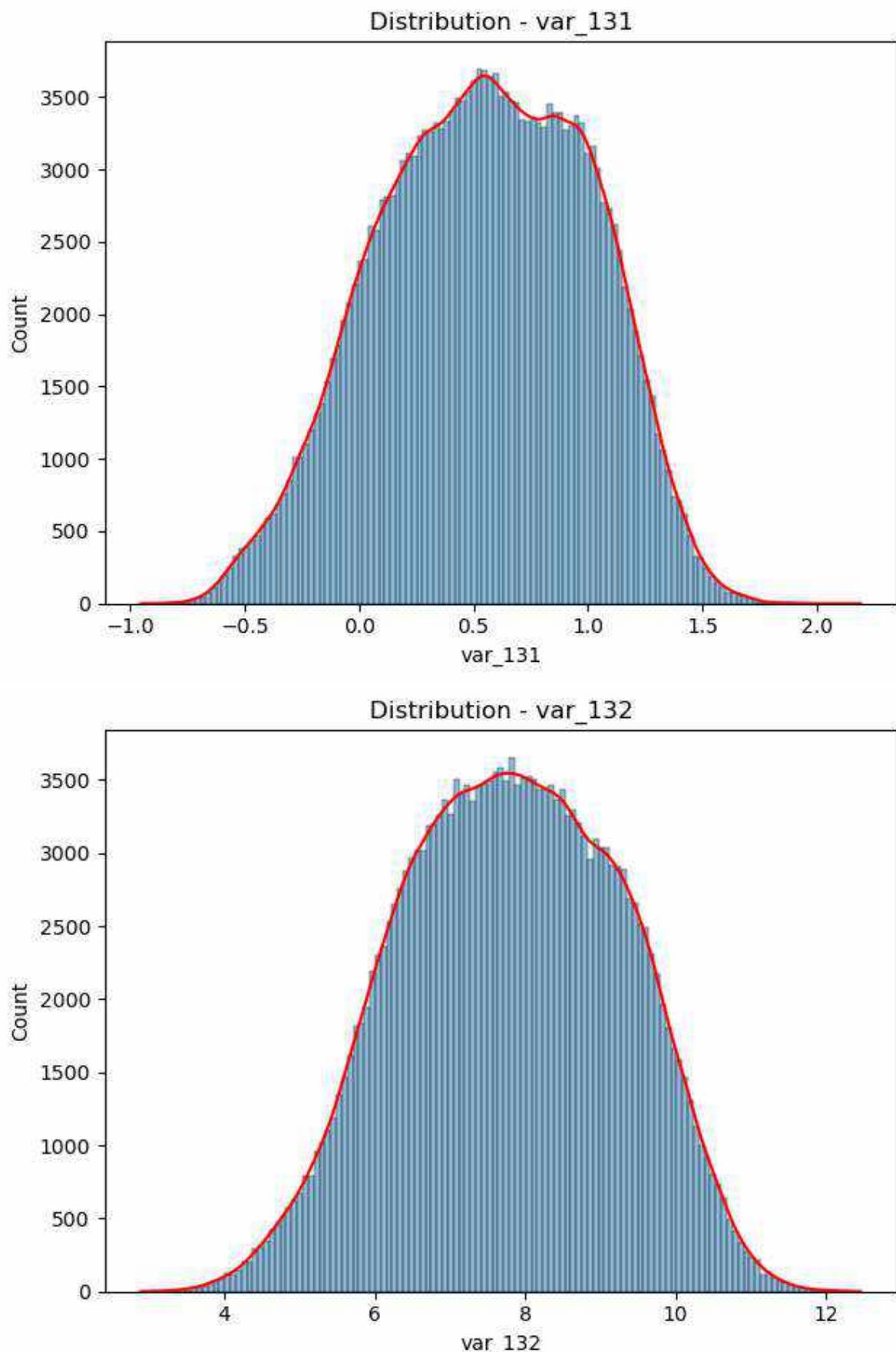


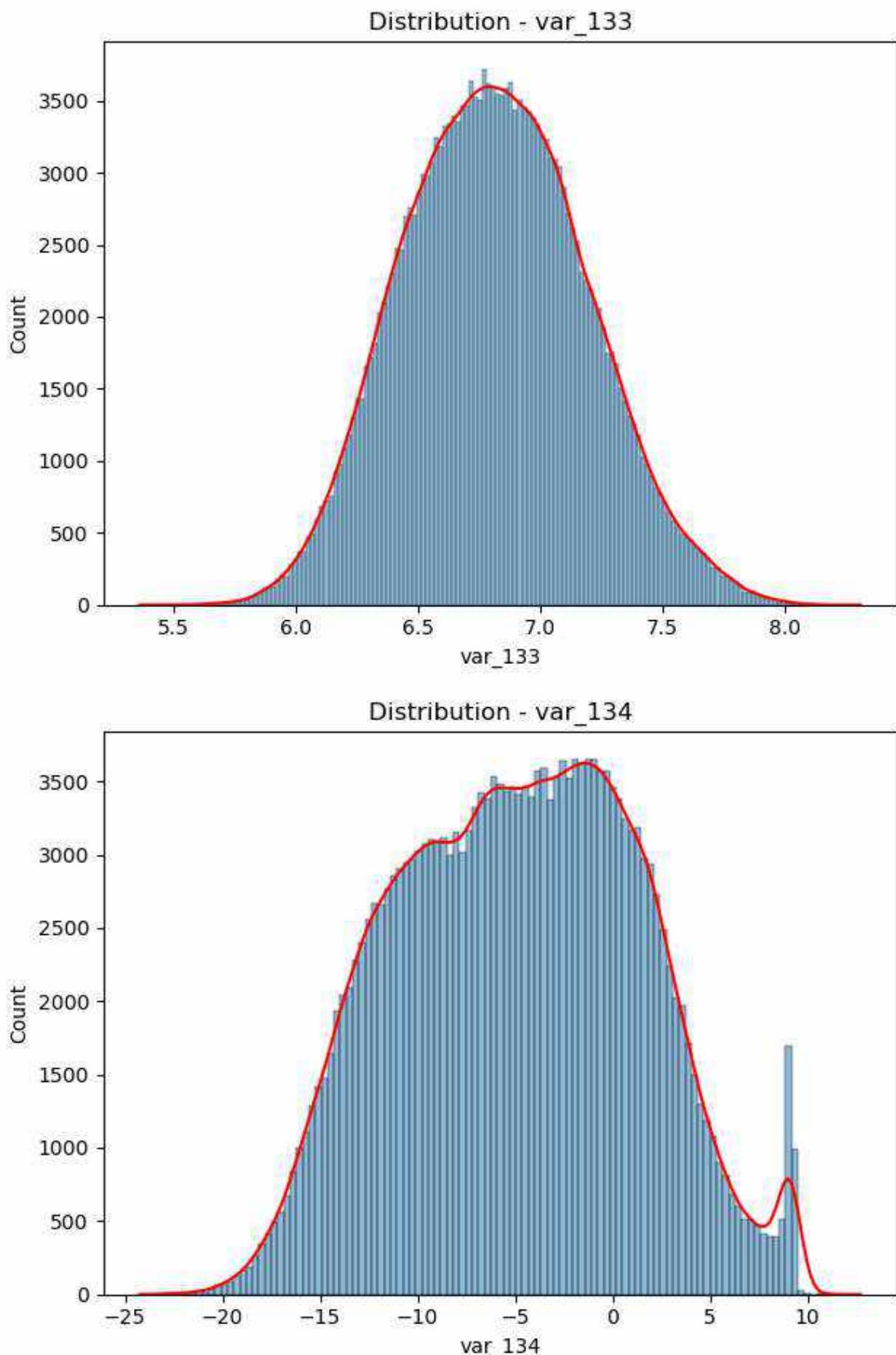


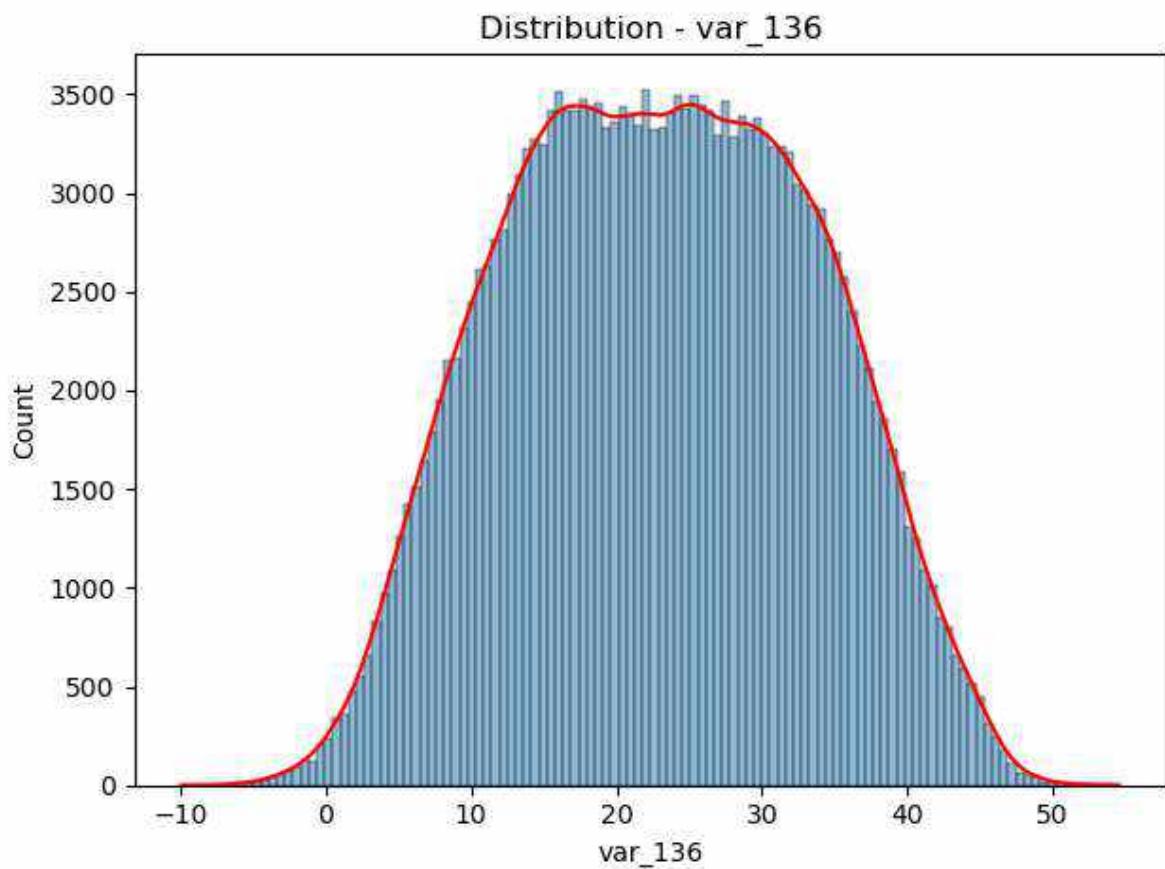
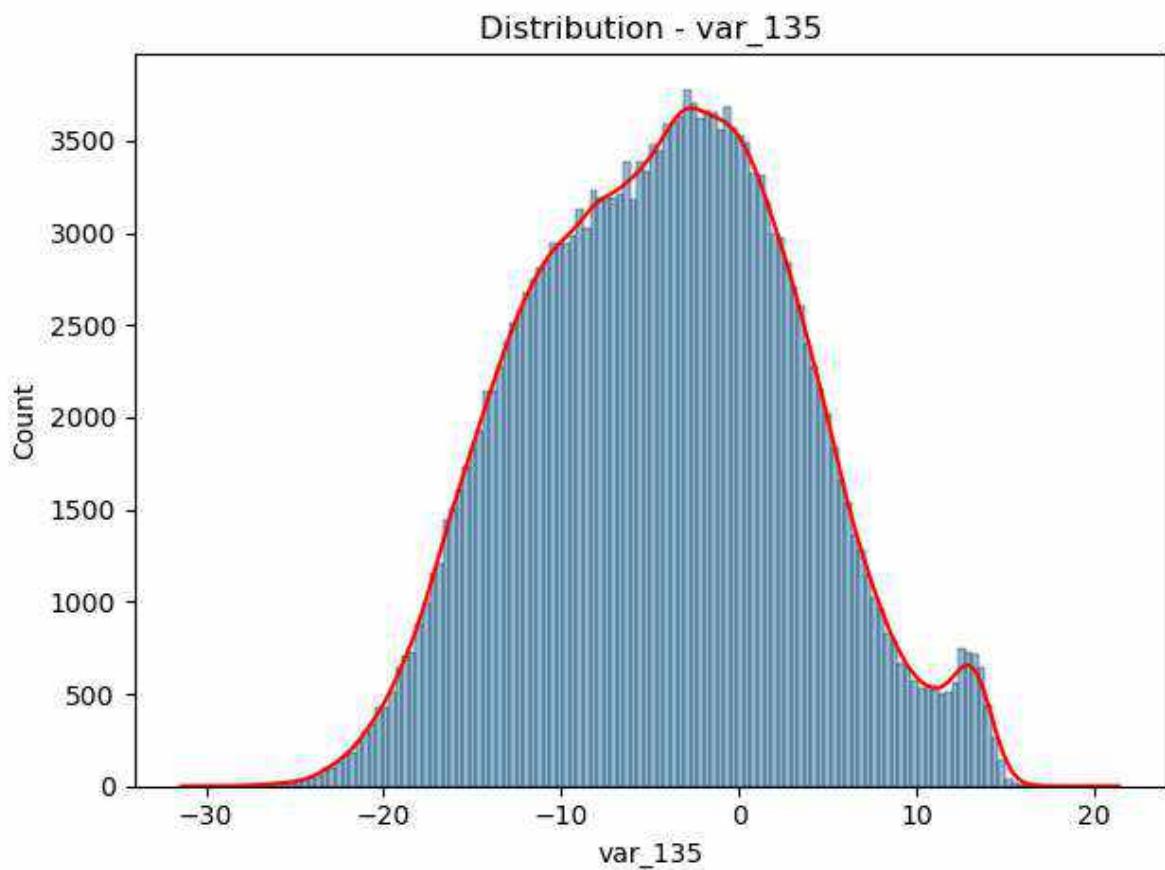


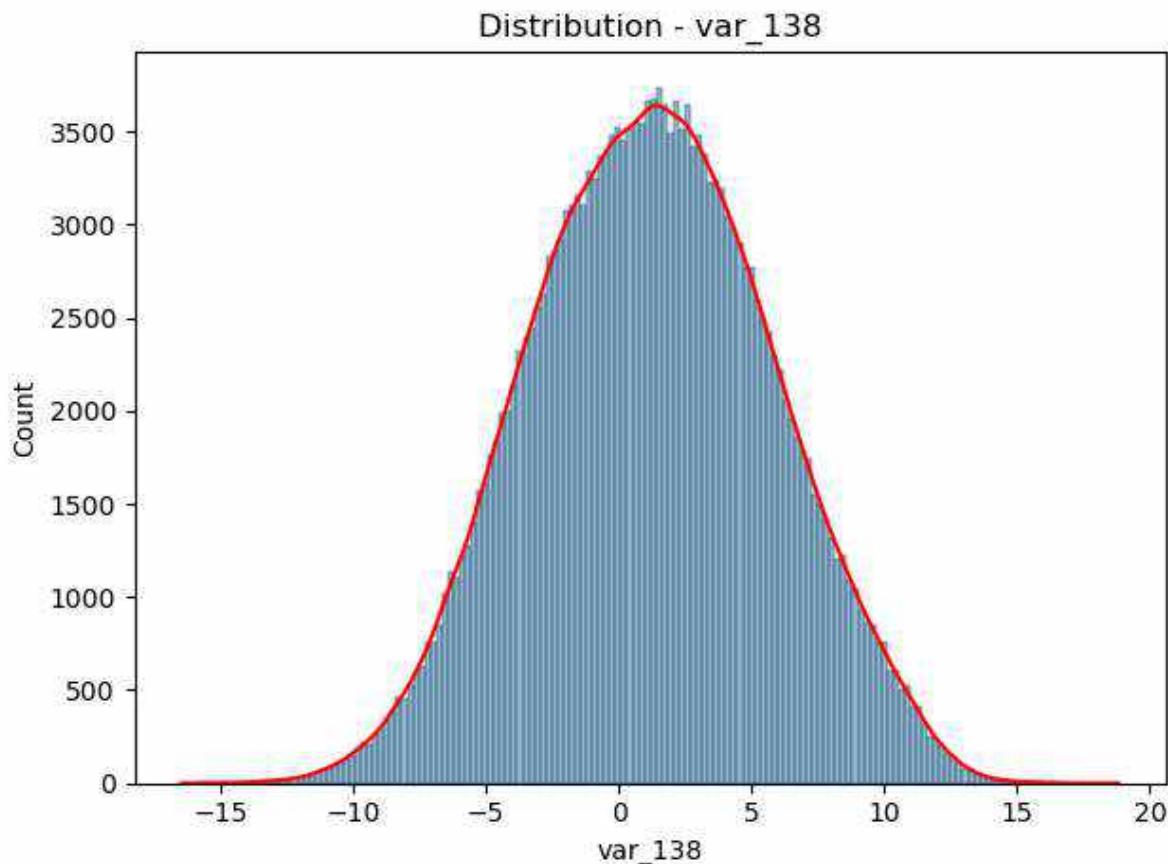
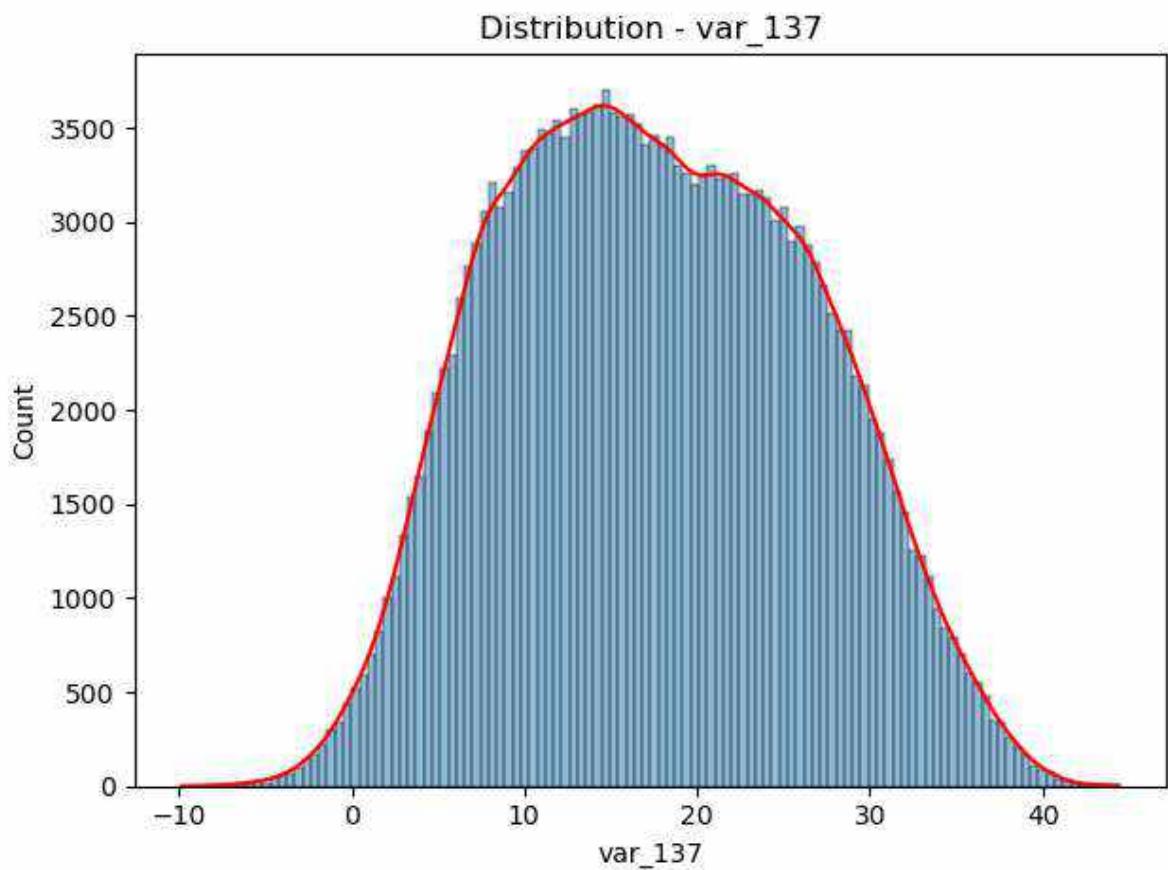


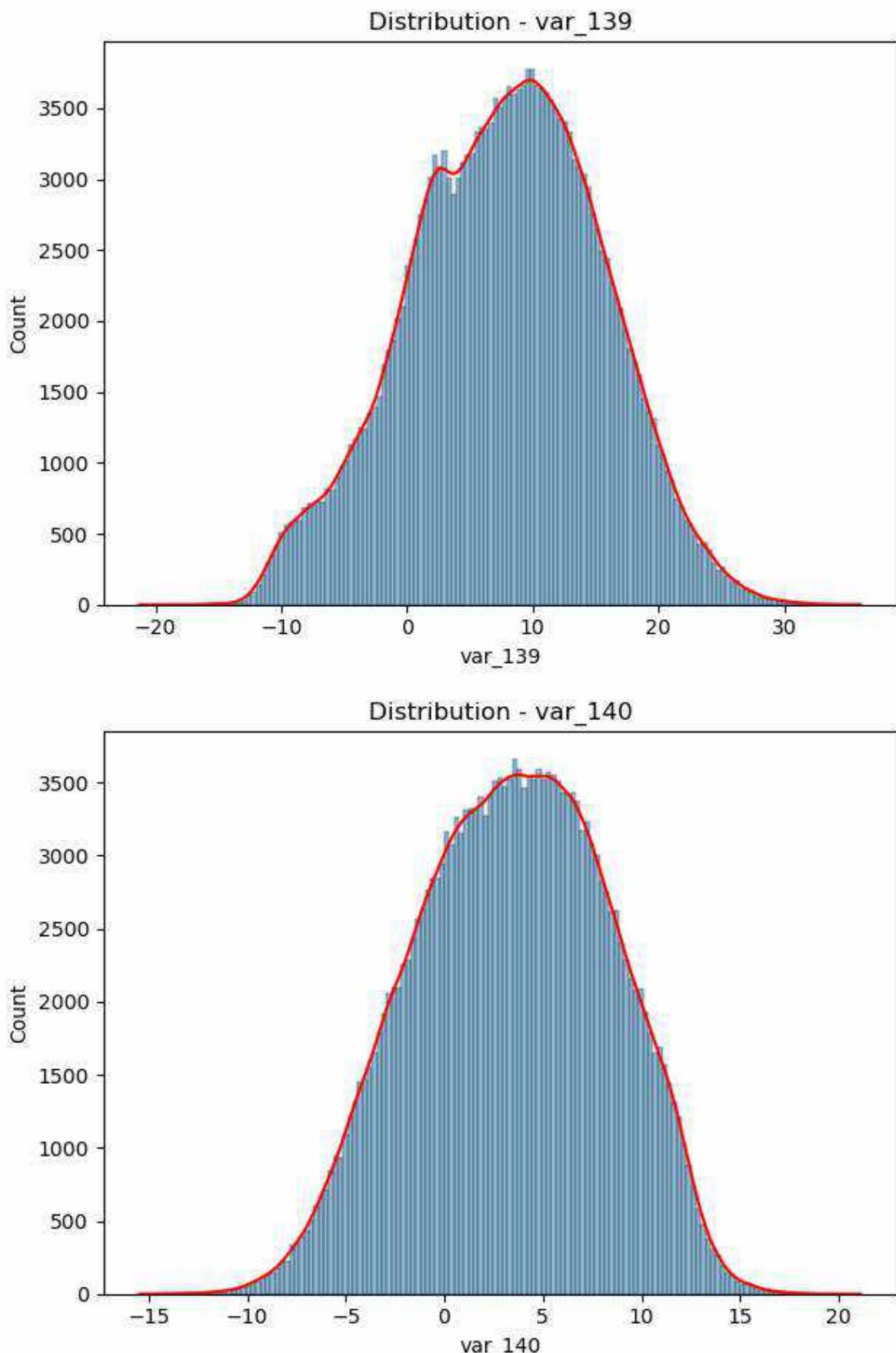


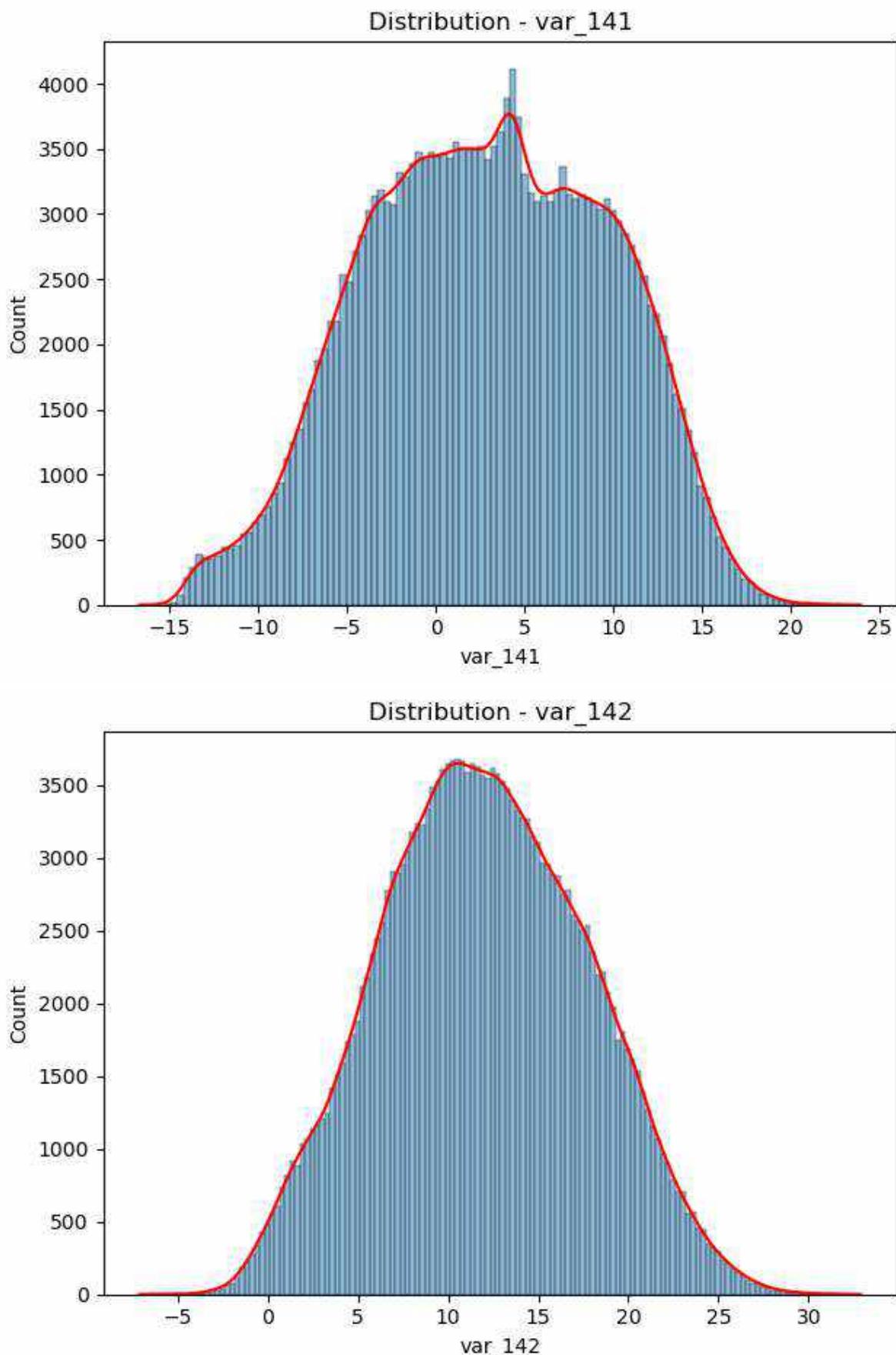


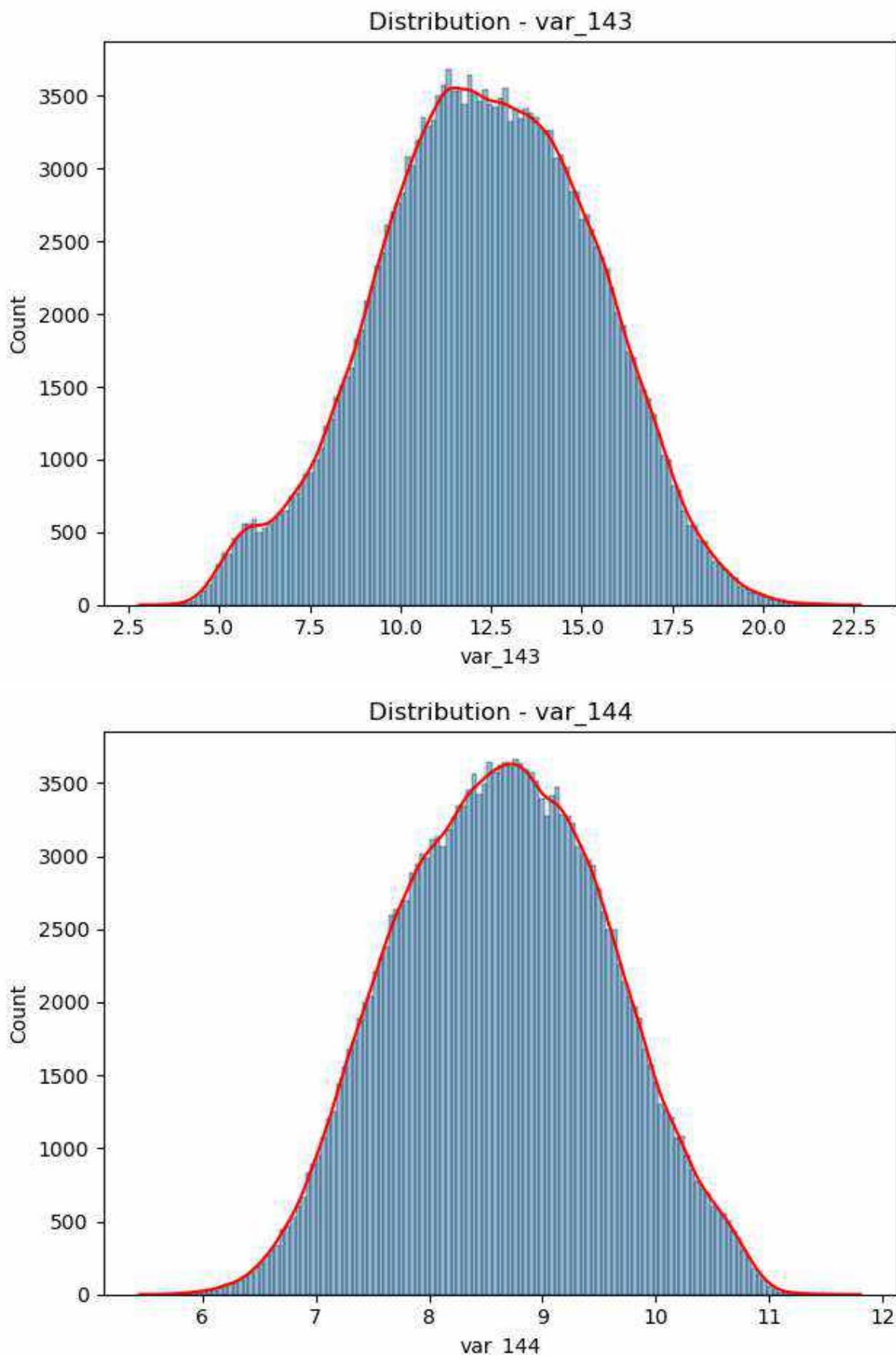


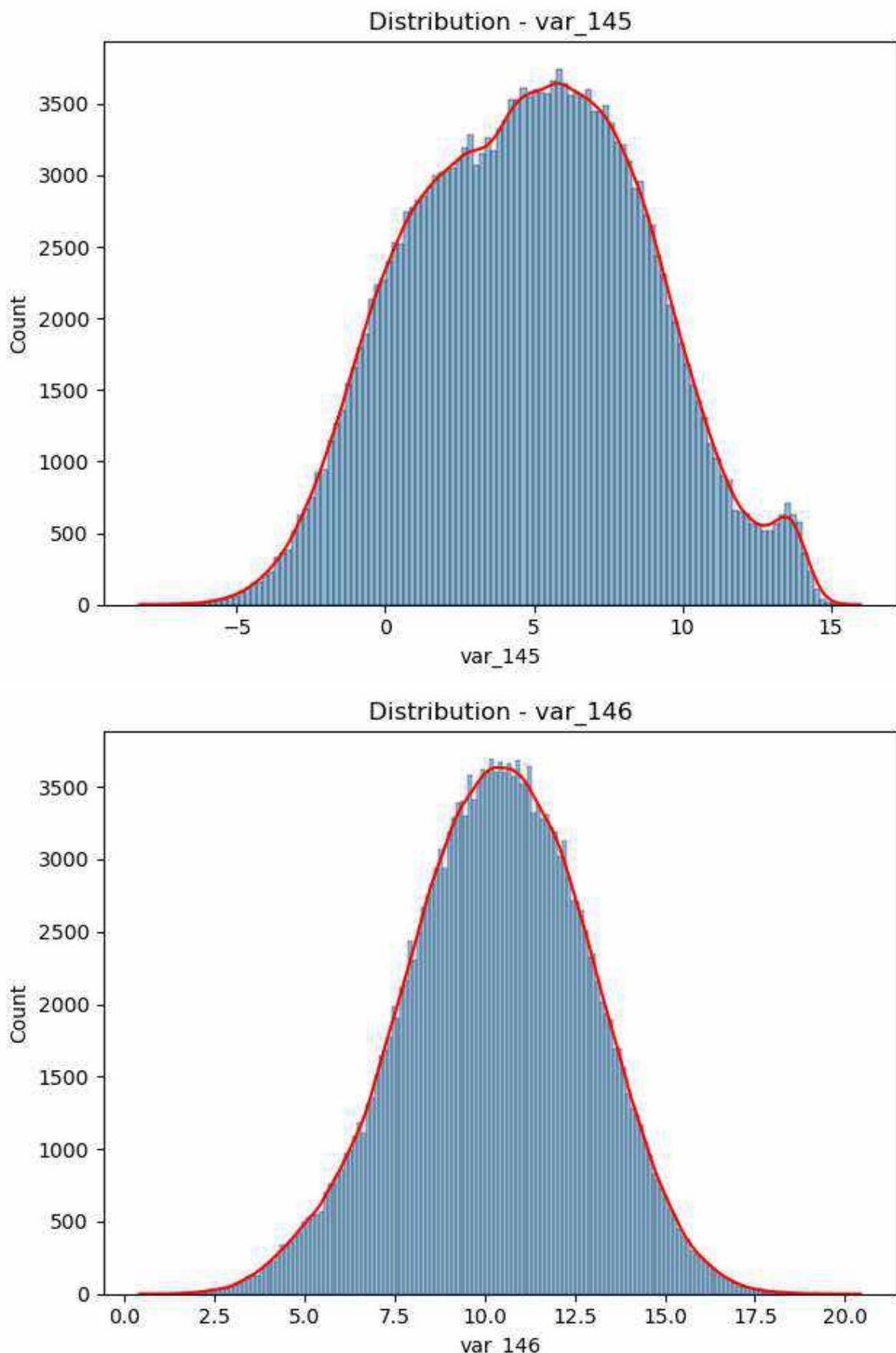


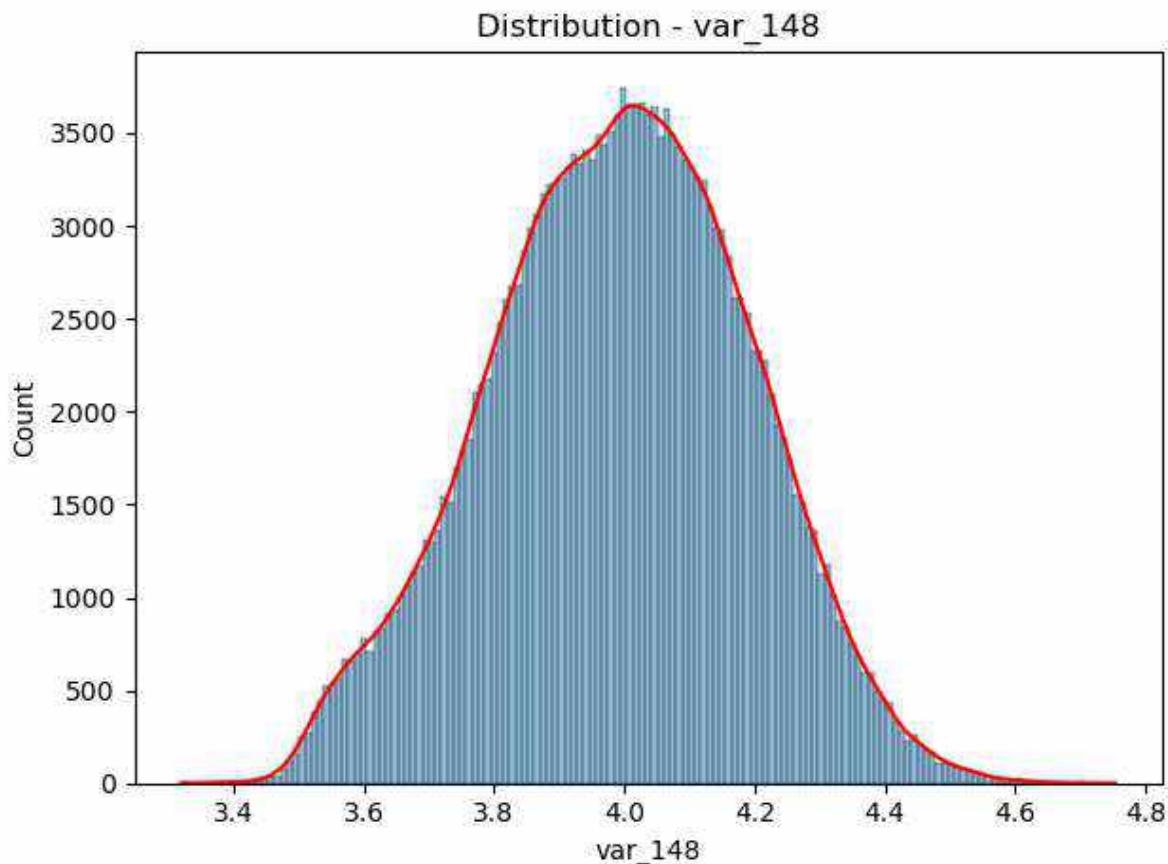
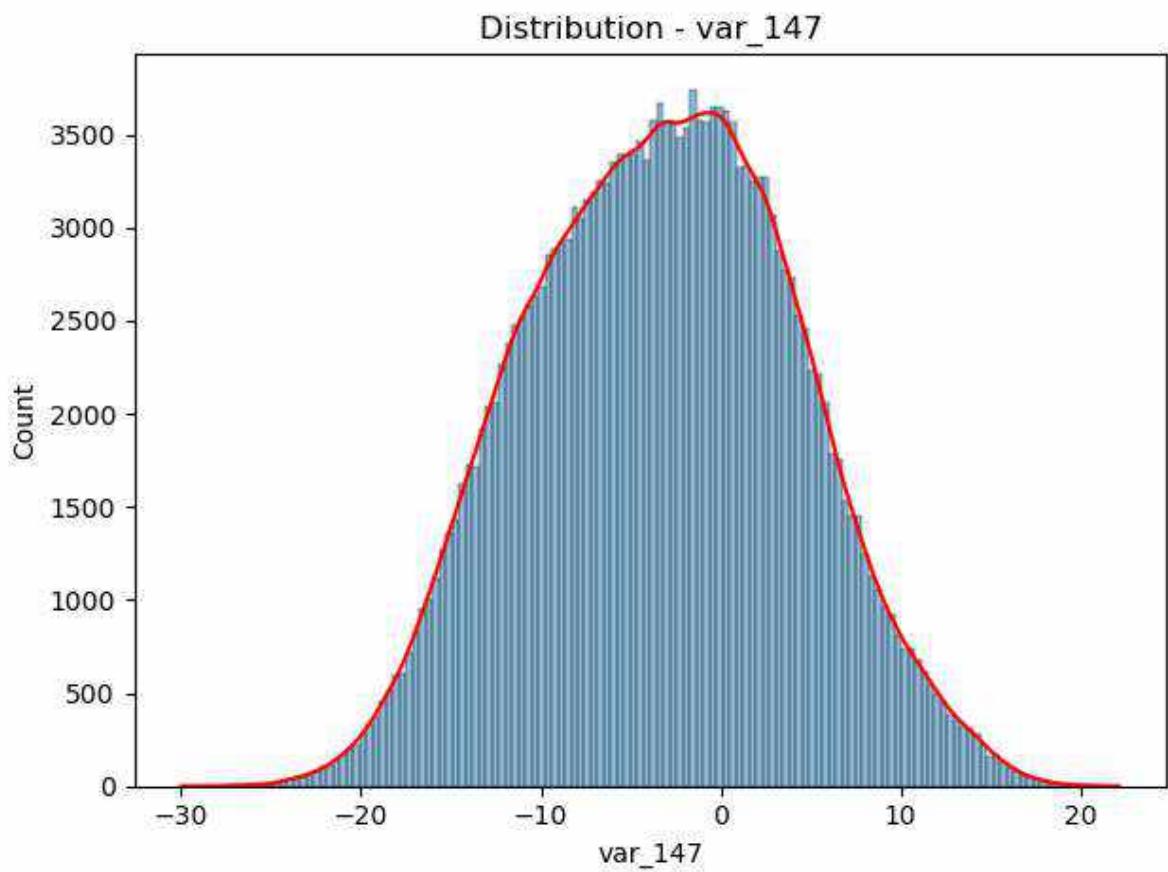


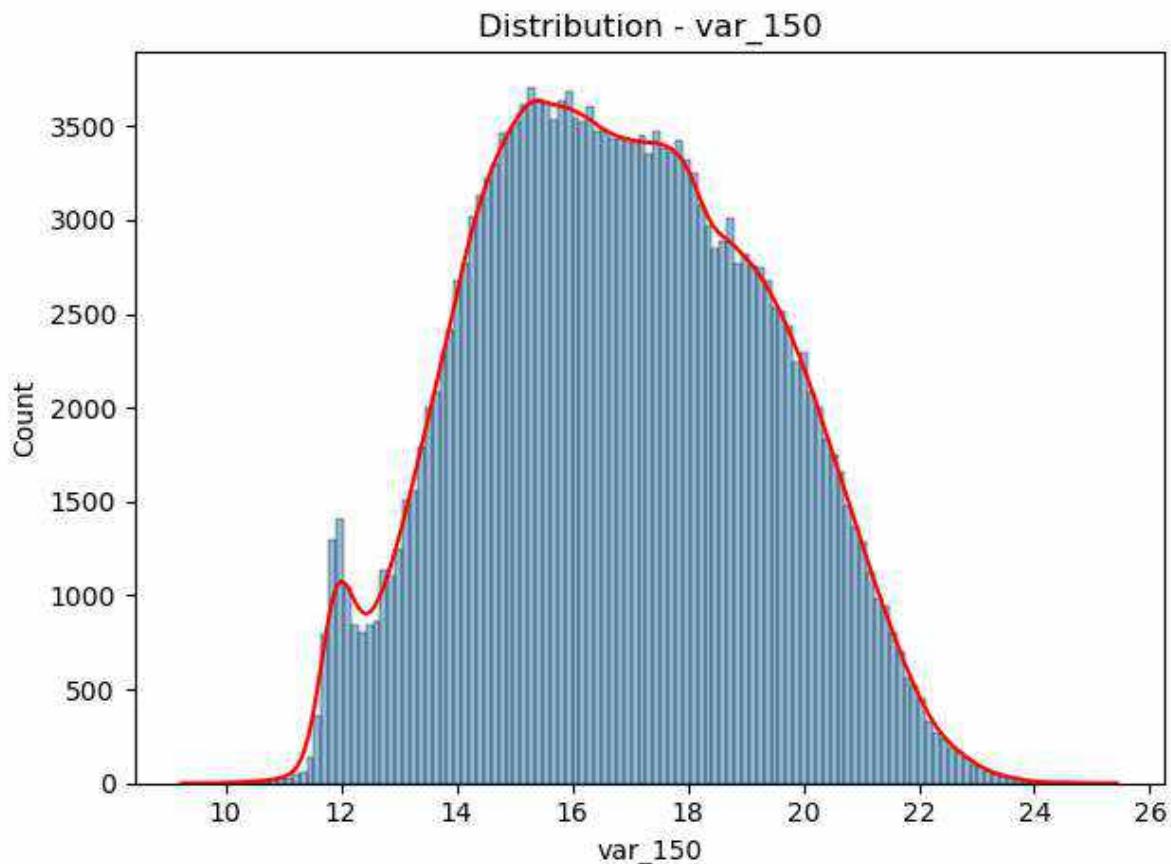
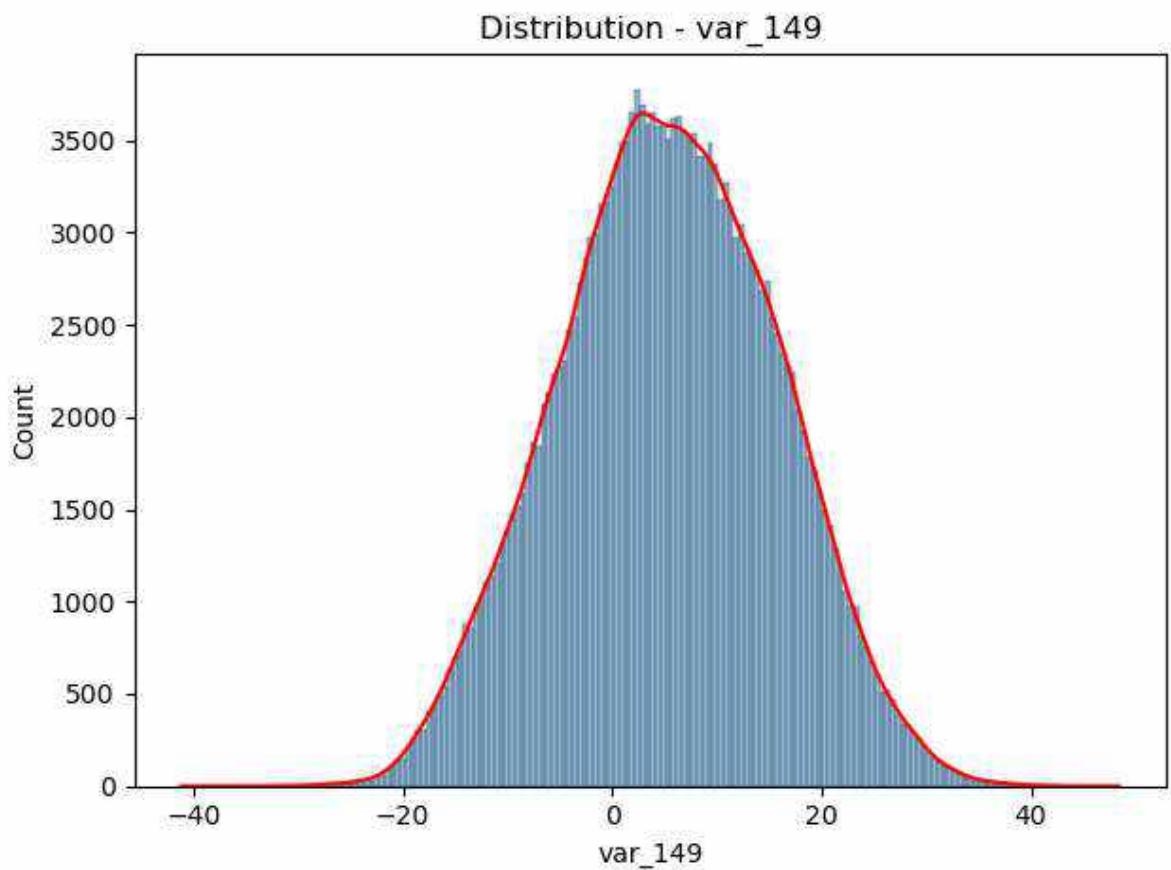


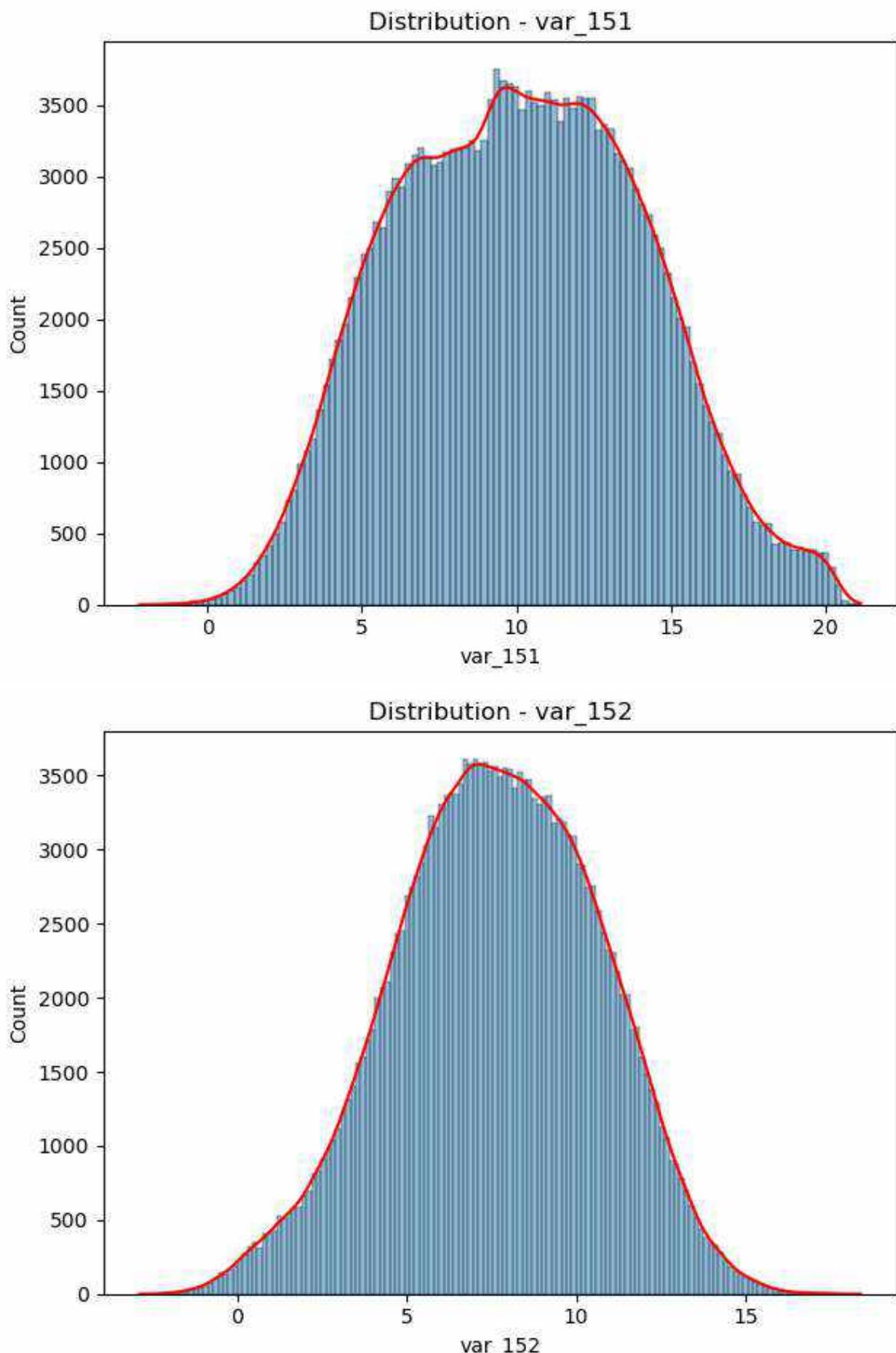


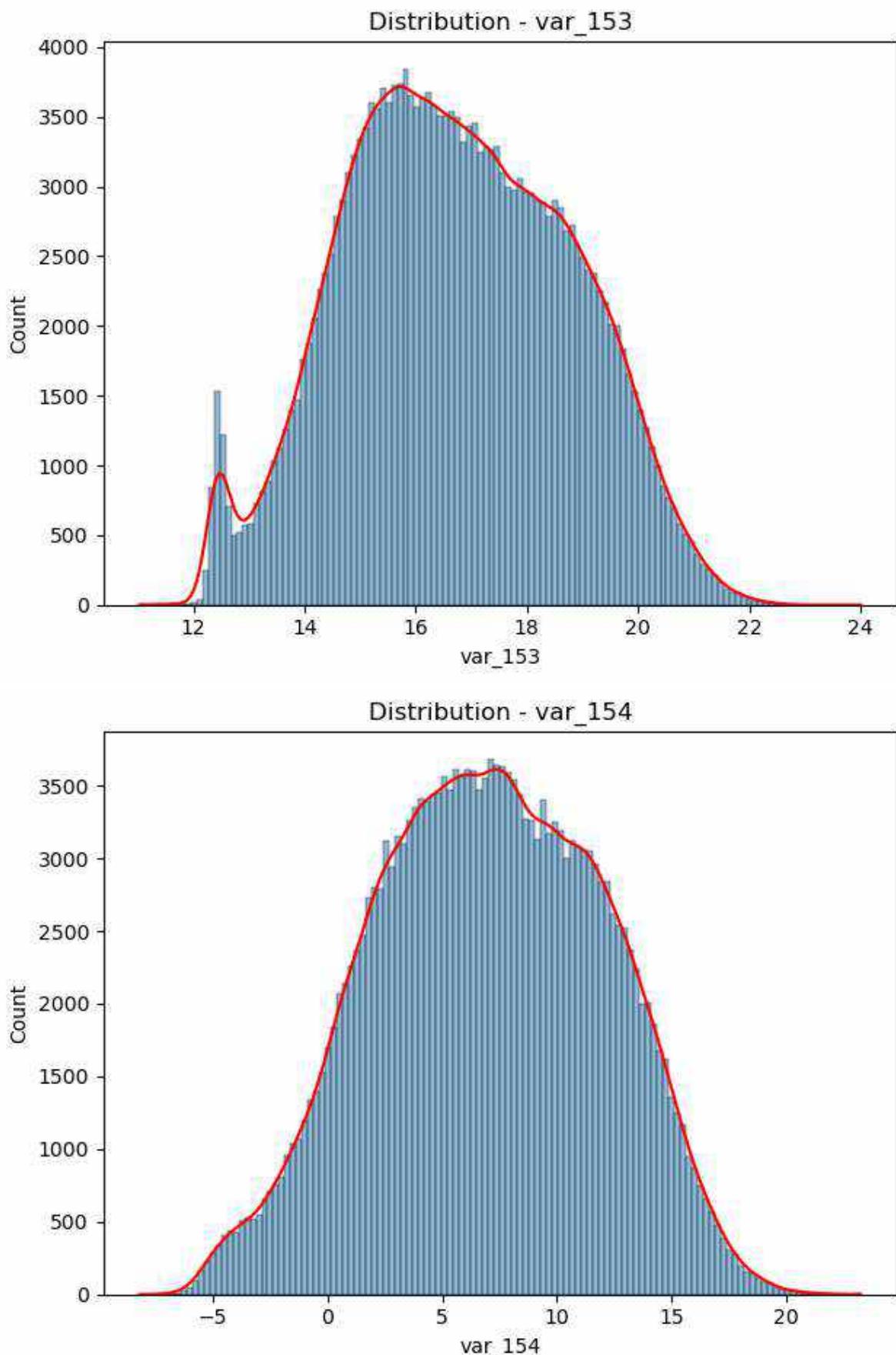


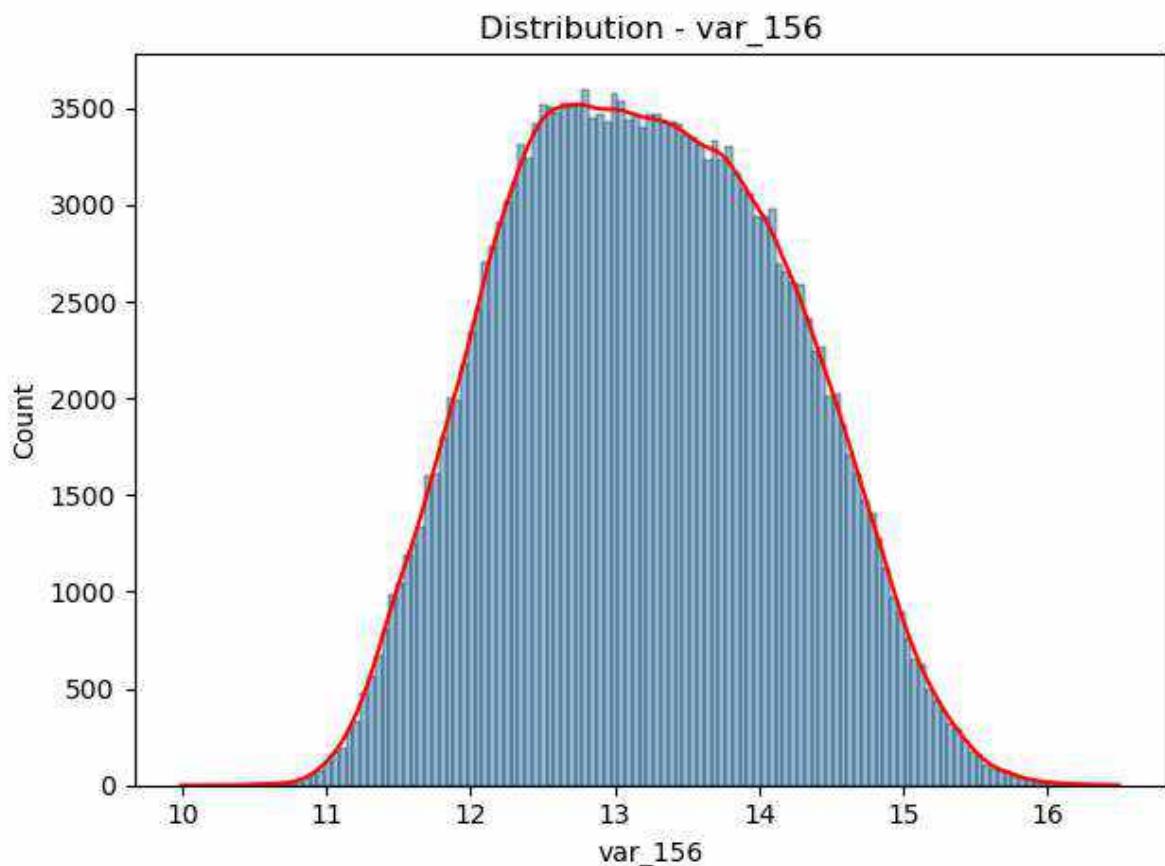
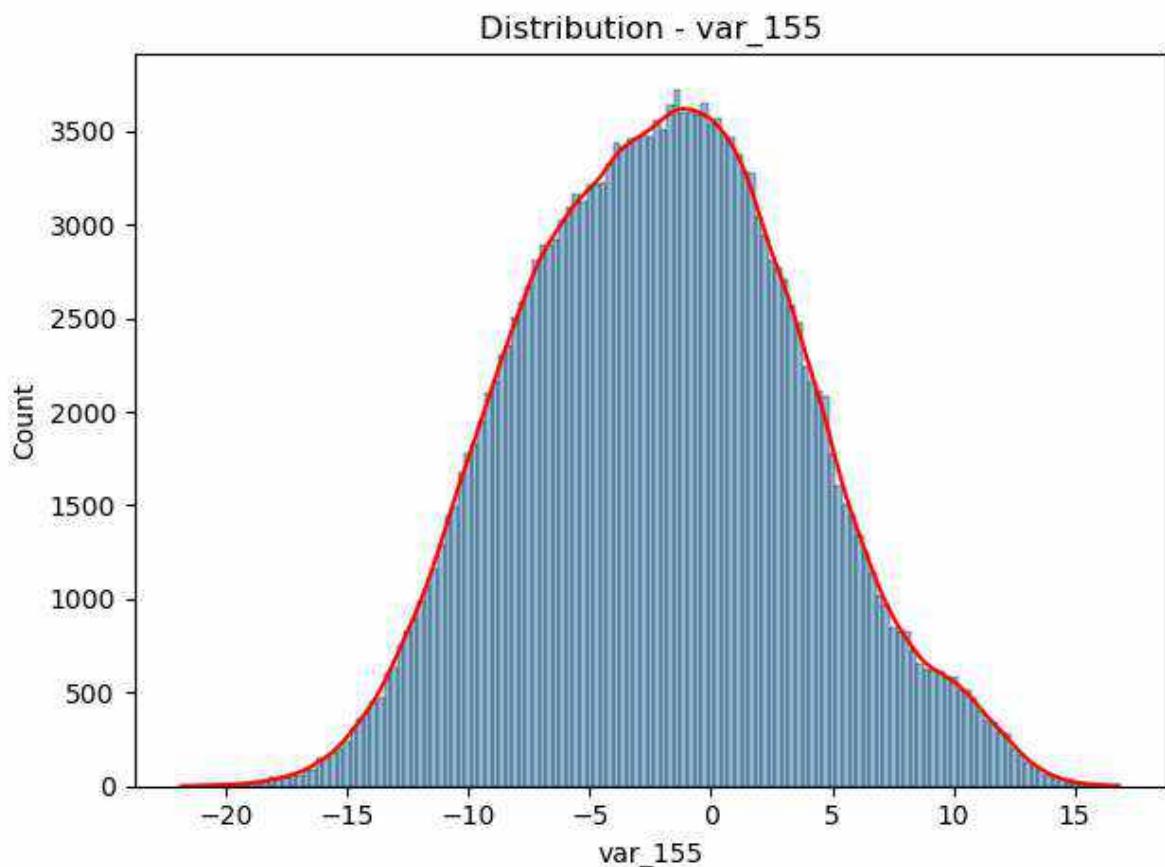


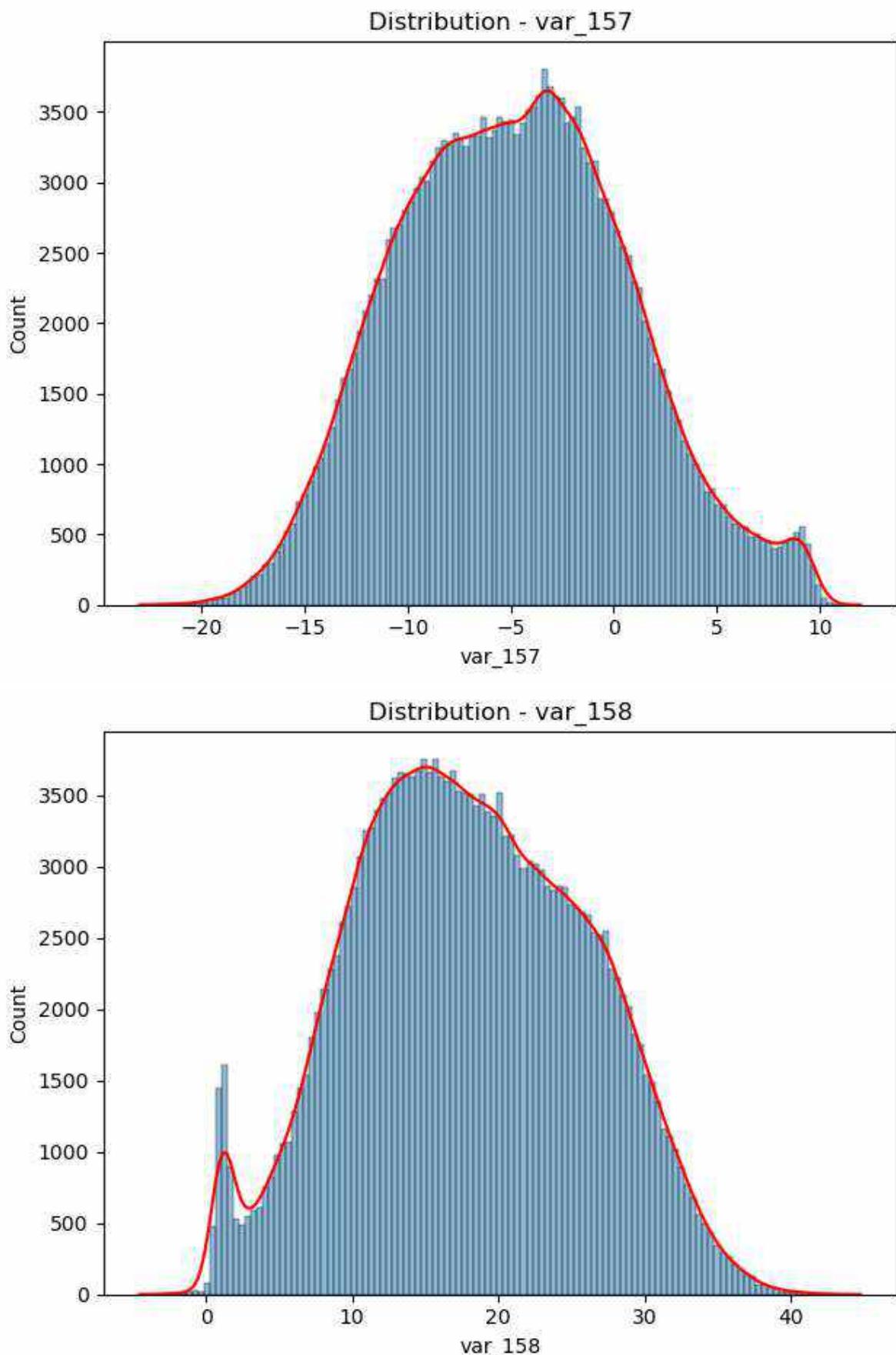


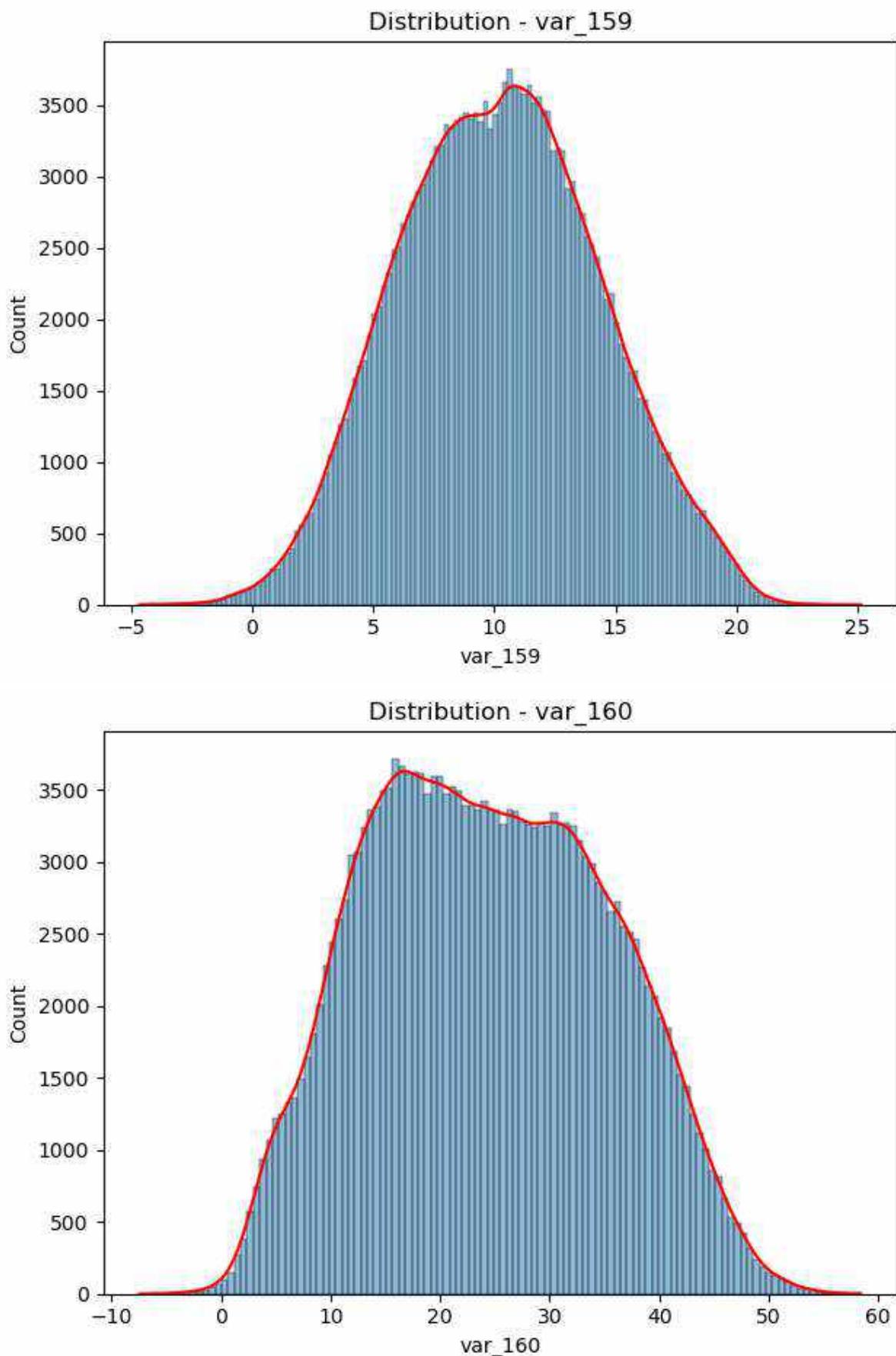


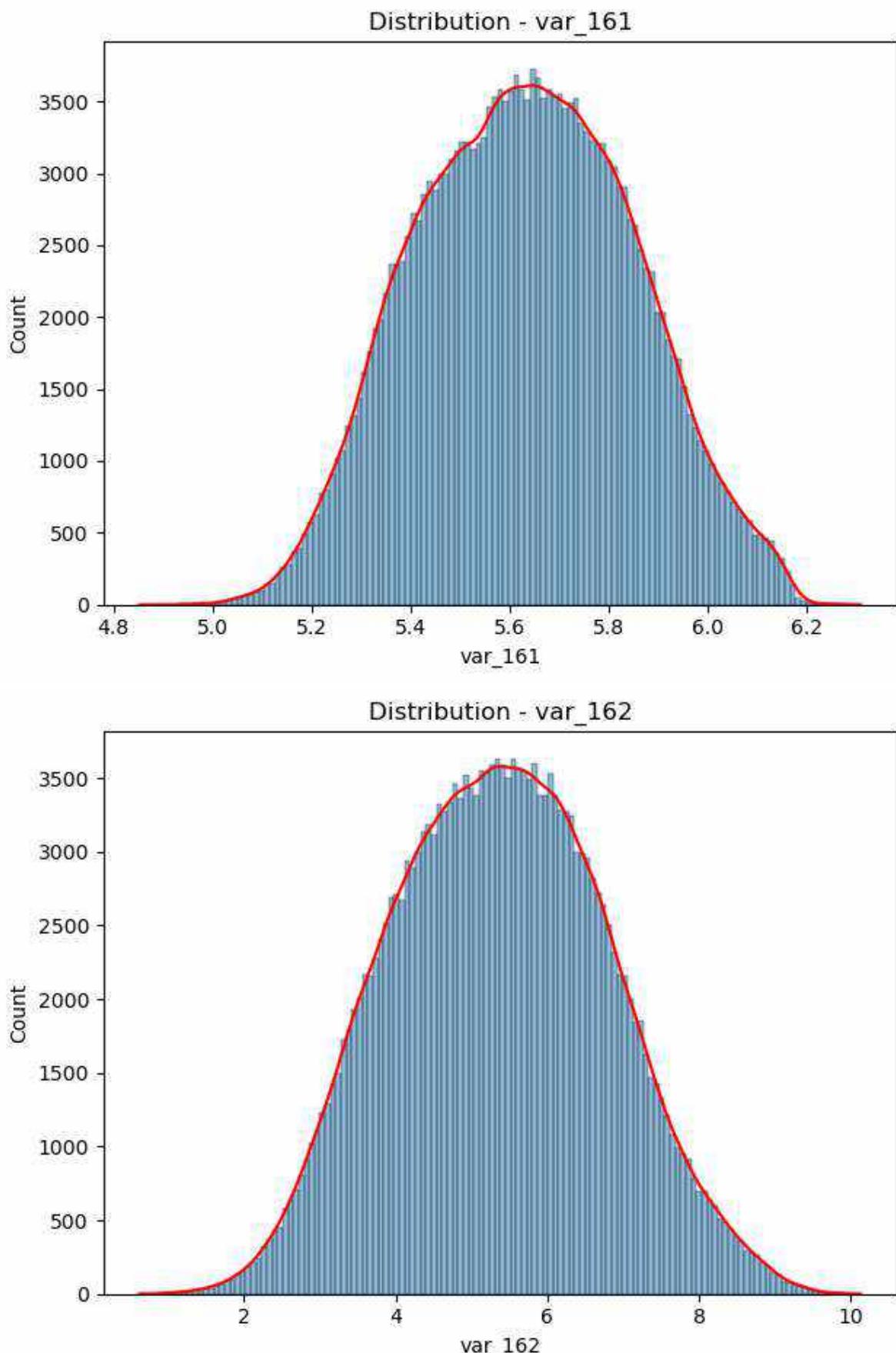


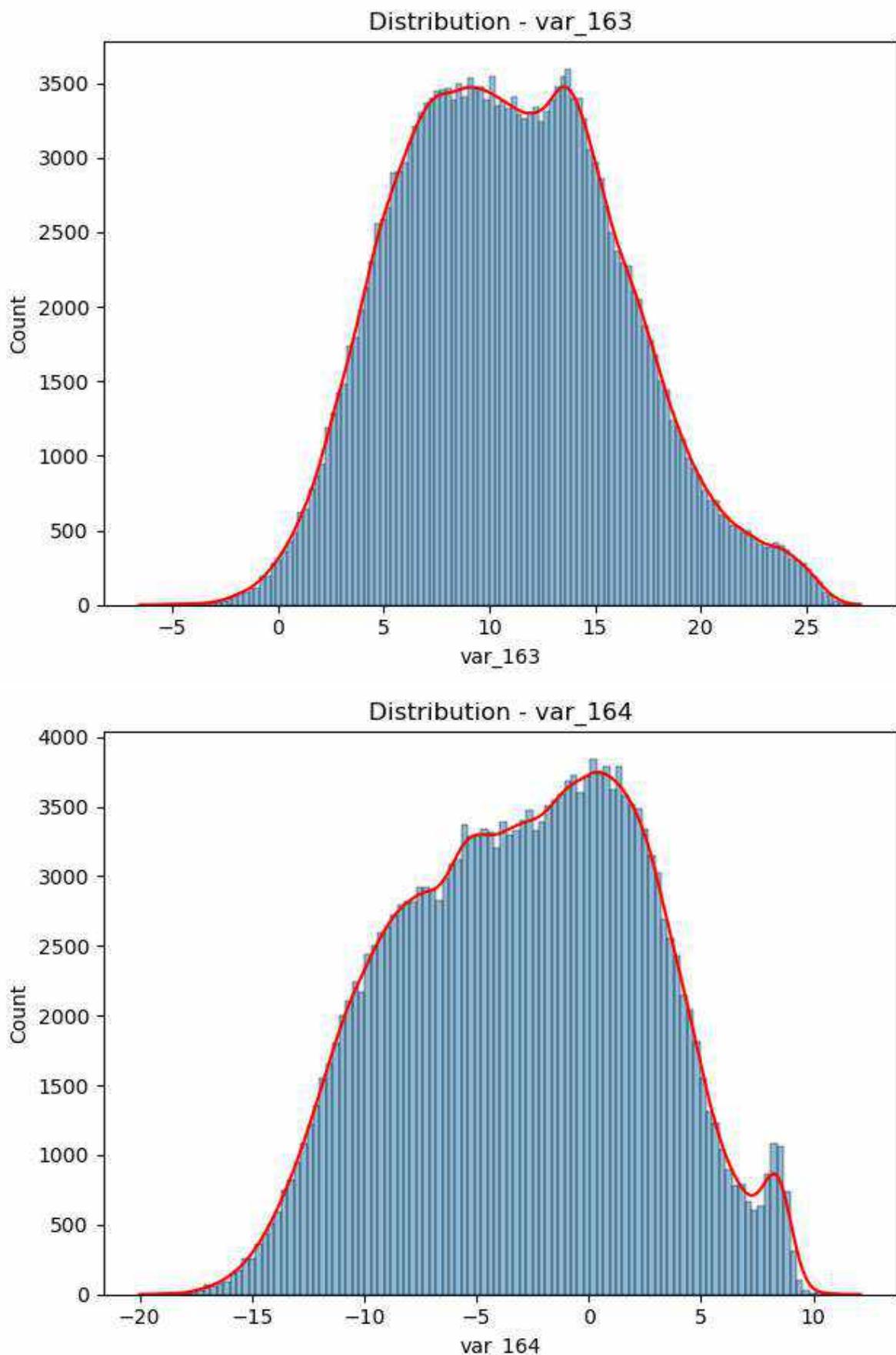


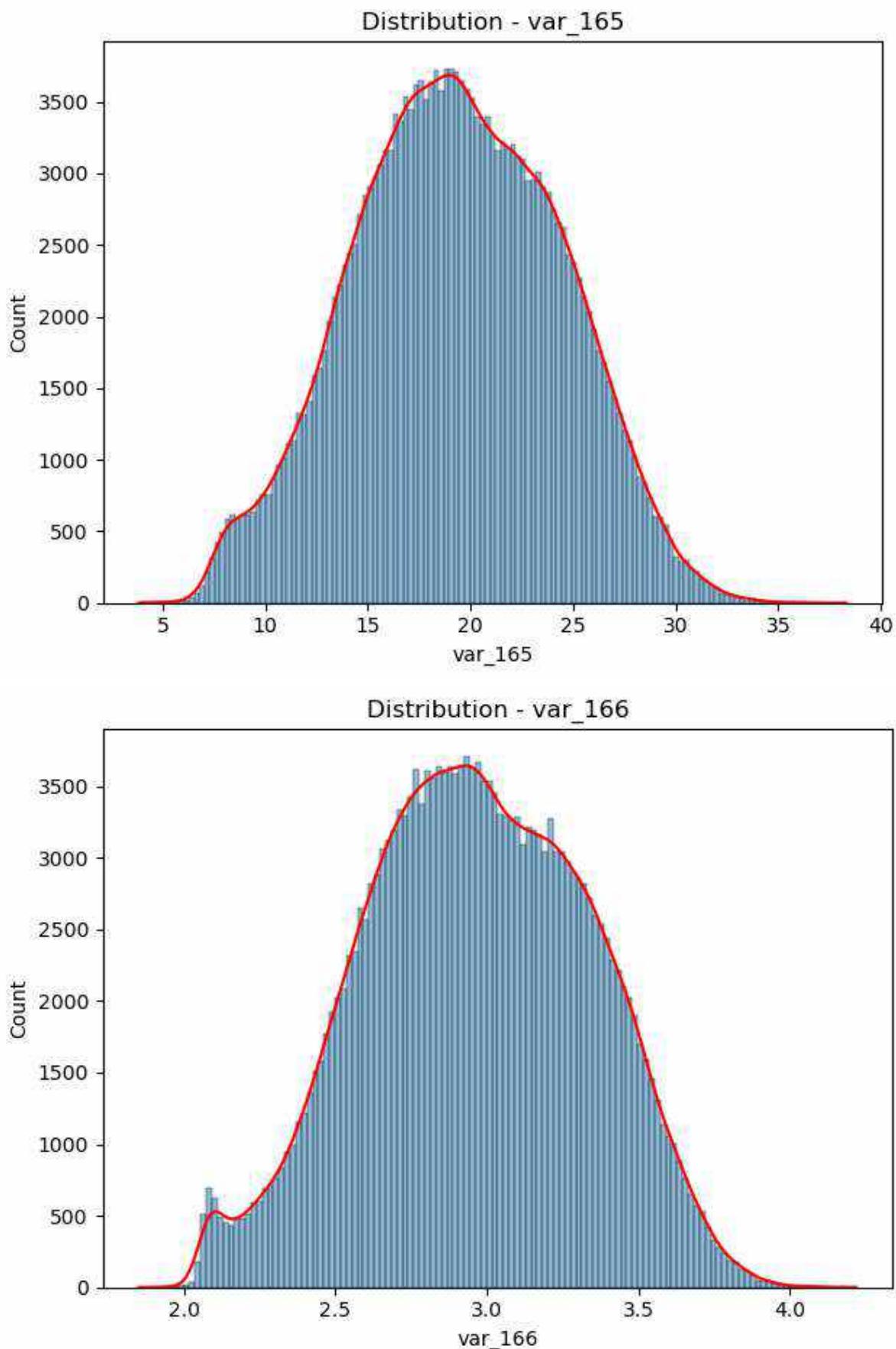


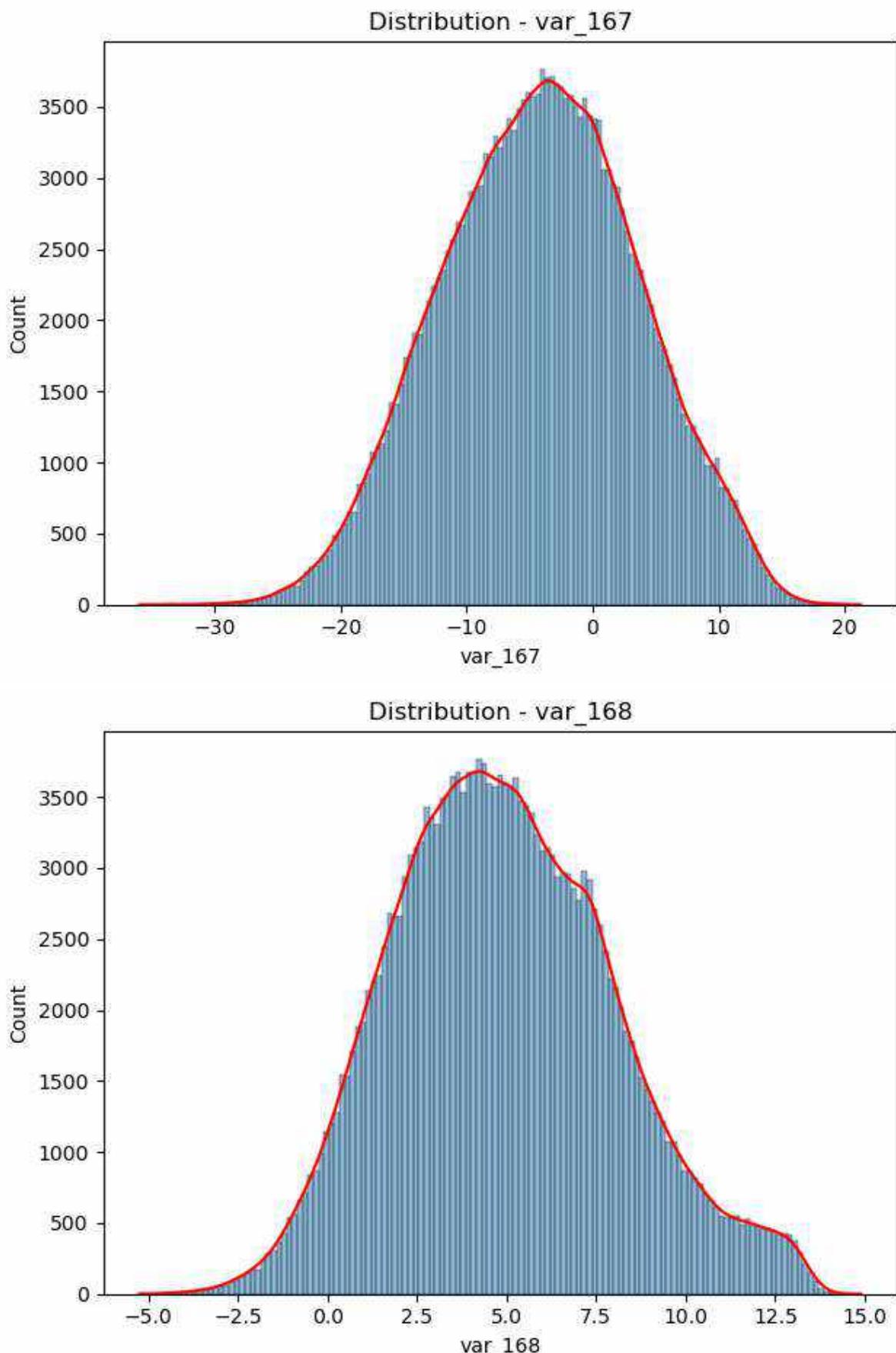


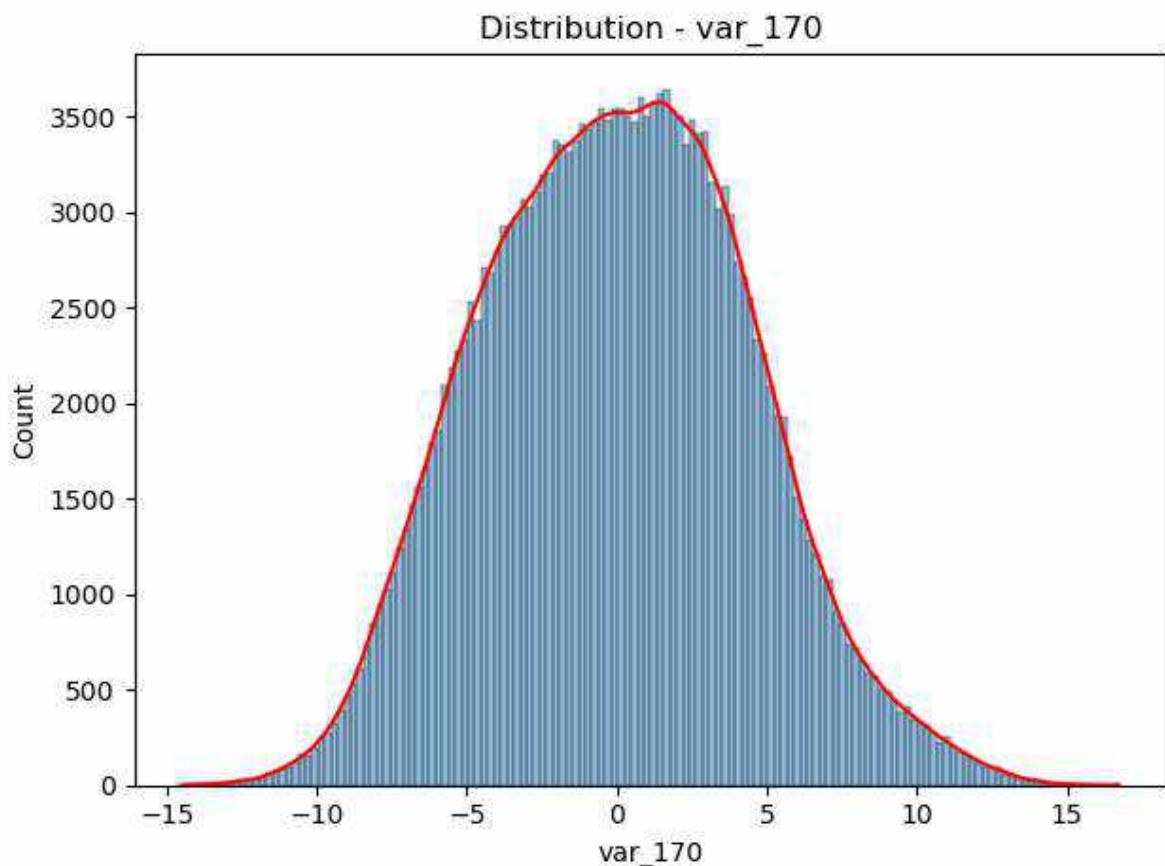
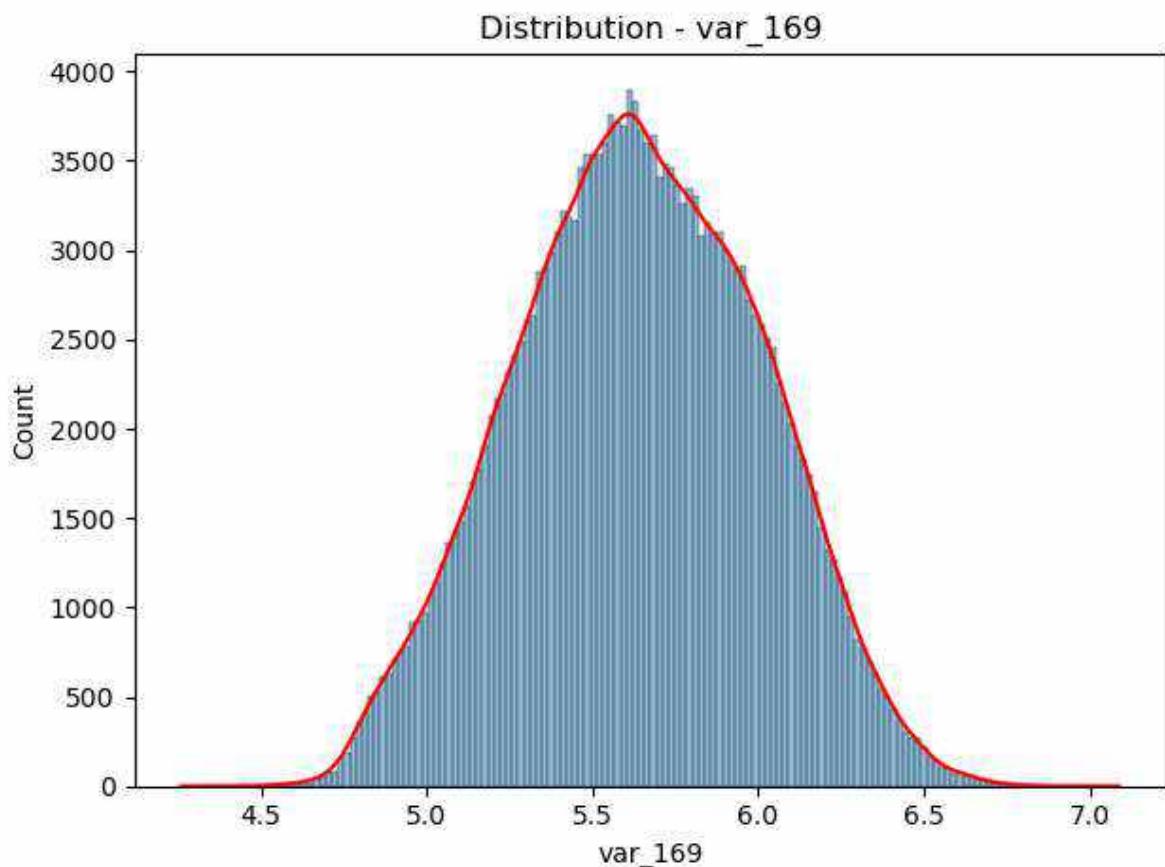


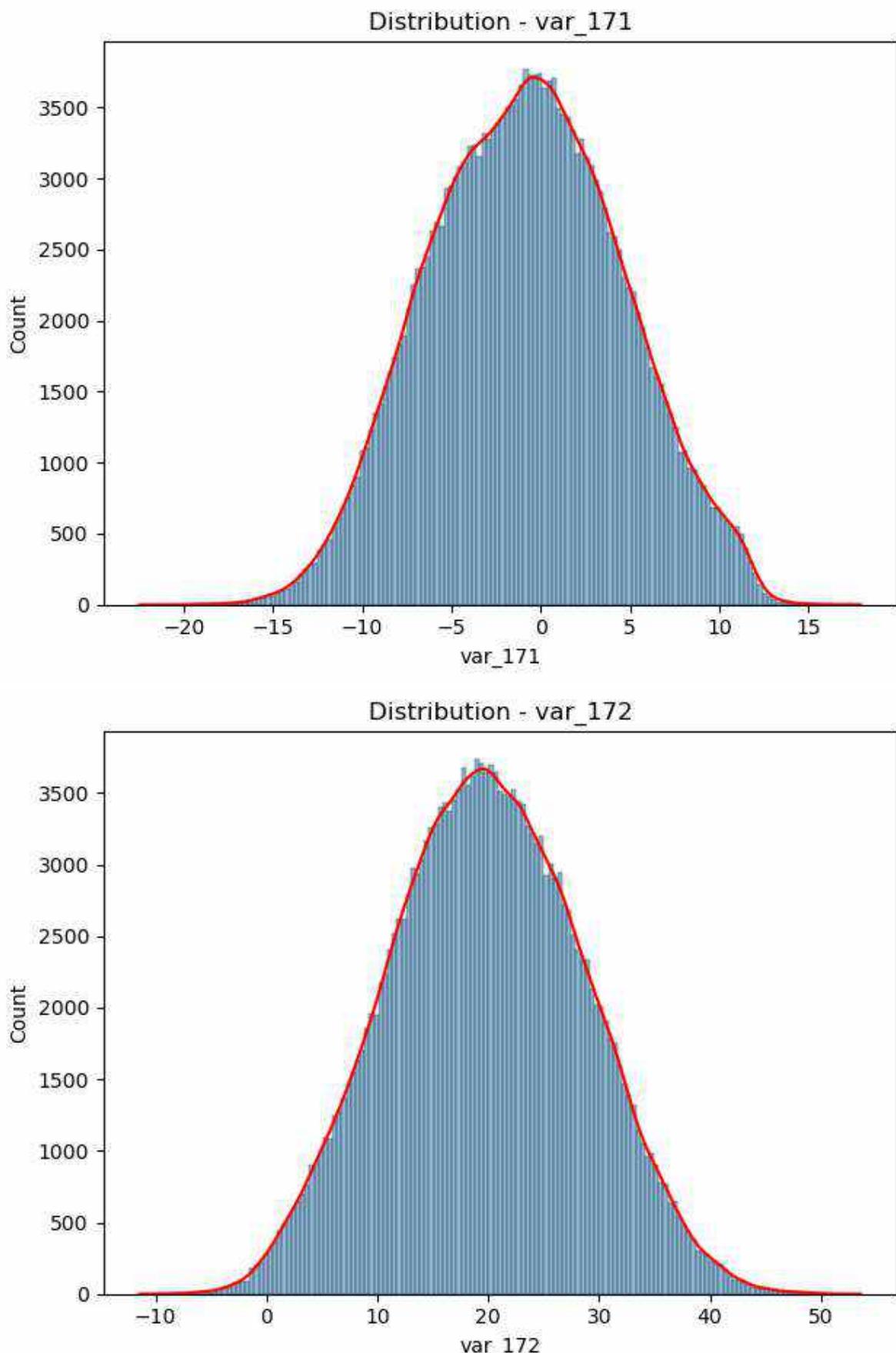


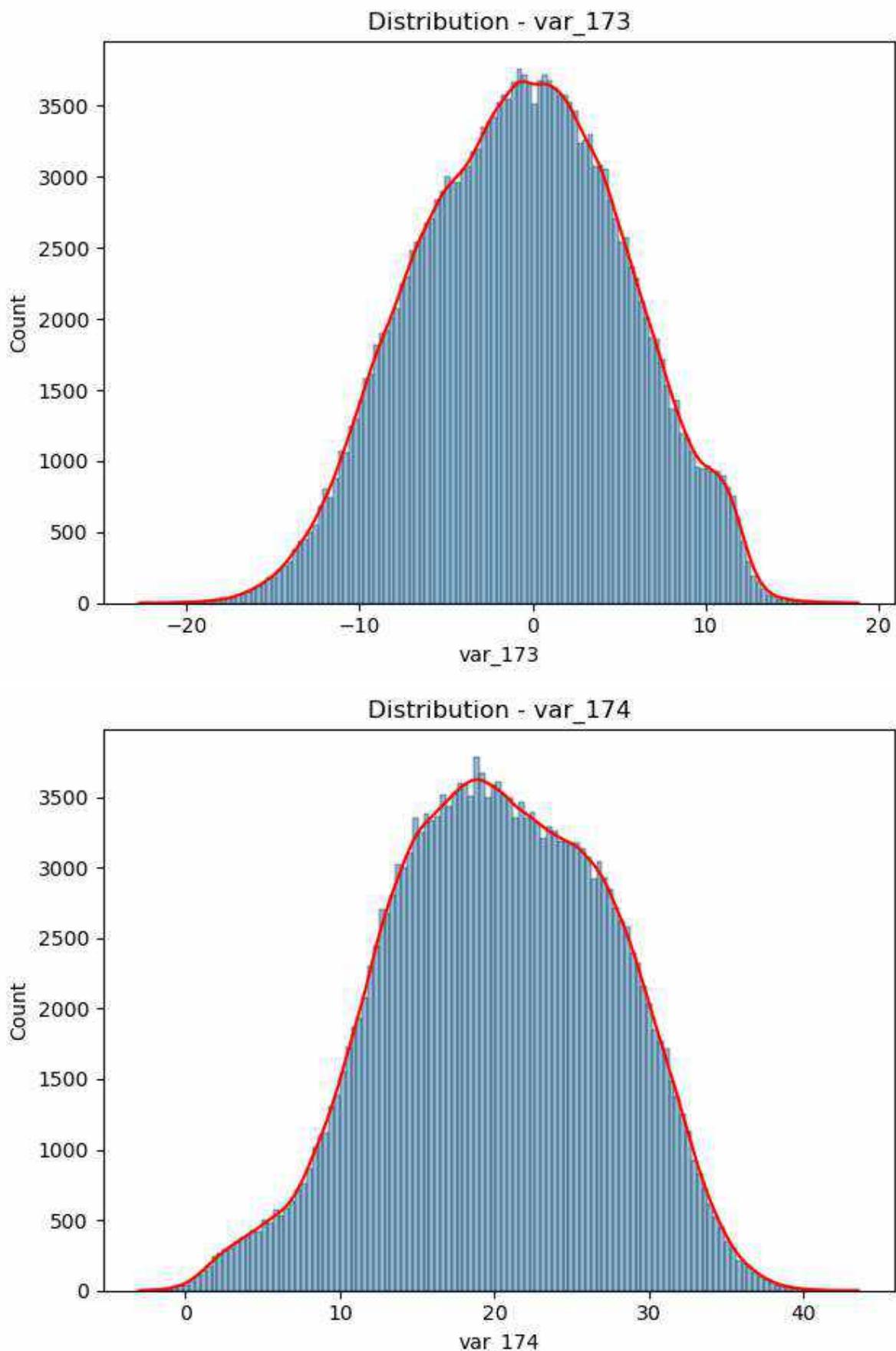


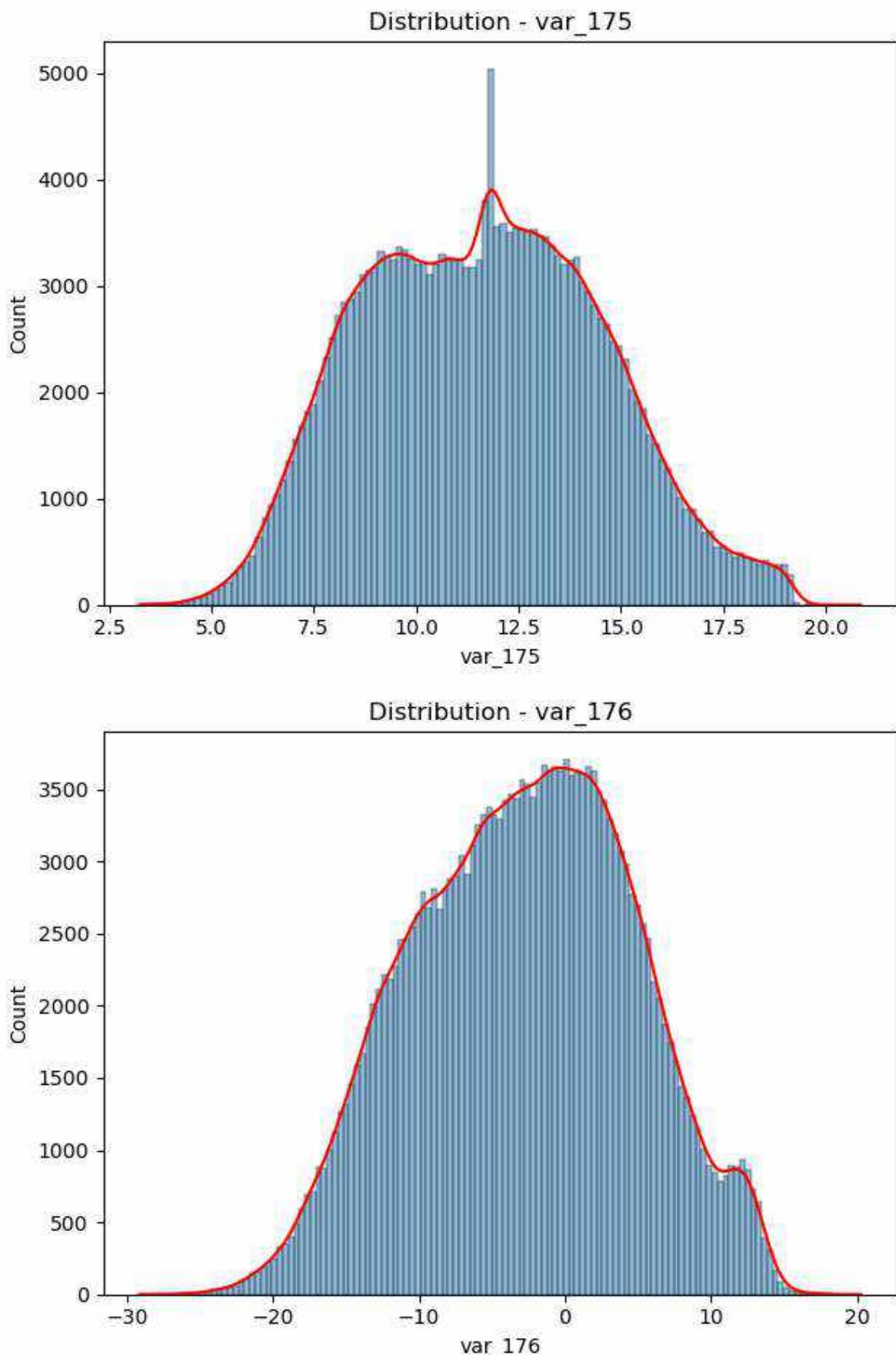


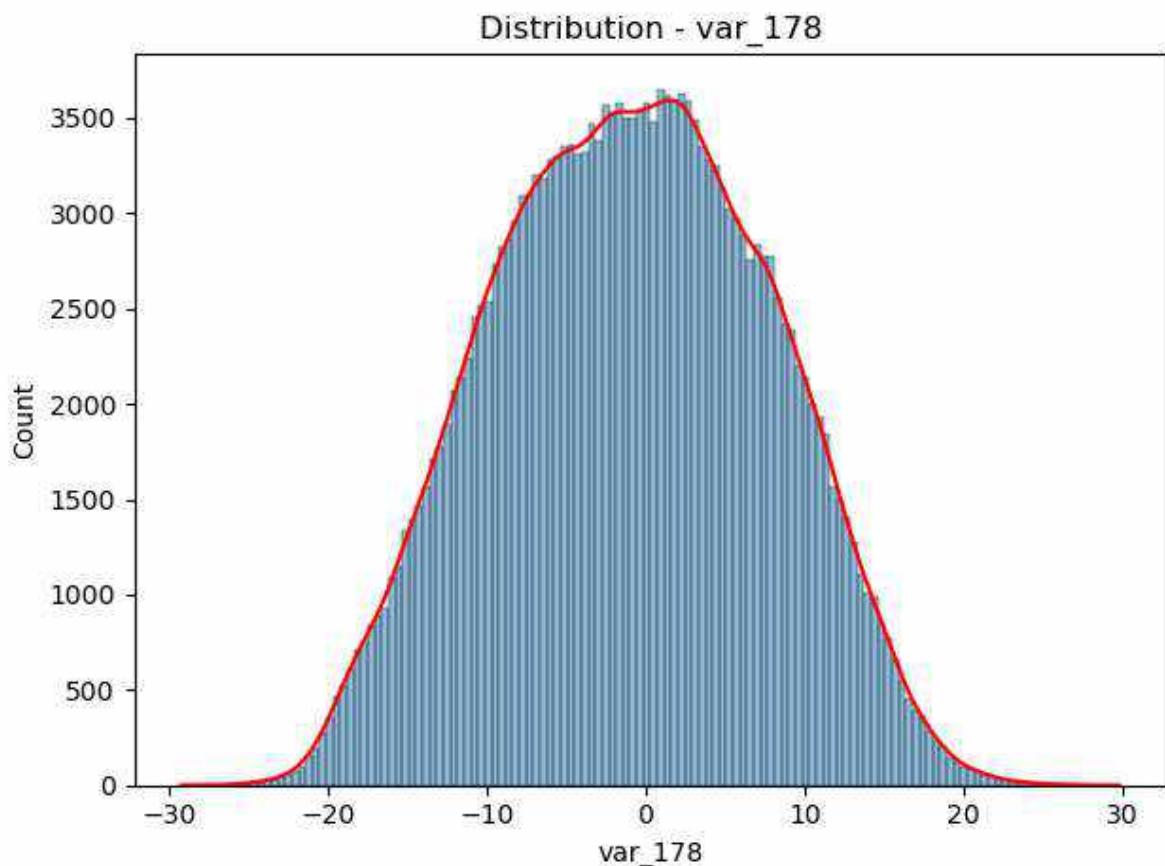
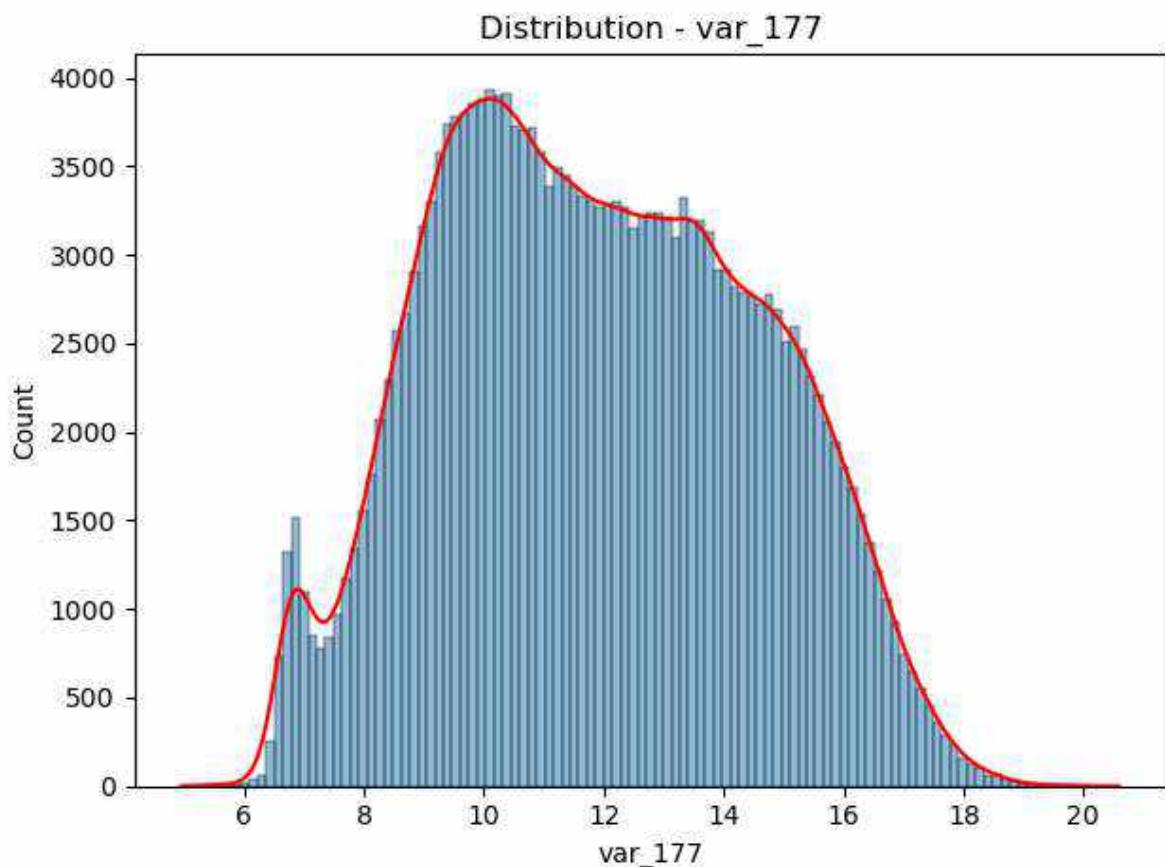


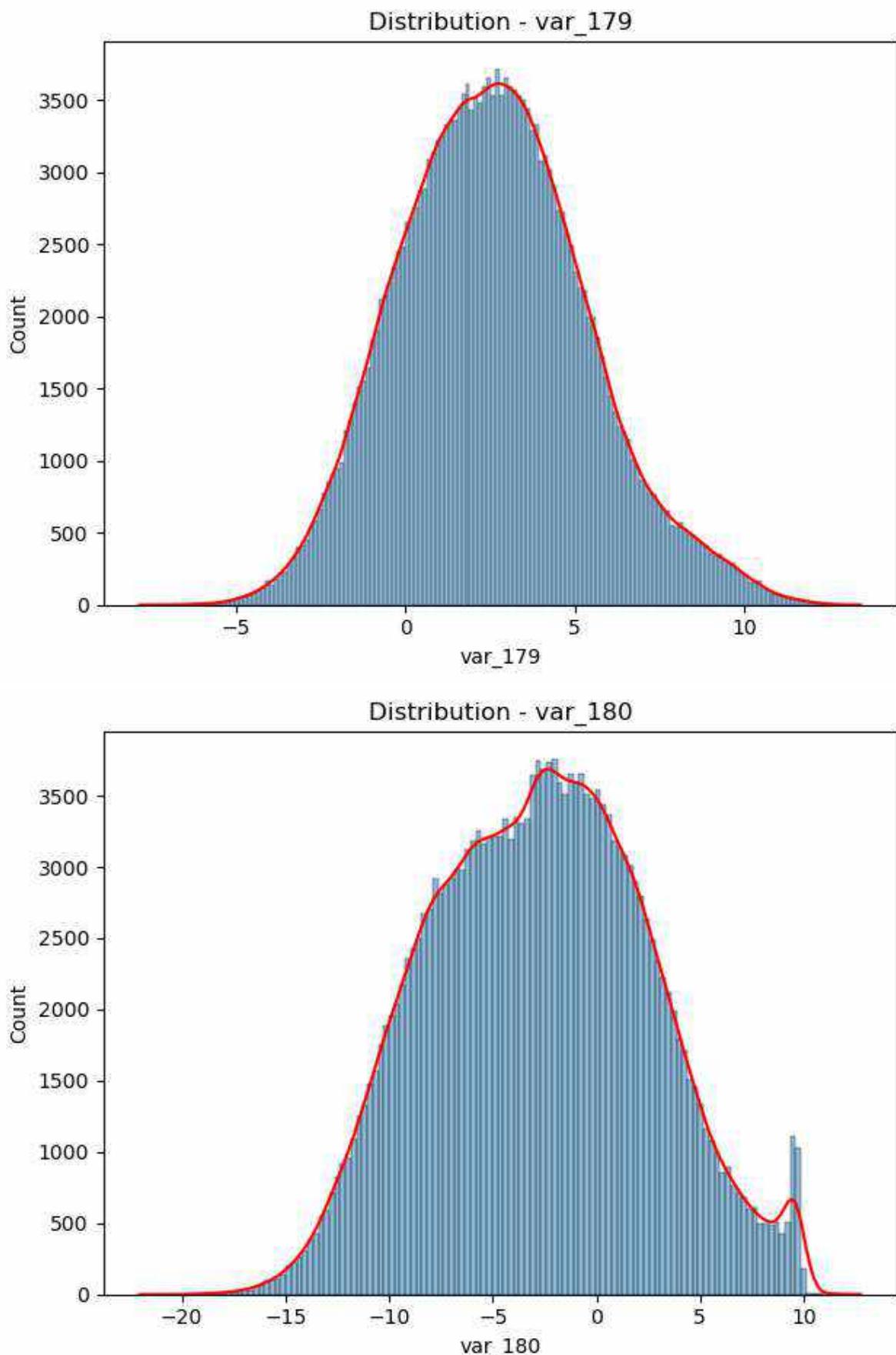


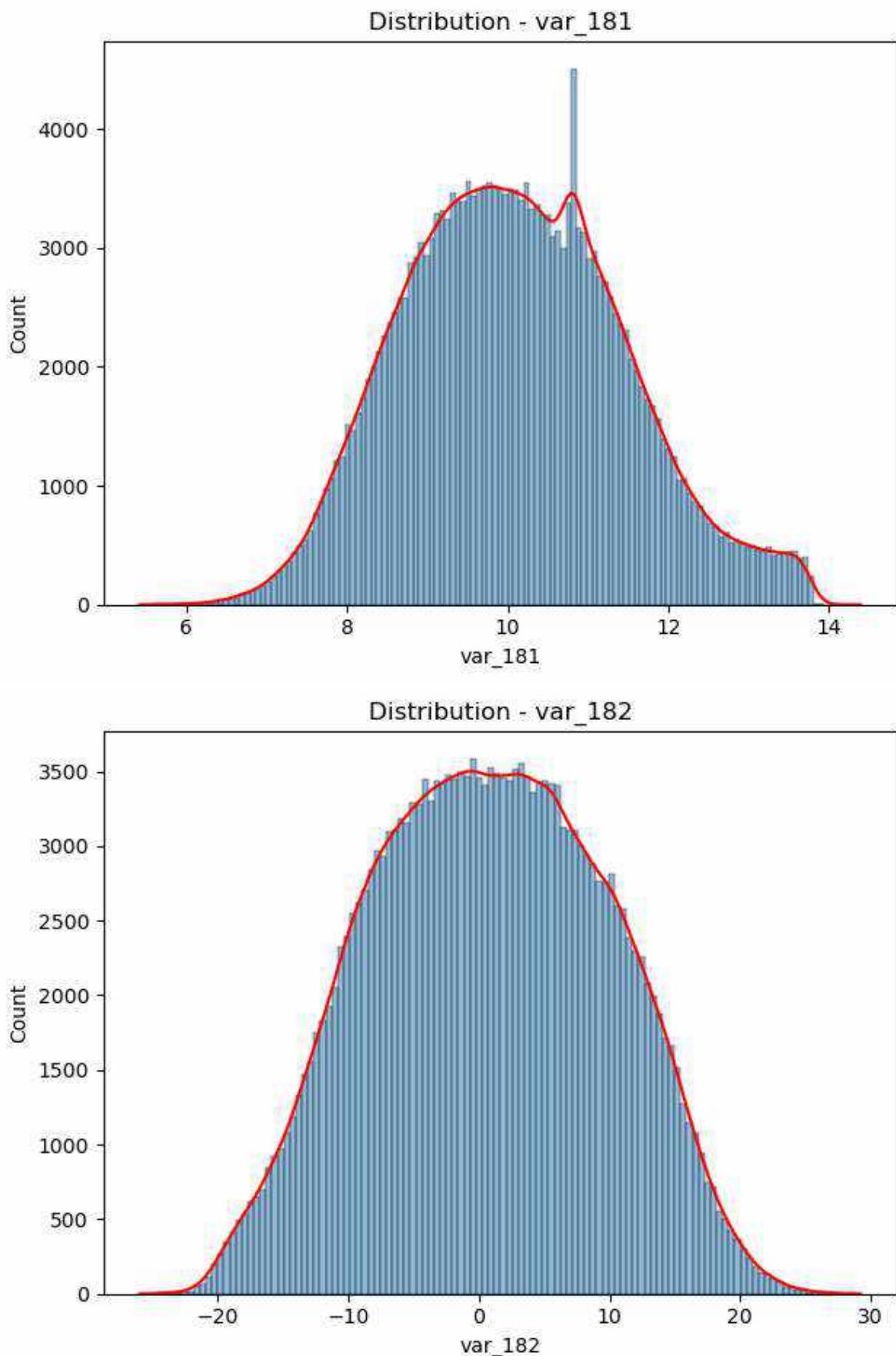


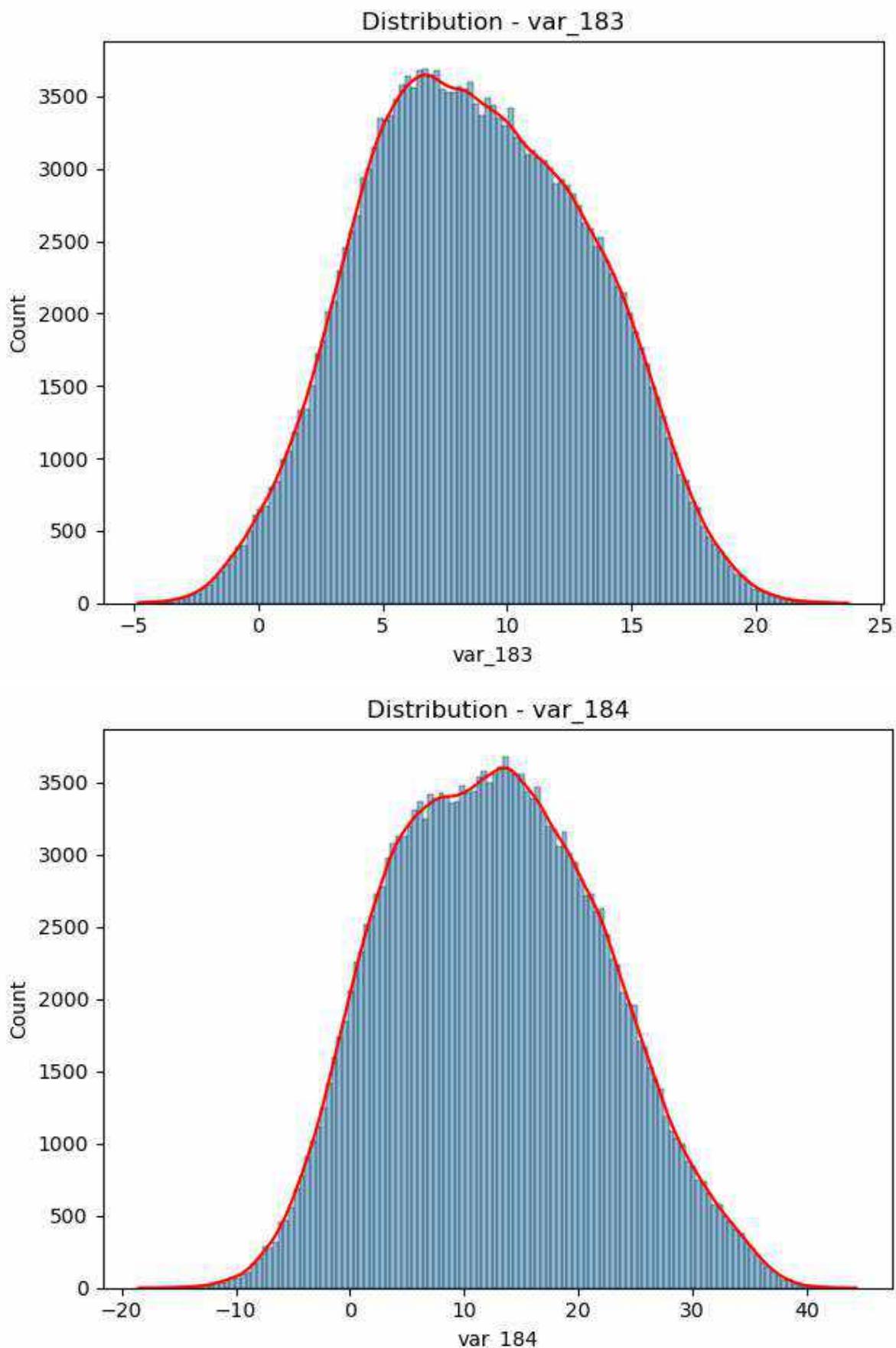


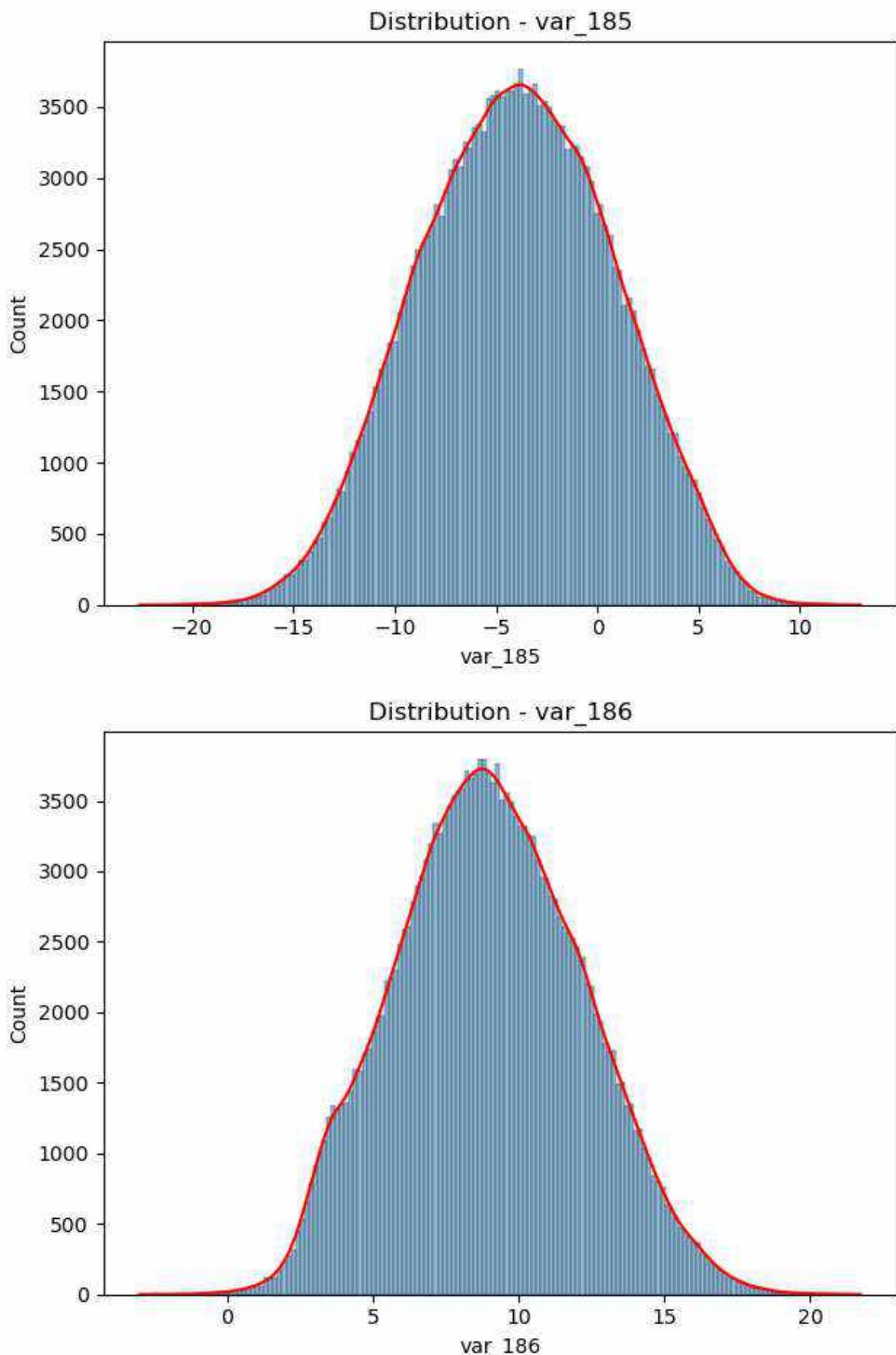


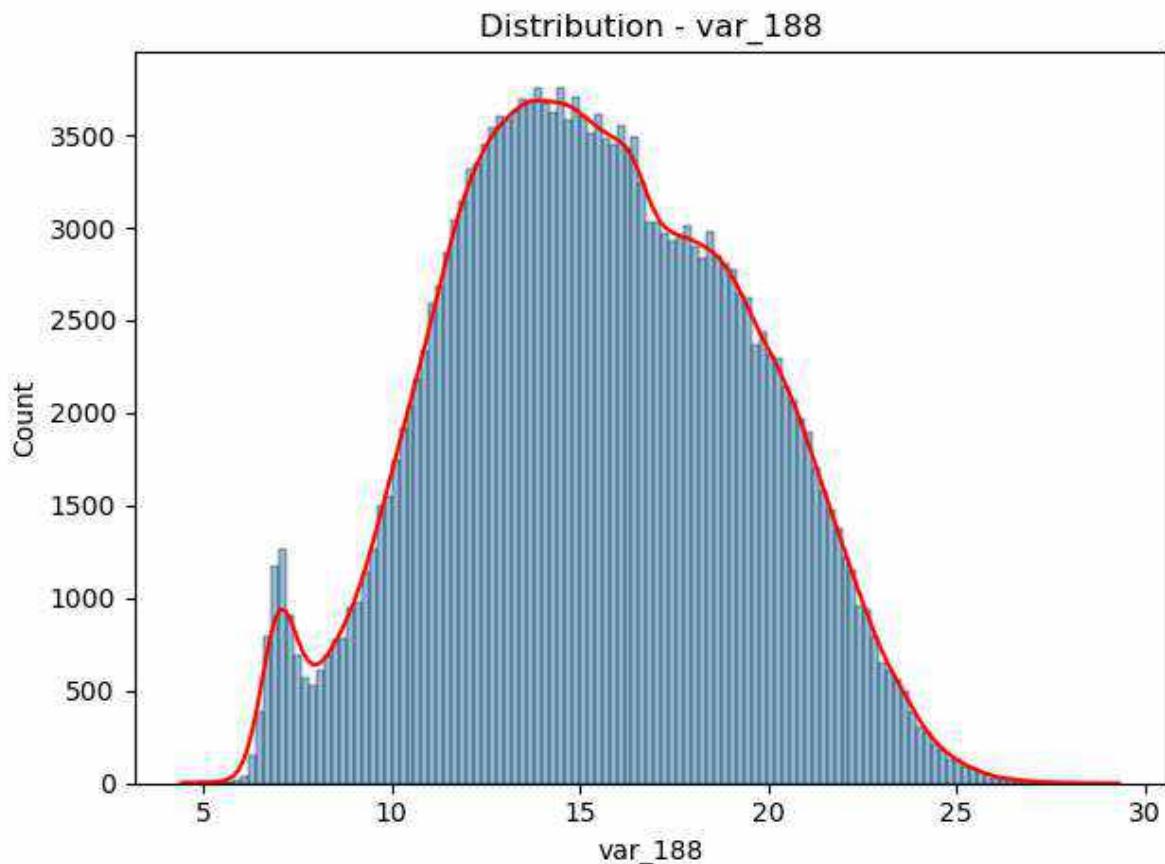
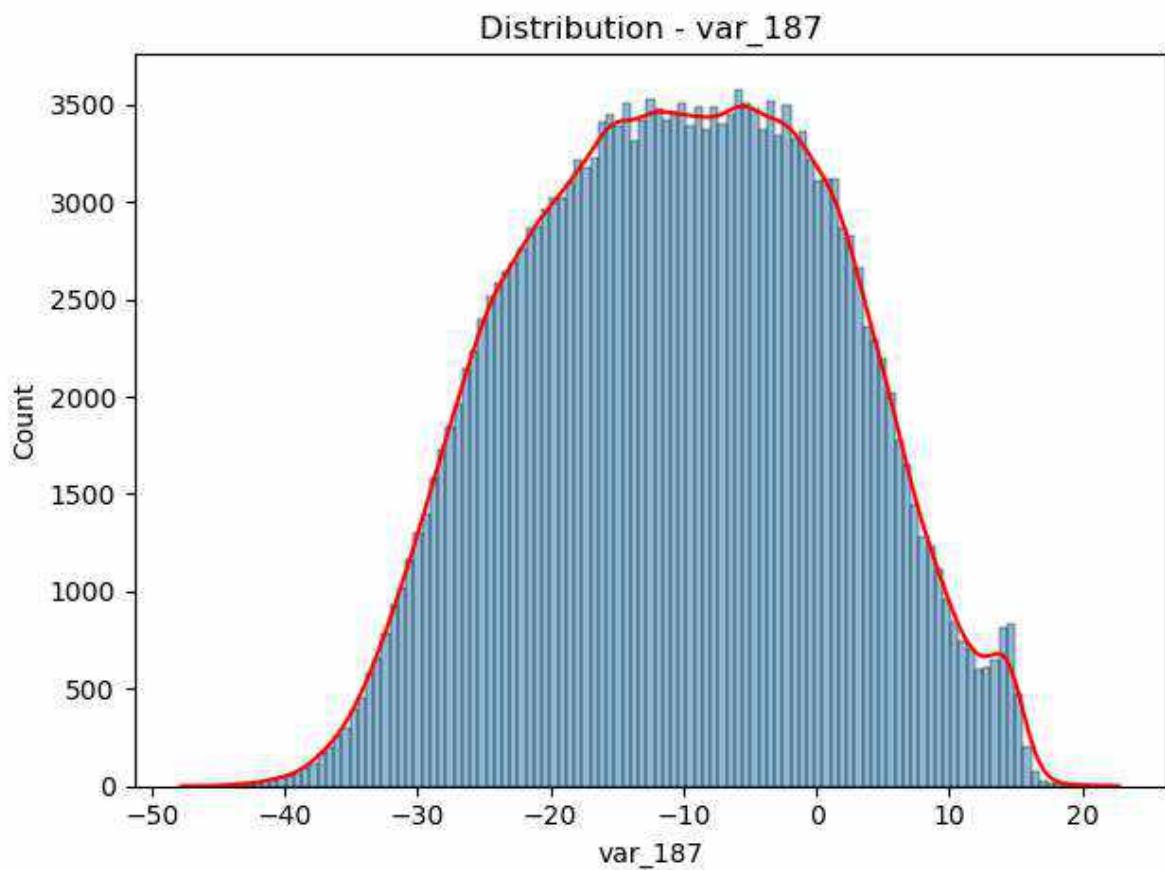


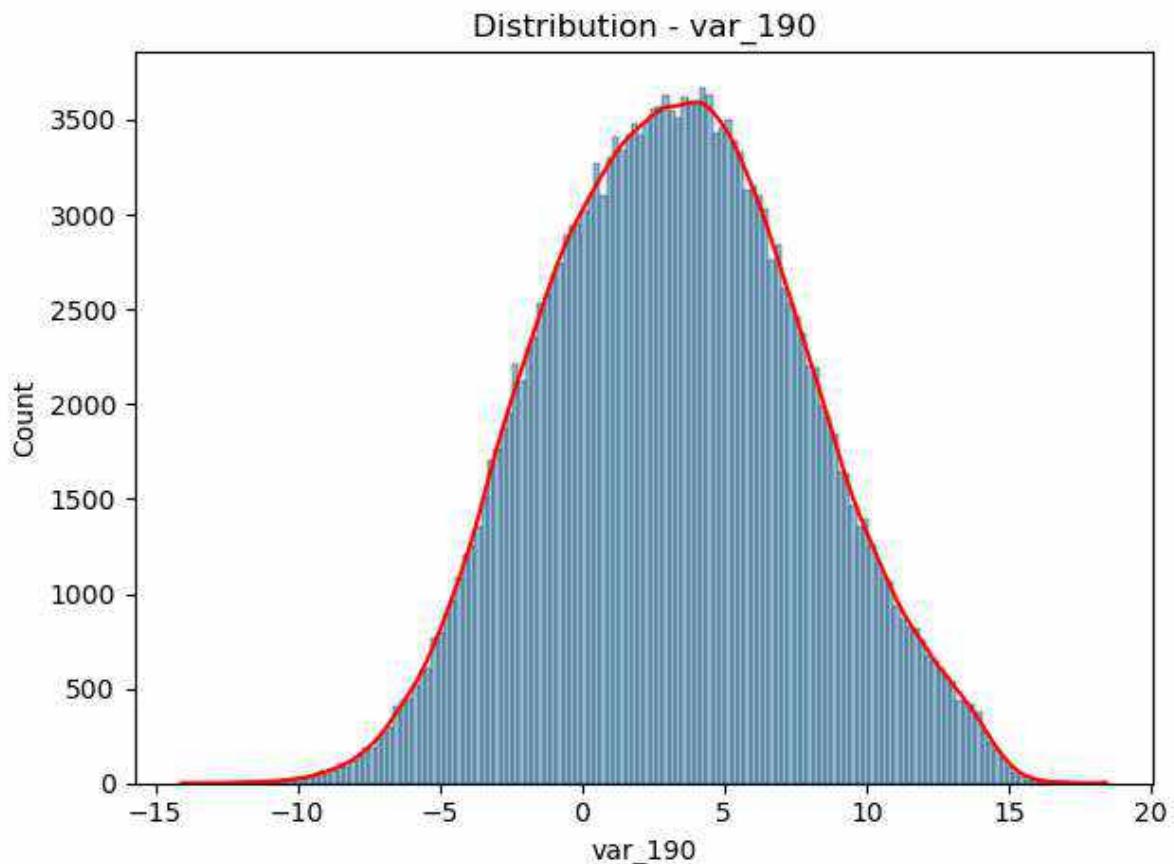
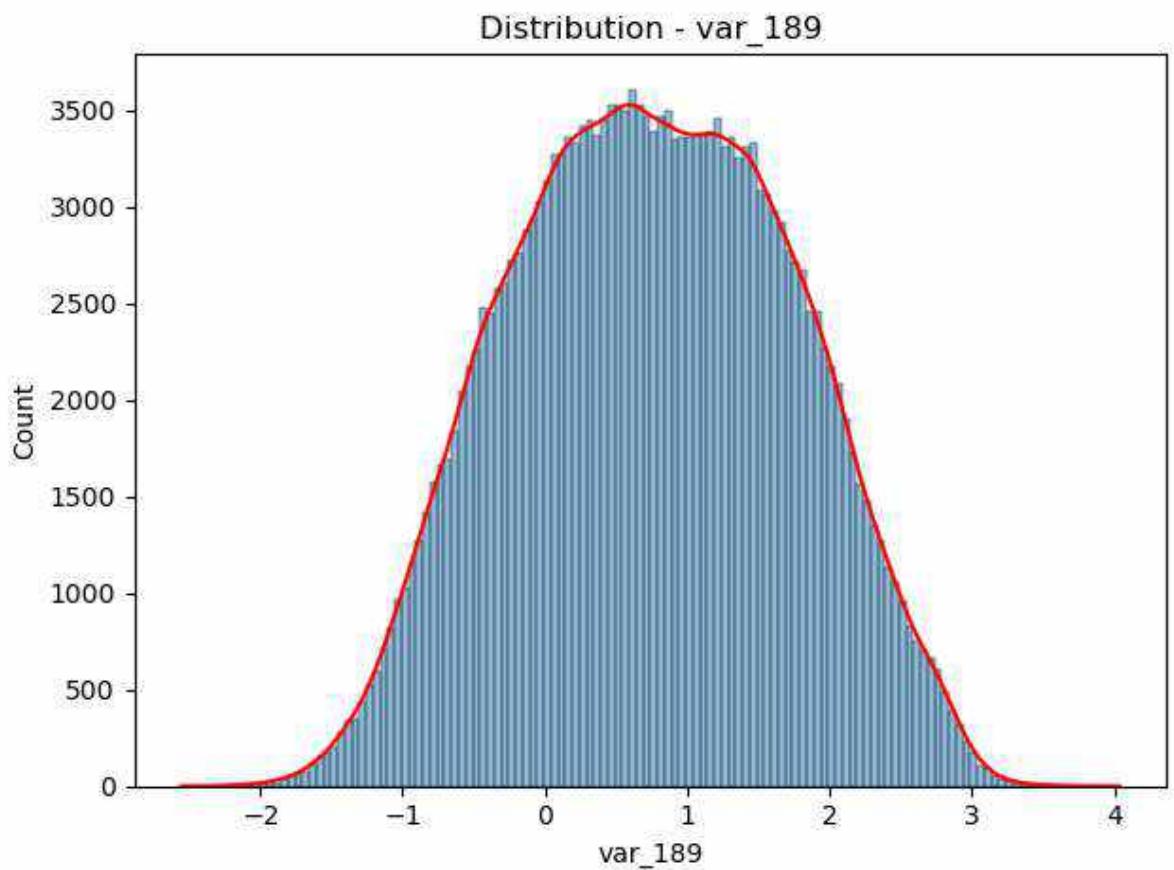


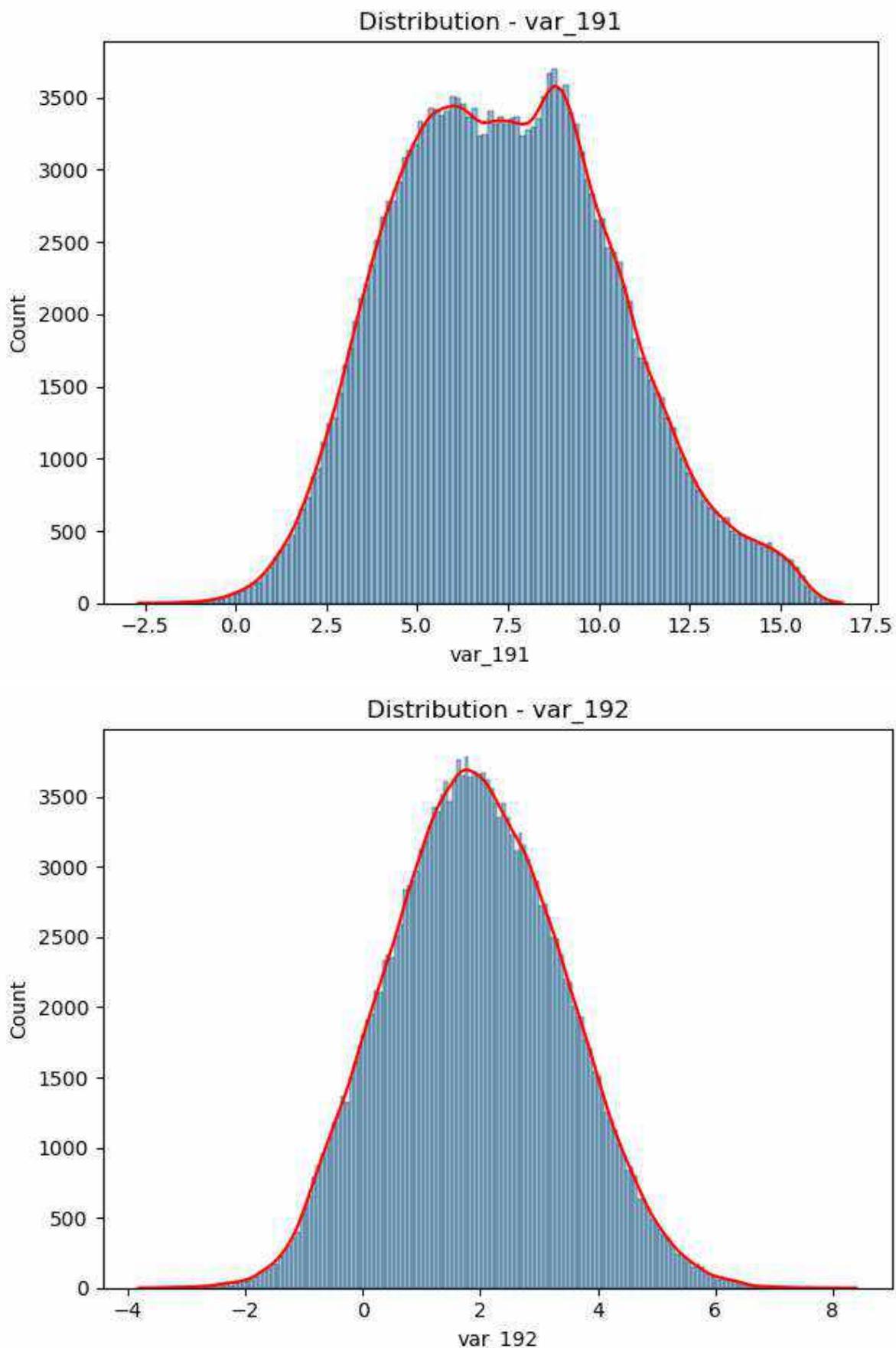


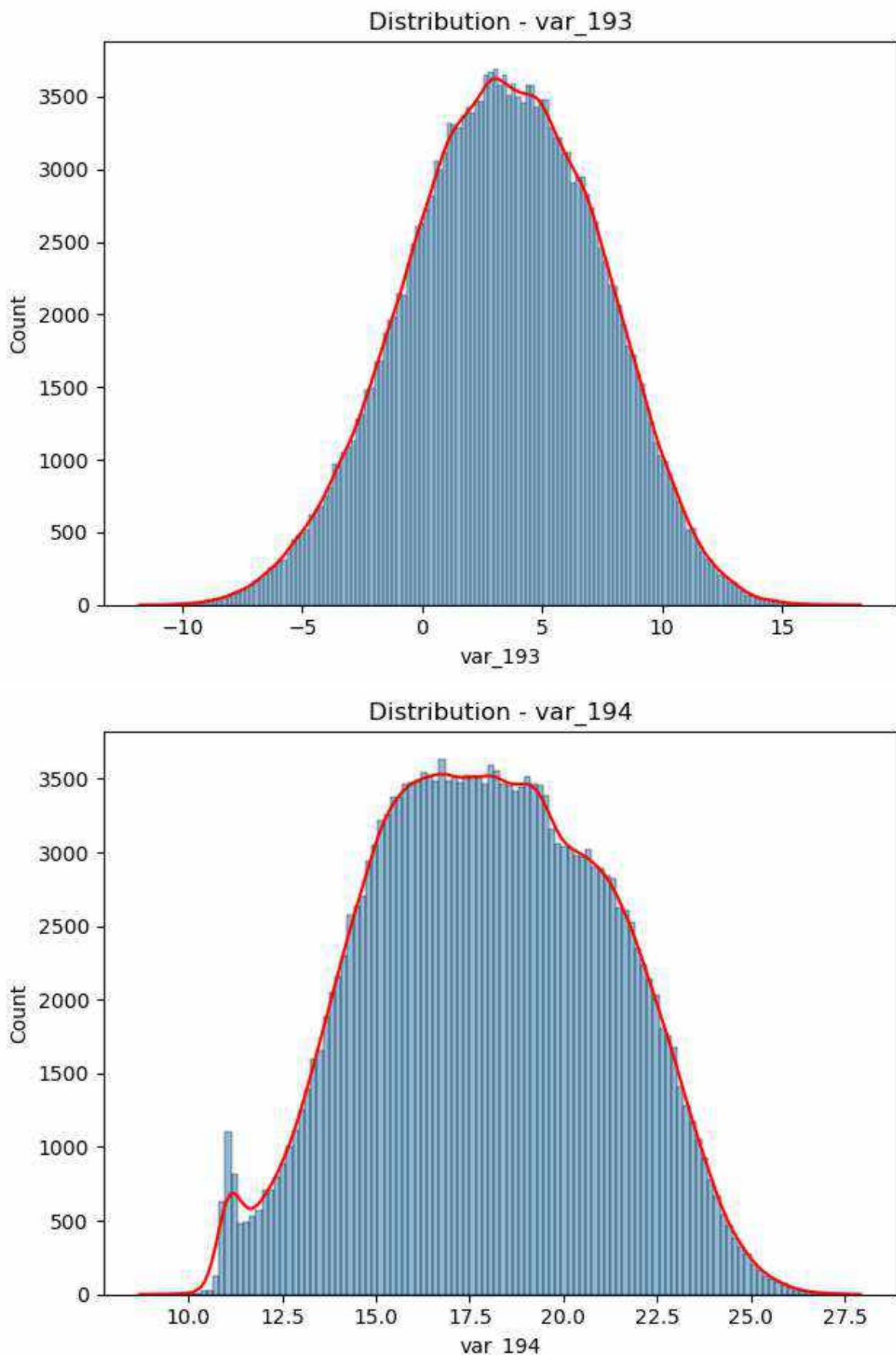


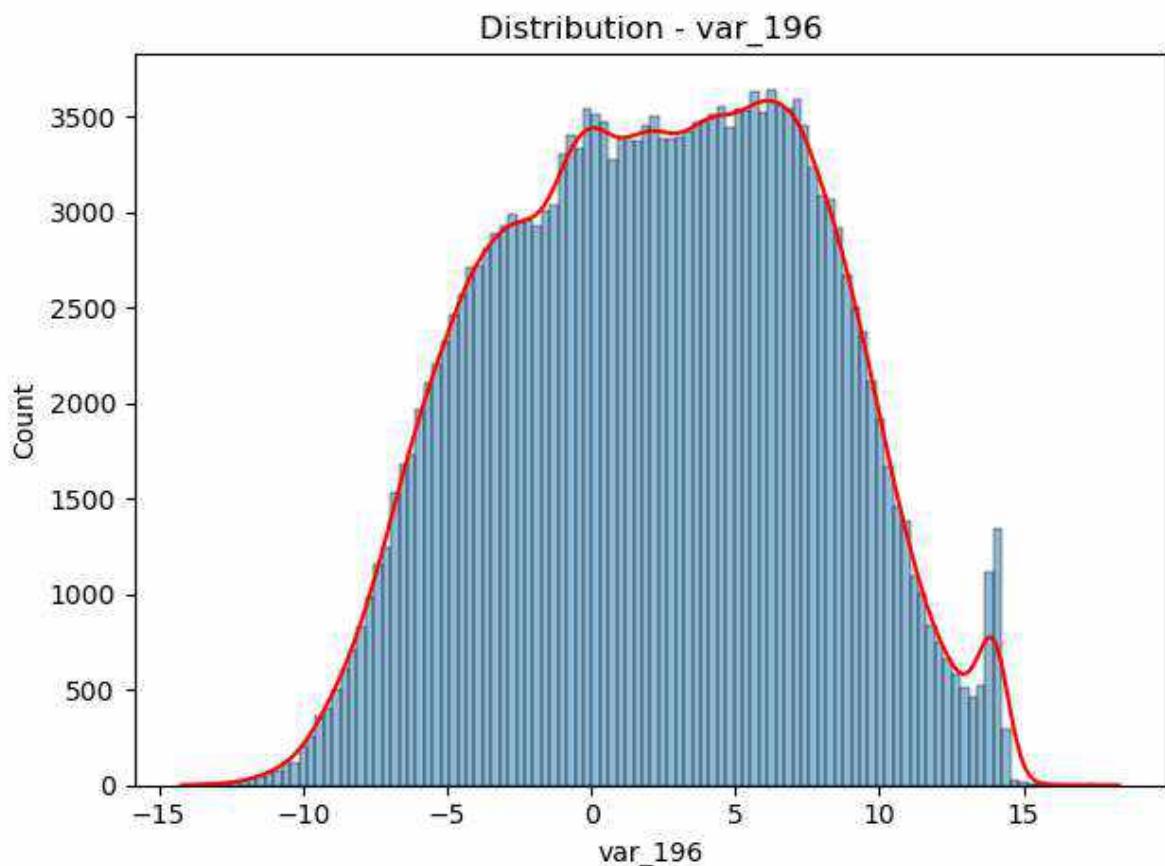
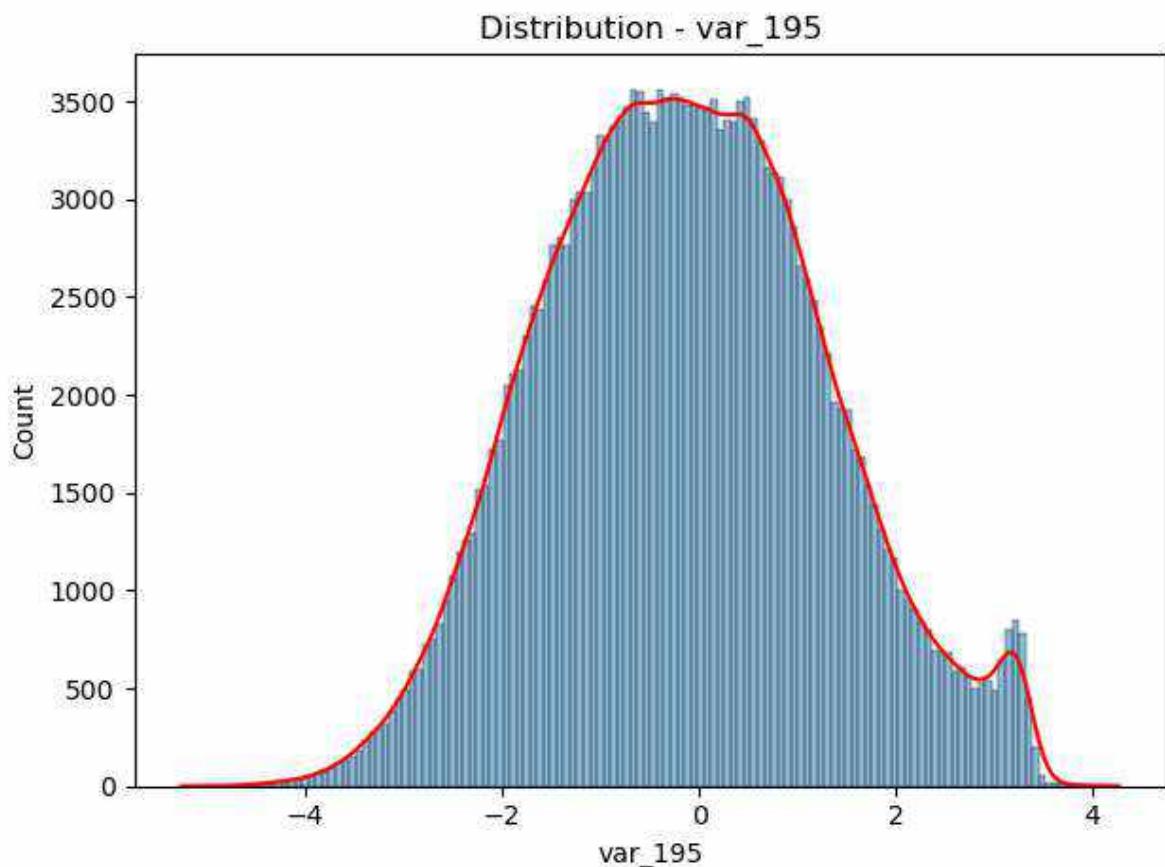


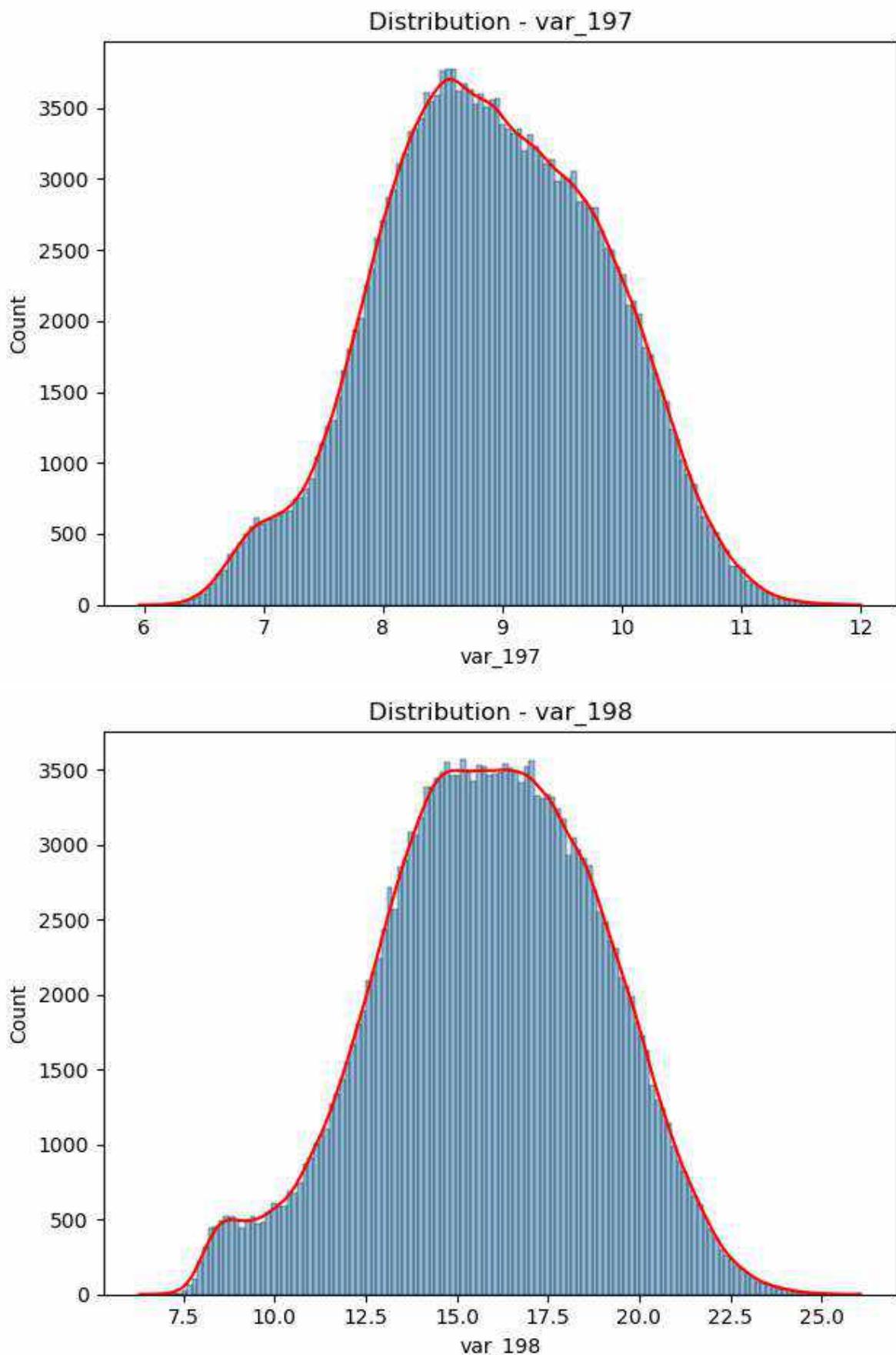


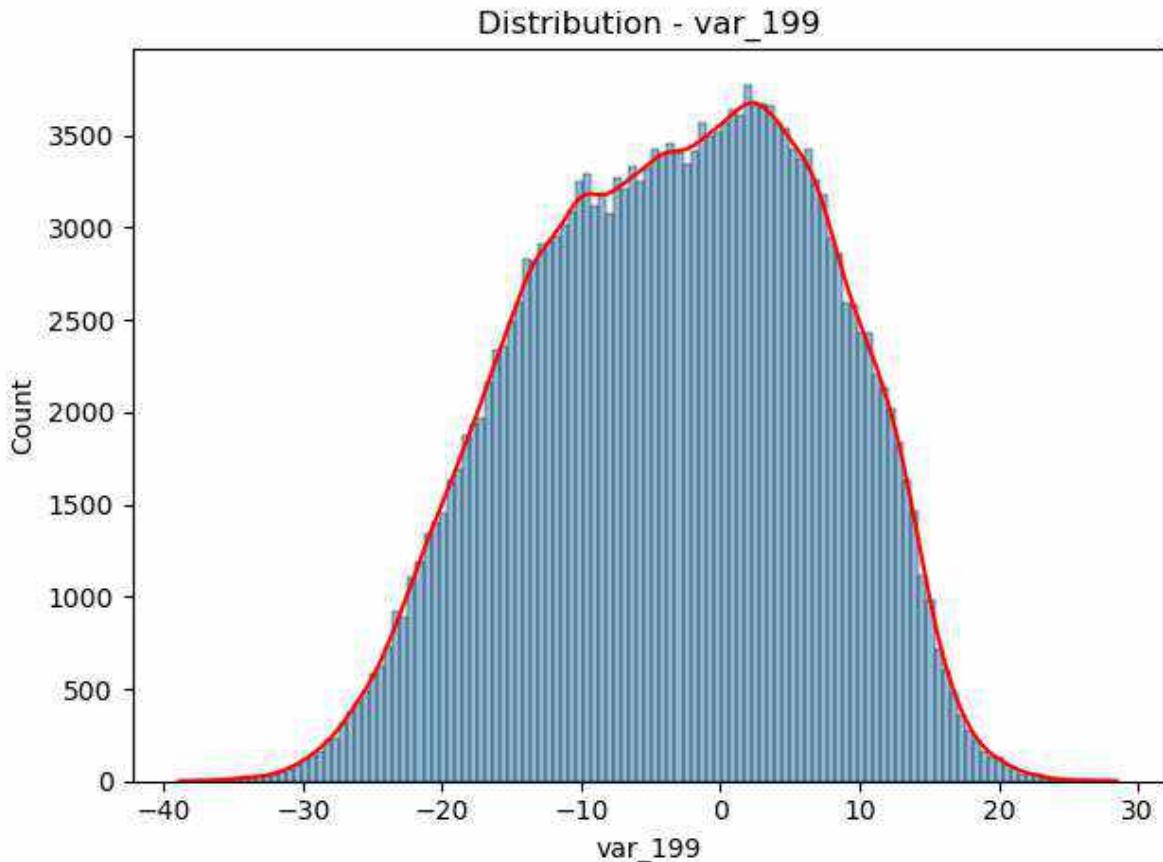












### Distribution Summary based on columns : training\_set

```
In [18]: # Exclude ID_code and target columns if present
numeric_columns = [column for column in data_training_set.columns if column not in

# the shape of the distribution and identify potential outliers for each column
distribution_summary = {'Normal': 0, 'Right-skewed': 0, 'Left-skewed': 0, 'Outliers': 0}
outlier_columns = []

for column in numeric_columns:
    # the shape of the distribution
    skewness = data_training_set[column].skew()
    if skewness > 1:
        distribution_summary['Right-skewed'] += 1
    elif skewness < -1:
        distribution_summary['Left-skewed'] += 1
    else:
        distribution_summary['Normal'] += 1

    # Identifying potential outliers
    q1 = data_training_set[column].quantile(0.25)
    q3 = data_training_set[column].quantile(0.75)
    iqr = q3 - q1
    lower_bound = q1 - 1.5 * iqr
    upper_bound = q3 + 1.5 * iqr
    outliers = data_training_set[(data_training_set[column] < lower_bound) | (data_training_set[column] > upper_bound)]

    if not outliers.empty:
        distribution_summary['Outliers'] += 1
        outlier_columns.append(column)

# distribution summary
print("Distribution Summary based on columns:")
for shape, count in distribution_summary.items():
    print(f"{shape}: {count}")
```

Distribution Summary based on columns:

Normal: 200  
 Right-skewed: 0  
 Left-skewed: 0  
 Outliers: 189

For training\_set, there are 200 columns that exhibit a normal distribution. So, the majority of the columns exhibit a normal distribution, while 189 columns have outliers. None of the columns are right-skewed or Left-skewed.

## Identifying potential outliers for training\_set

```
In [19]: # Exclude ID_code and target columns if present
numeric_columns = [column for column in data_training_set.columns if column not in
# the number of rows and columns for subplots
num_cols = 3
num_rows = (len(numeric_columns) + num_cols - 1) // num_cols

fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 5*num_rows))
fig.tight_layout(pad=3.0)

outliers = []
for i, column in enumerate(numeric_columns):
    row = i // num_cols
    col = i % num_cols
    ax = axes[row, col] if num_rows > 1 else axes[col]

    sns.boxplot(data=data_training_set[column], ax=ax, color='skyblue')

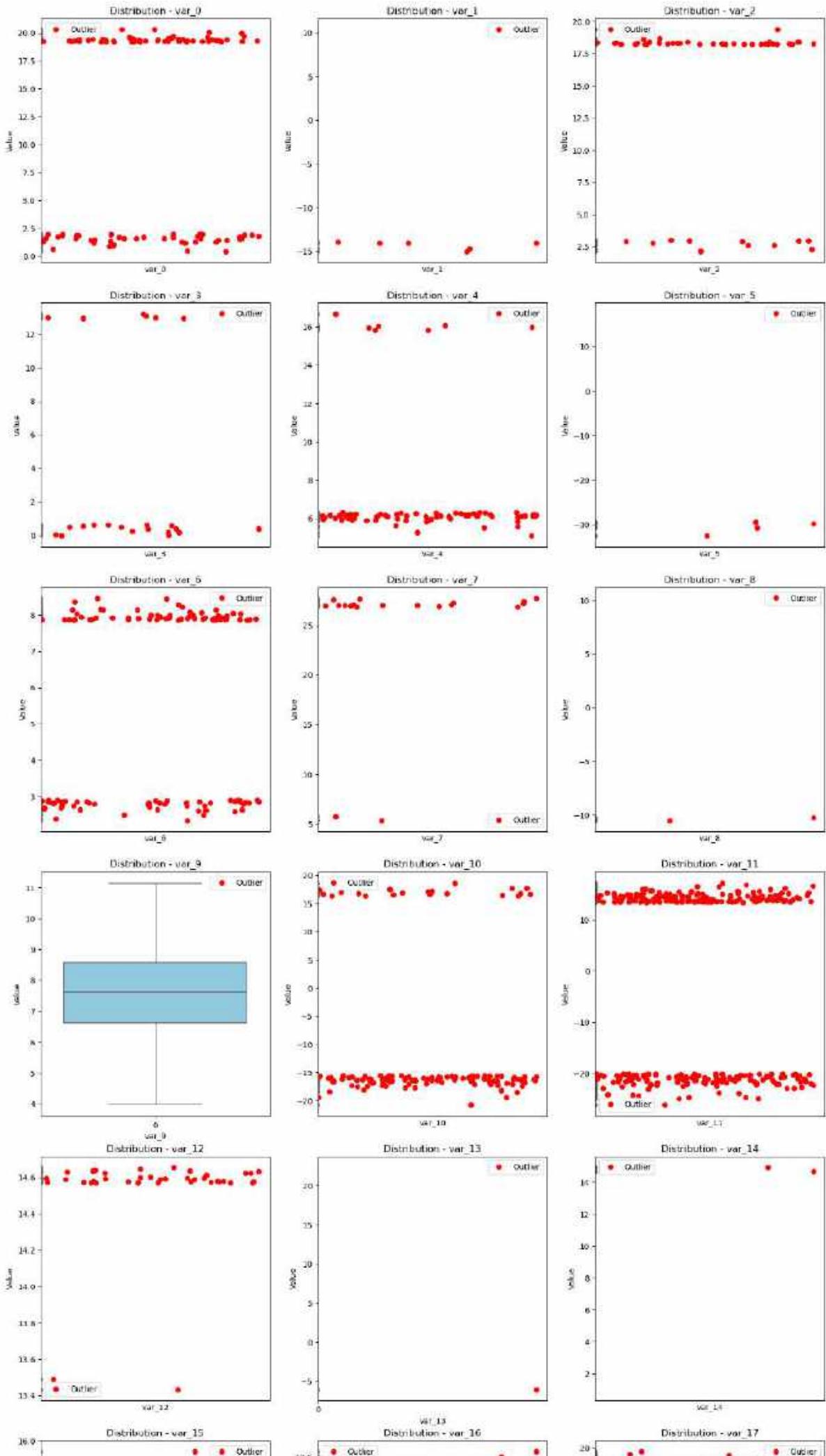
    # Identifying potential outliers
    q1 = data_training_set[column].quantile(0.25)
    q3 = data_training_set[column].quantile(0.75)
    iqr = q3 - q1
    lower_bound = q1 - 1.5 * iqr
    upper_bound = q3 + 1.5 * iqr
    column_outliers = data_training_set[(data_training_set[column] < lower_bound) | (data_training_set[column] > upper_bound)]
    outliers.extend(column_outliers.index)

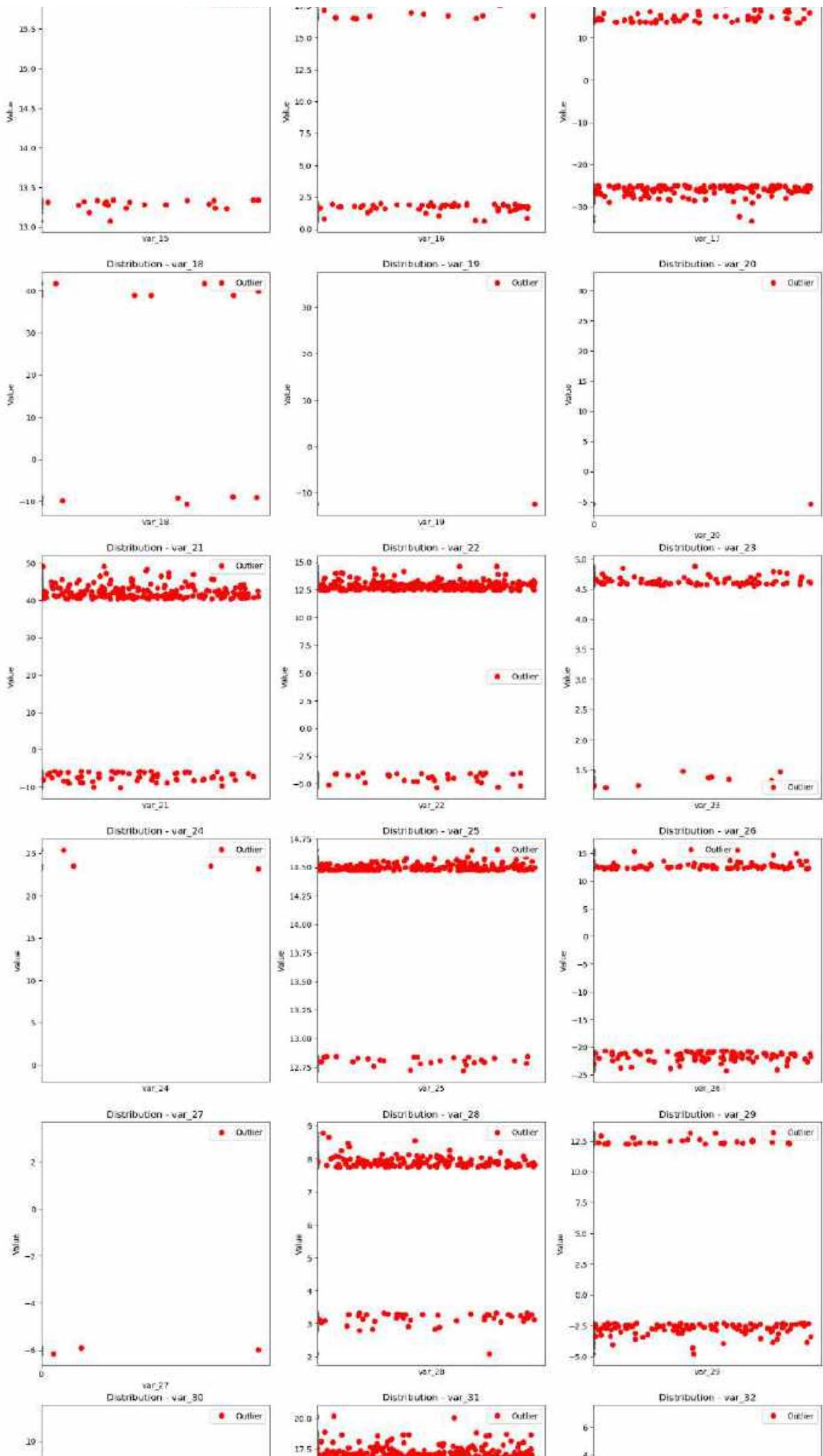
    ax.plot(column_outliers.index, column_outliers[column], 'ro', label='Outlier')
    ax.legend()

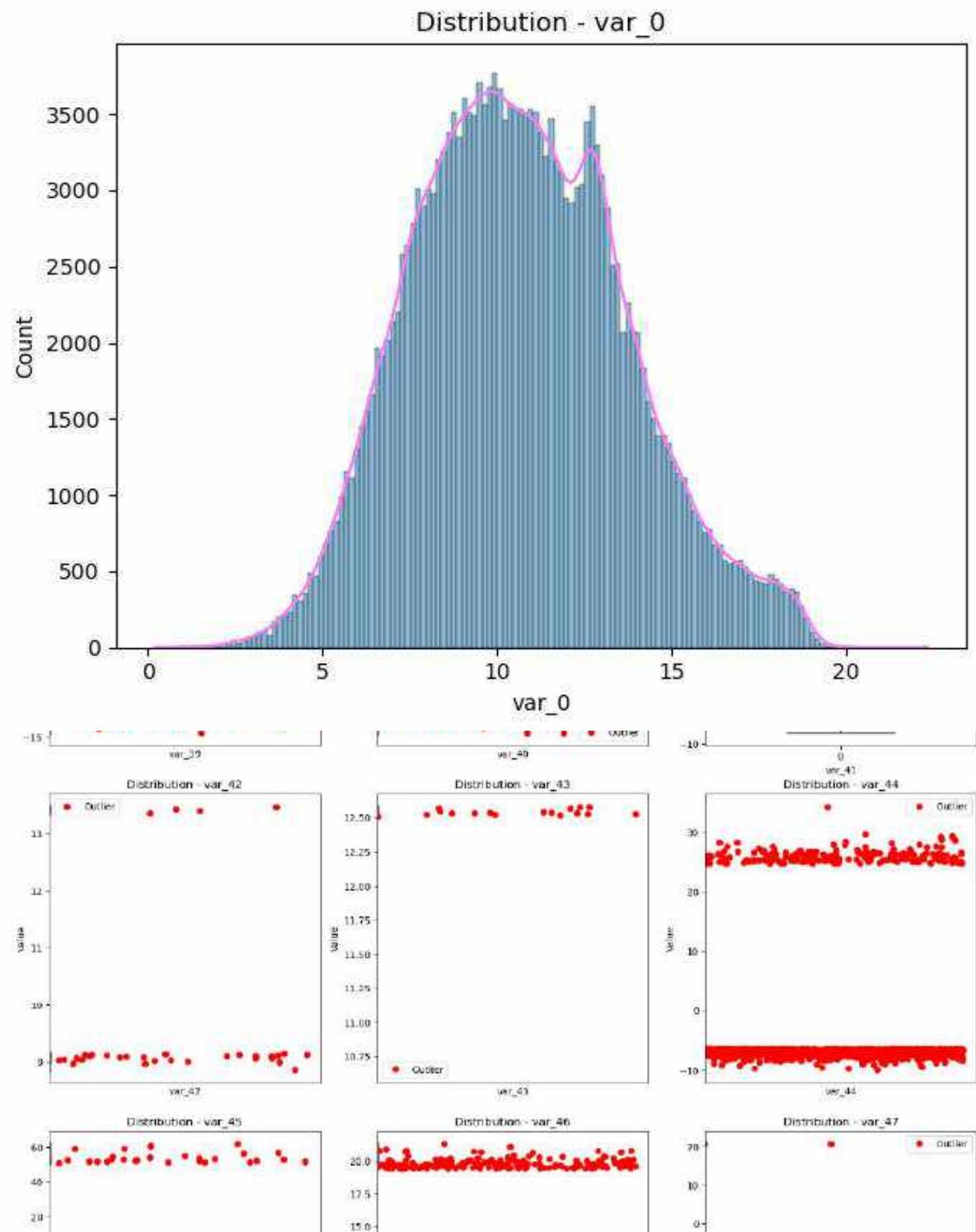
    ax.set_title(f'Distribution - {column}')
    ax.set_xlabel(column)
    ax.set_ylabel('Value')

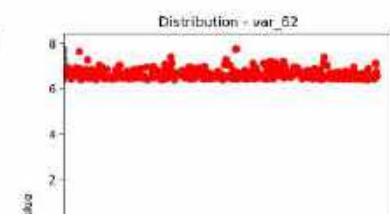
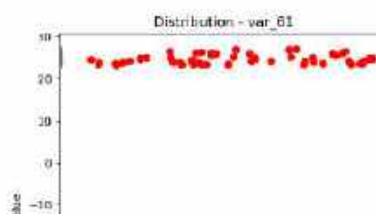
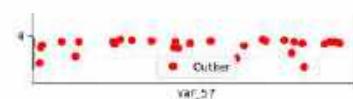
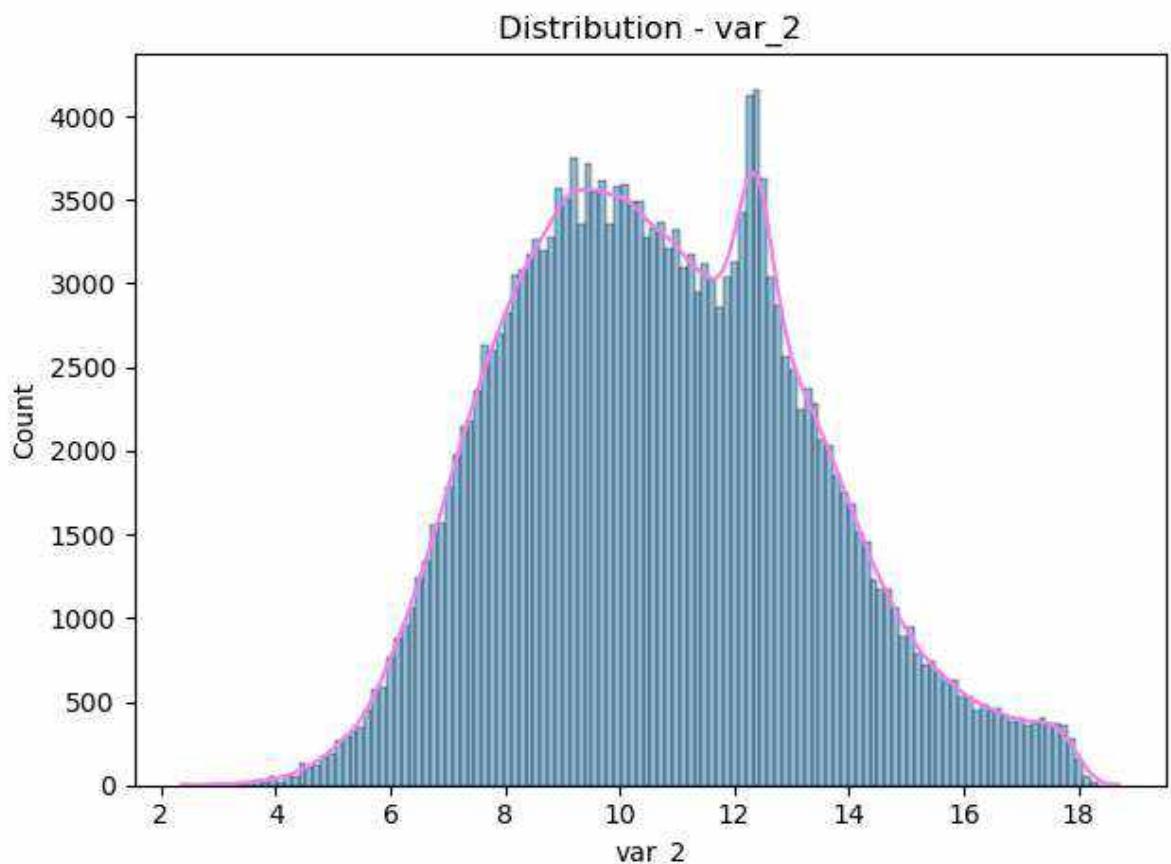
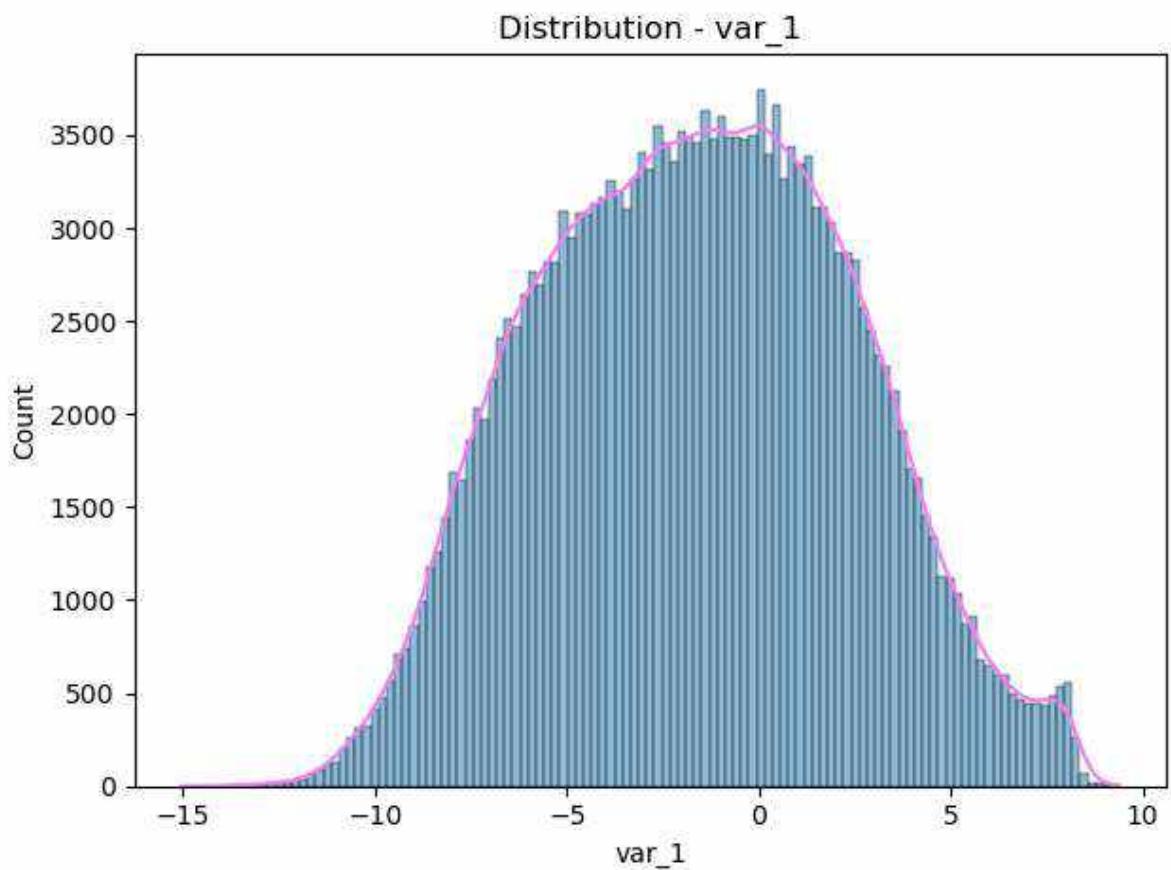
if len(numeric_columns) % num_cols != 0:
    empty_cols = num_cols - len(numeric_columns) % num_cols
    if num_rows > 1:
        for i in range(empty_cols):
            axes[num_rows-1, num_cols-i-1].axis('off')
    else:
        for i in range(empty_cols):
            axes[num_cols-i-1].axis('off')

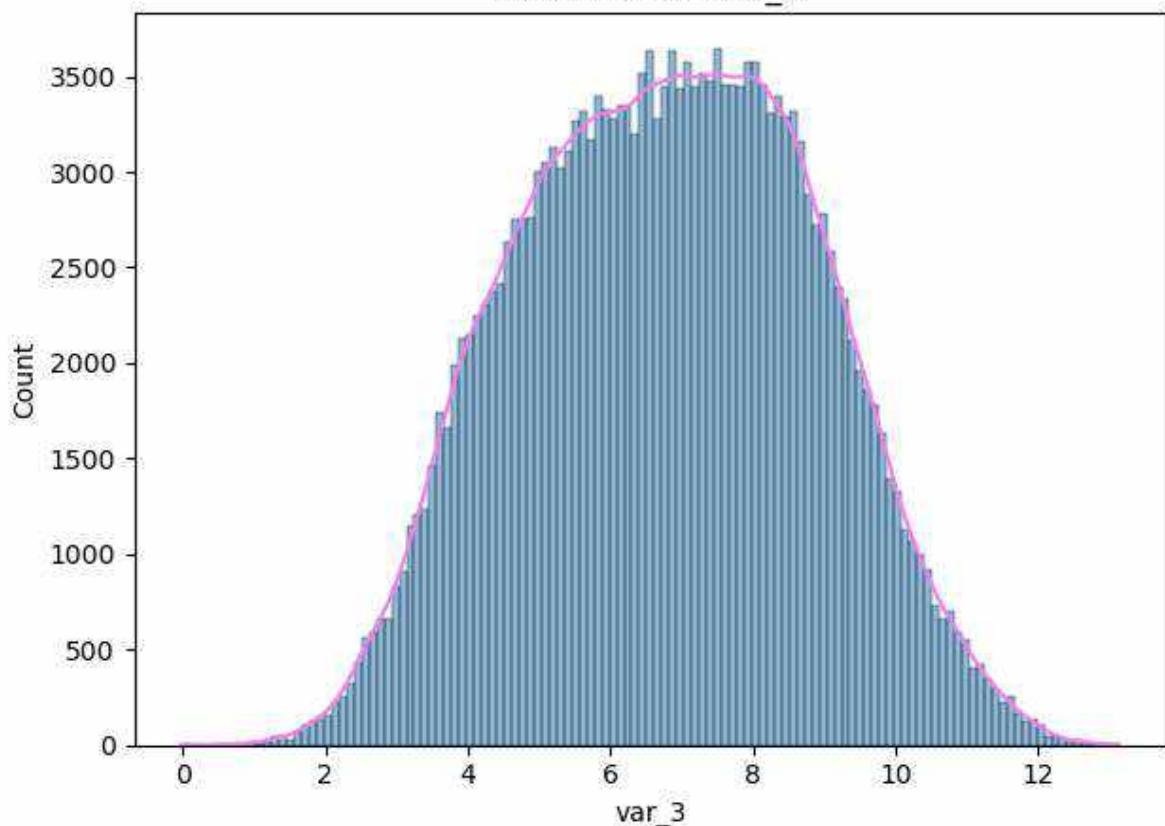
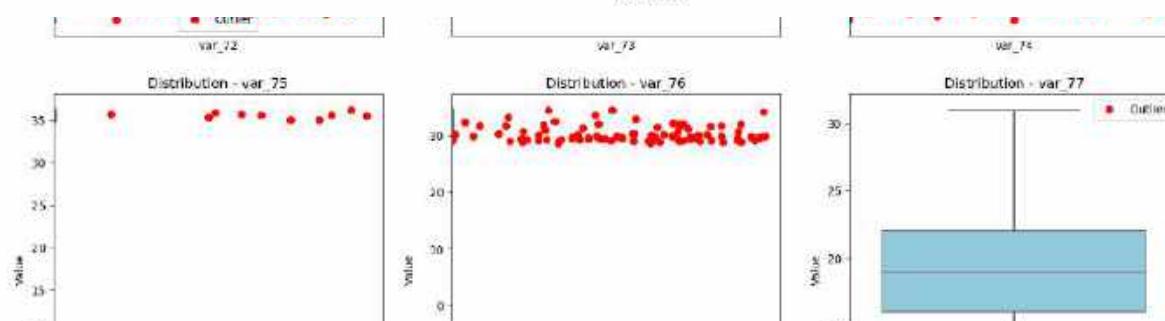
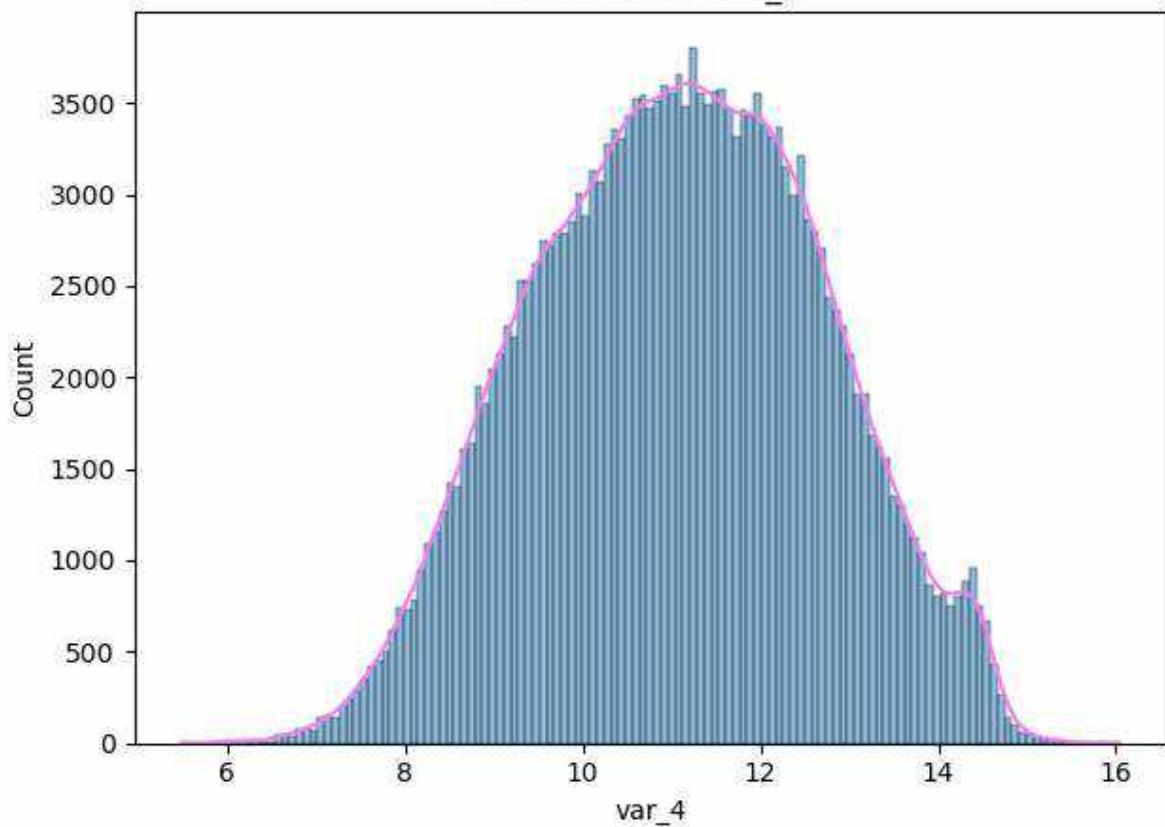
plt.show()
```

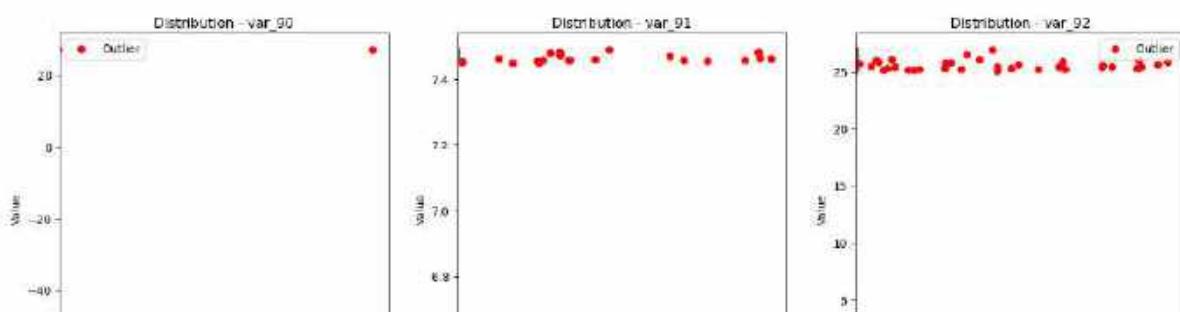
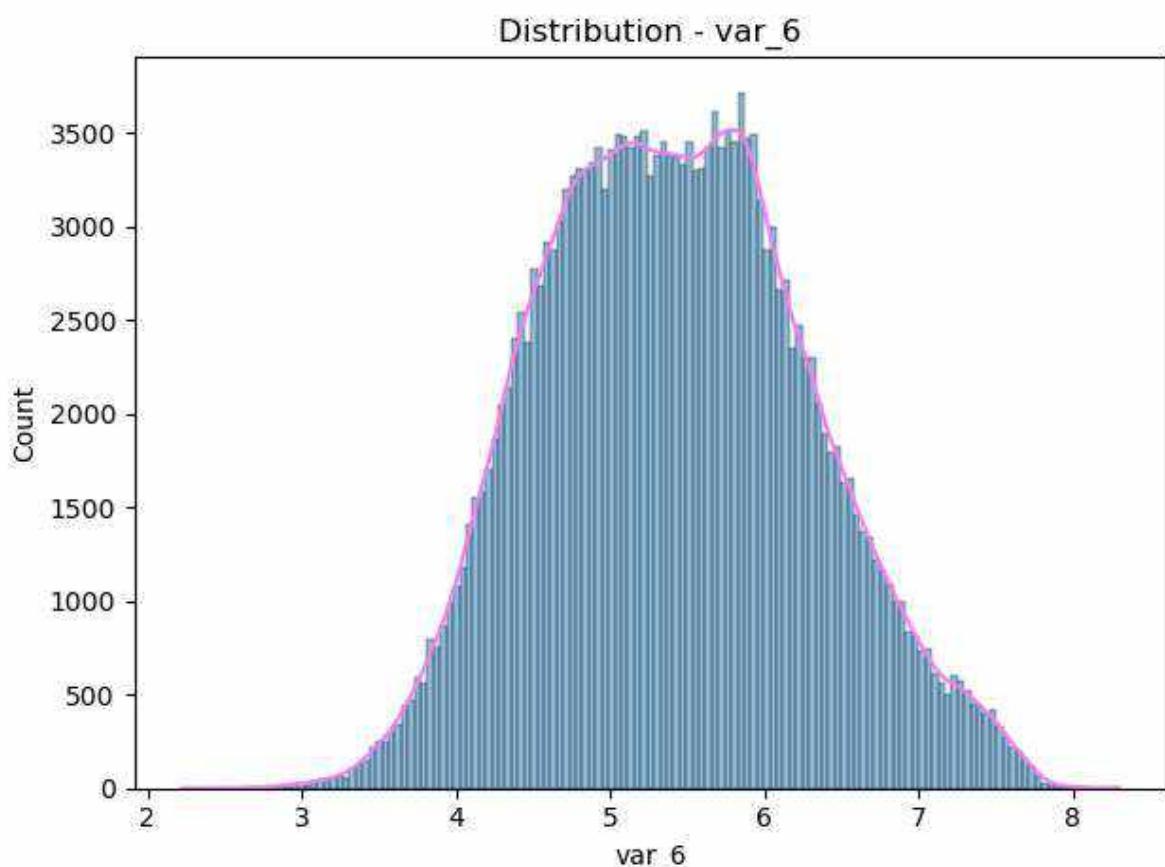
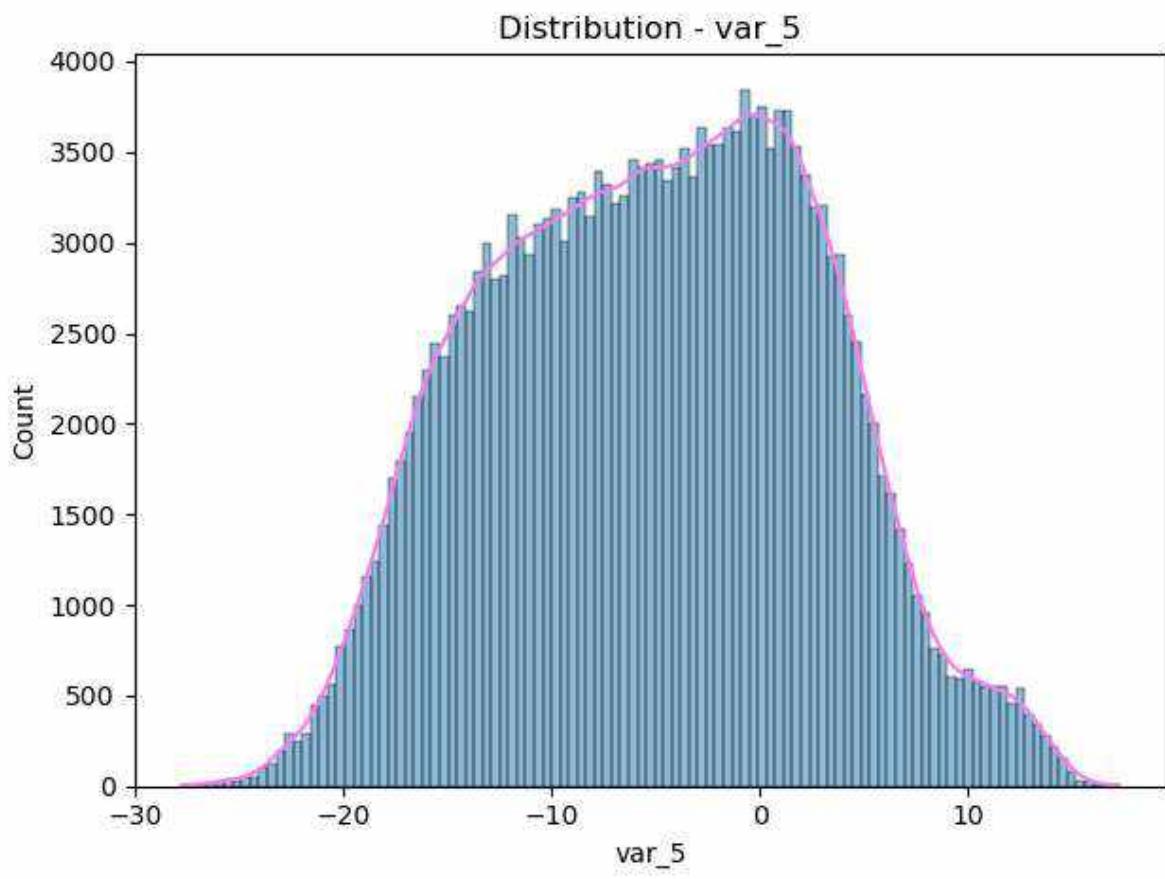


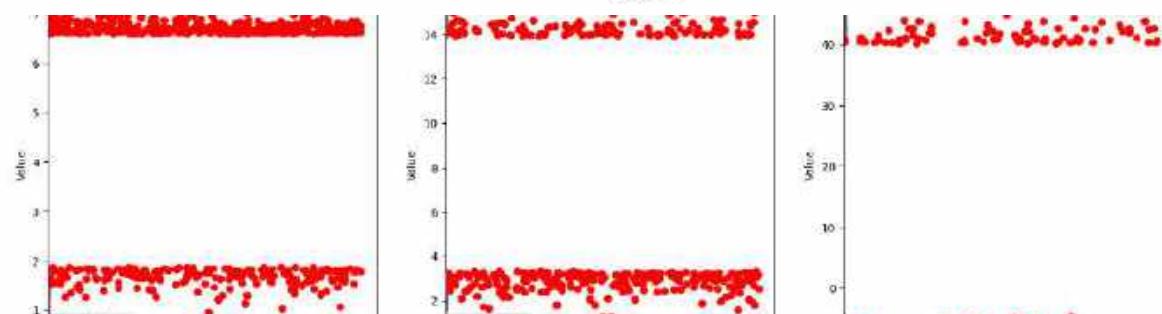
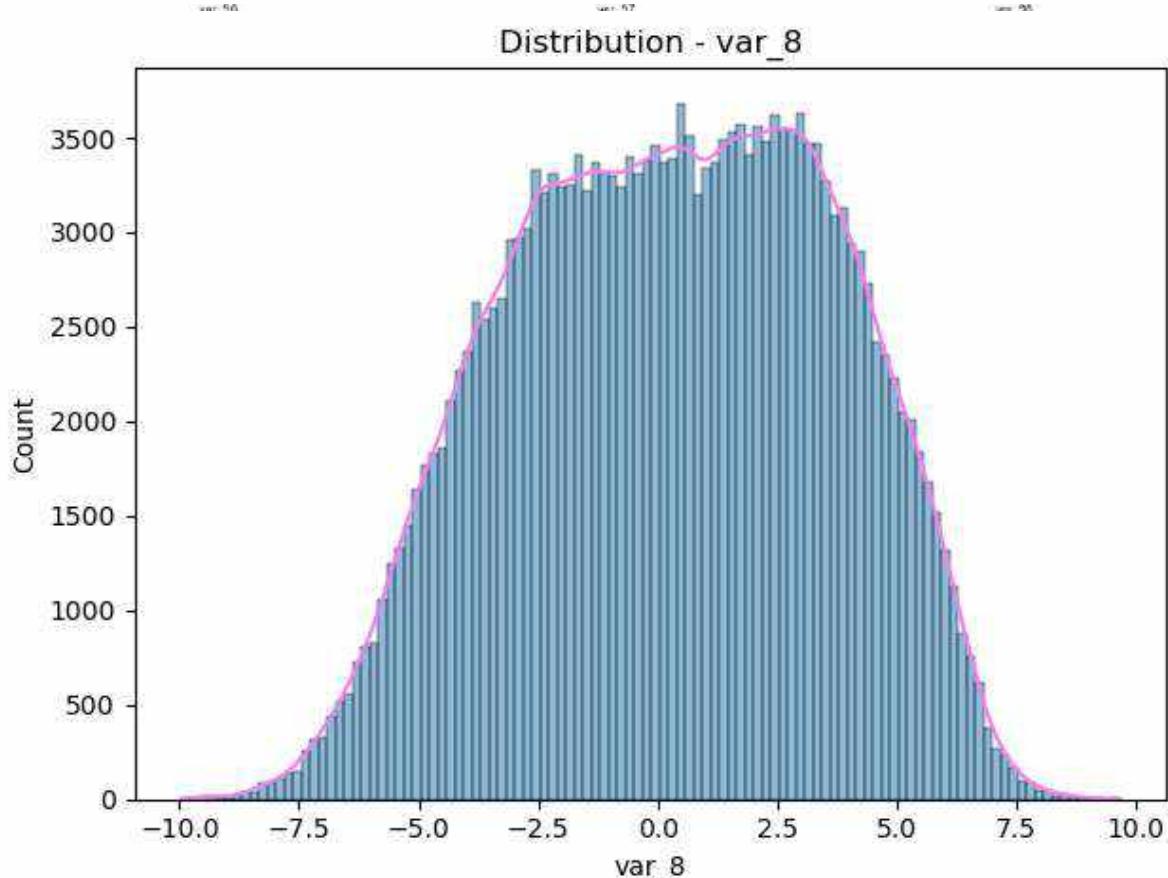
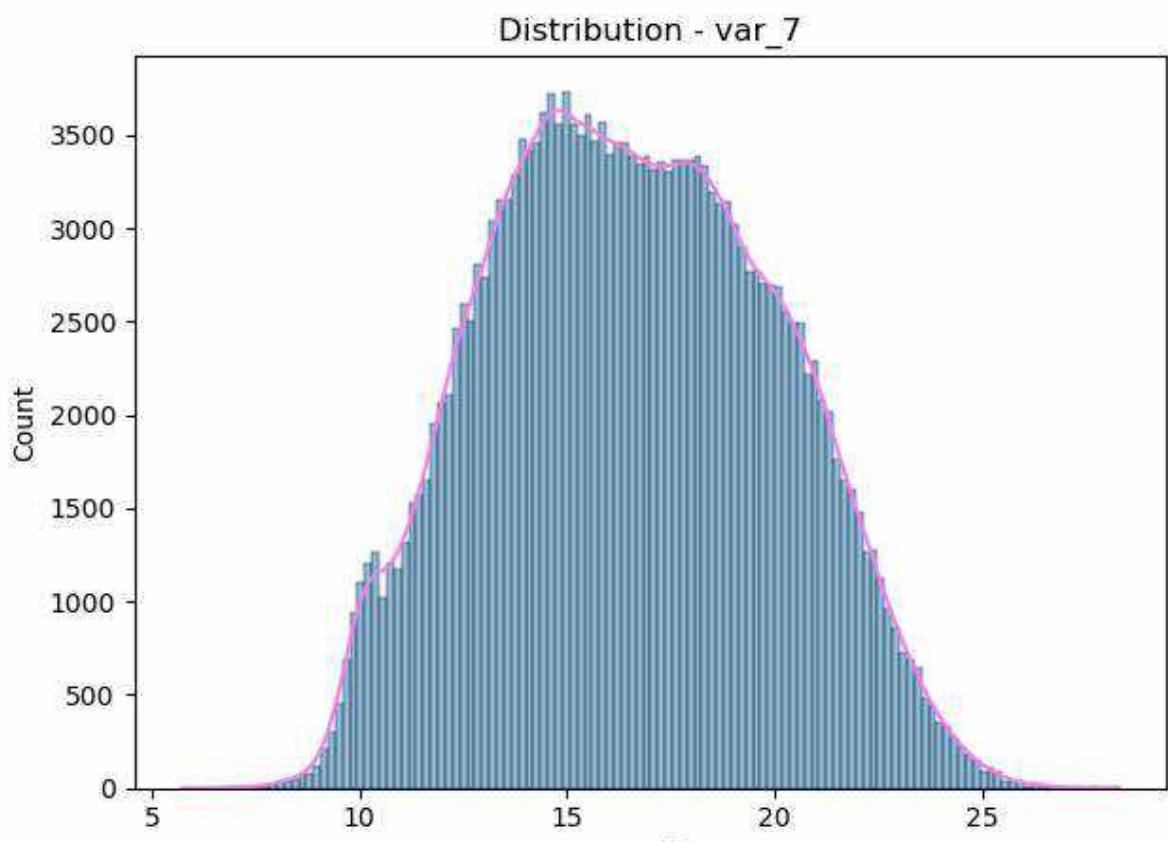


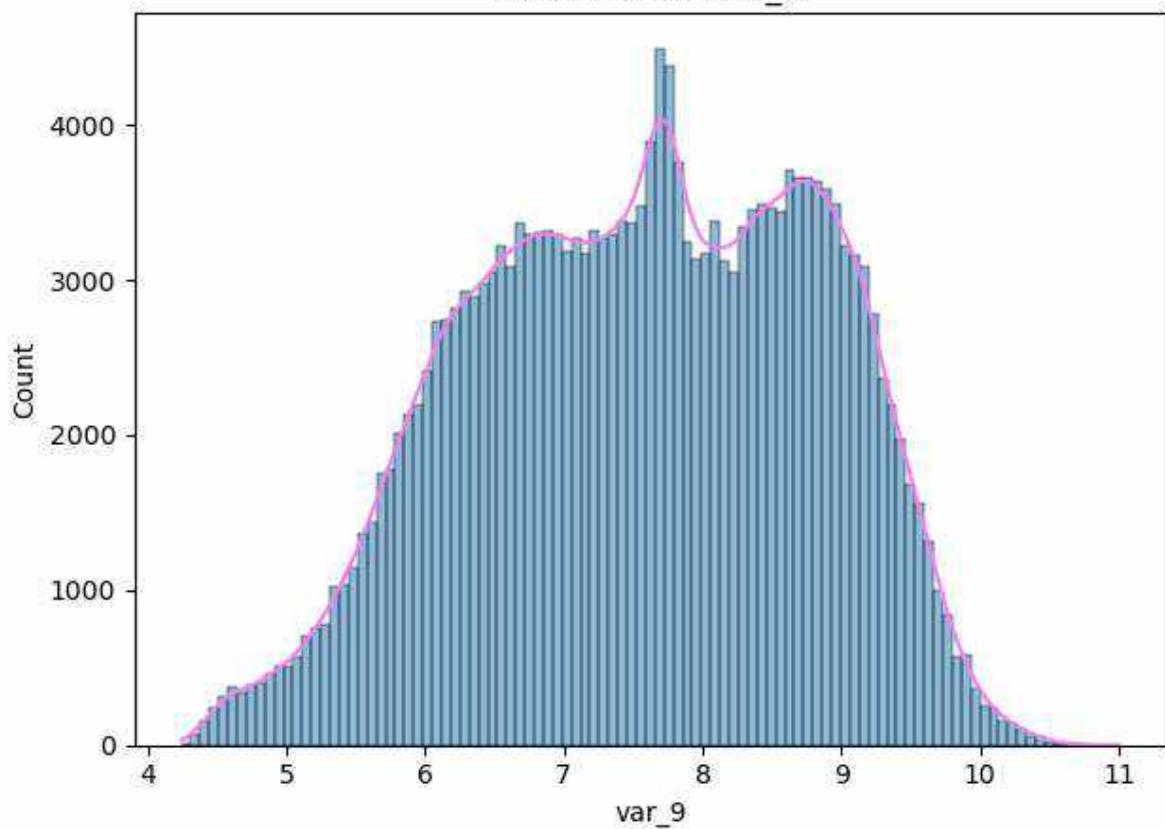
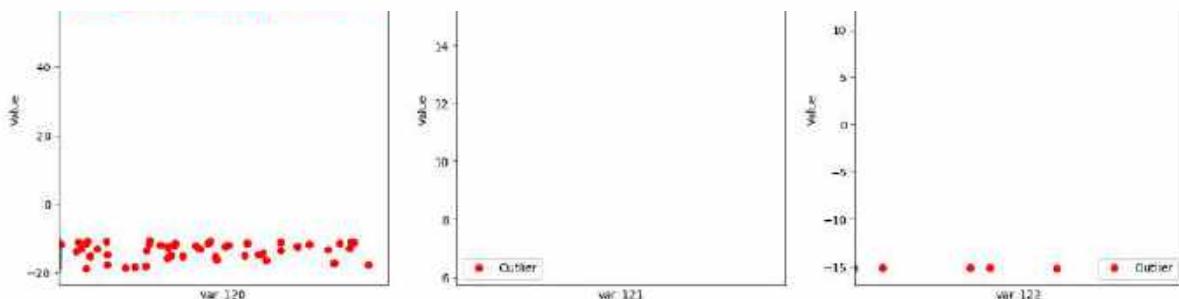
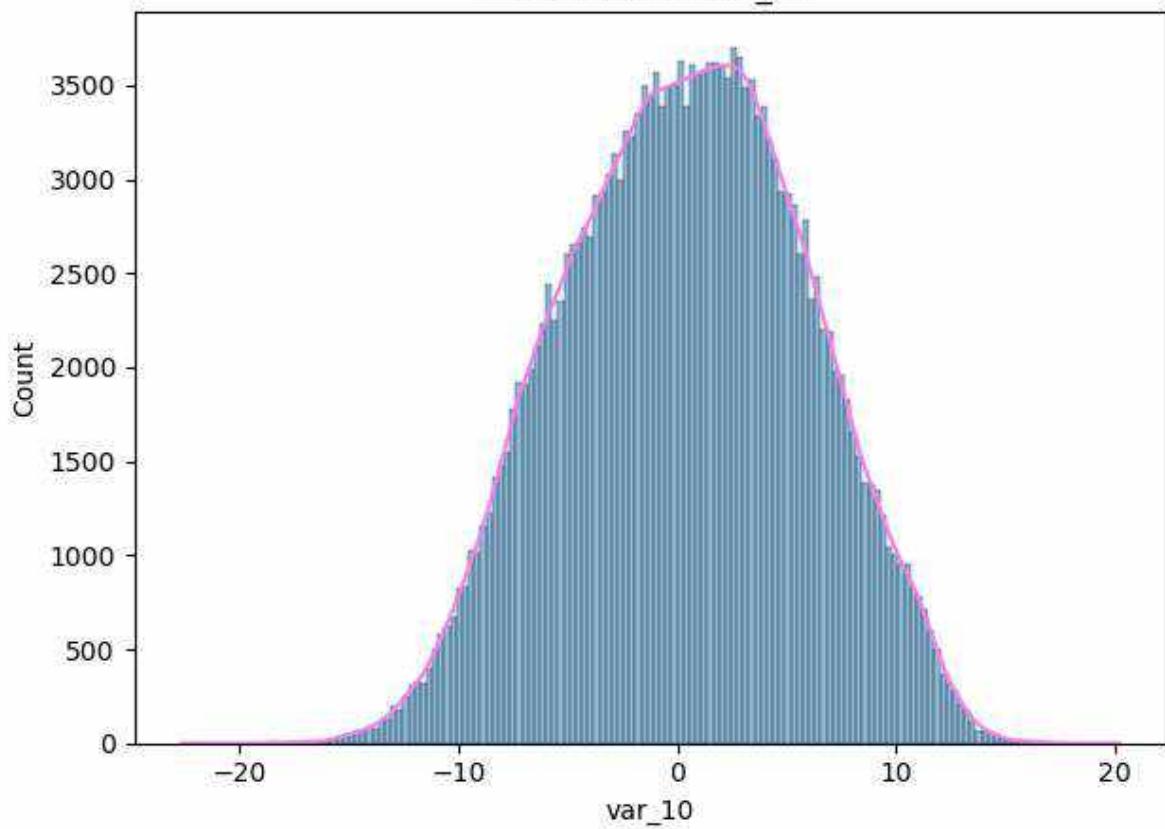


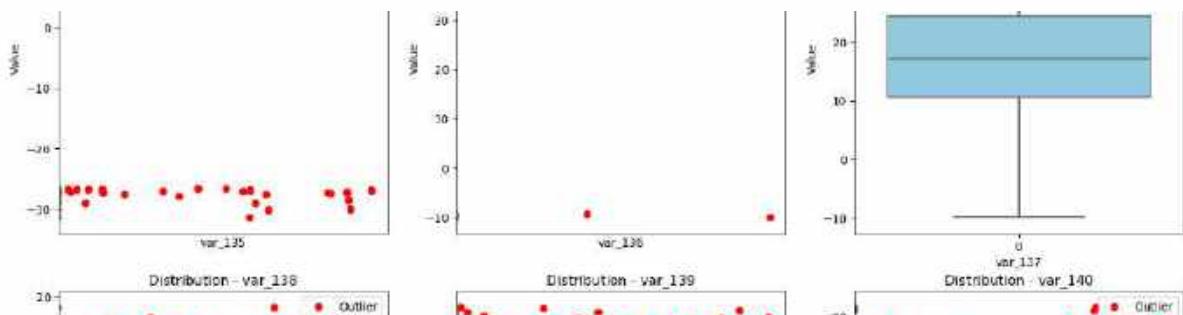
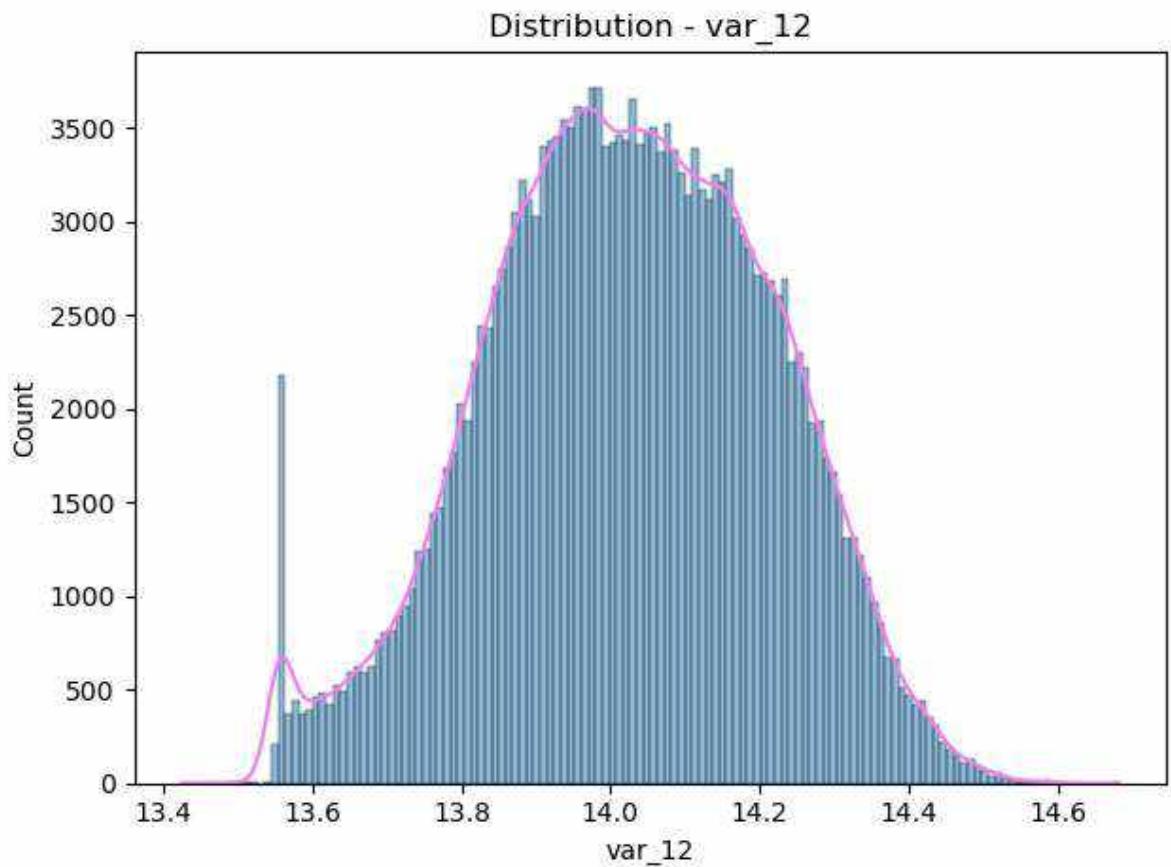
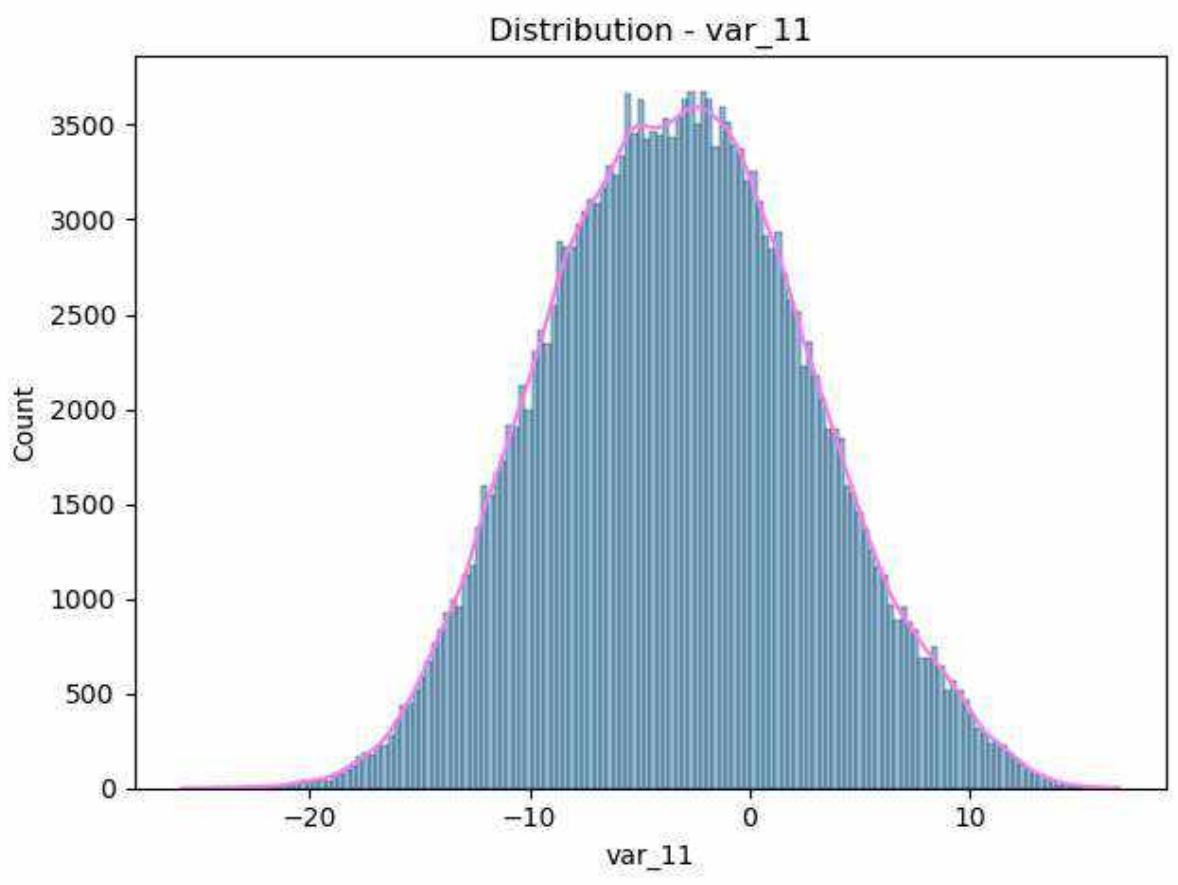


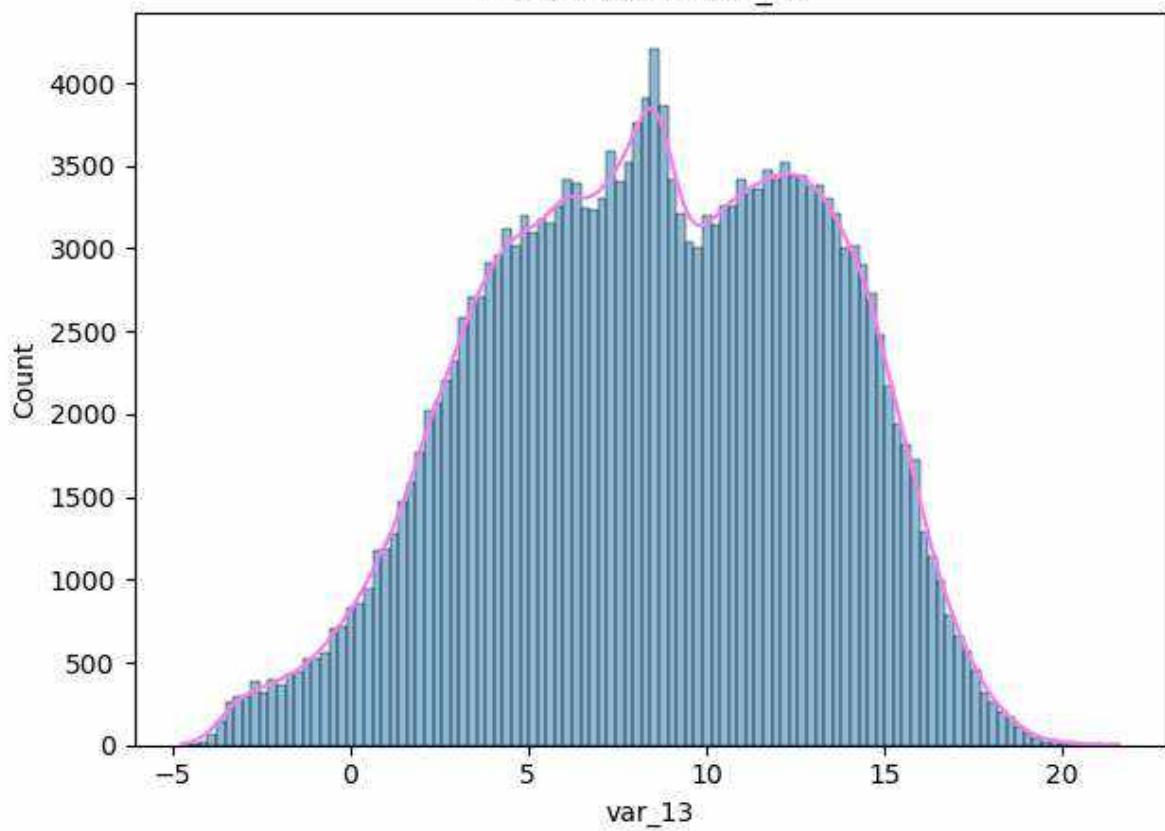
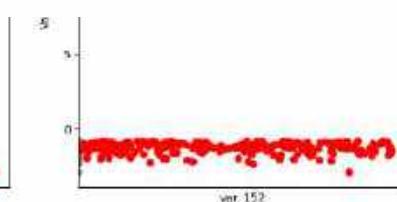
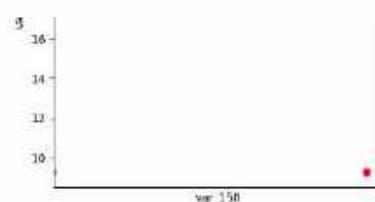
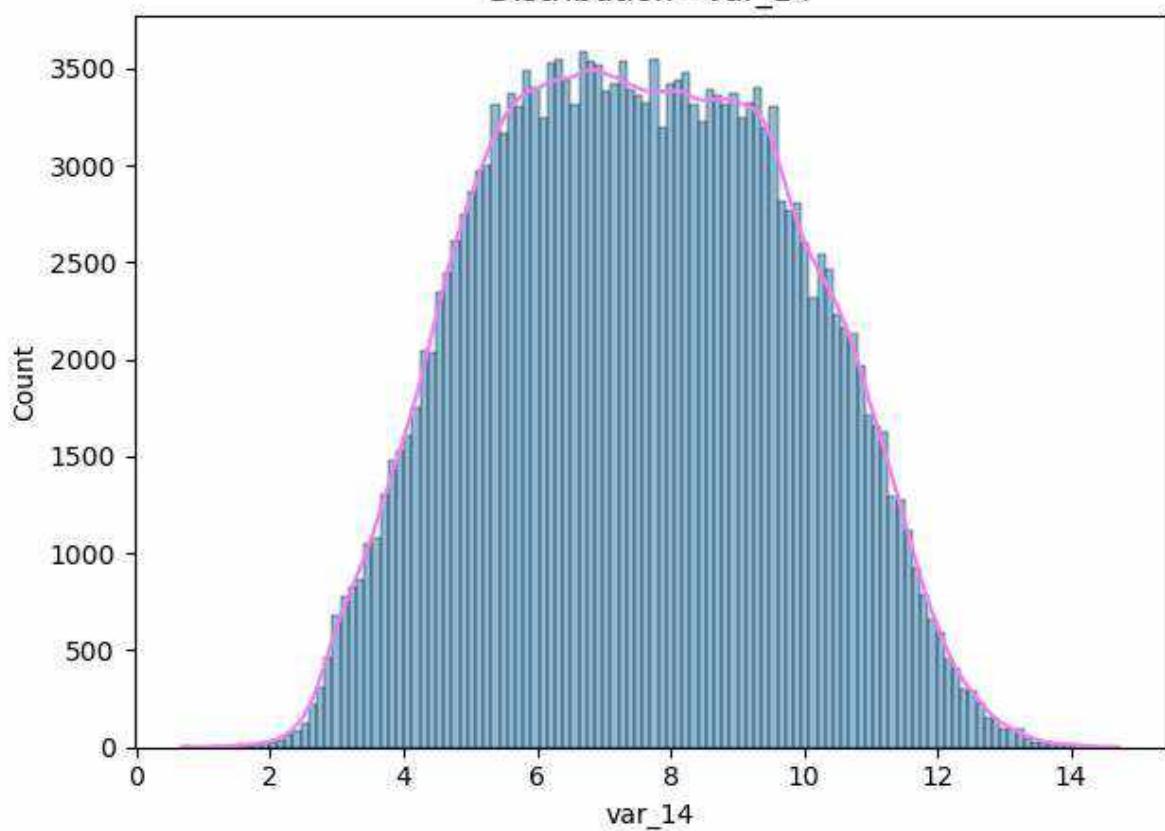
**Distribution - var\_3****Distribution - var\_4**





**Distribution - var\_9****Distribution - var\_10**

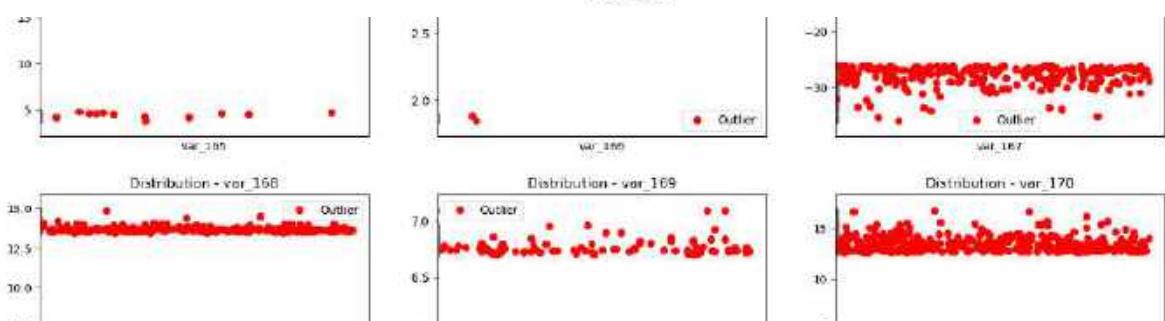
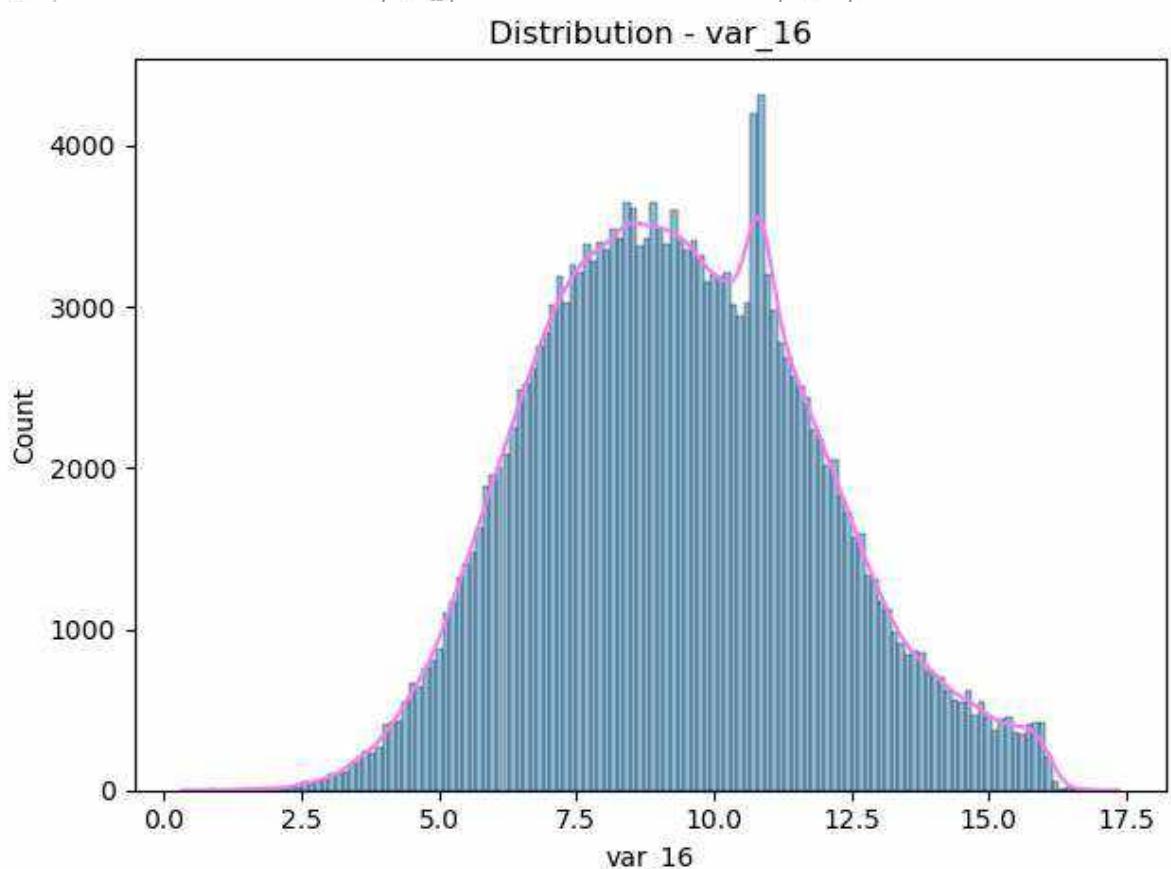
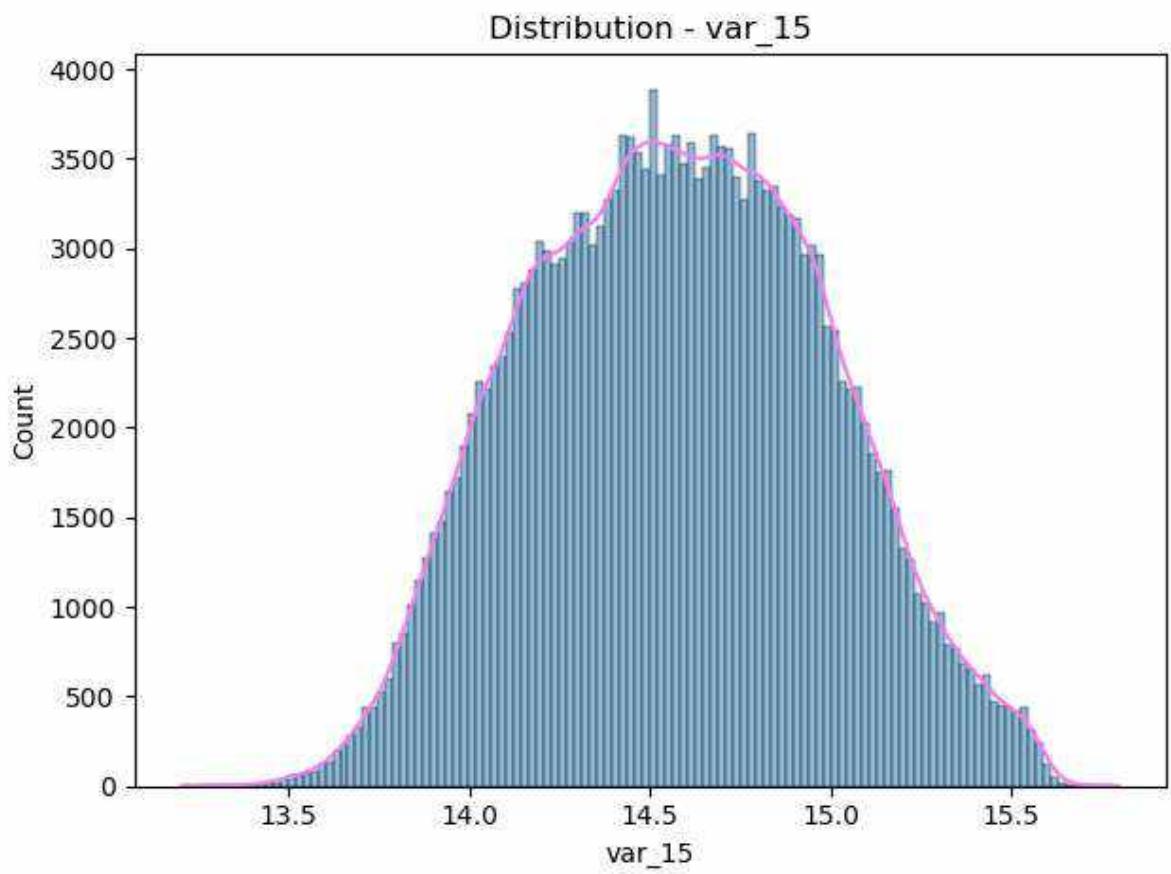


**Distribution - var\_13****Distribution - var\_14**

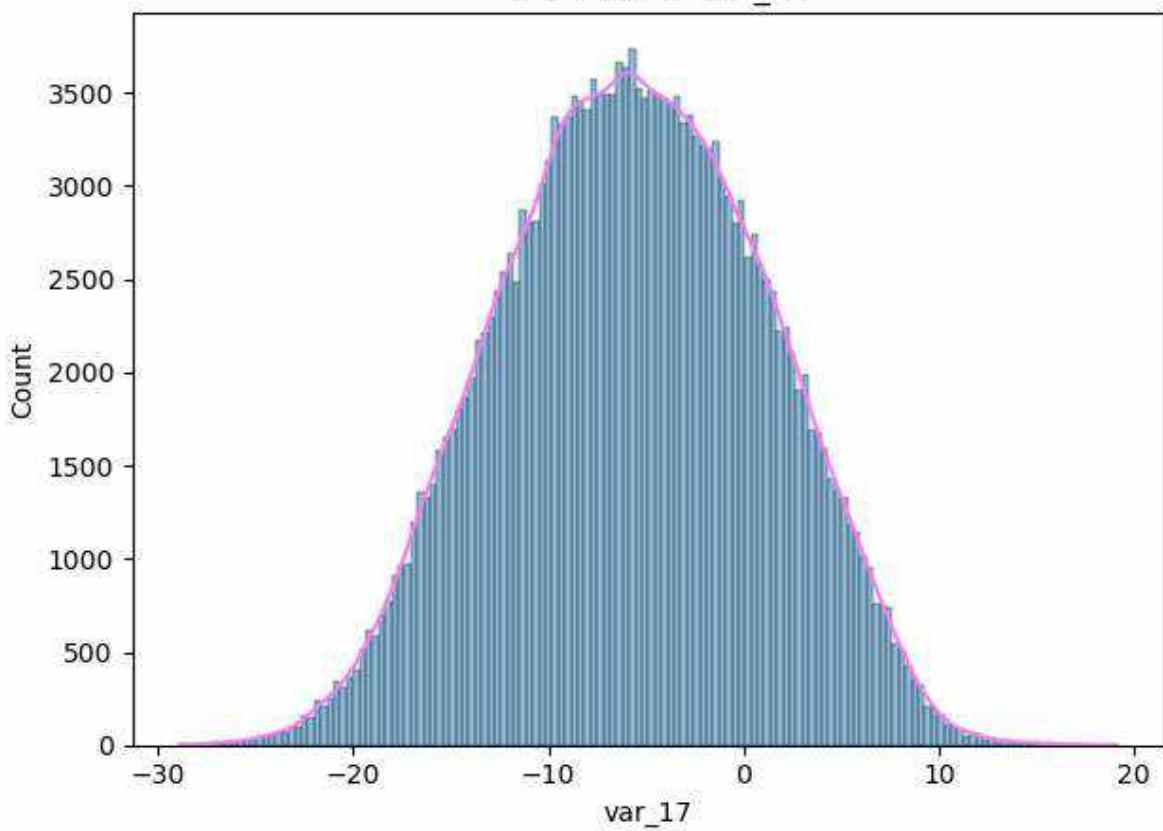
Distribution - var\_153

Distribution - var\_154

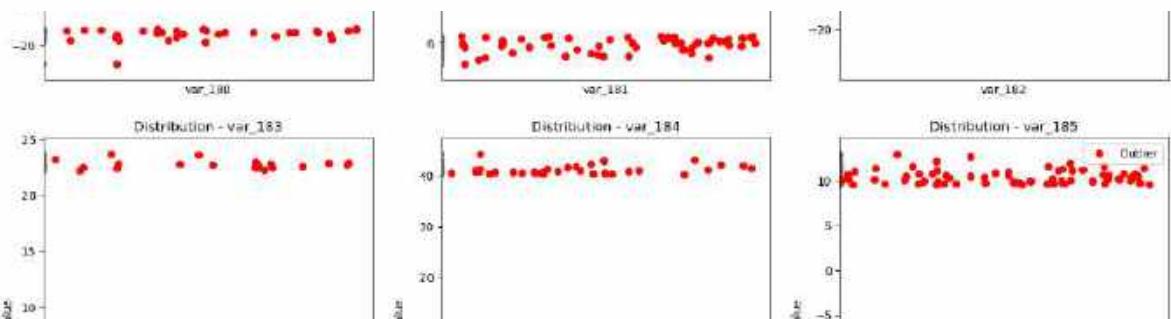
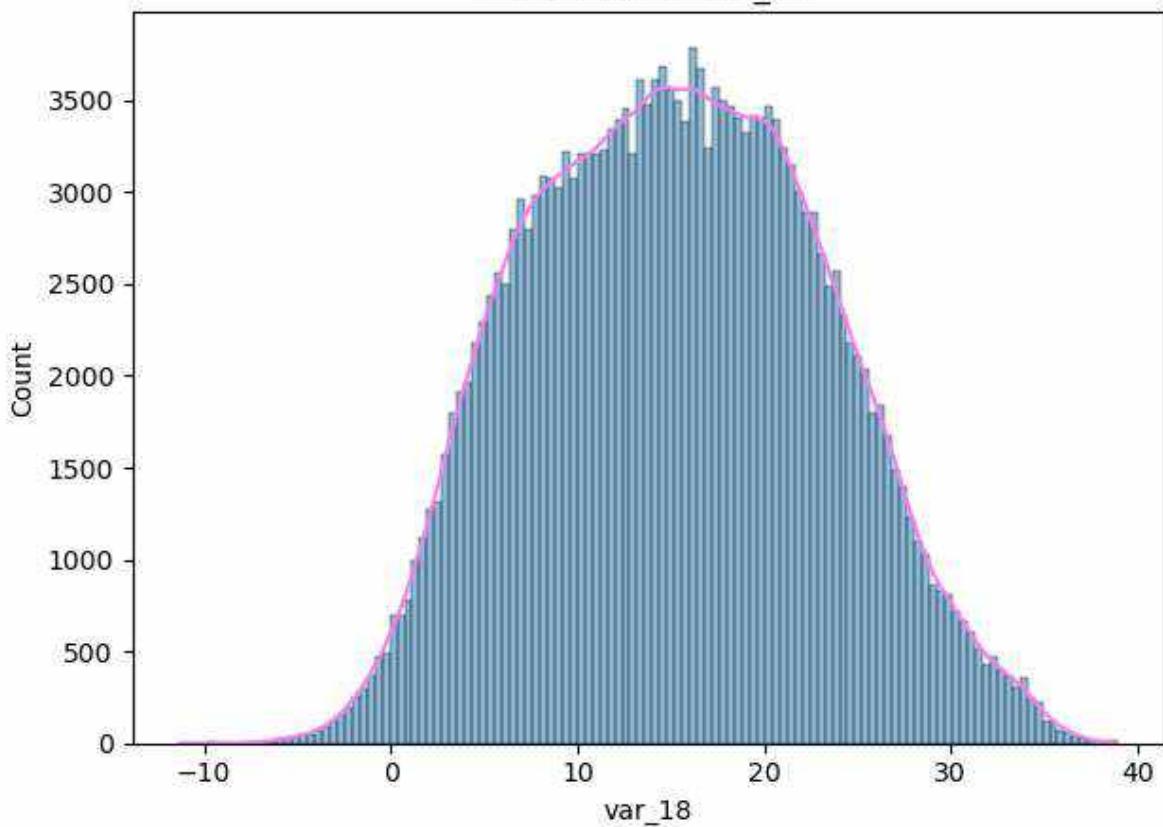
Distribution - var\_155



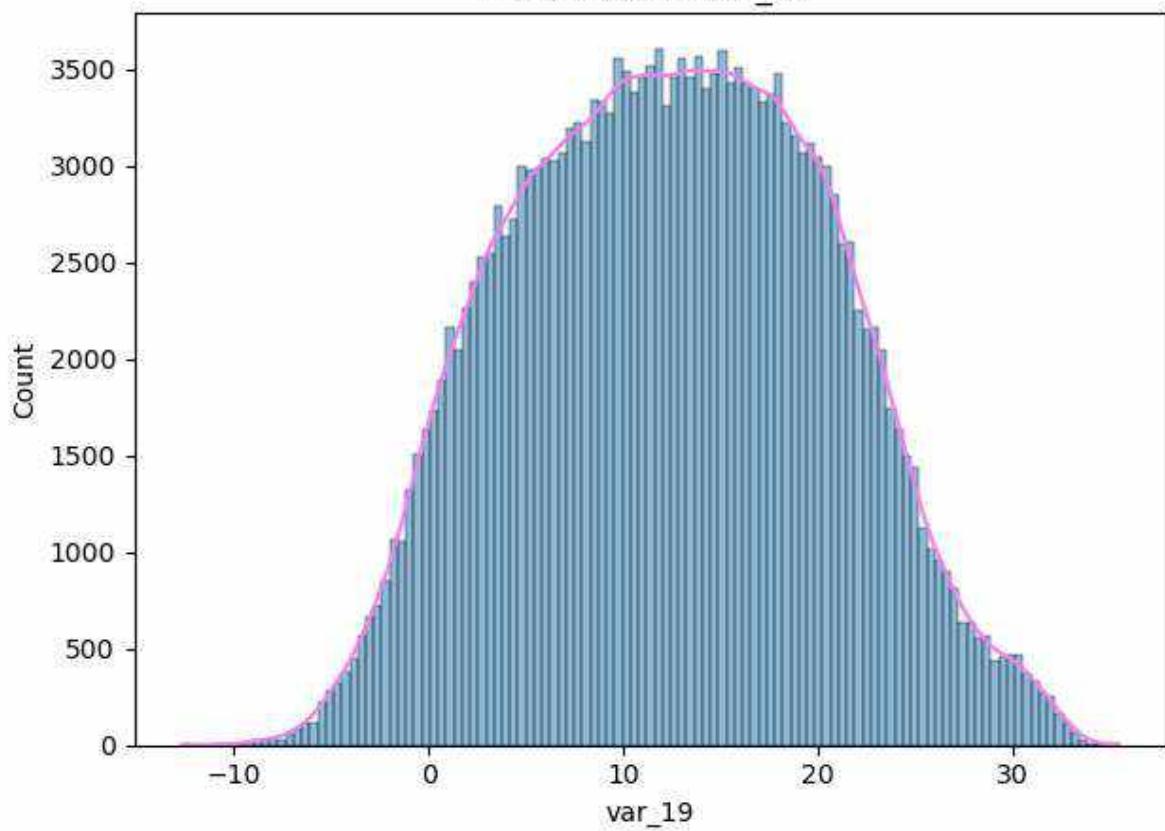
Distribution - var\_17



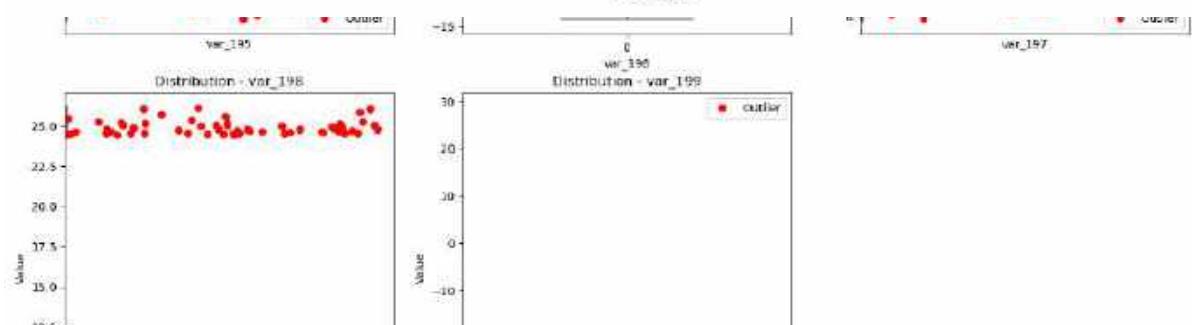
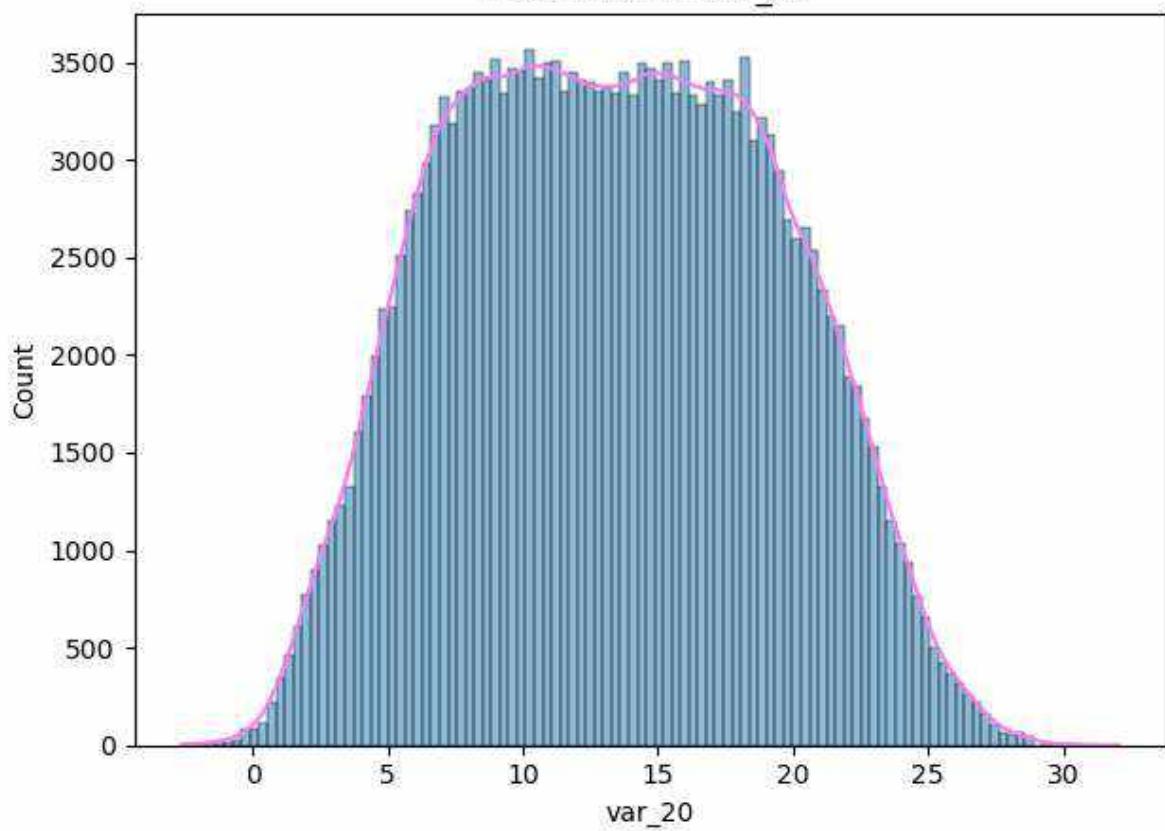
Distribution - var\_18

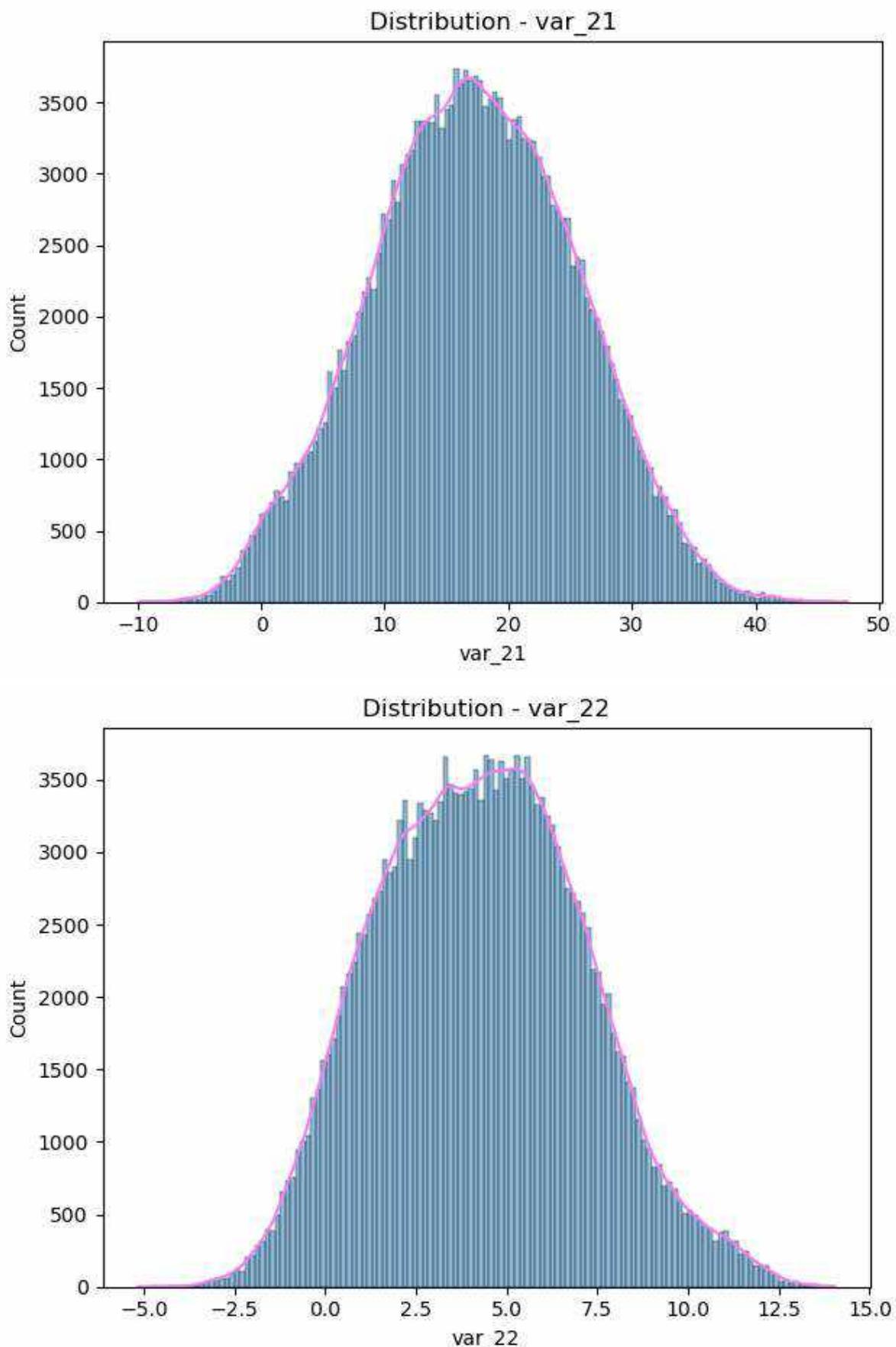


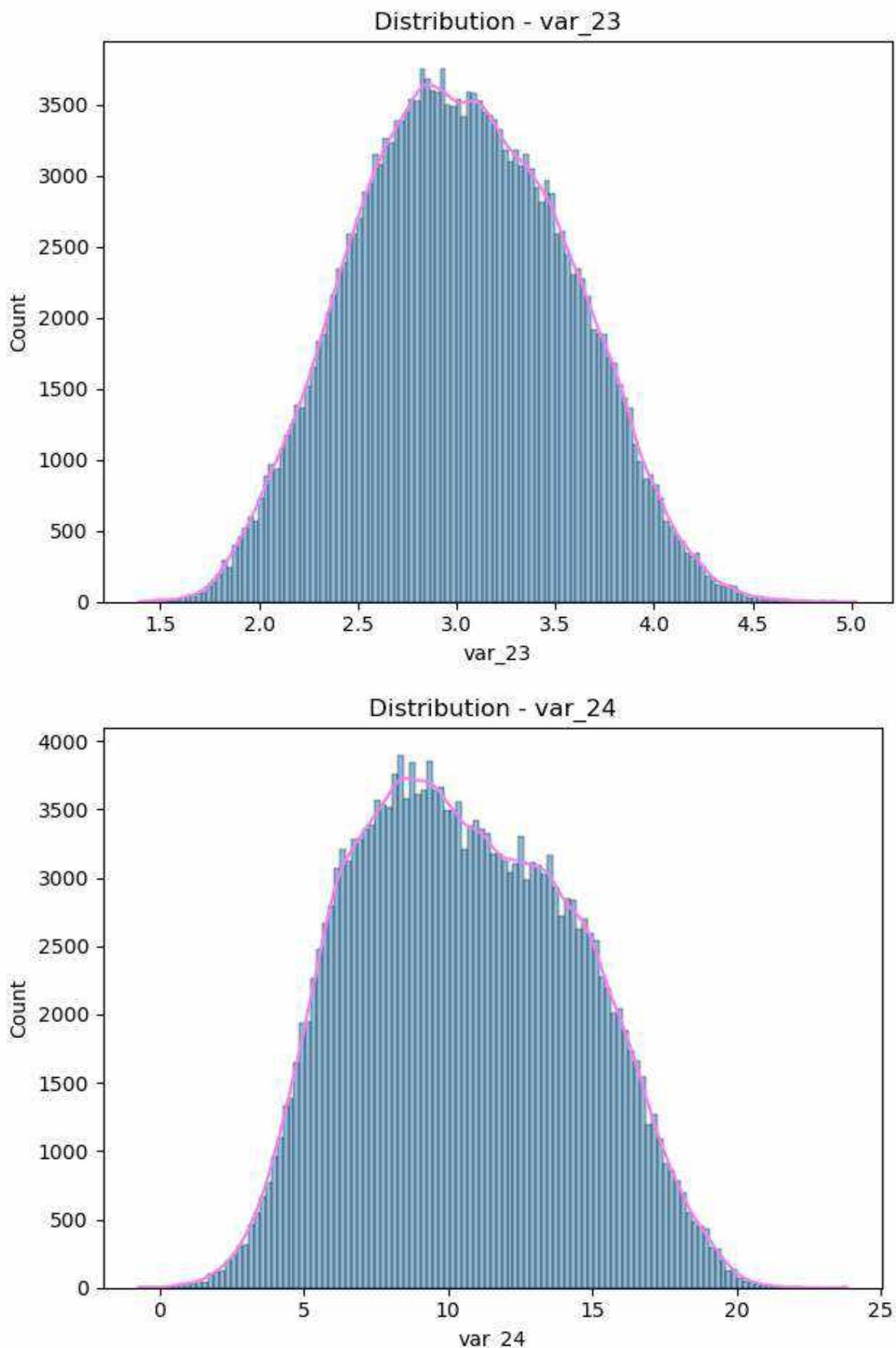
Distribution - var\_19

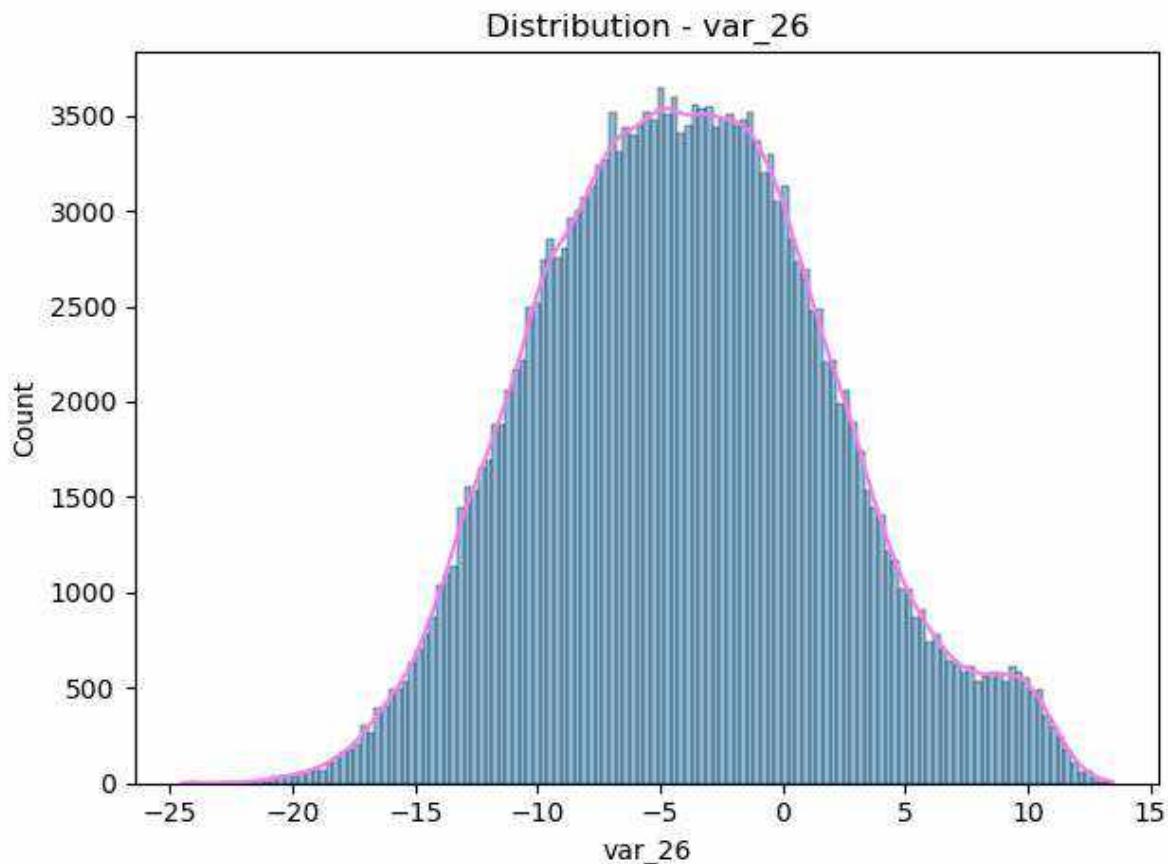
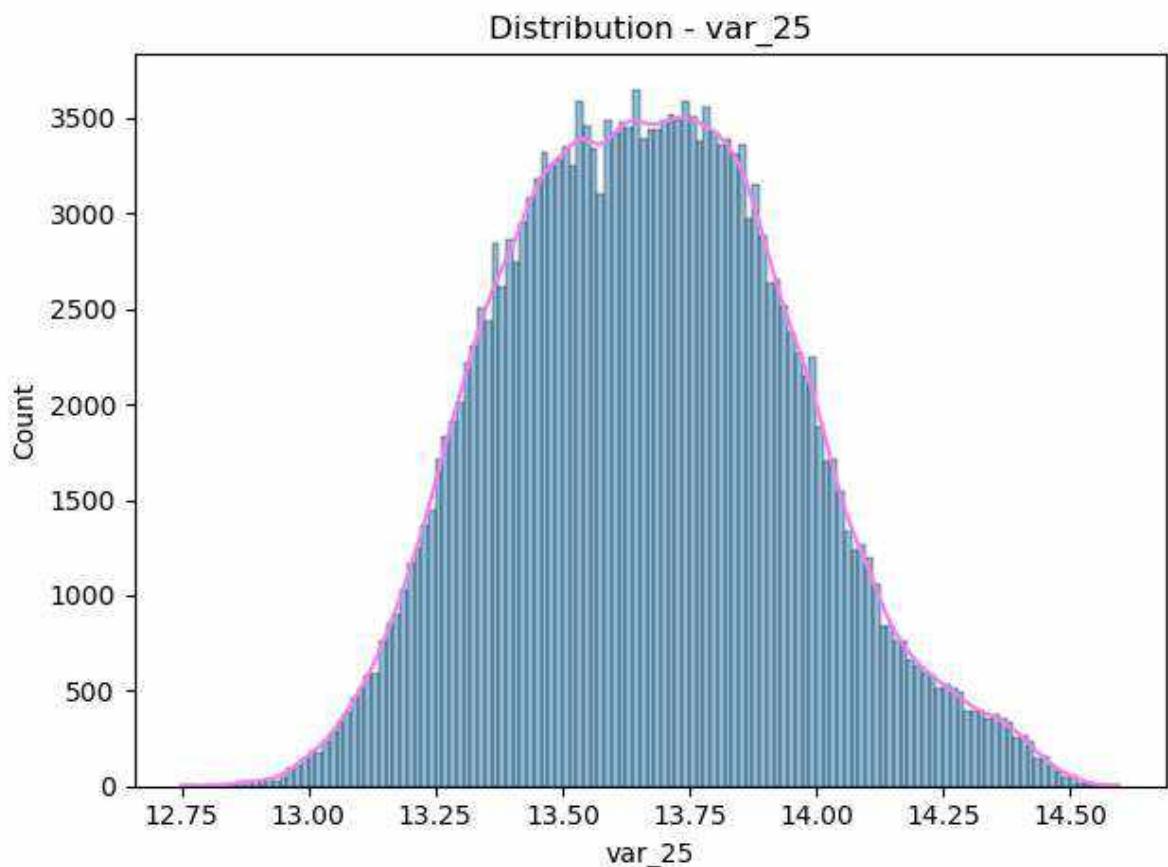


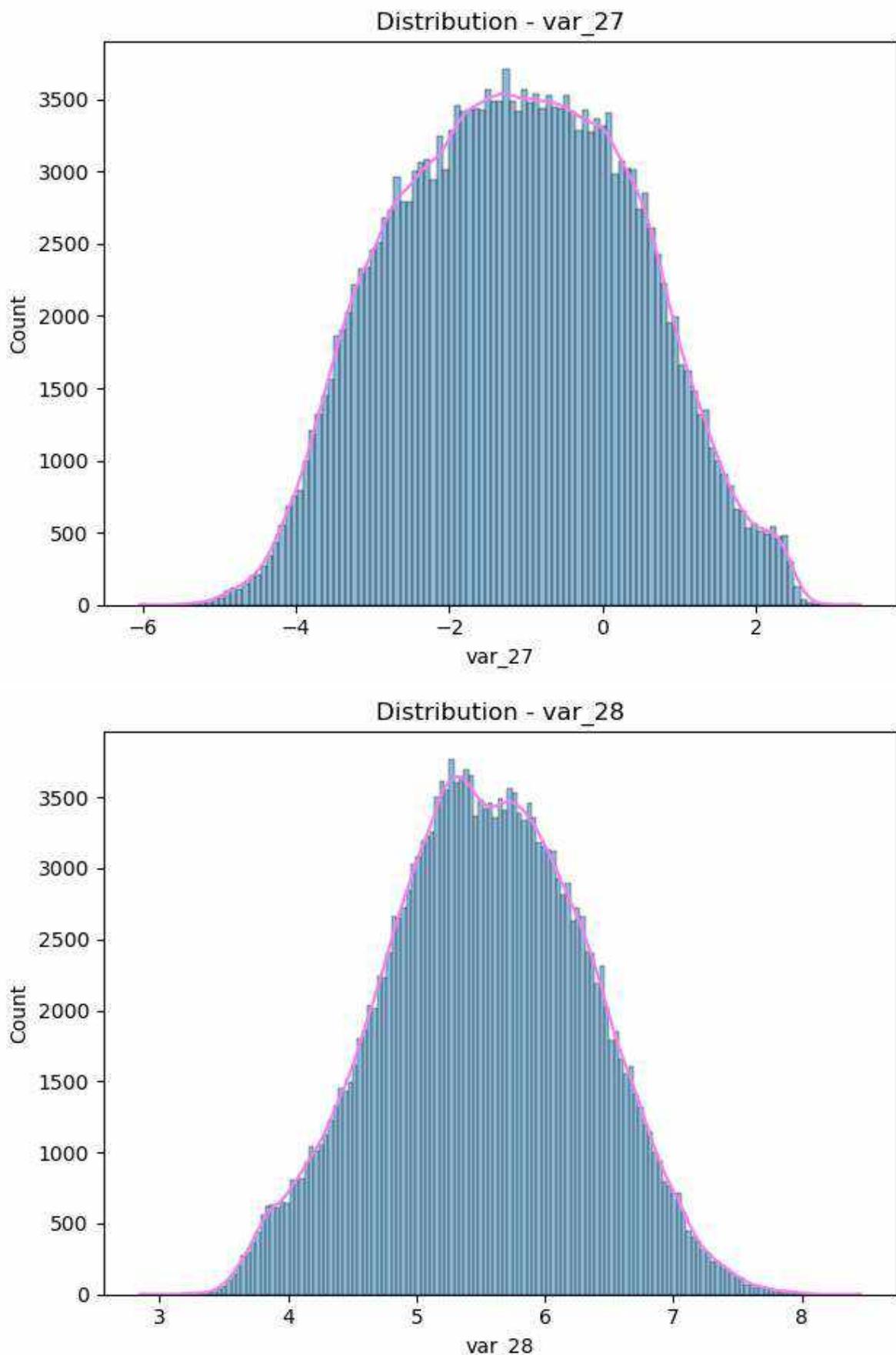
Distribution - var\_20

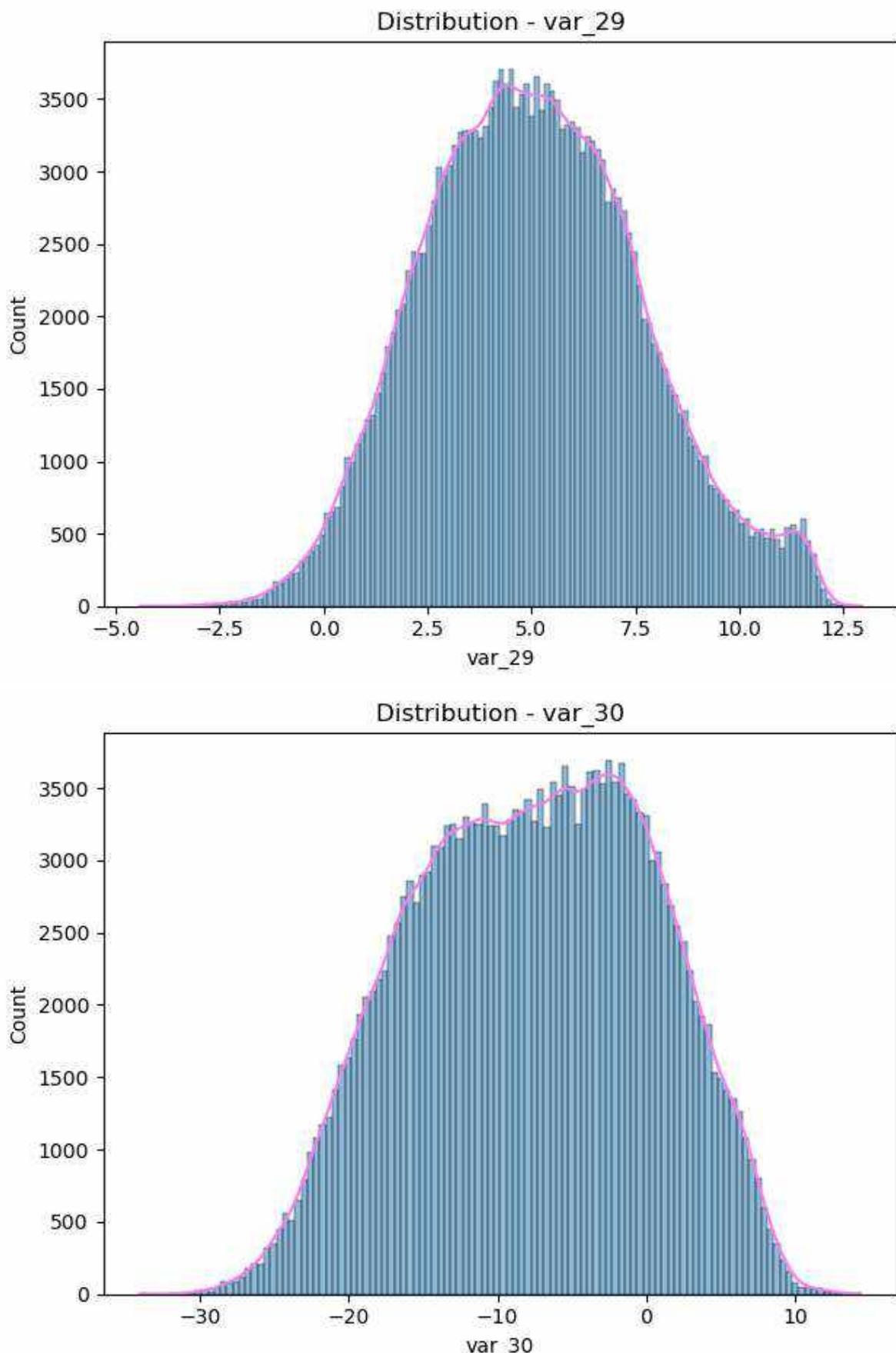


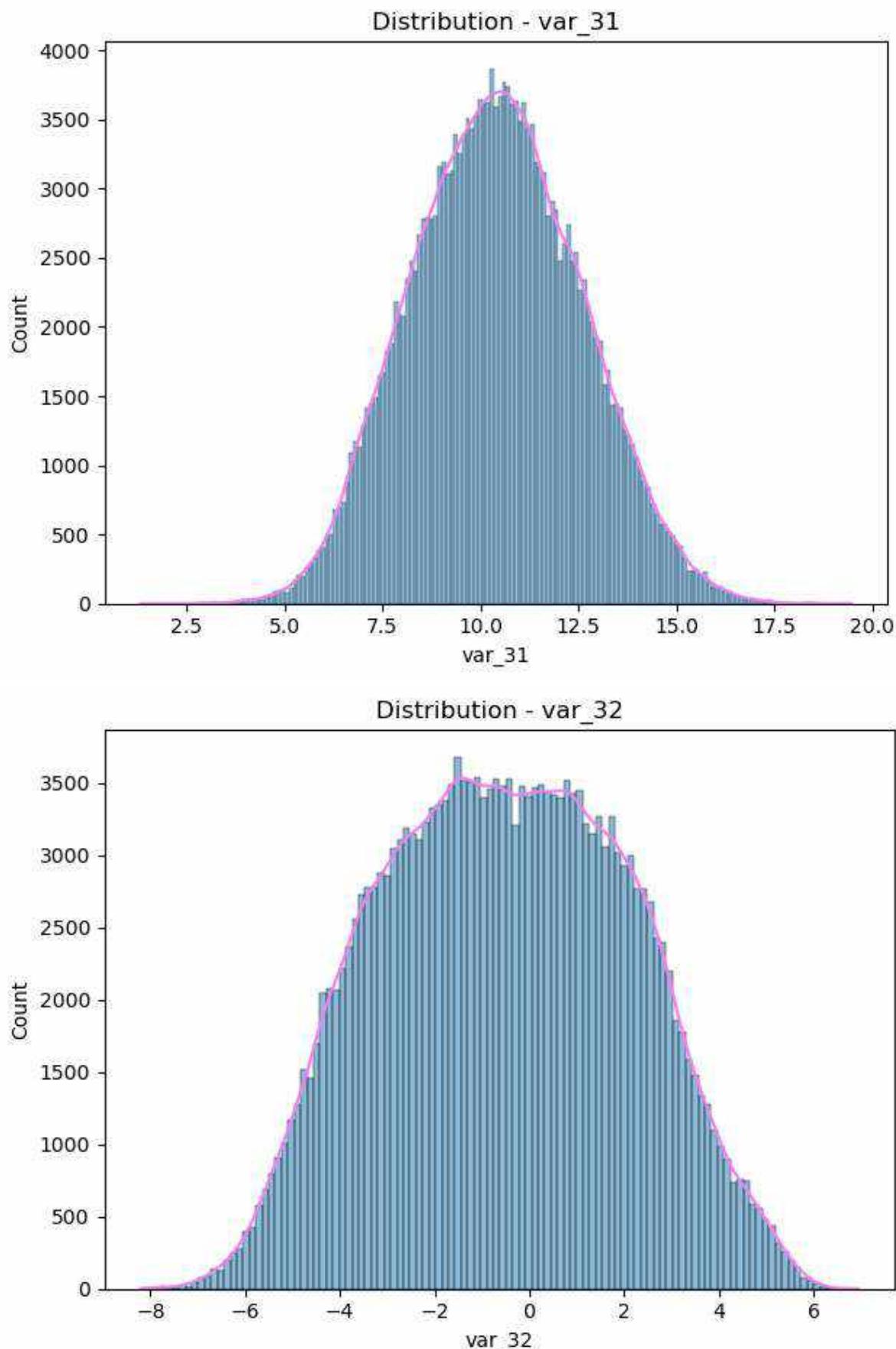


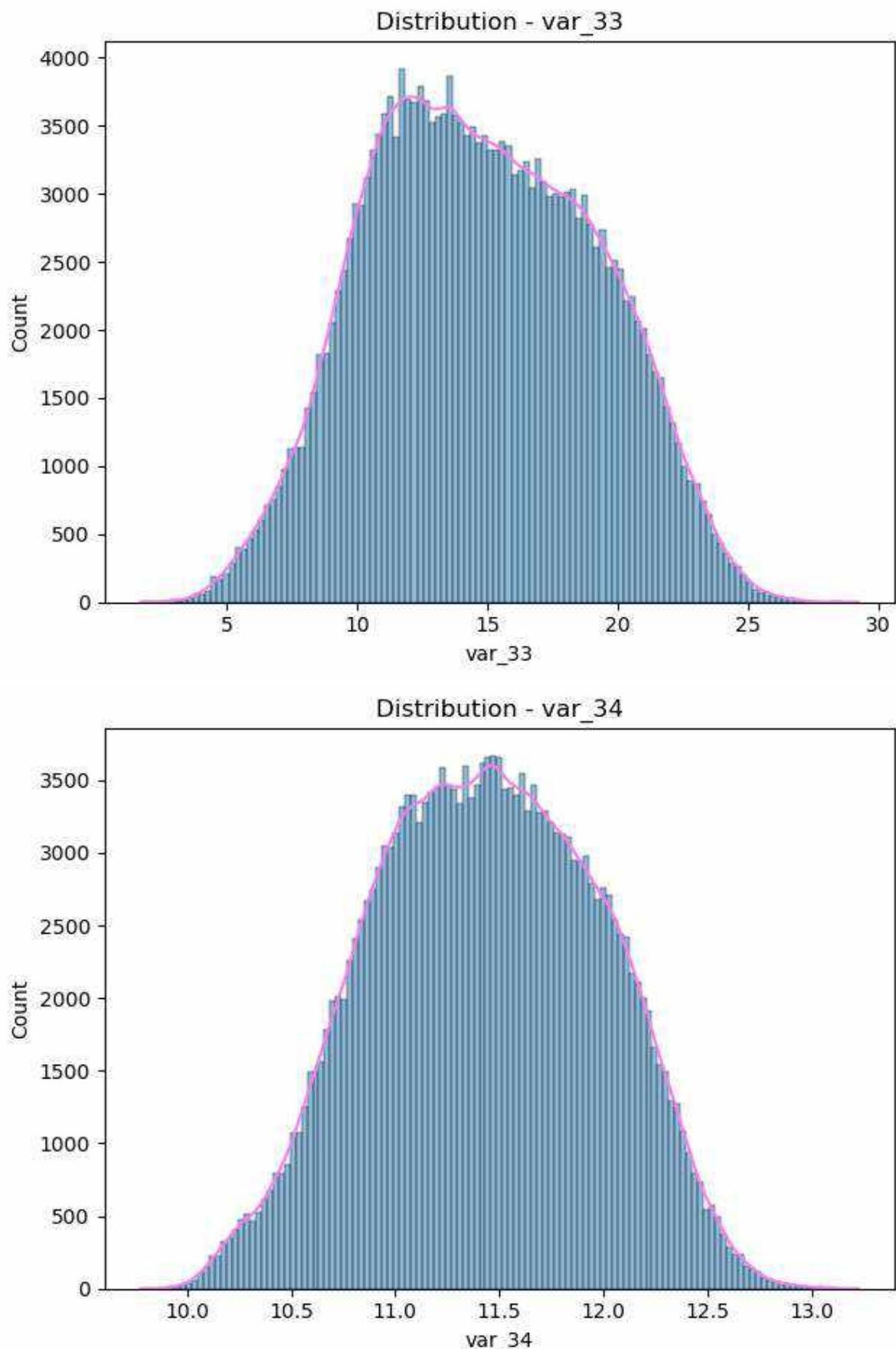


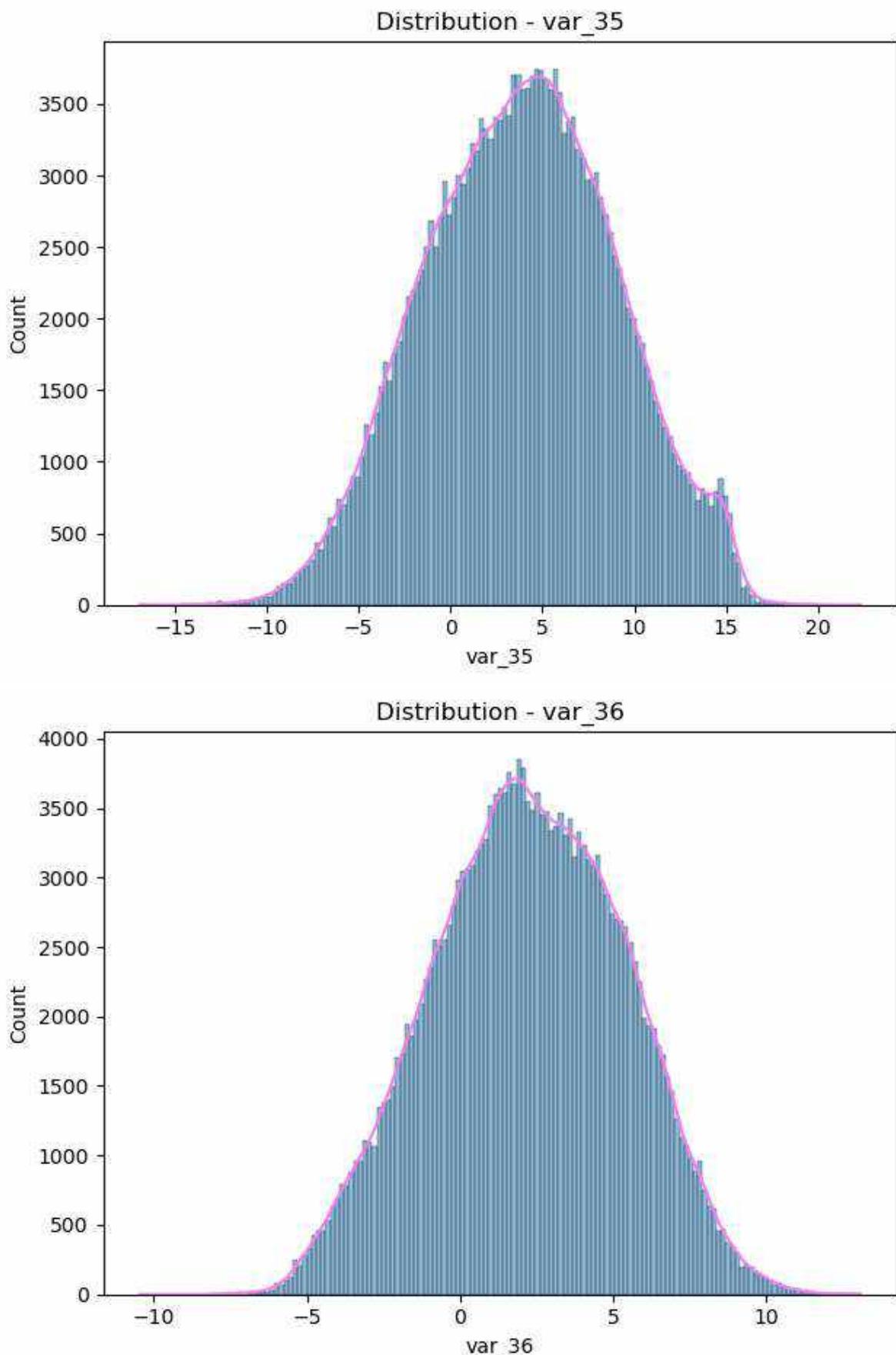


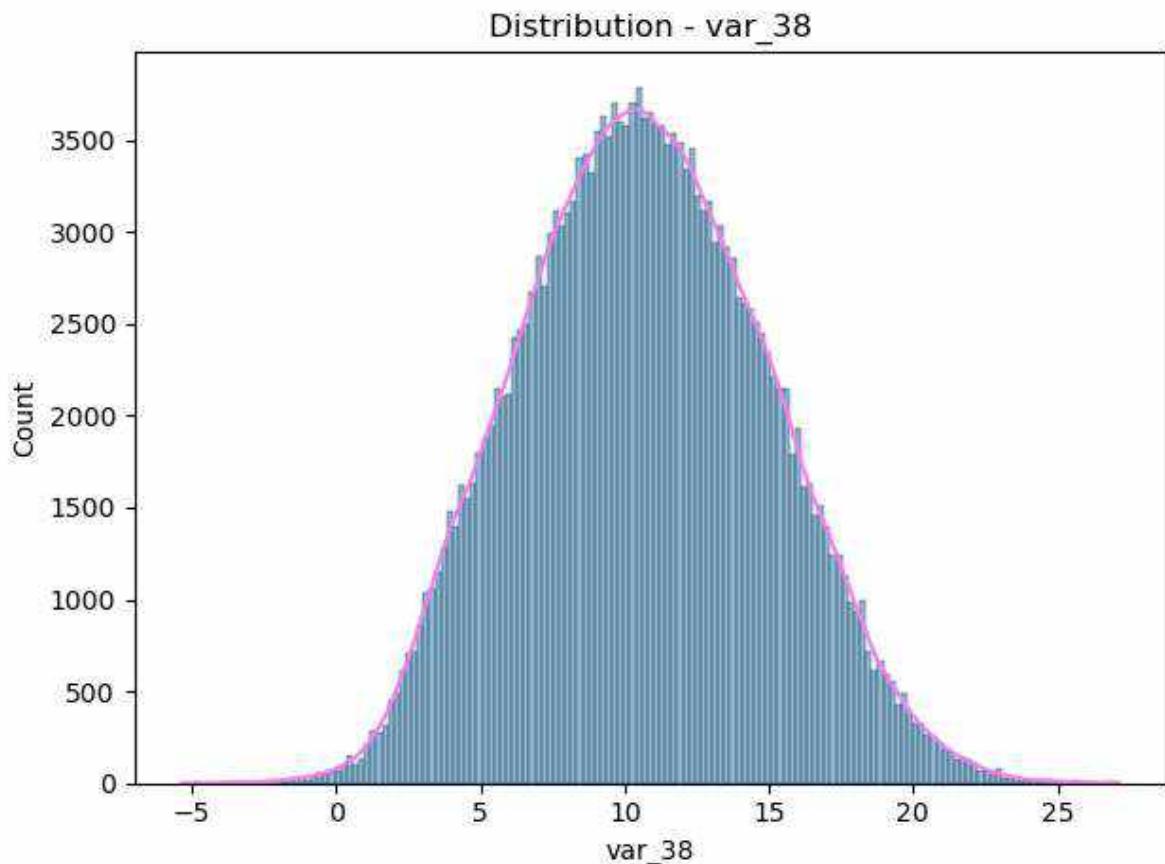
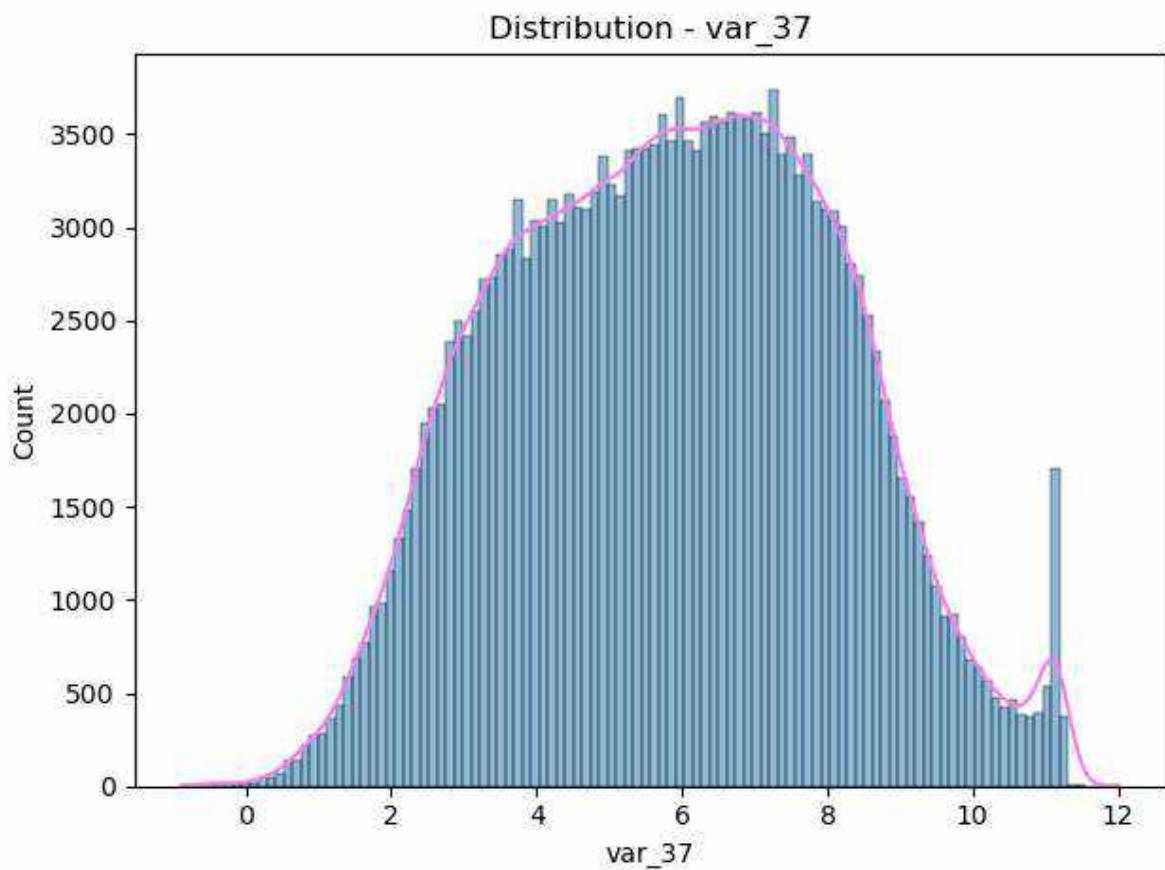


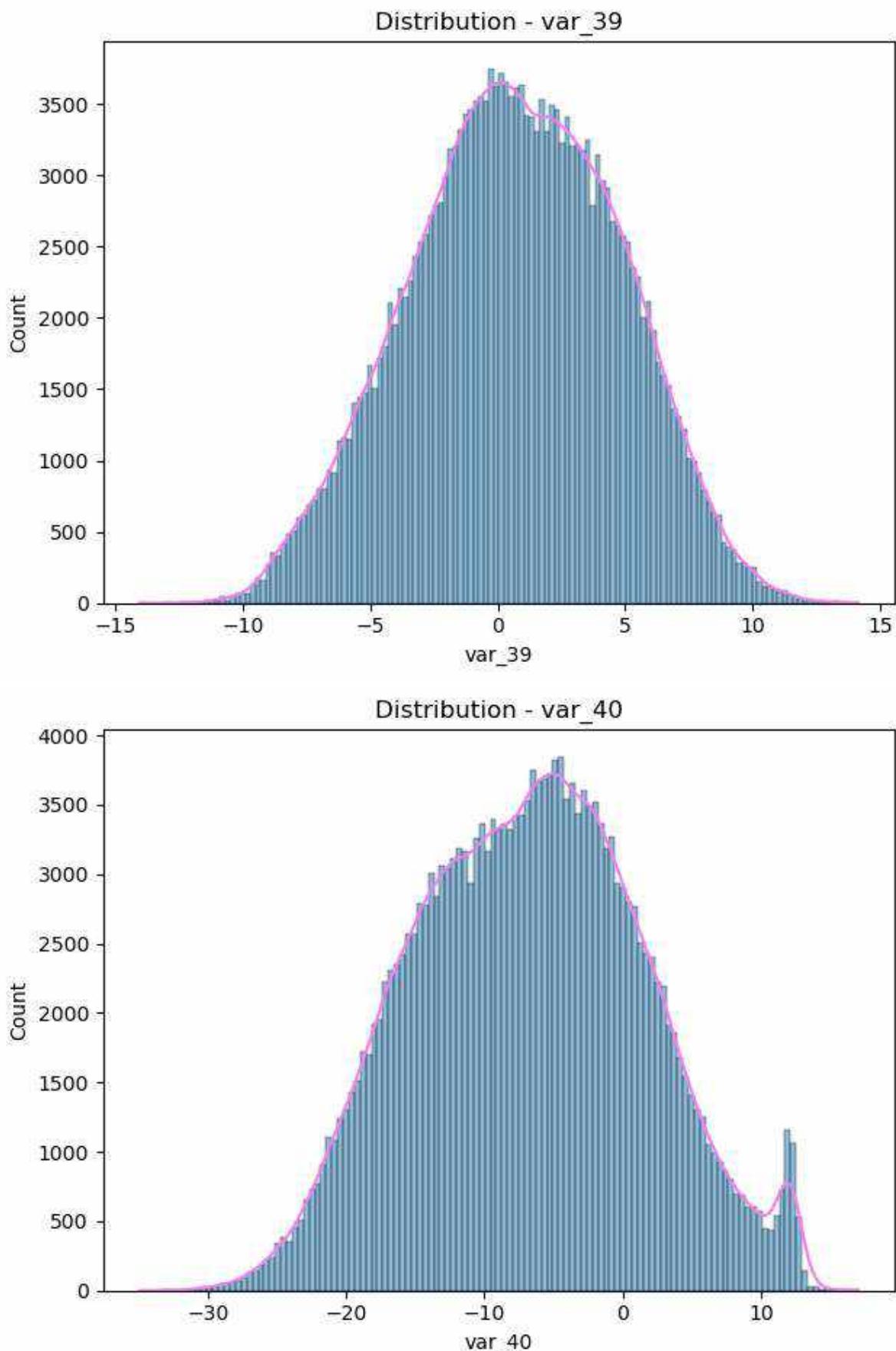


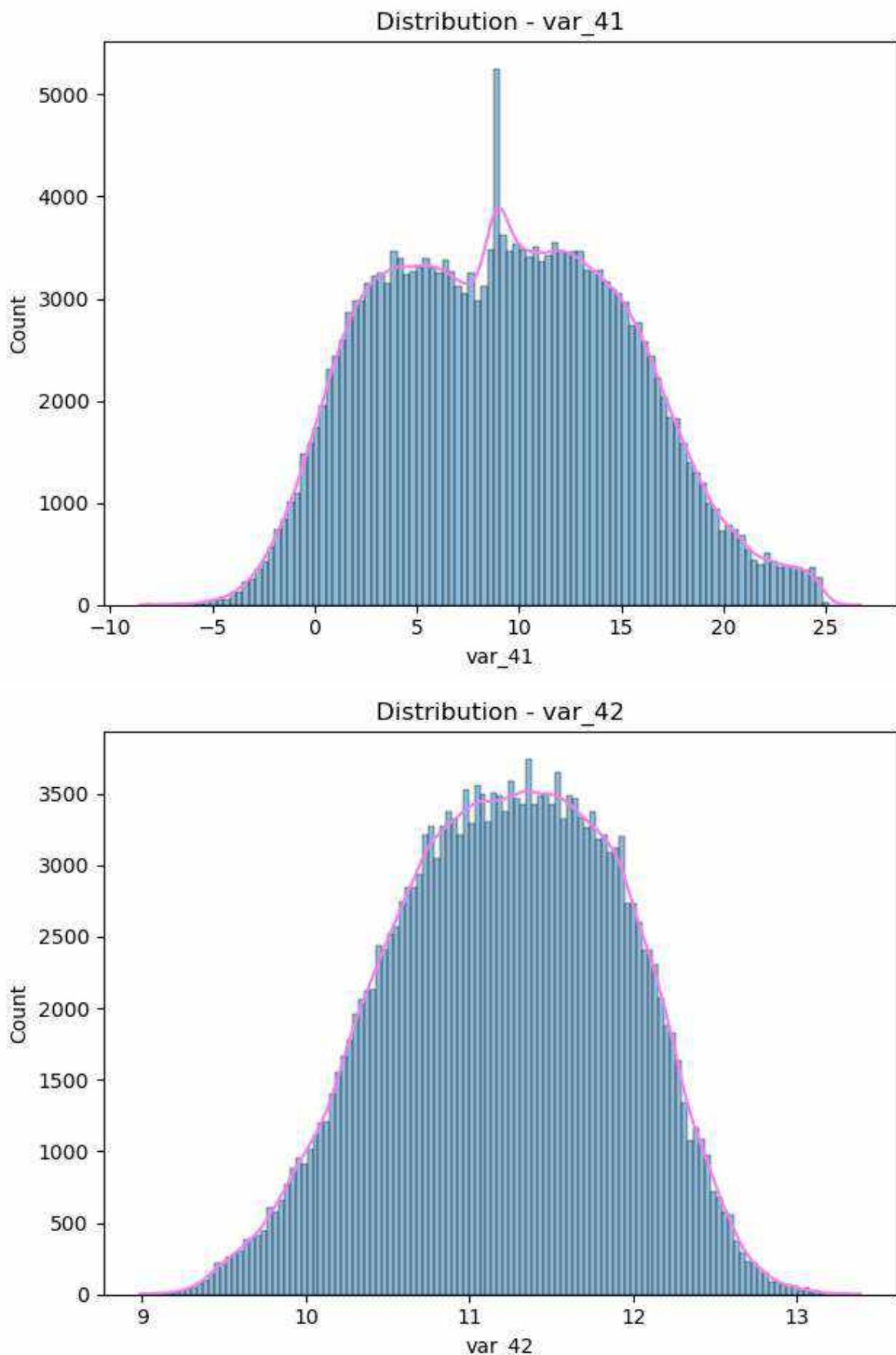


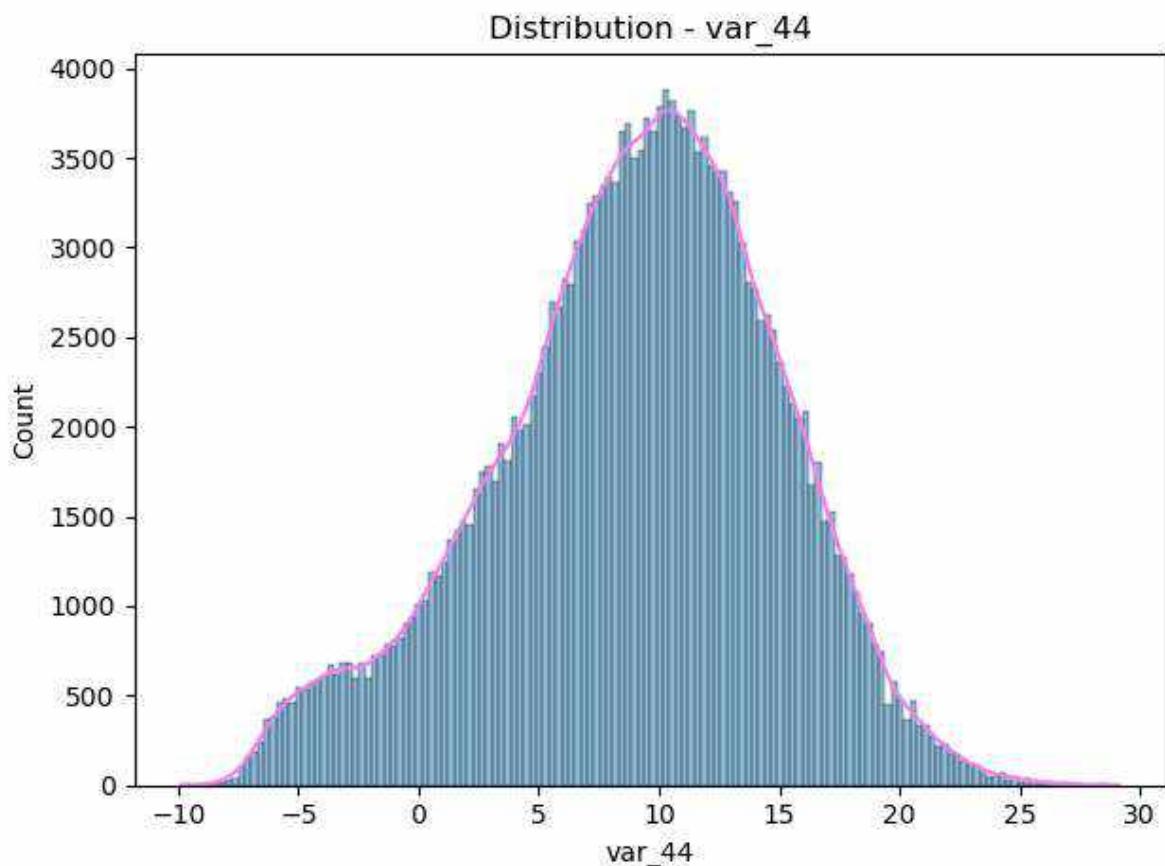
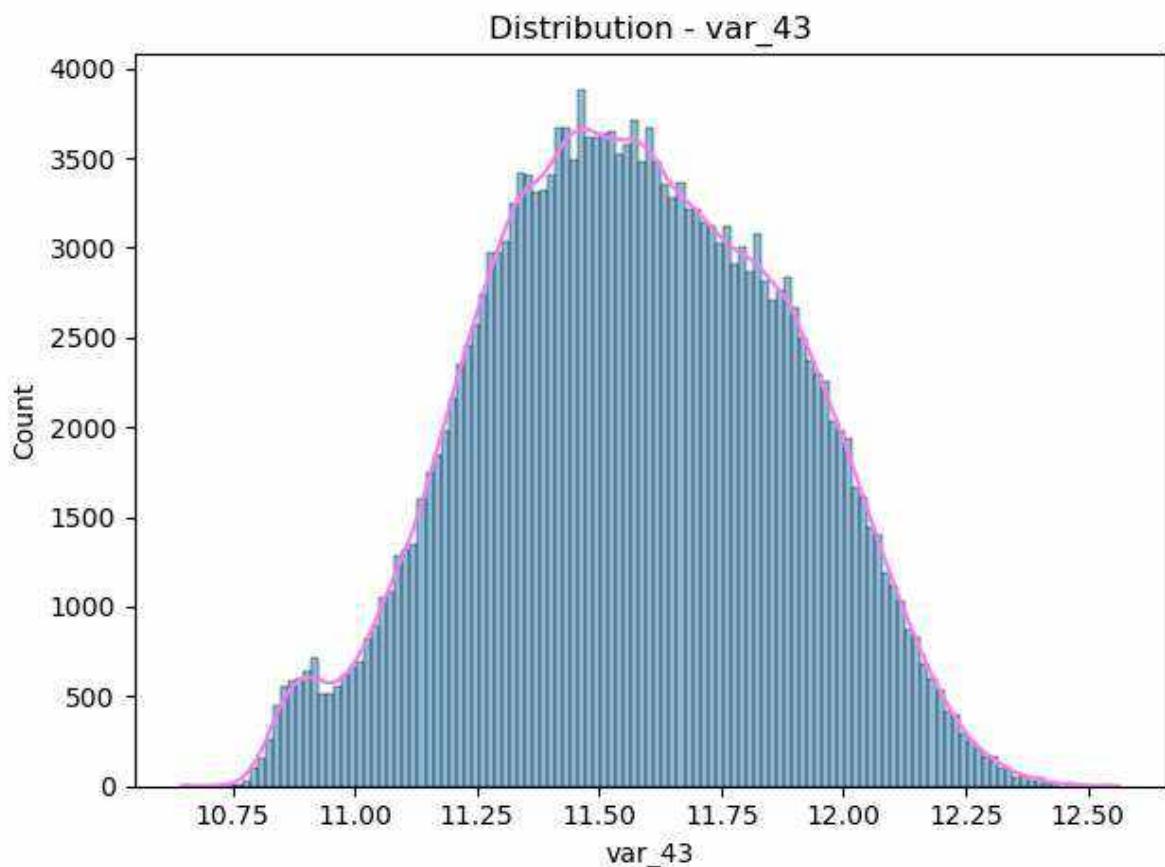


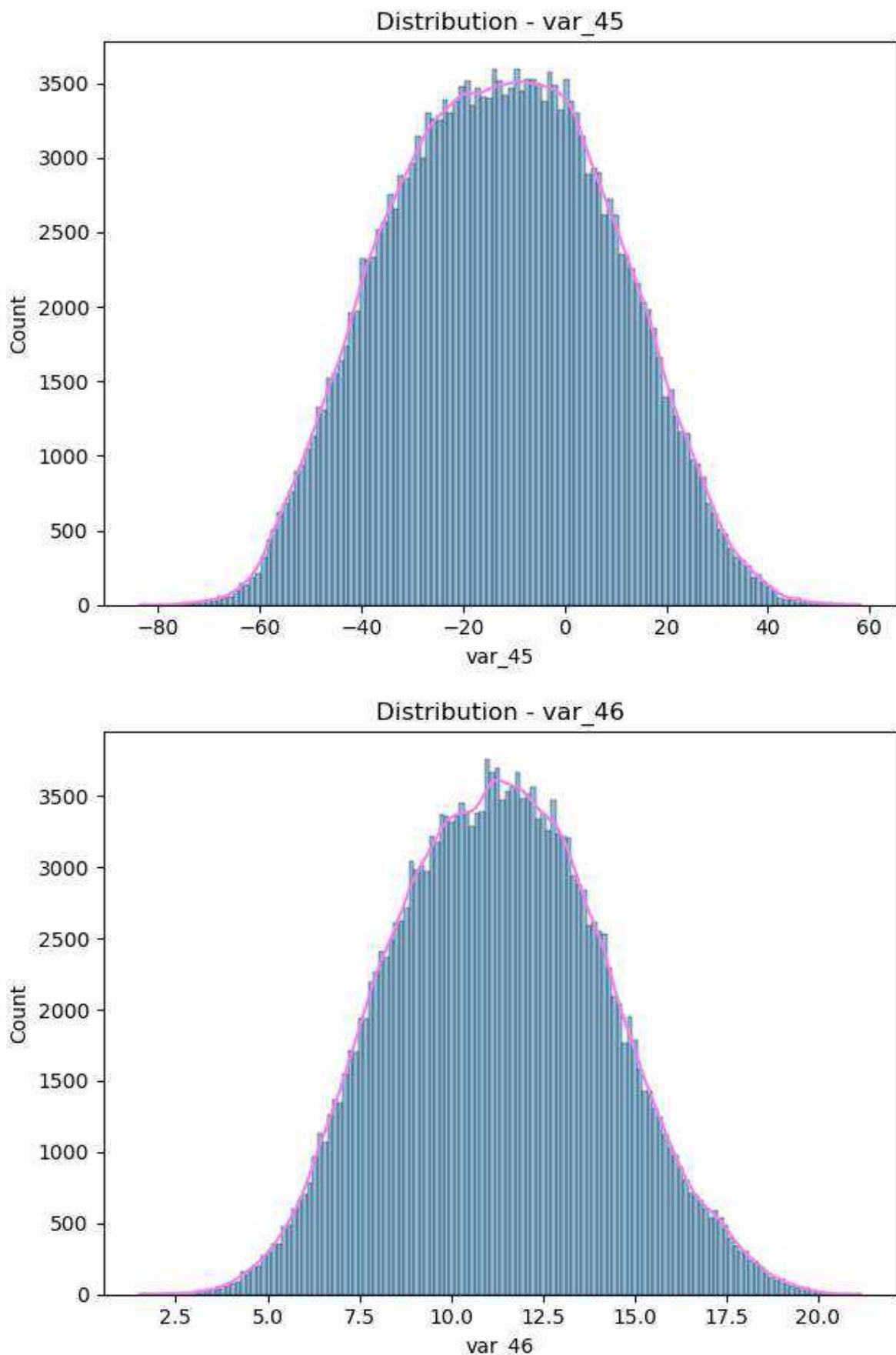


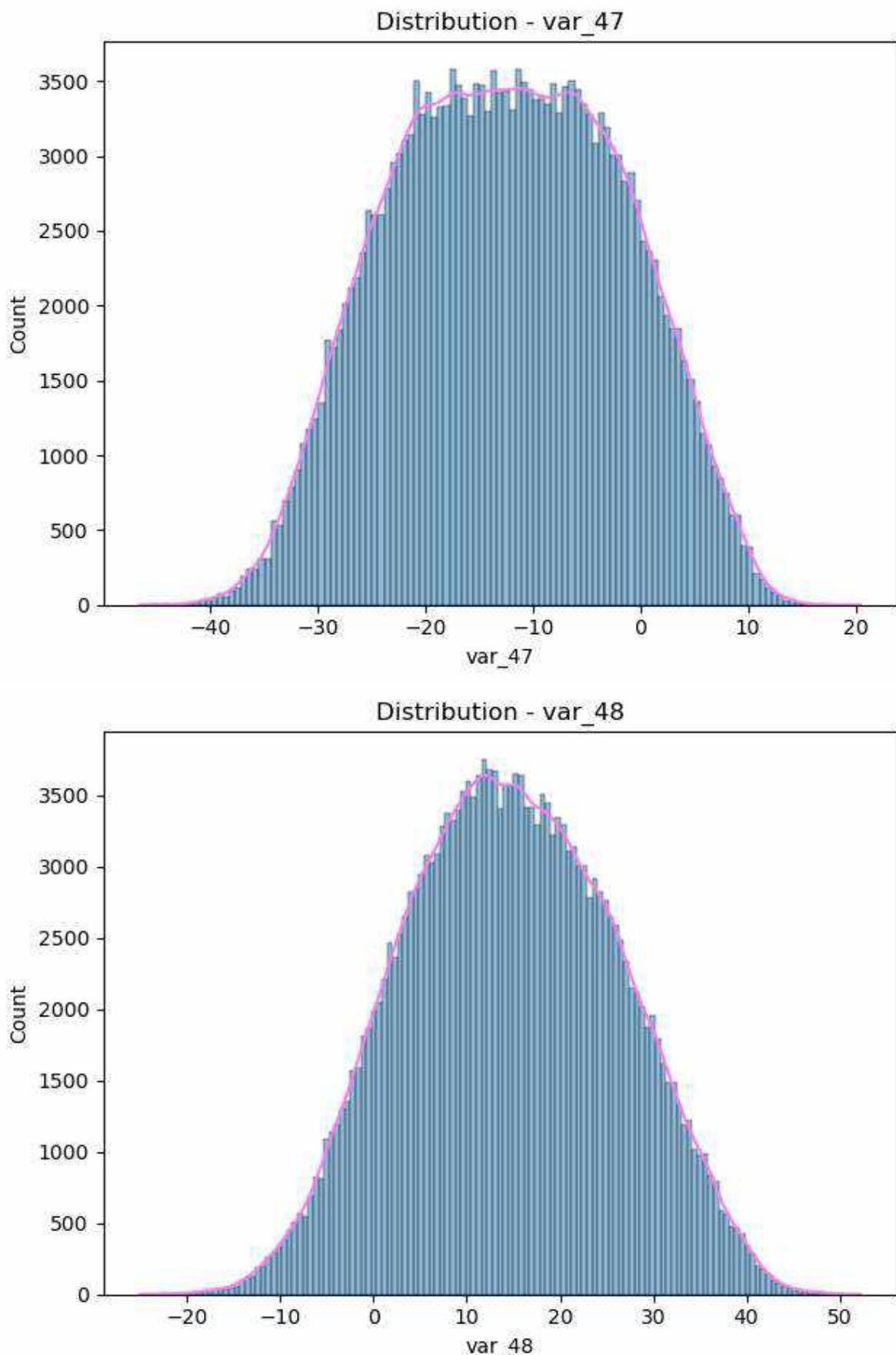


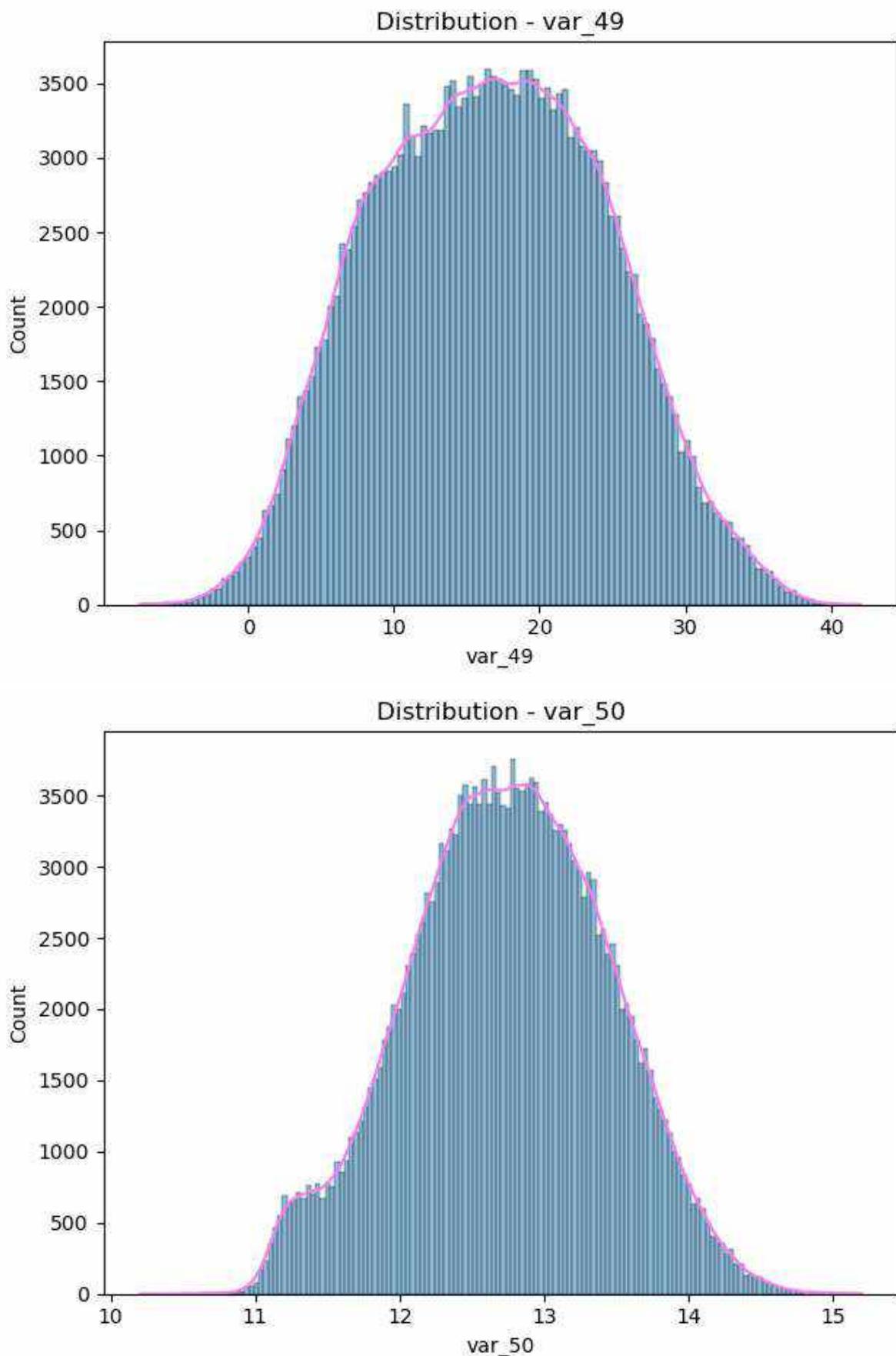


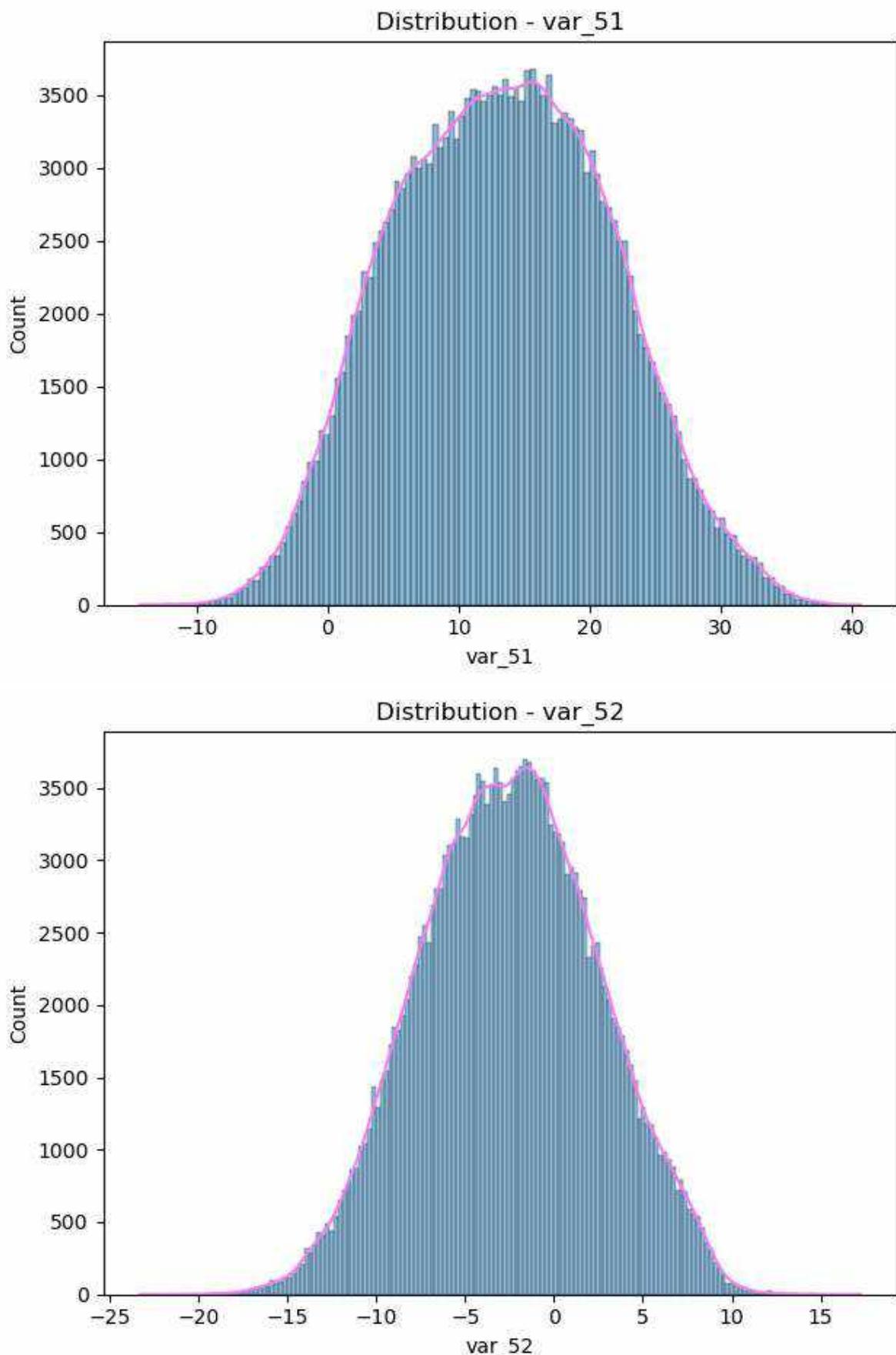


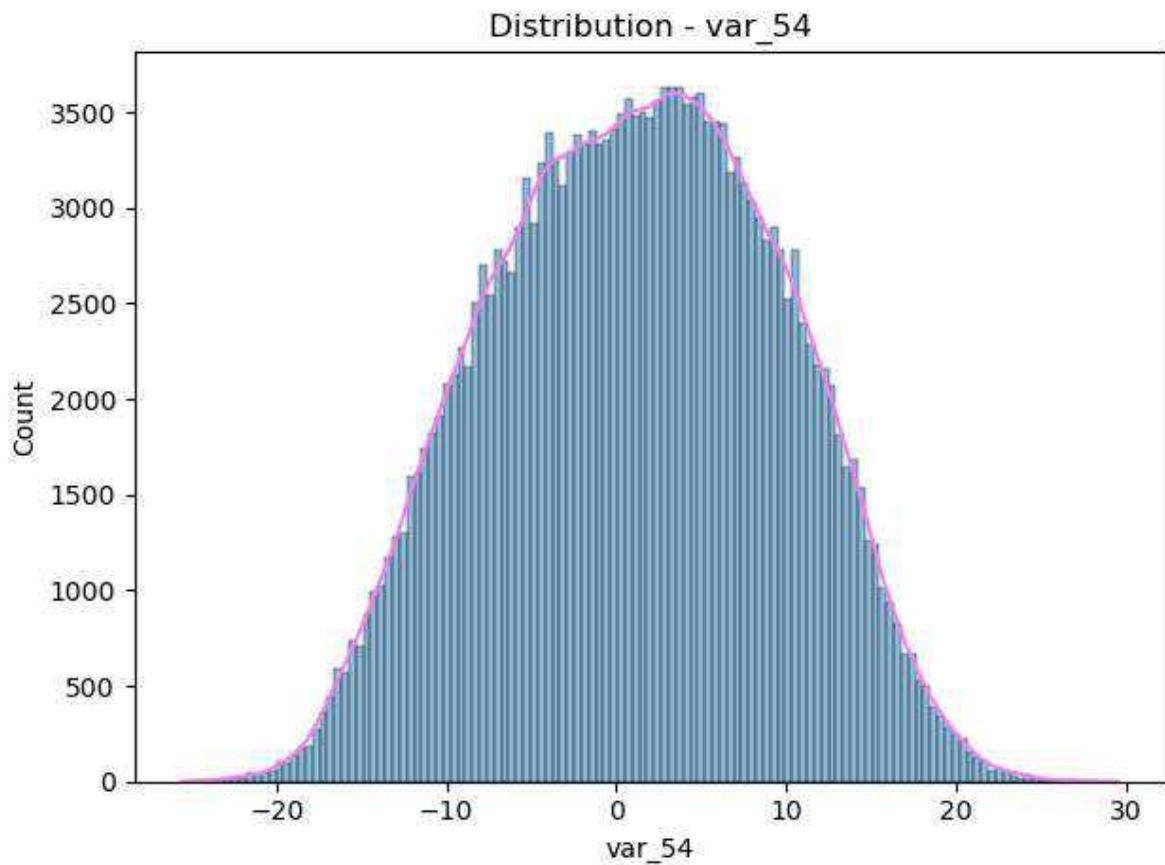
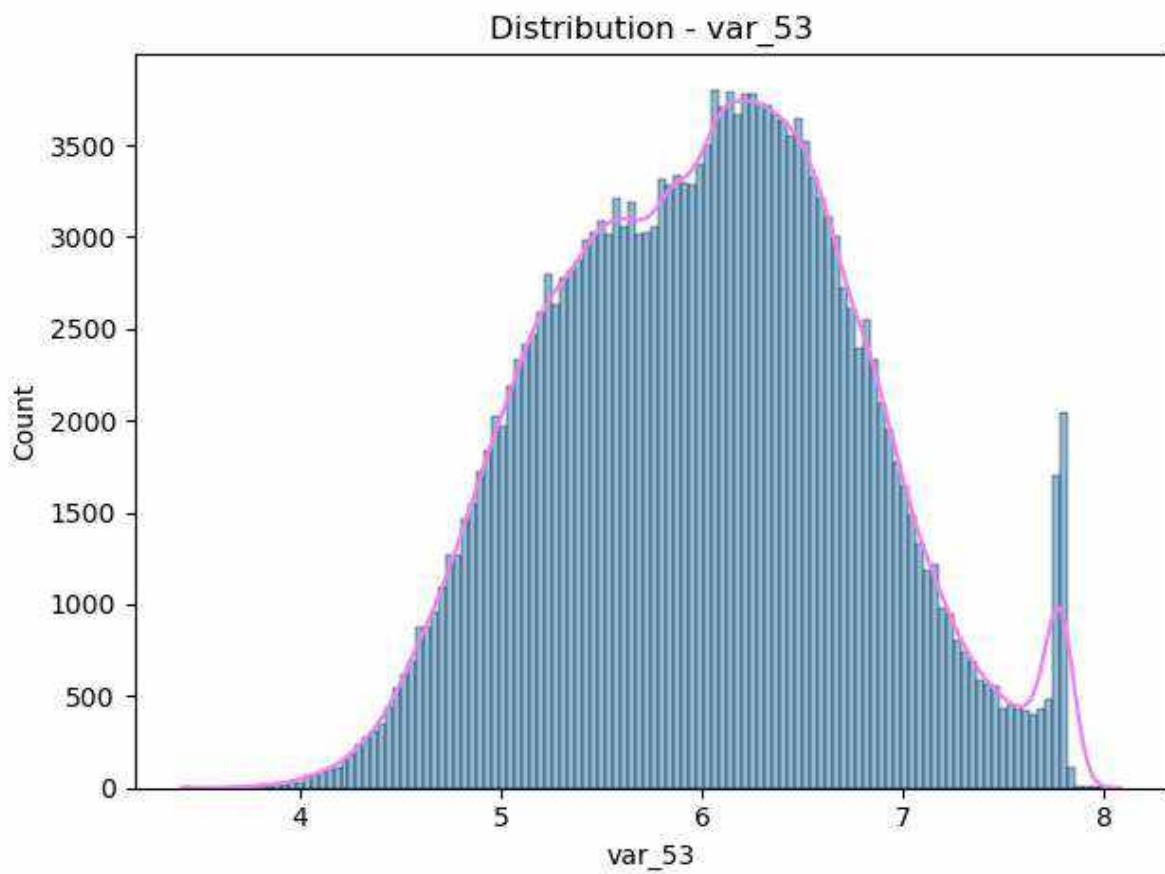


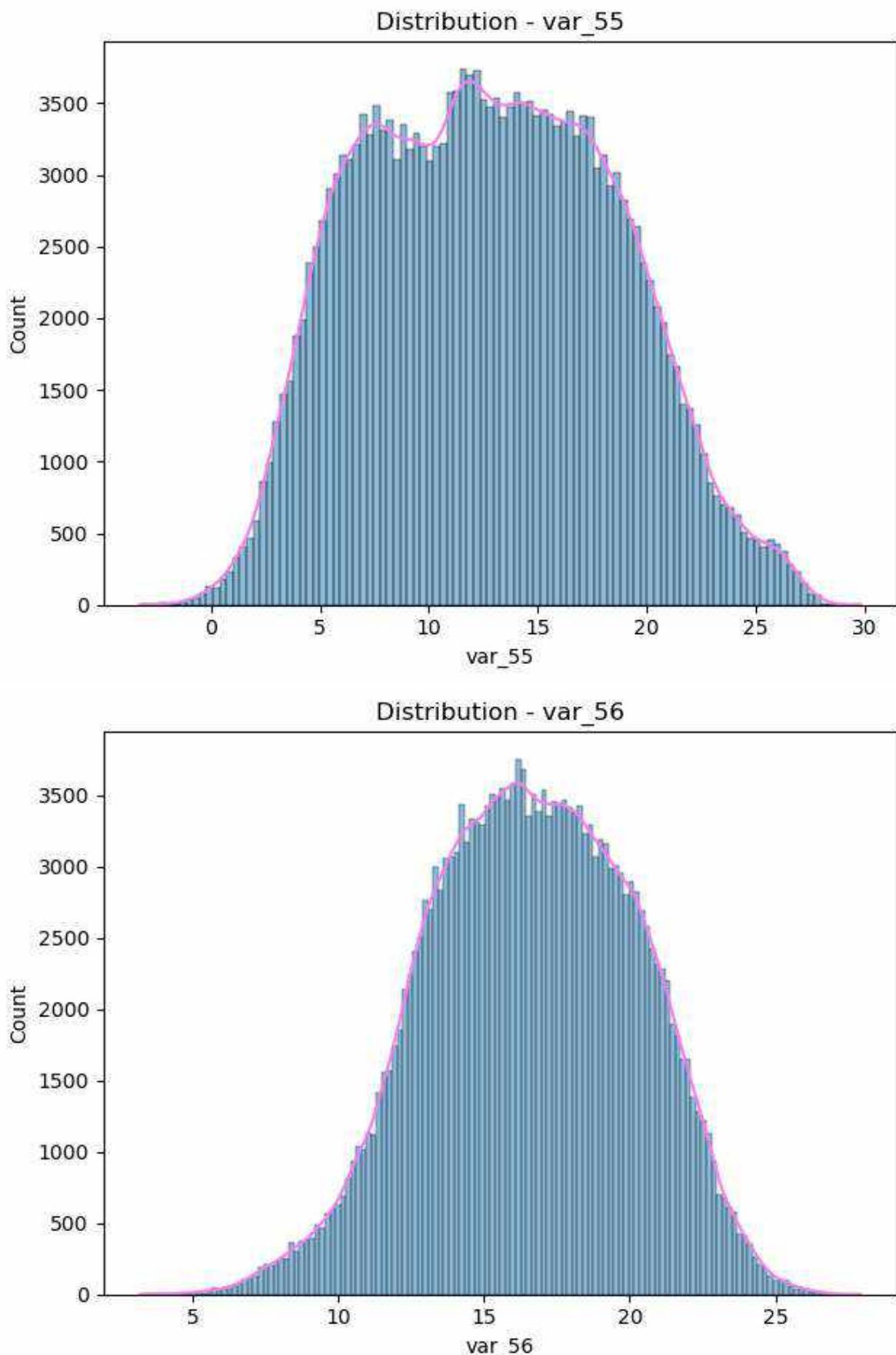


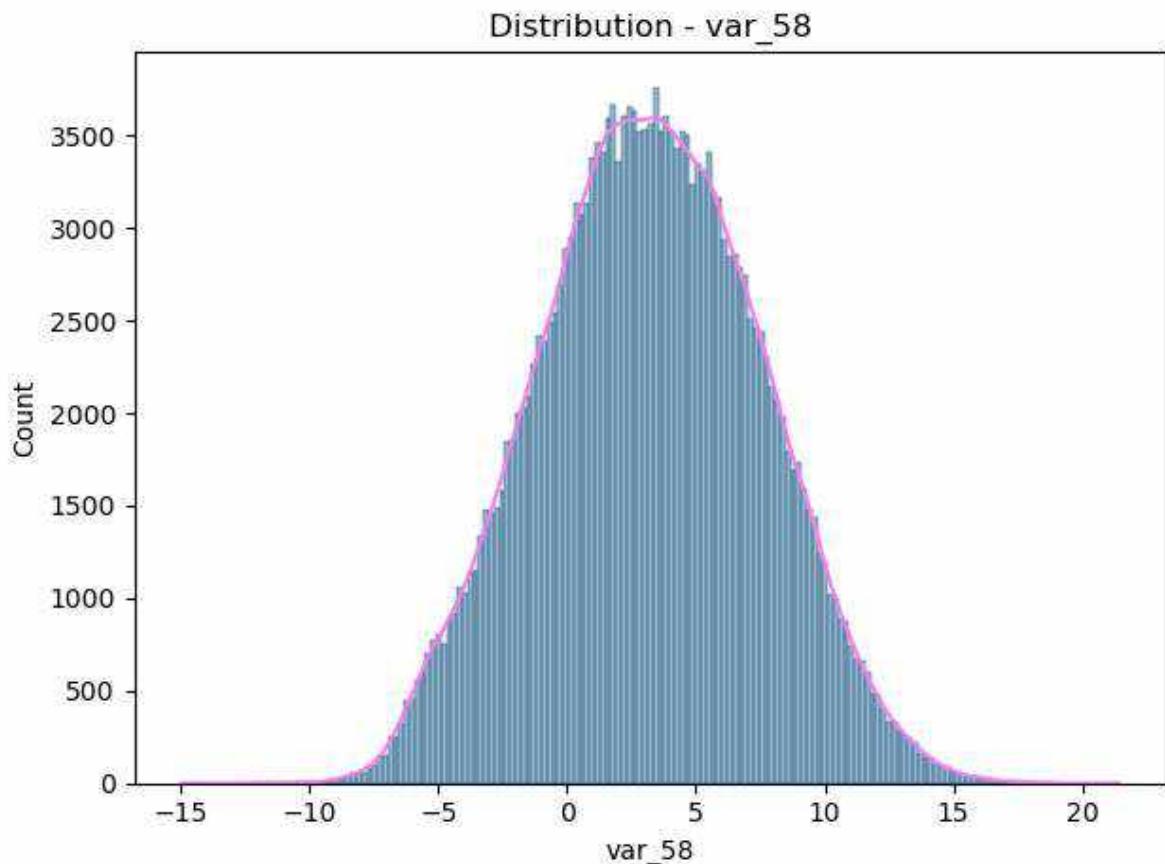
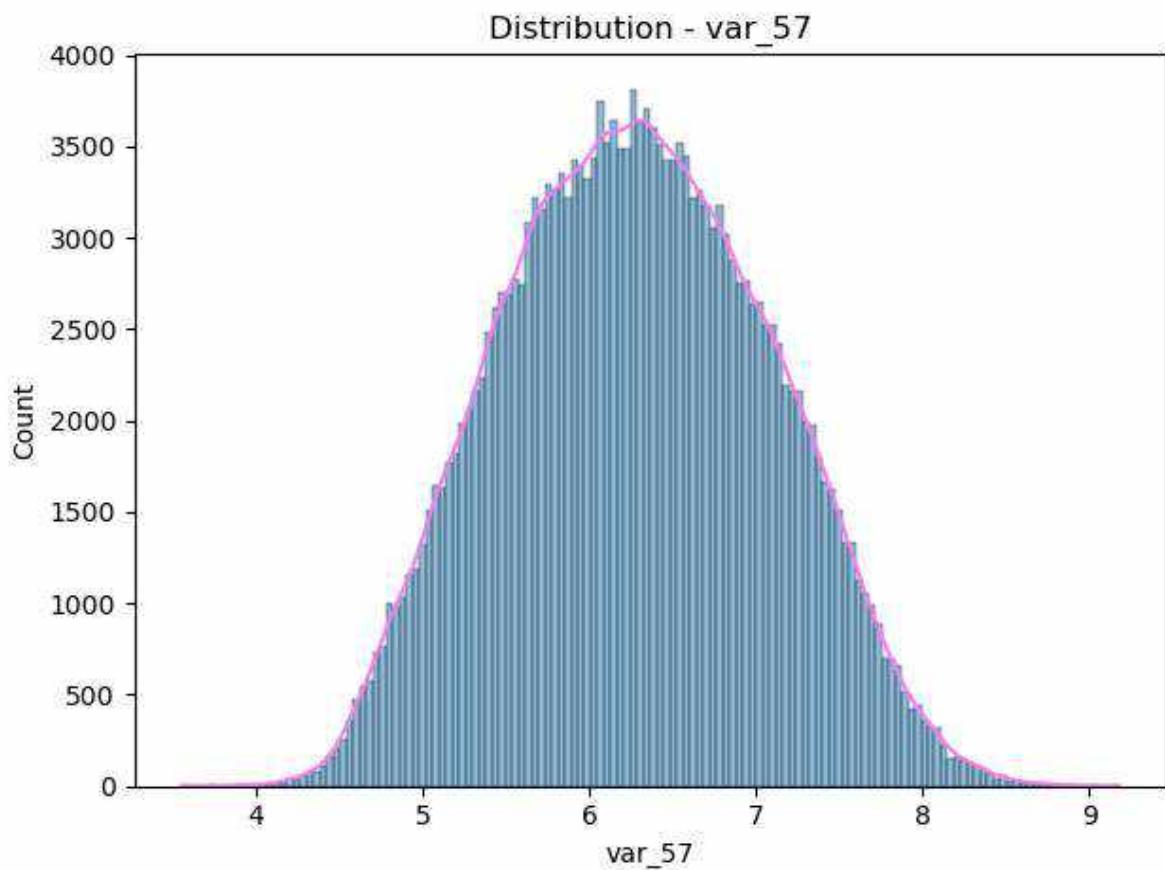


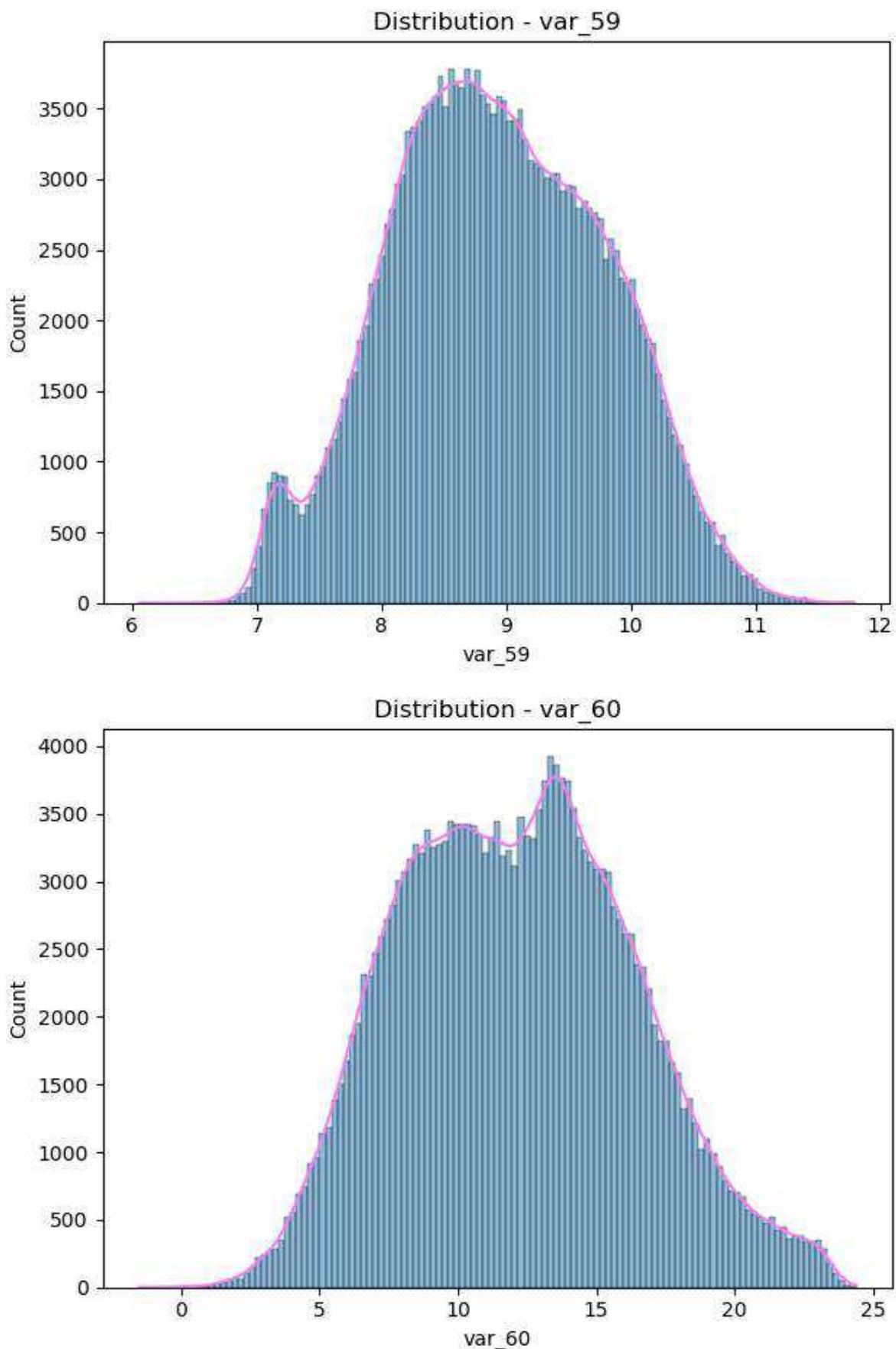


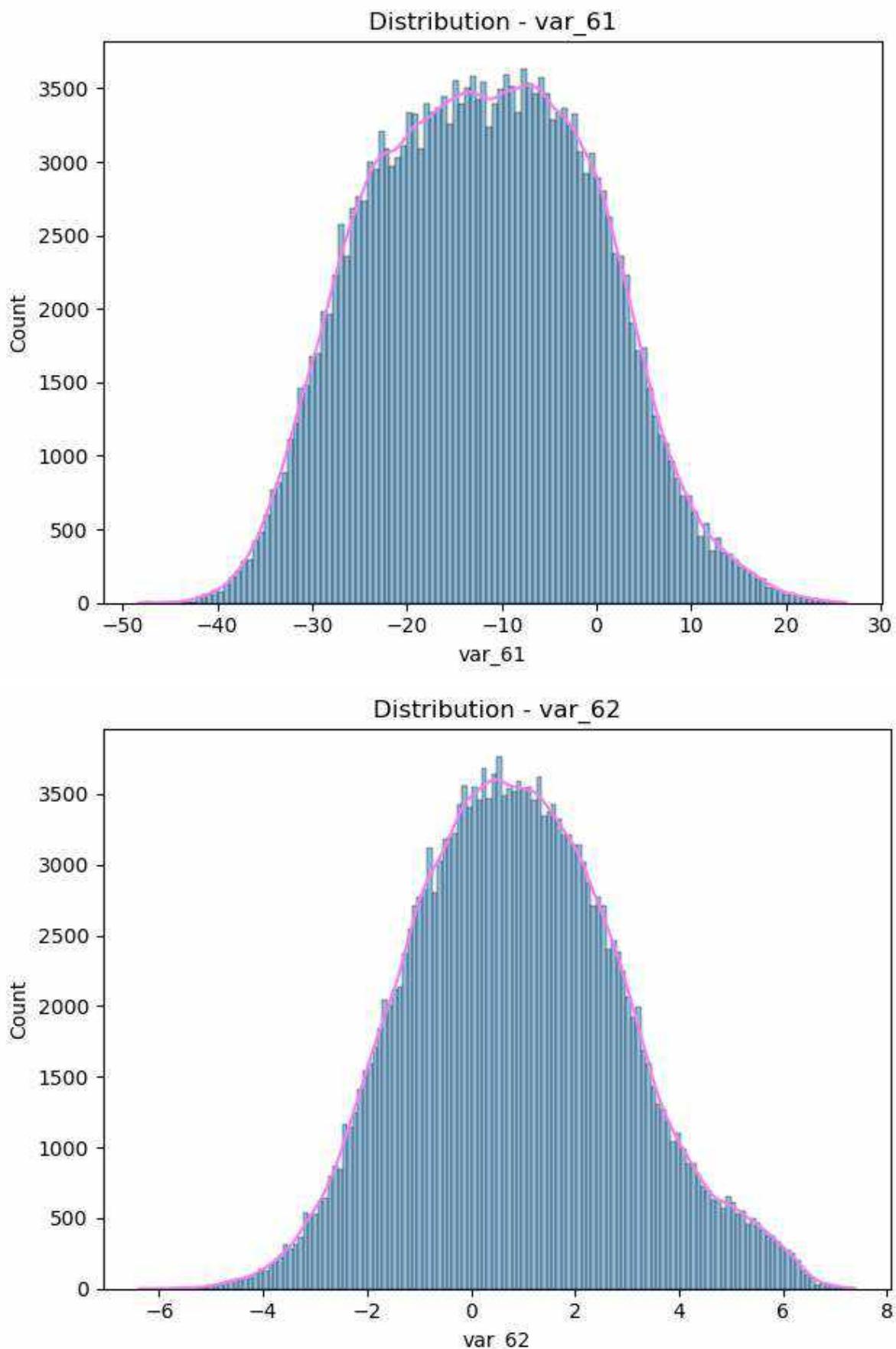


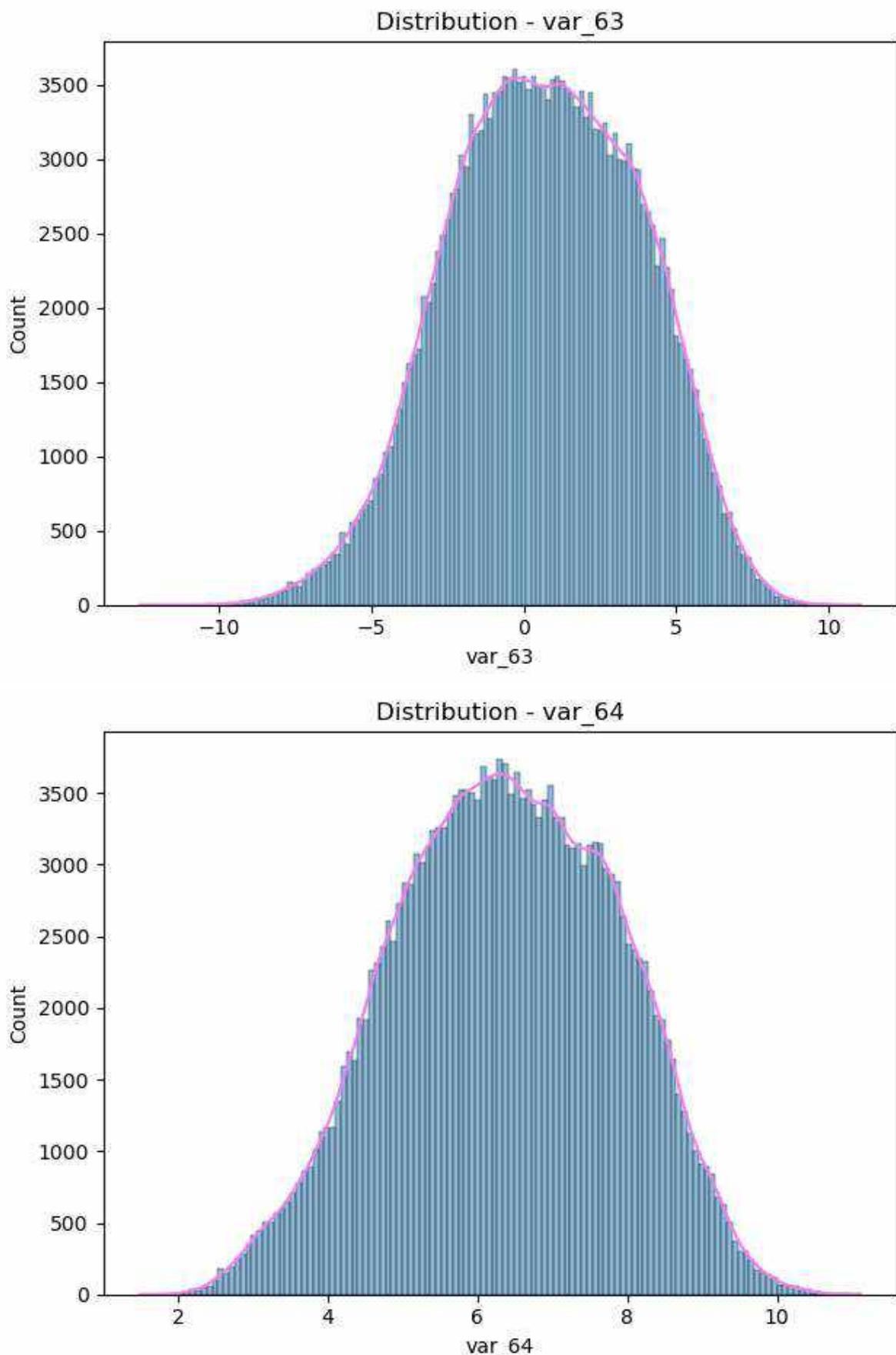


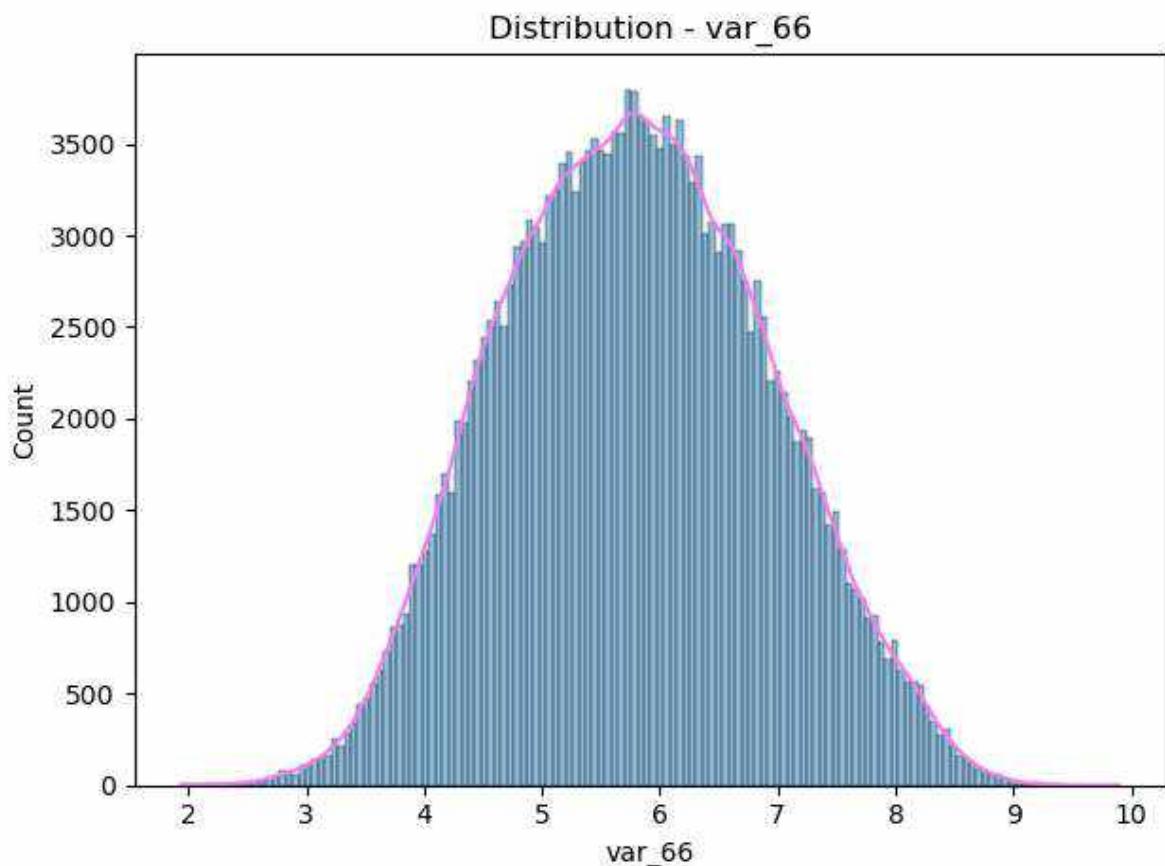
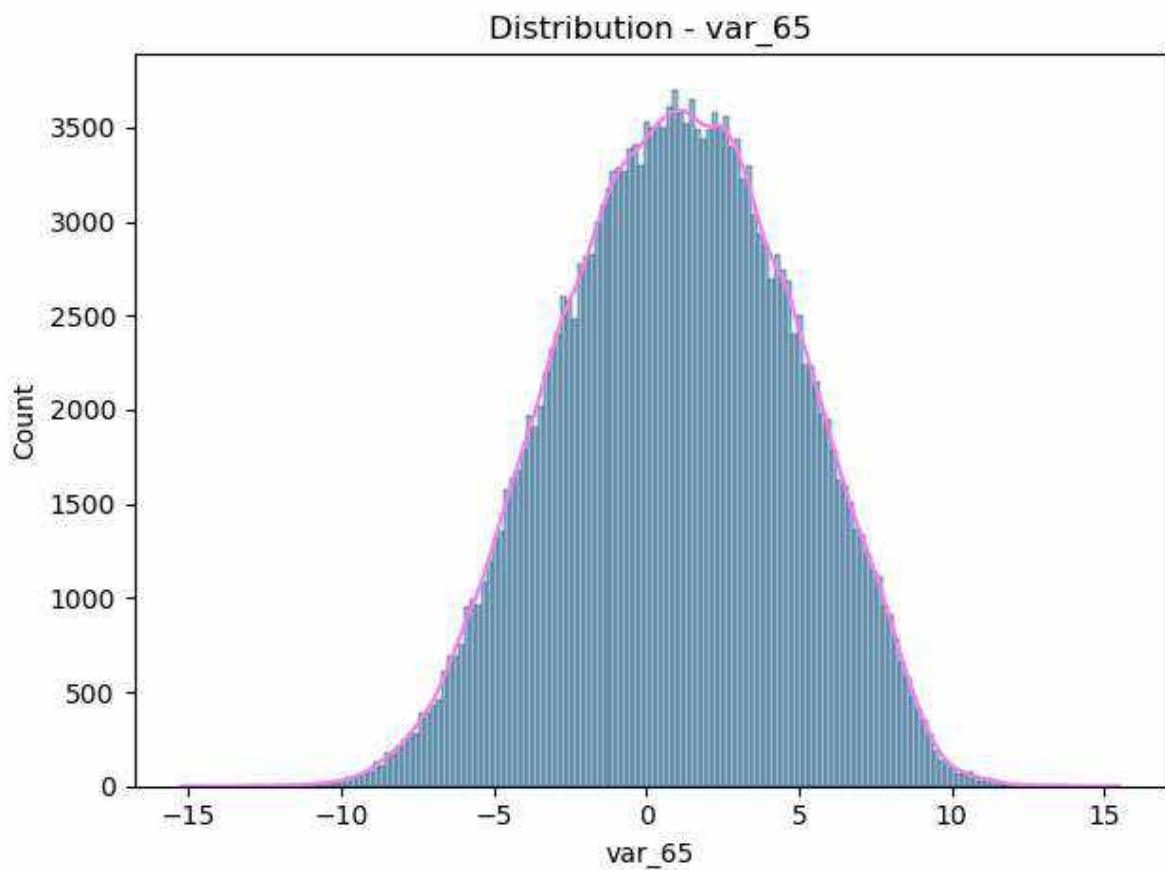


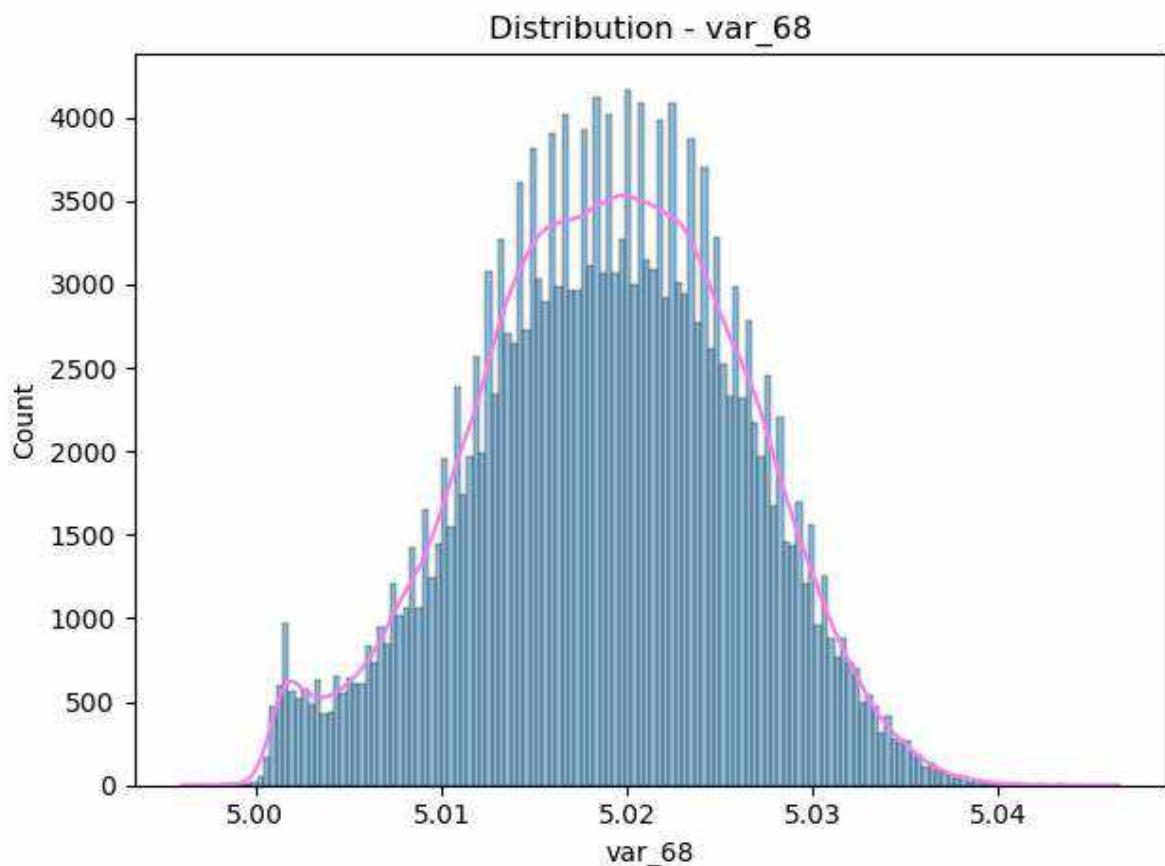
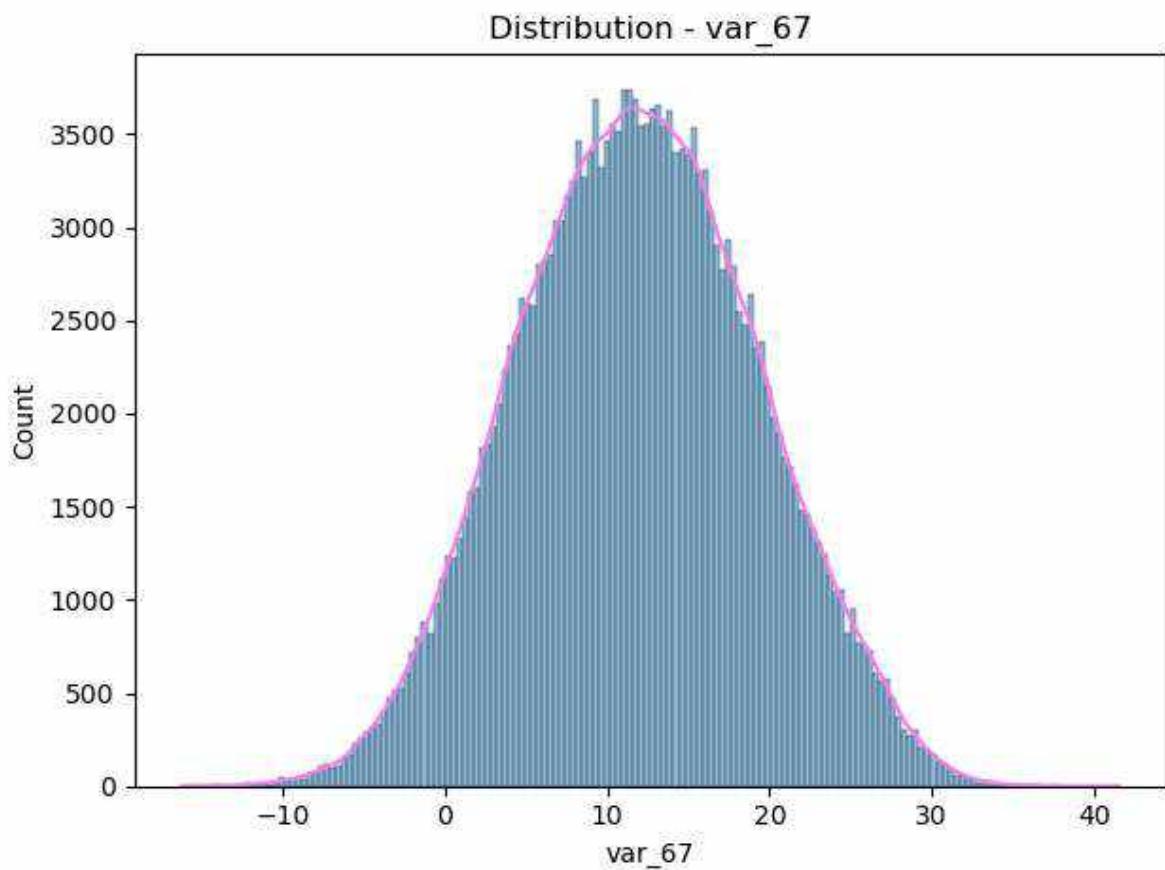


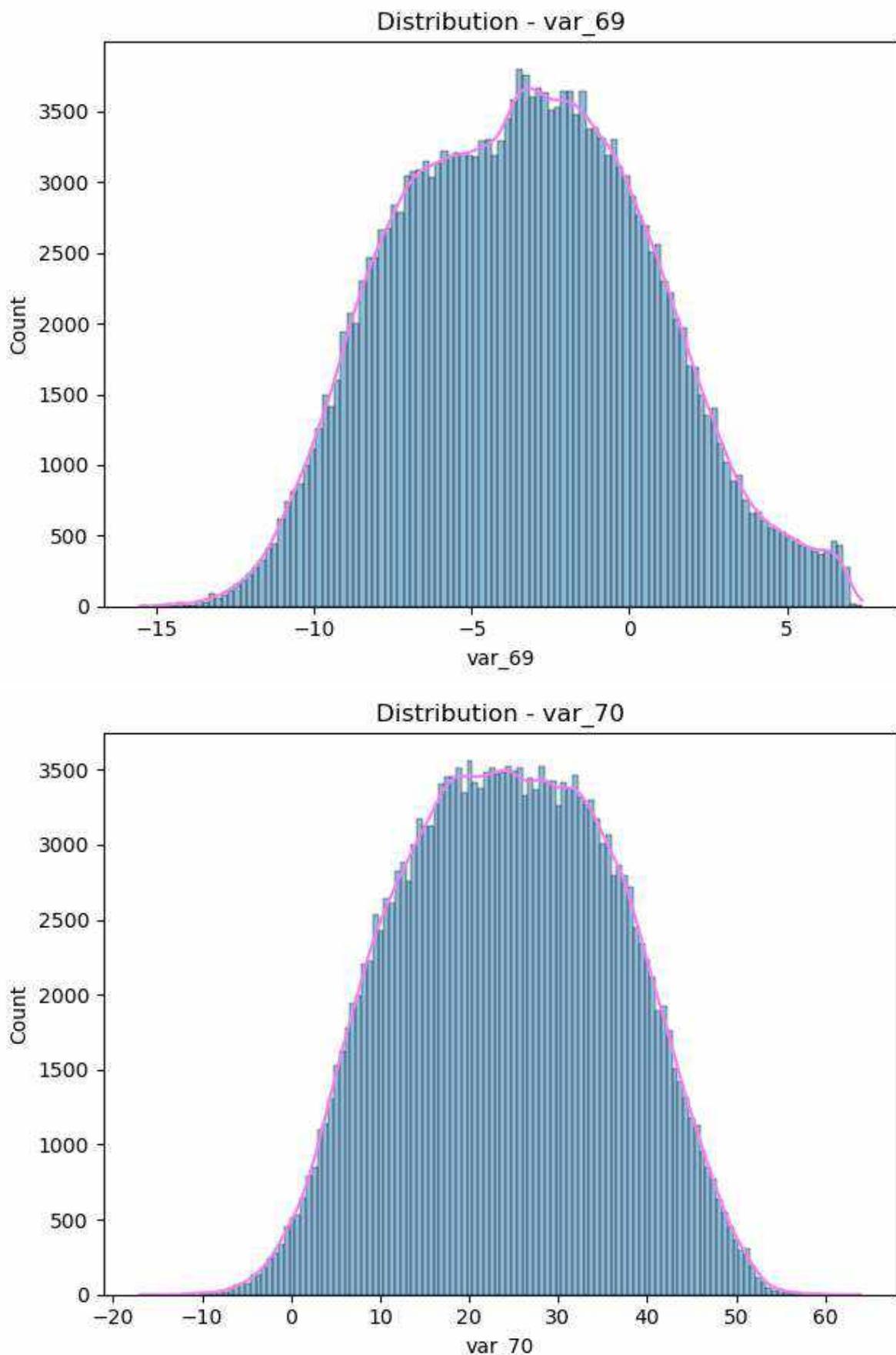


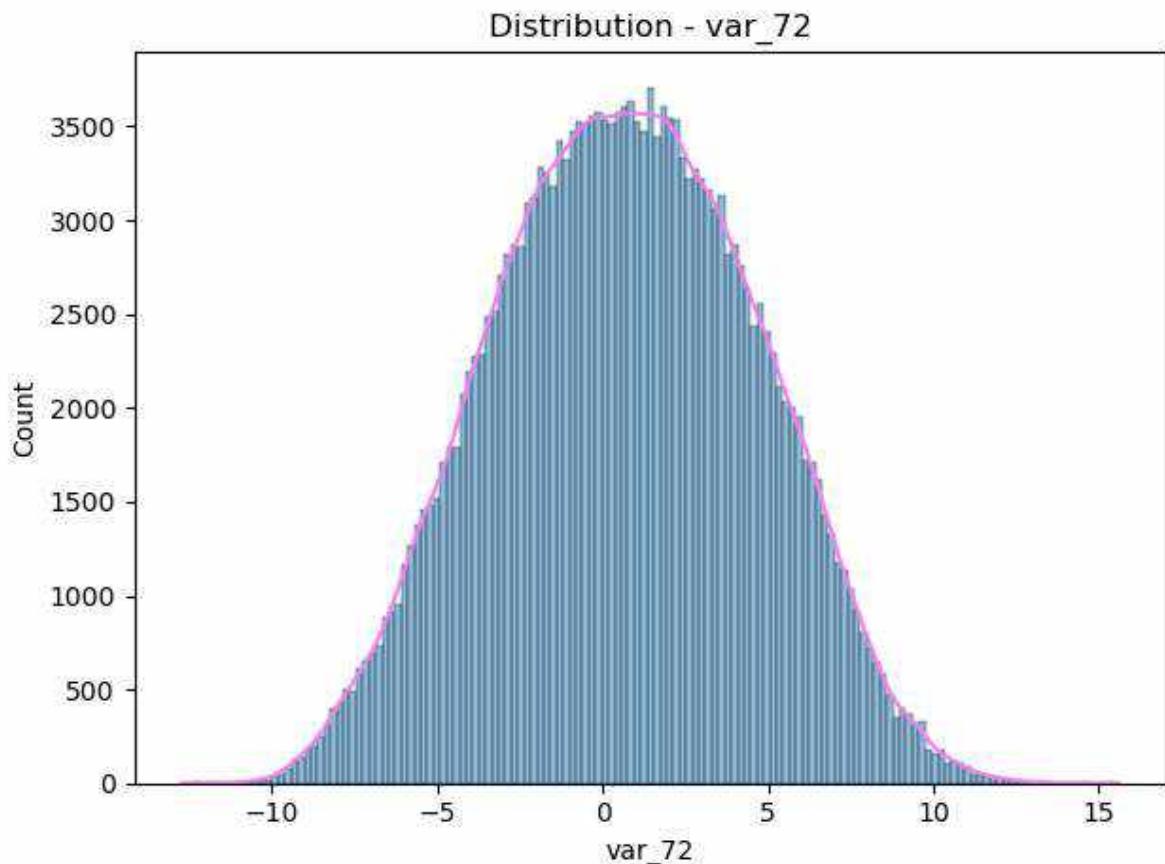
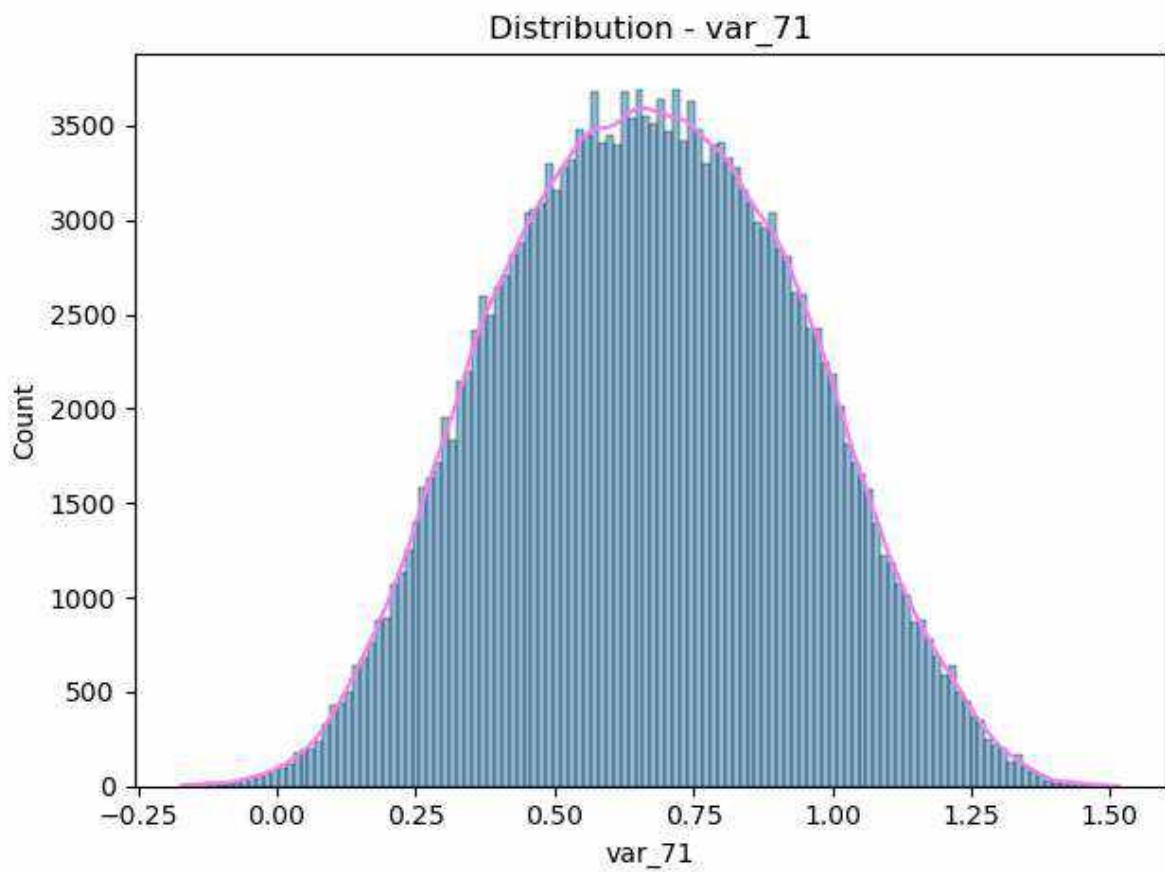


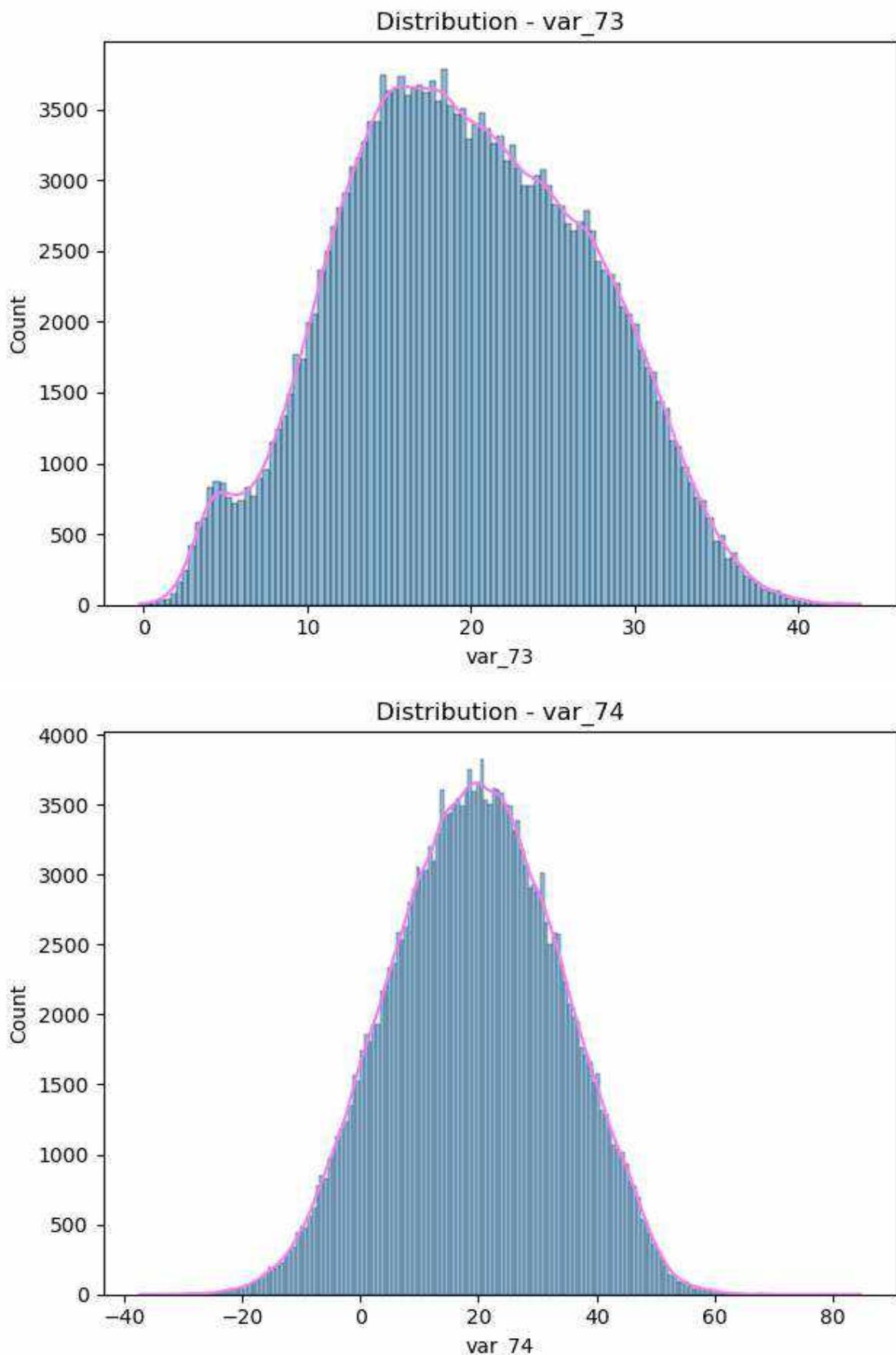


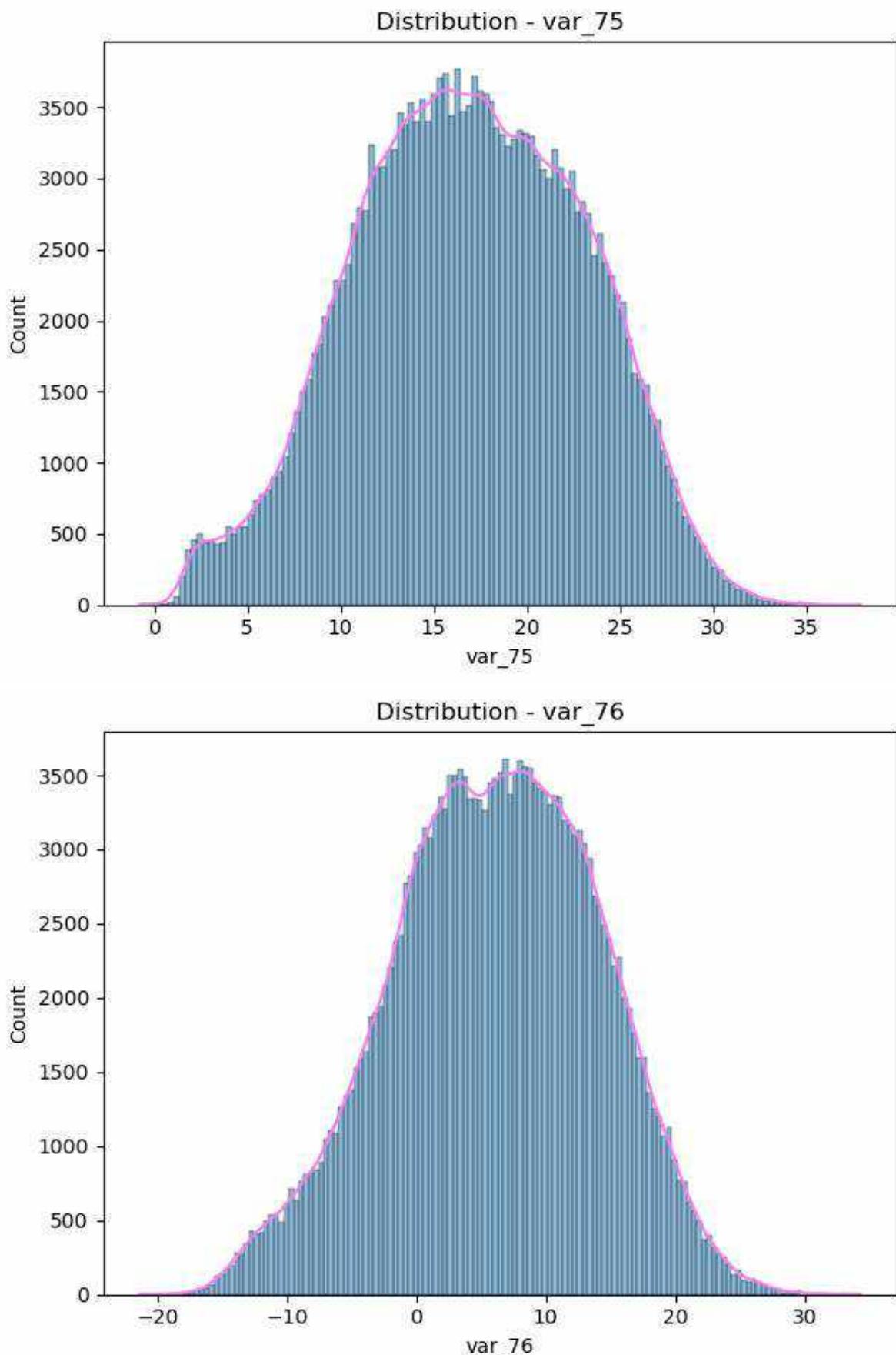


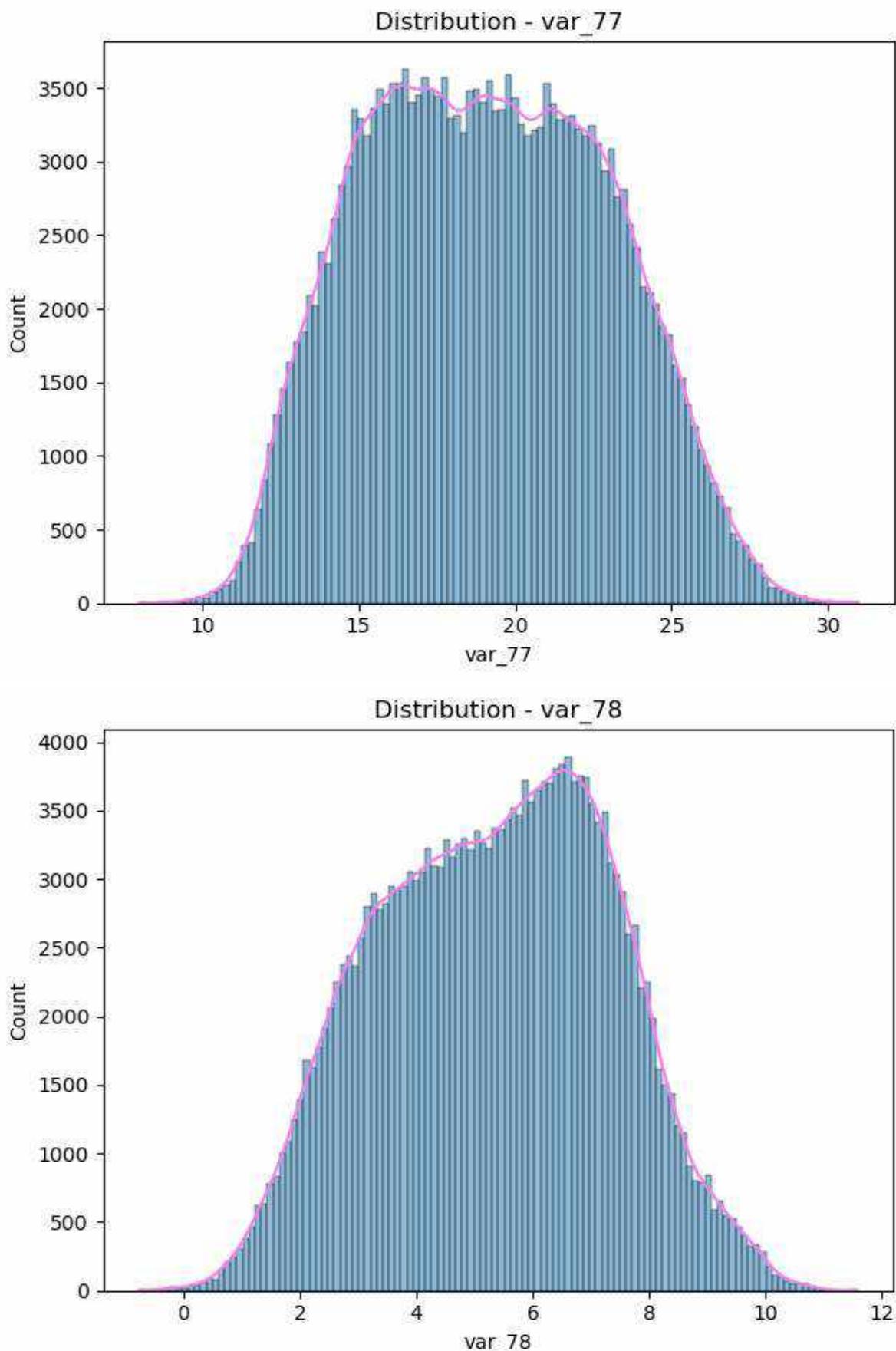


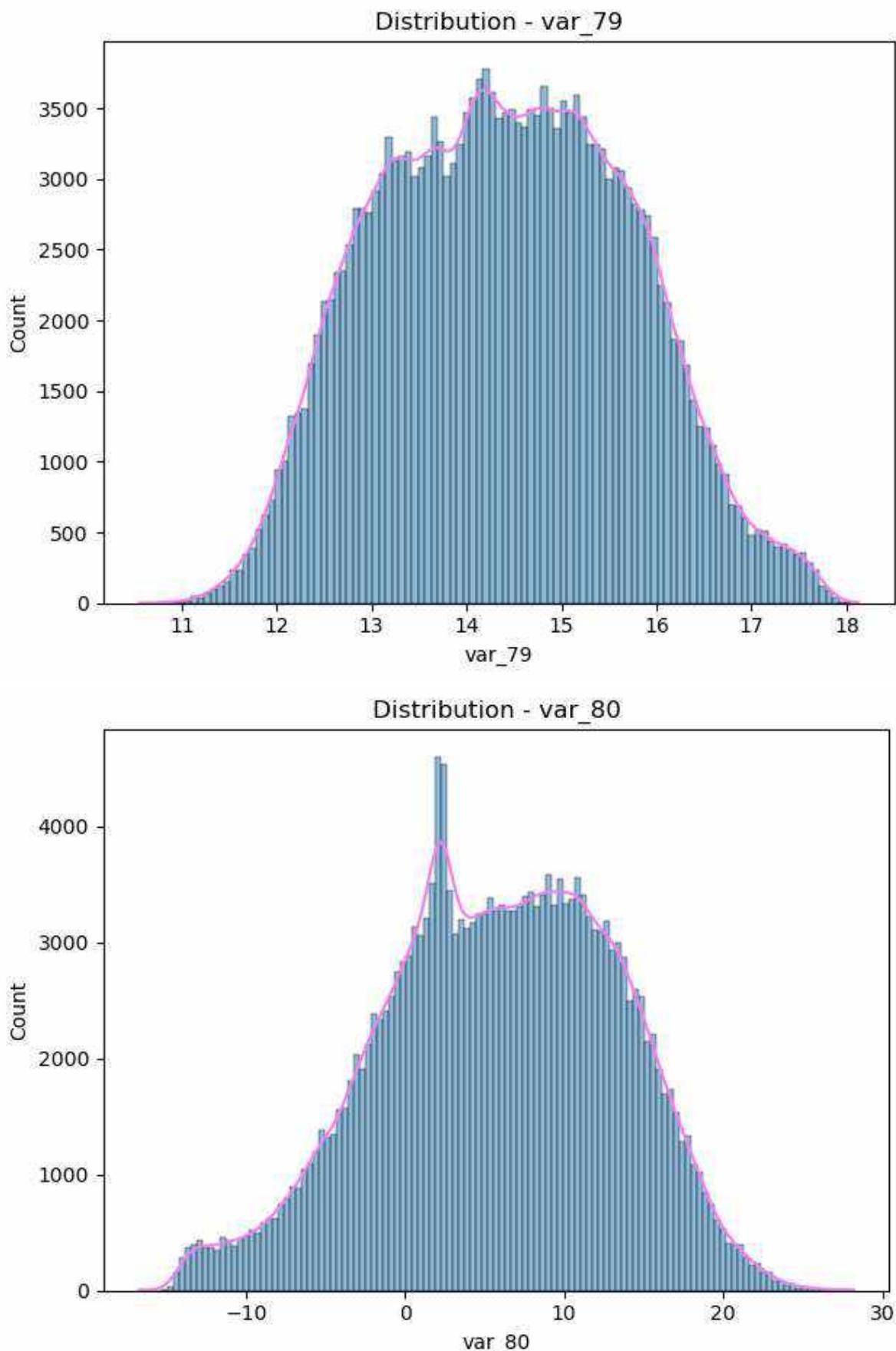


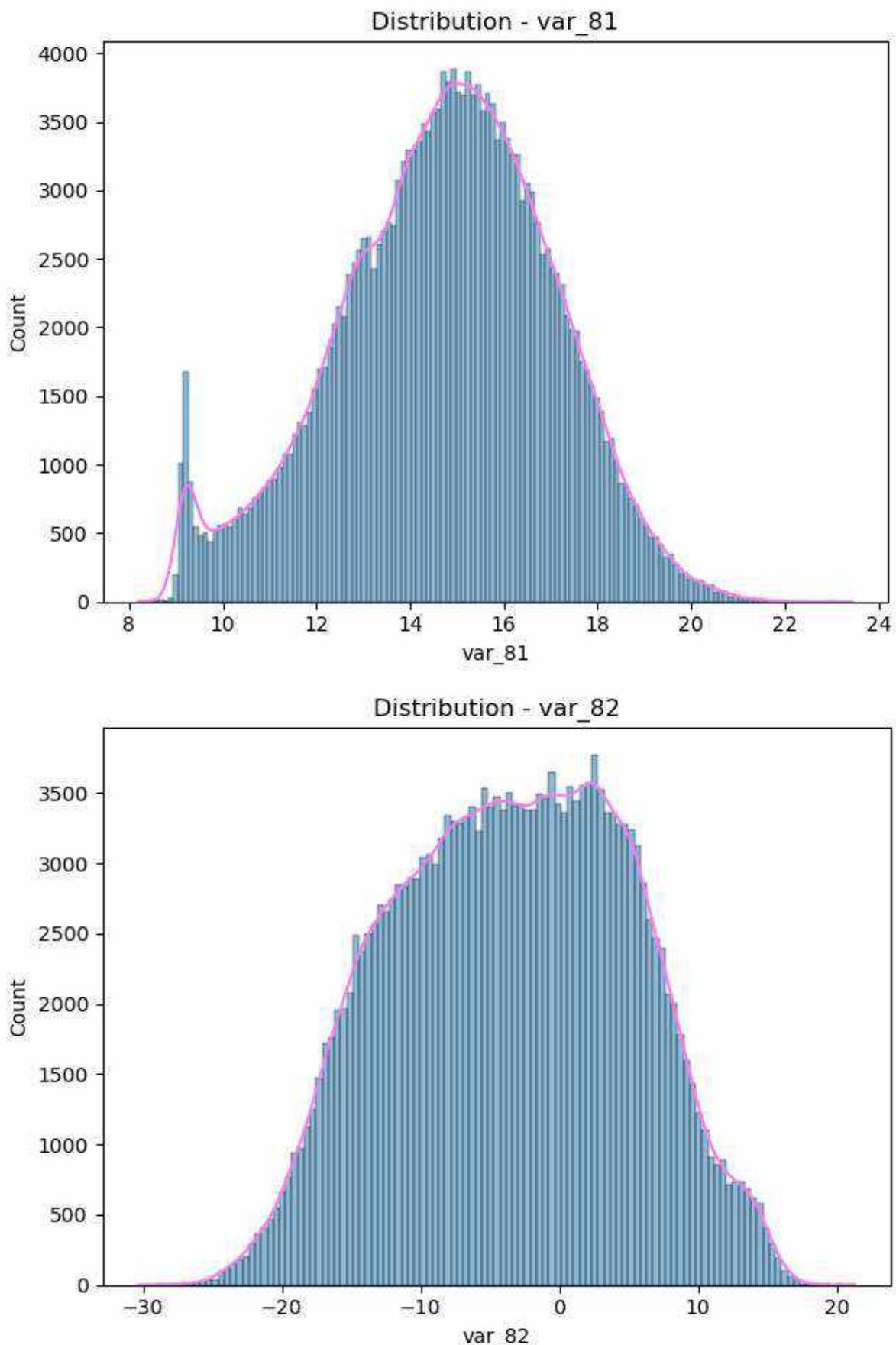


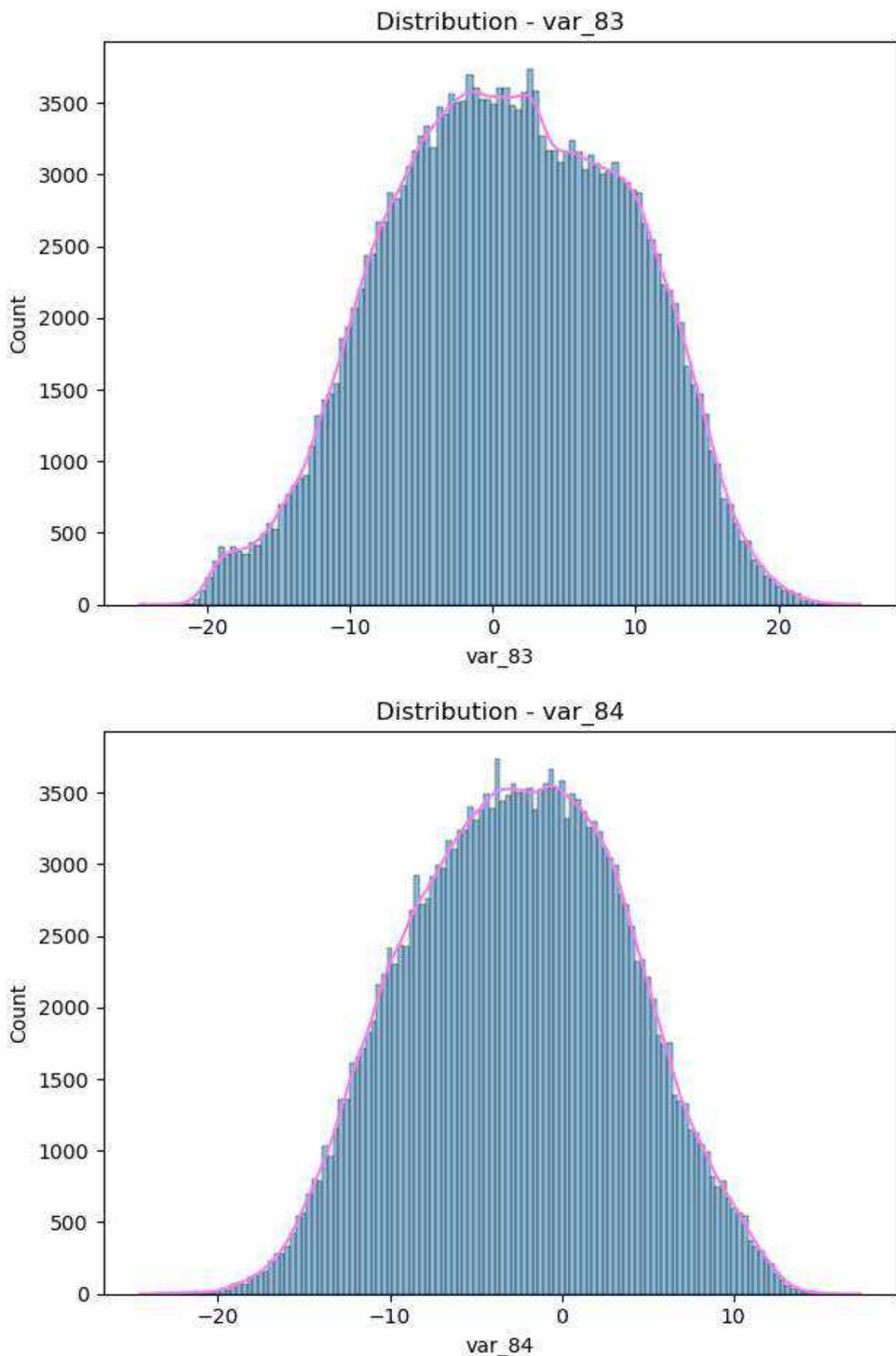


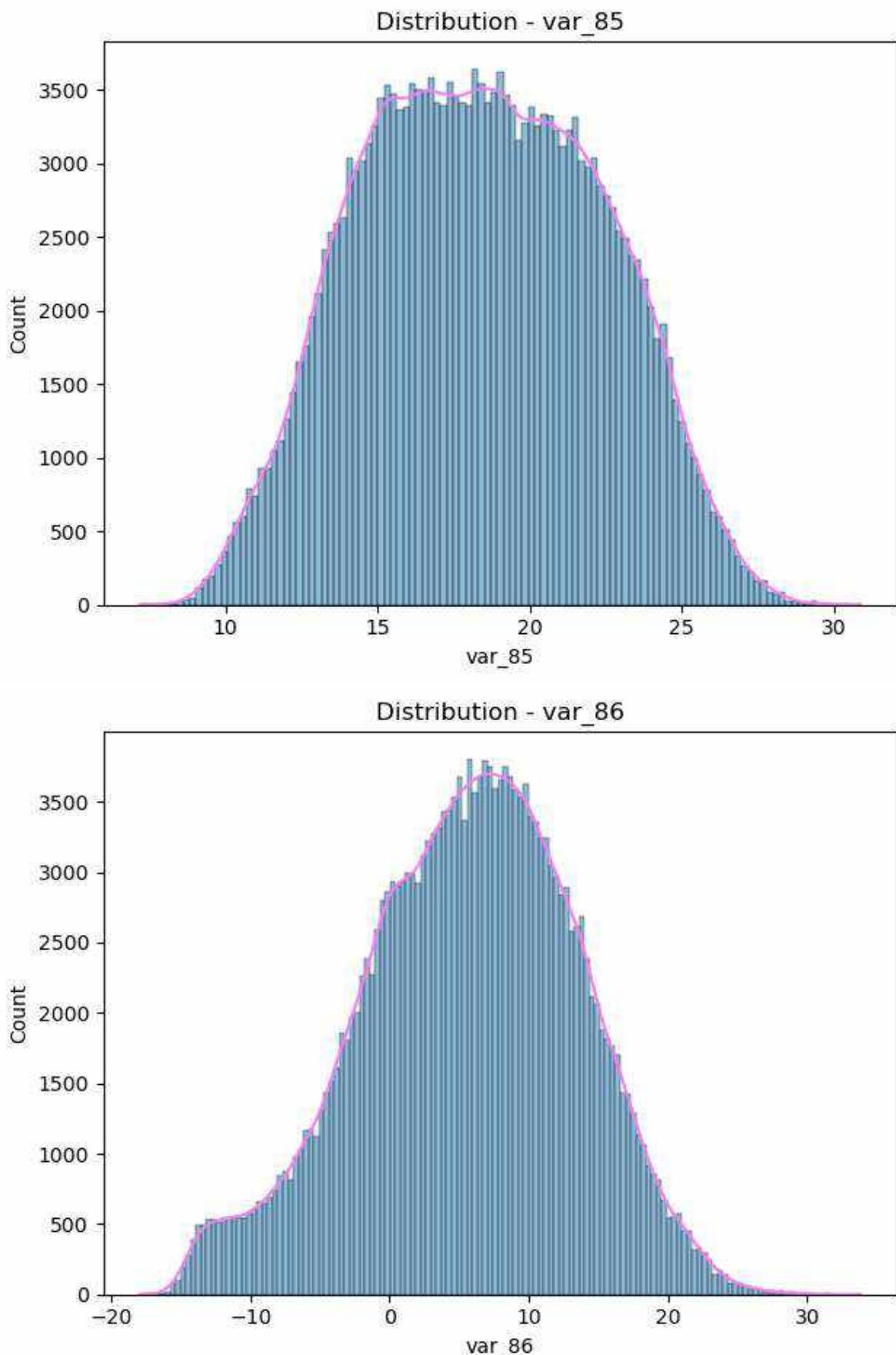


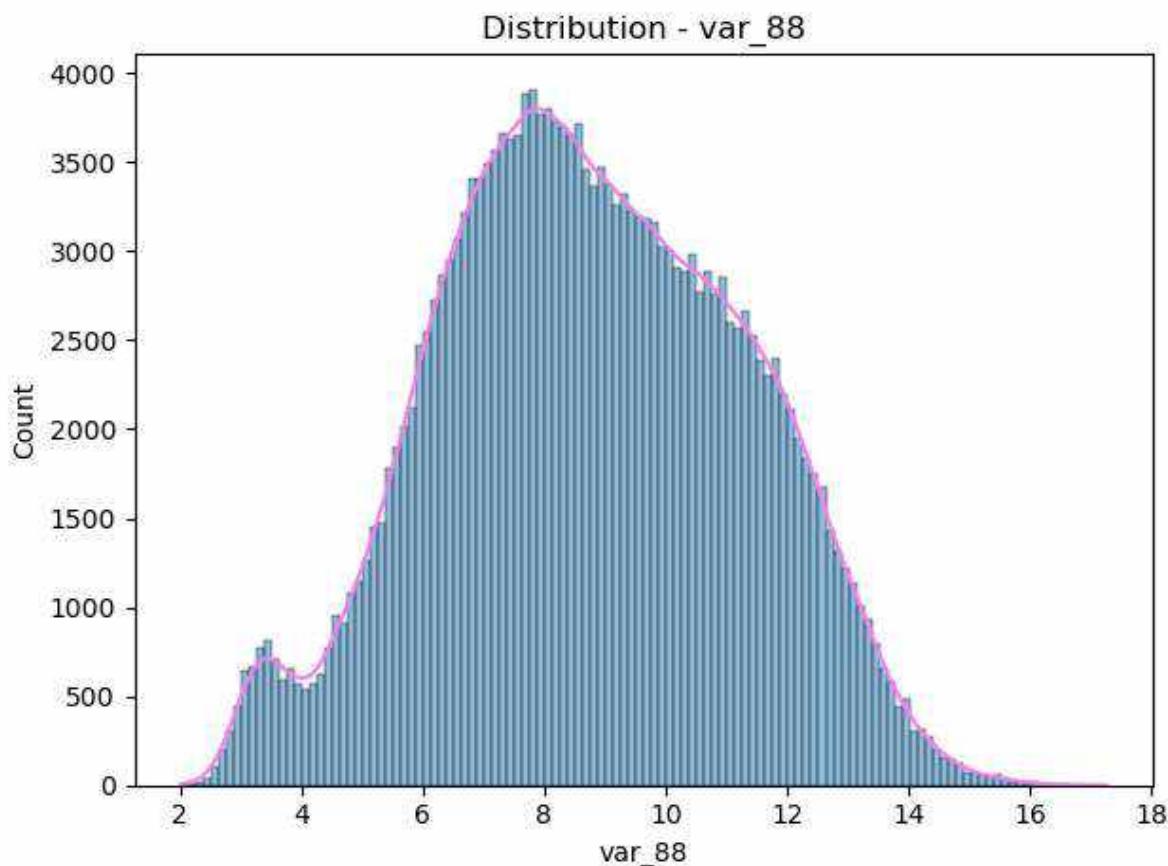
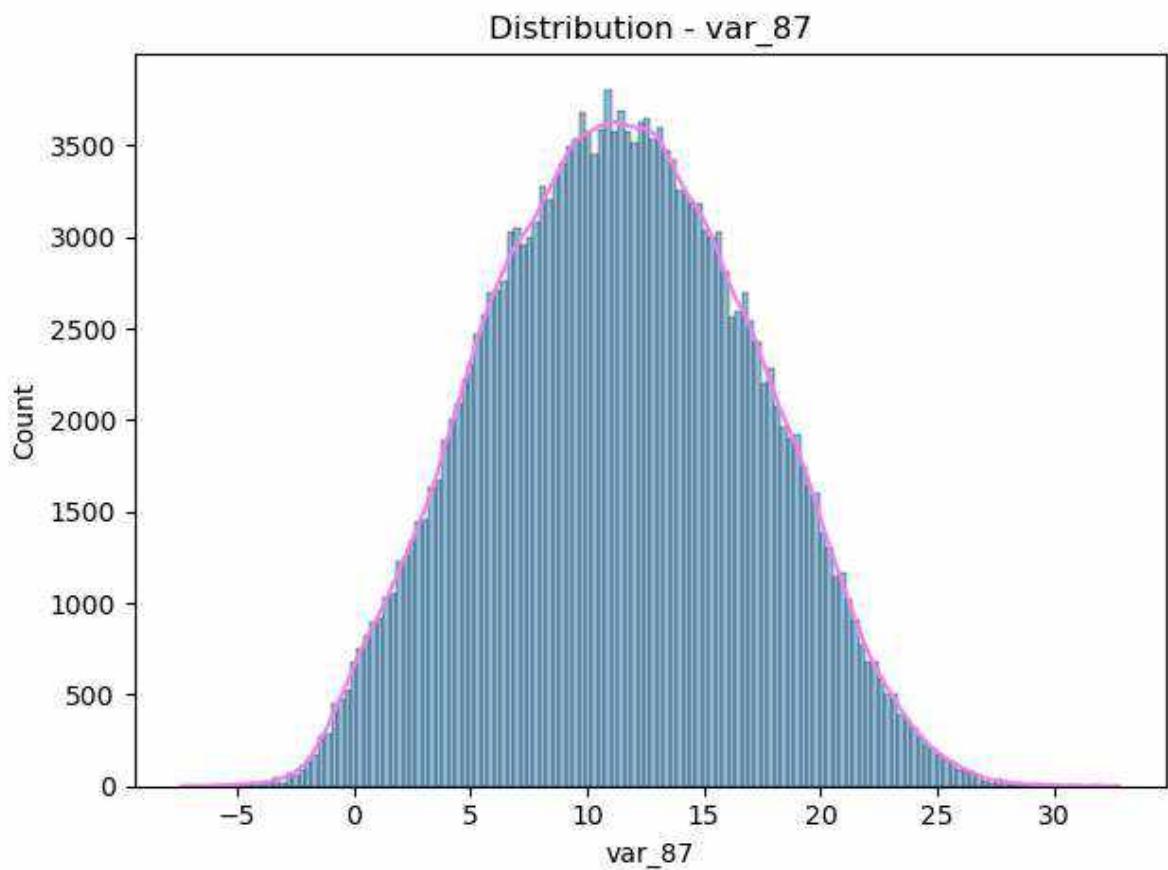


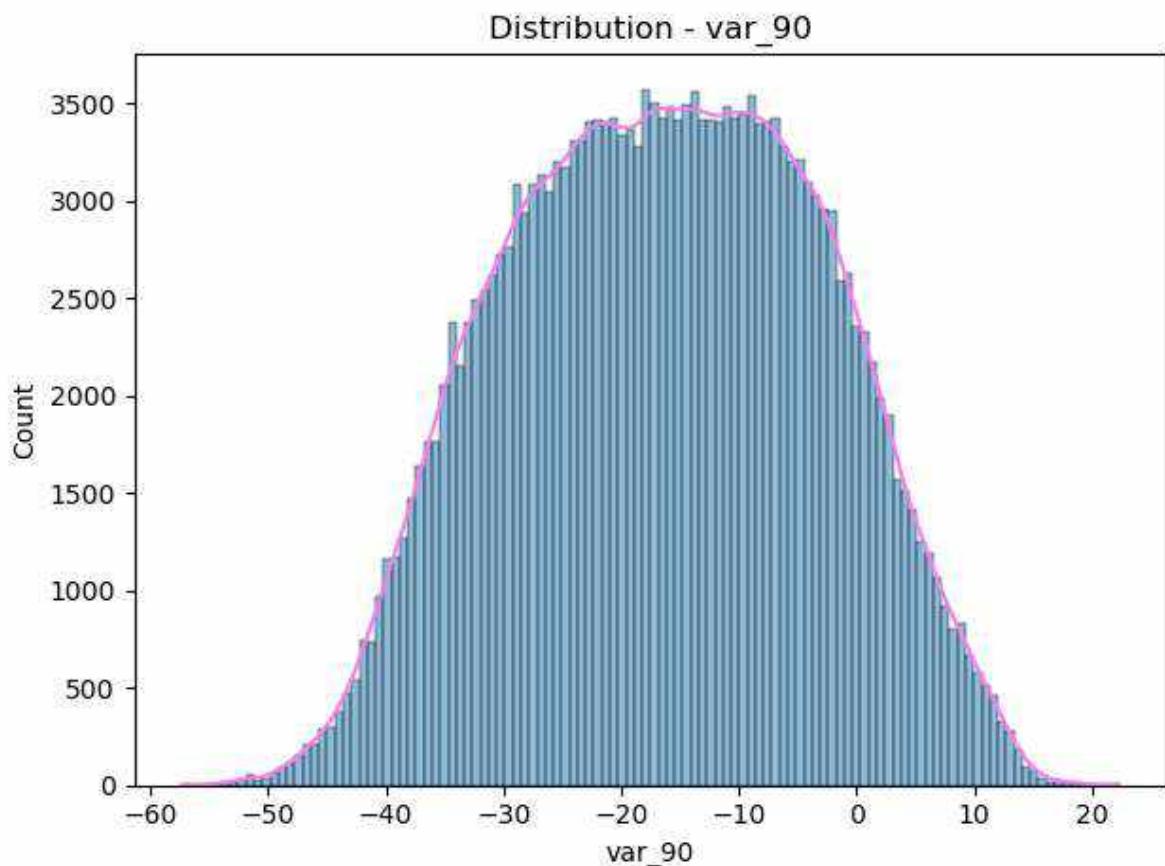
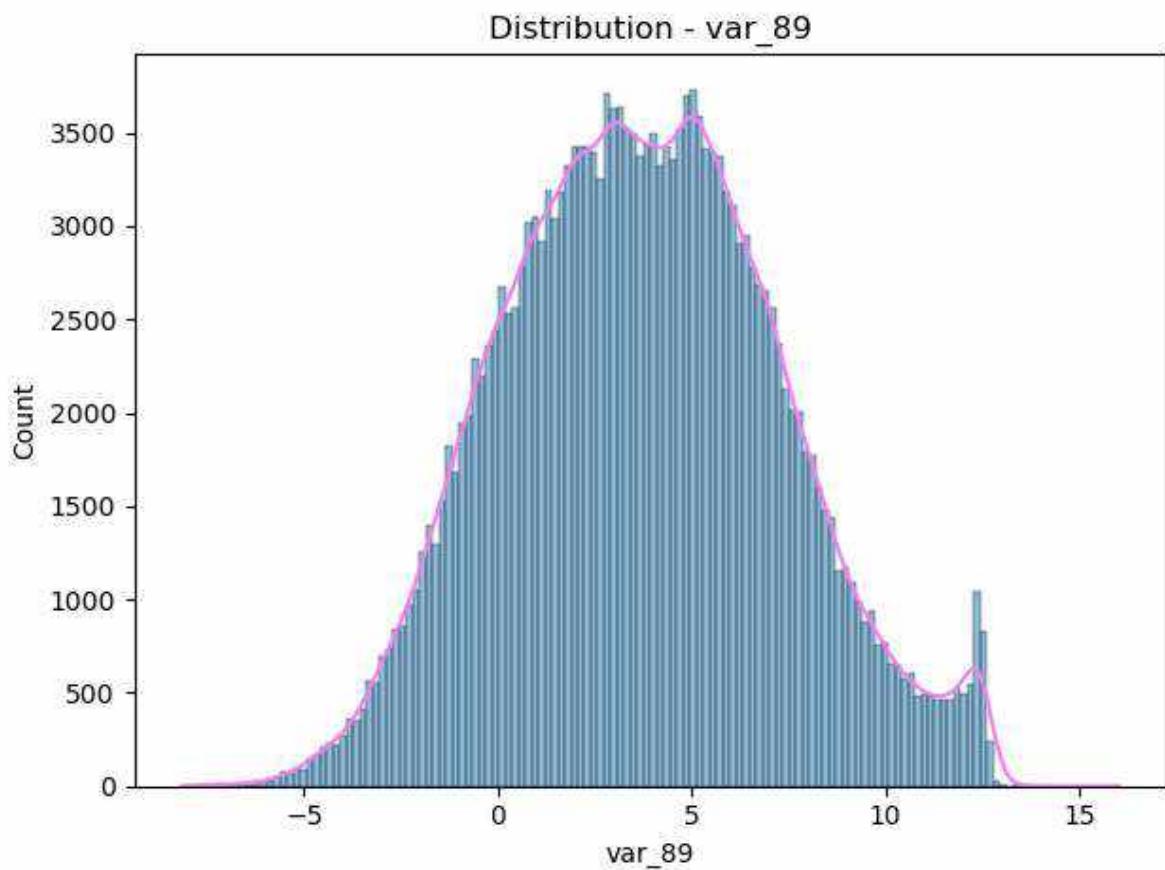


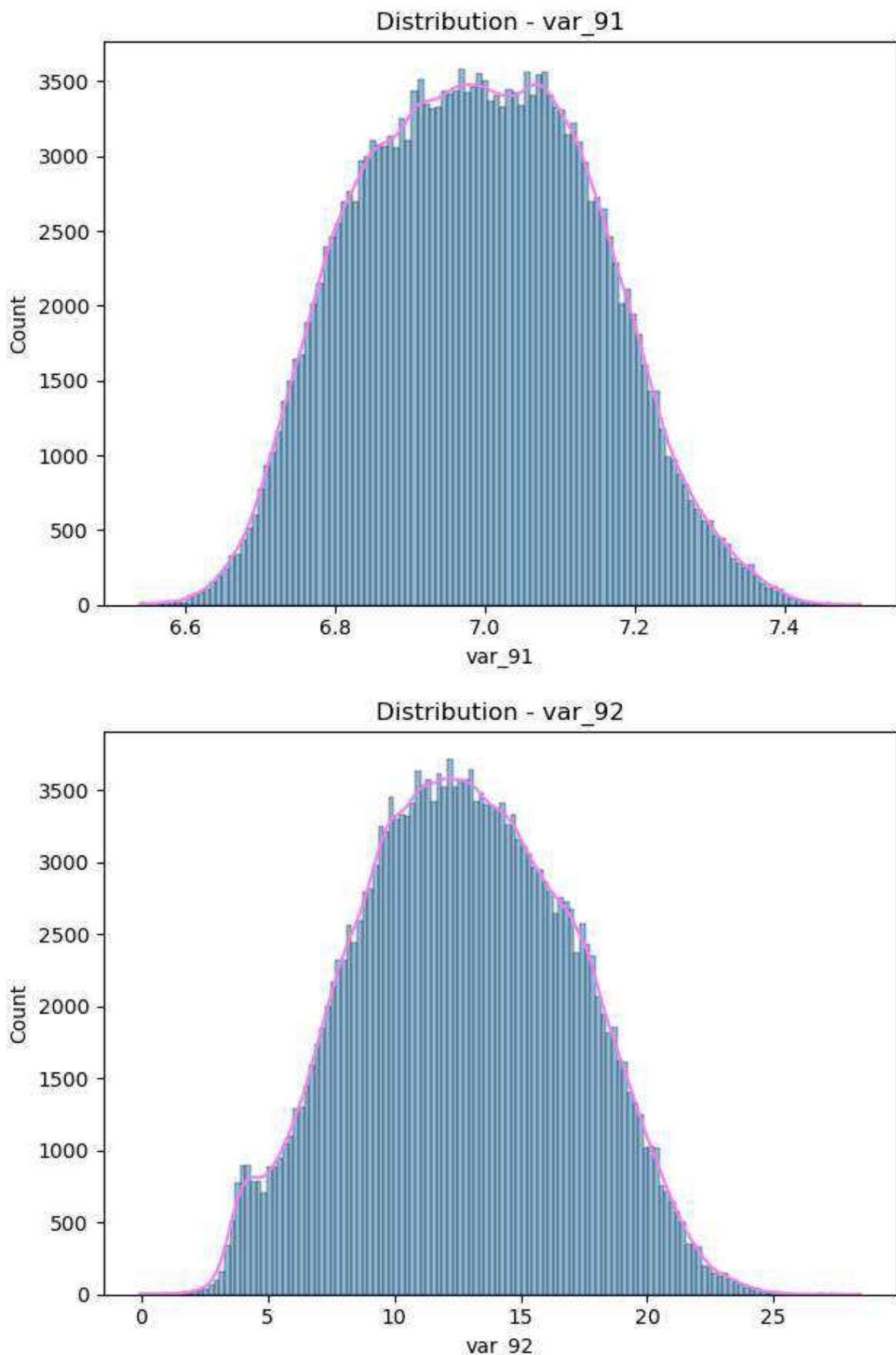


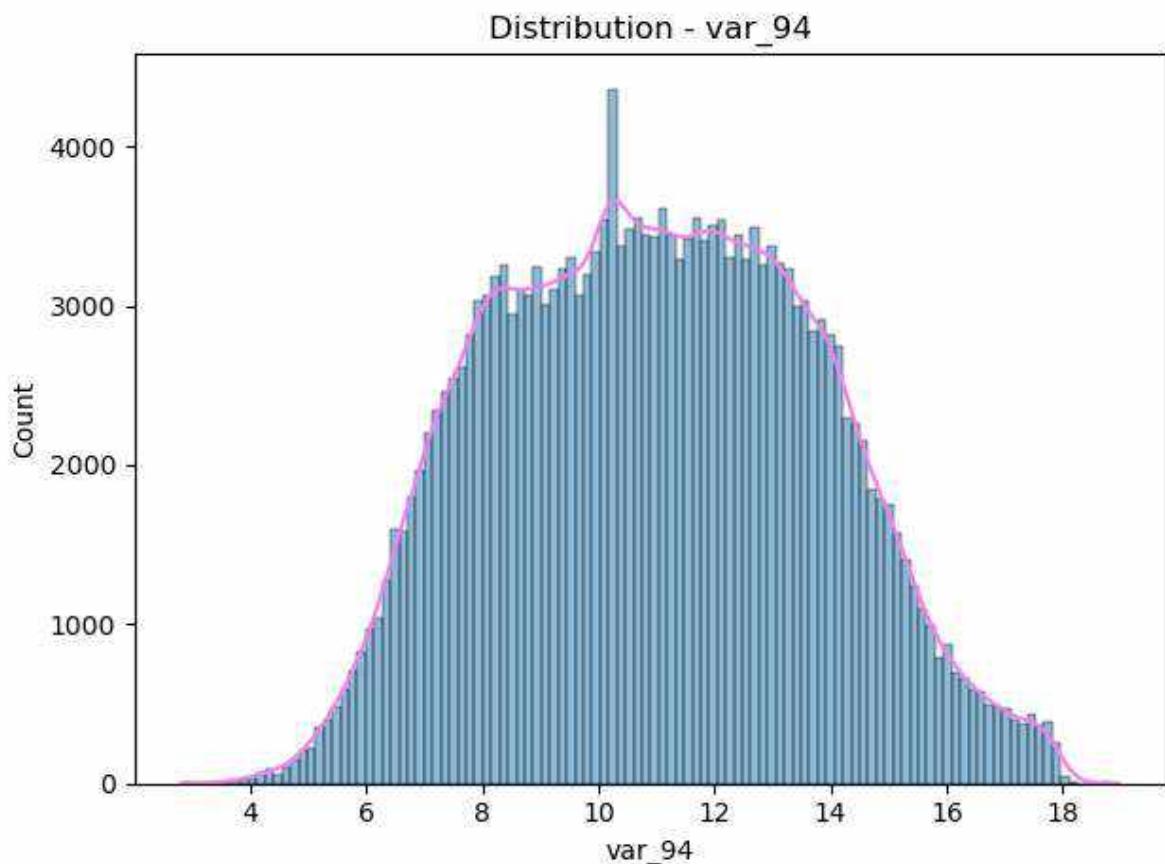
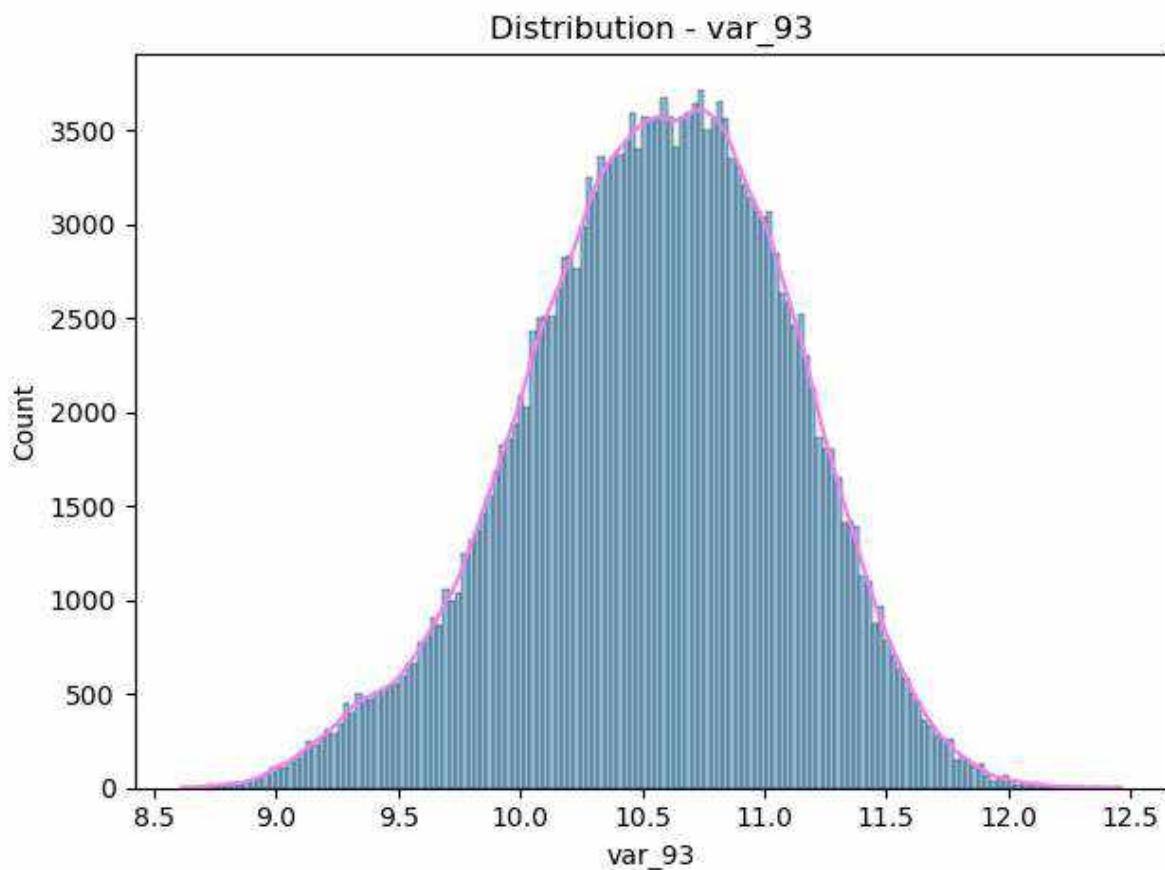


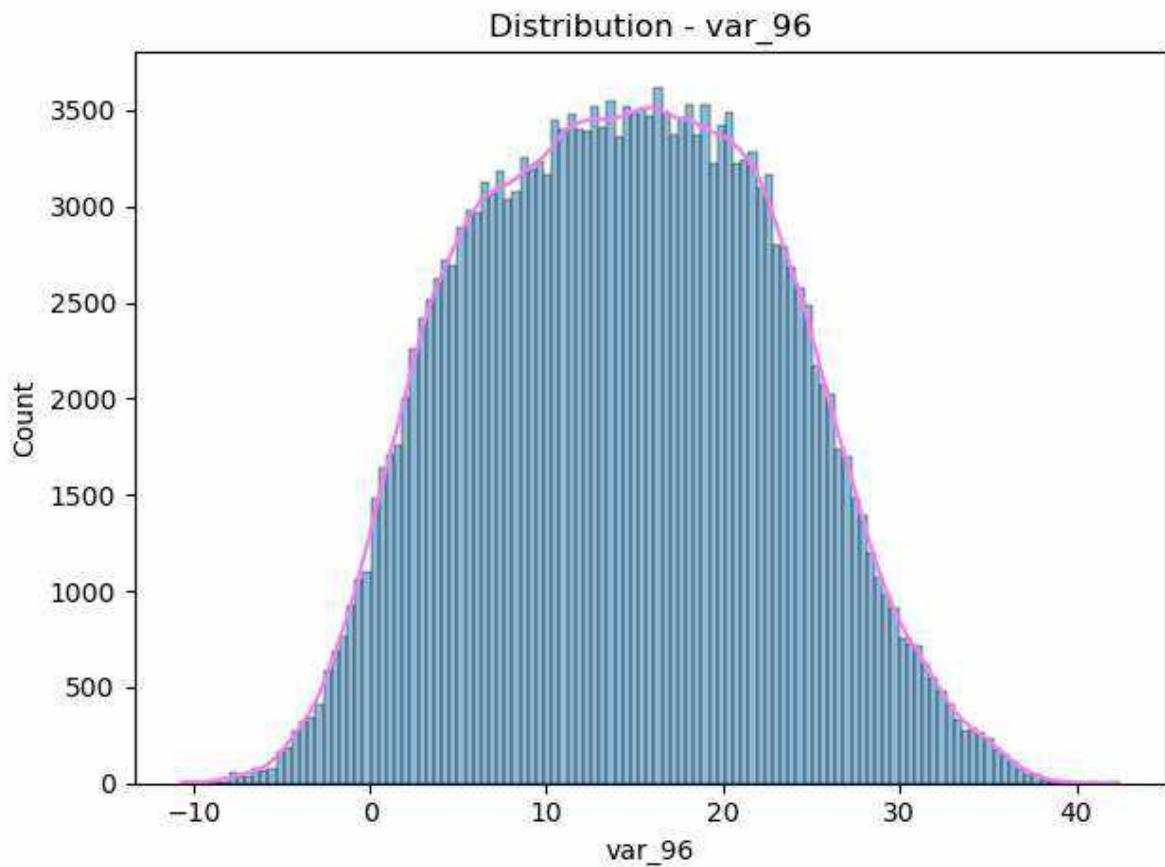
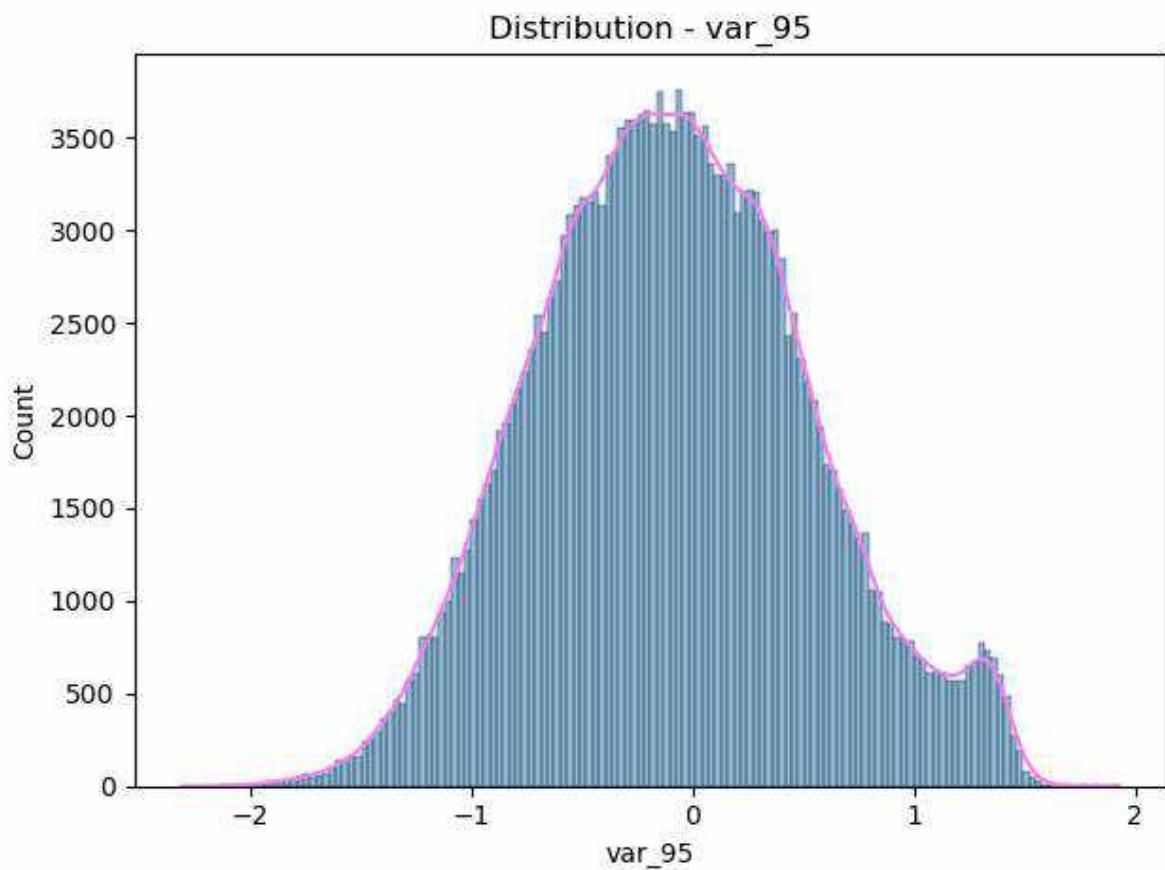


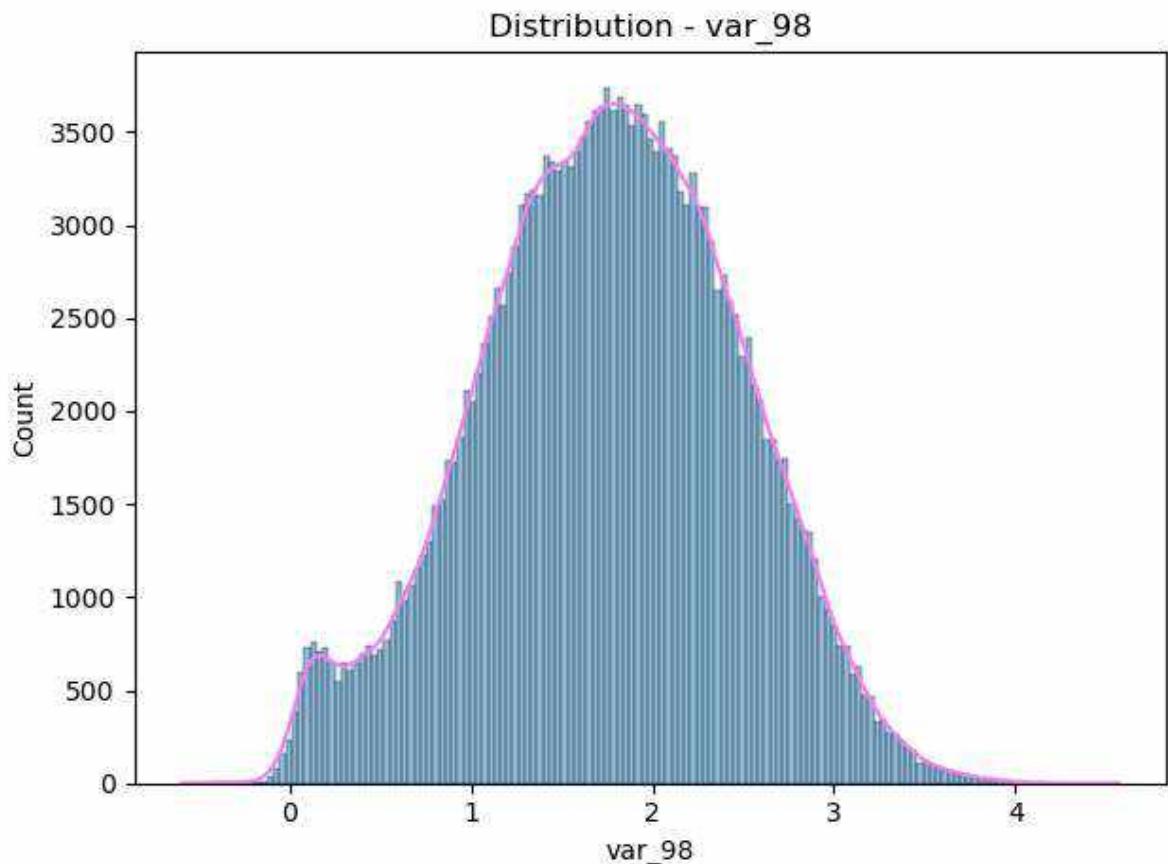
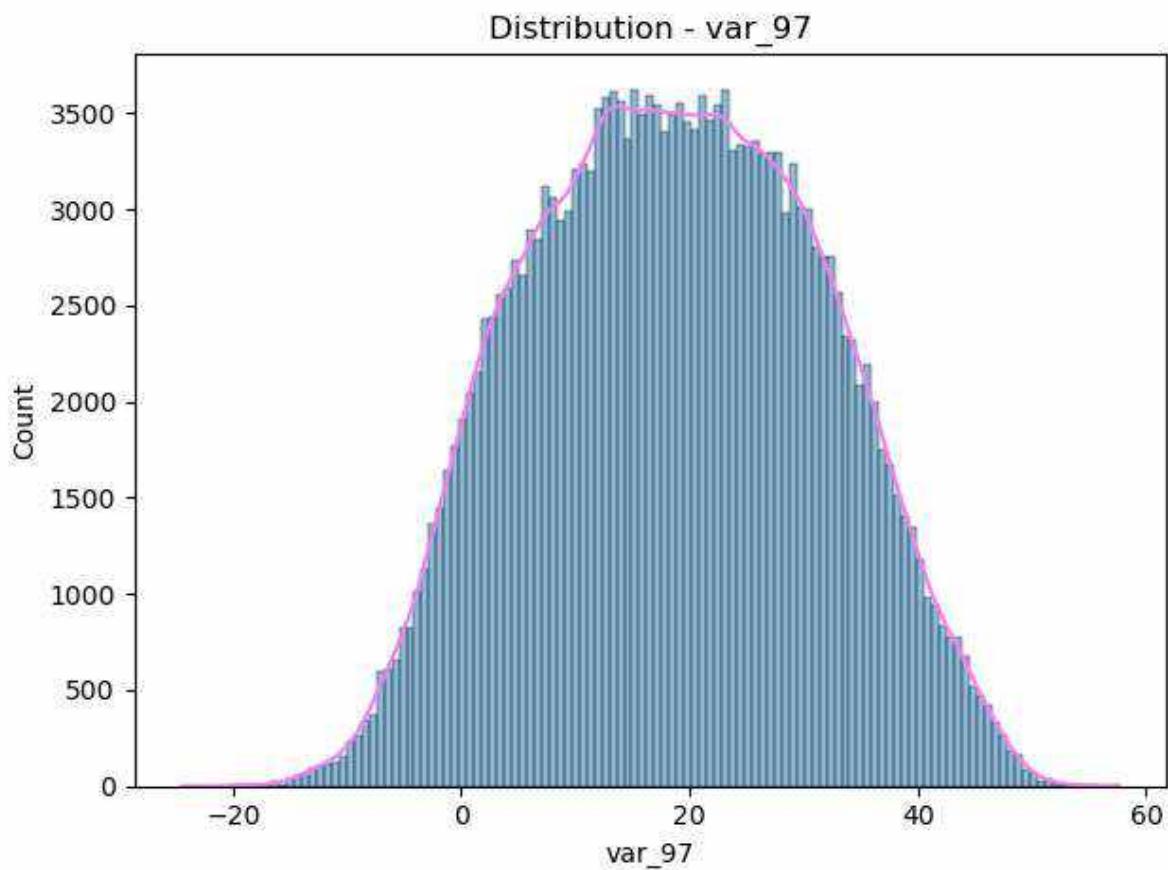


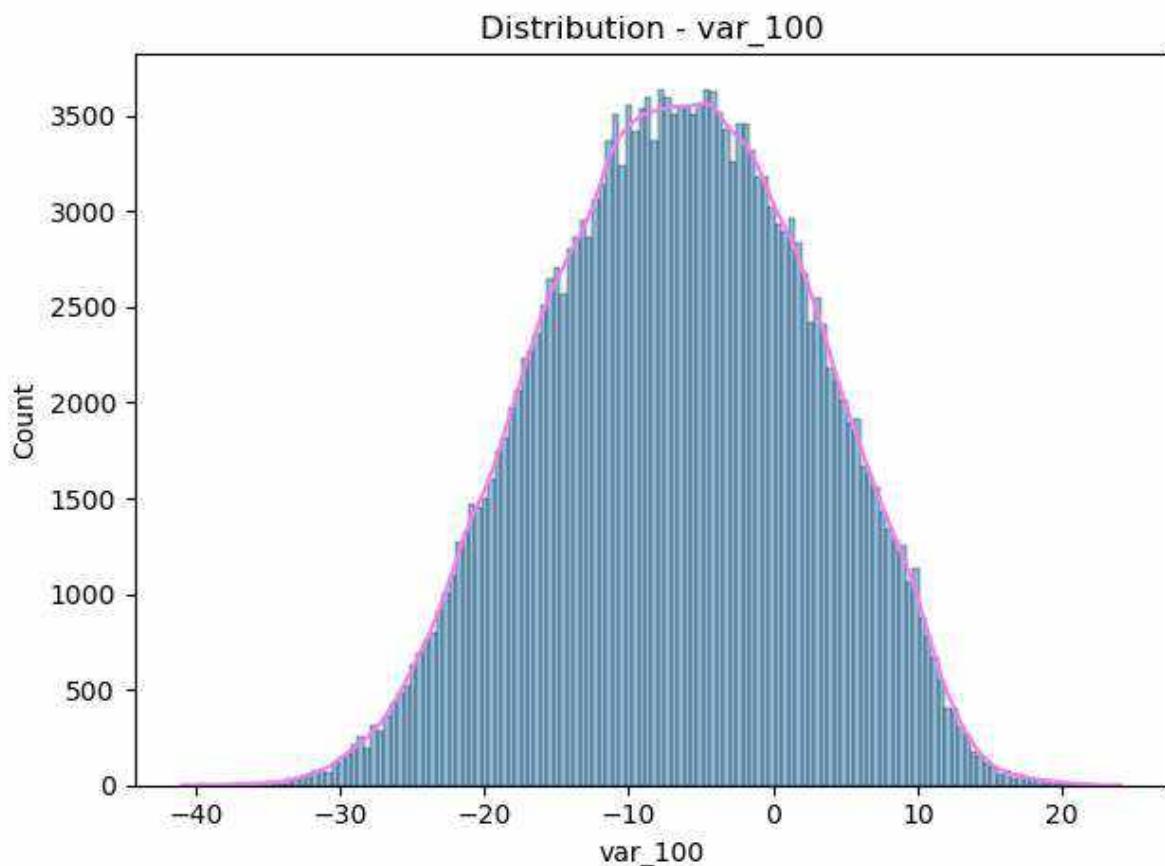
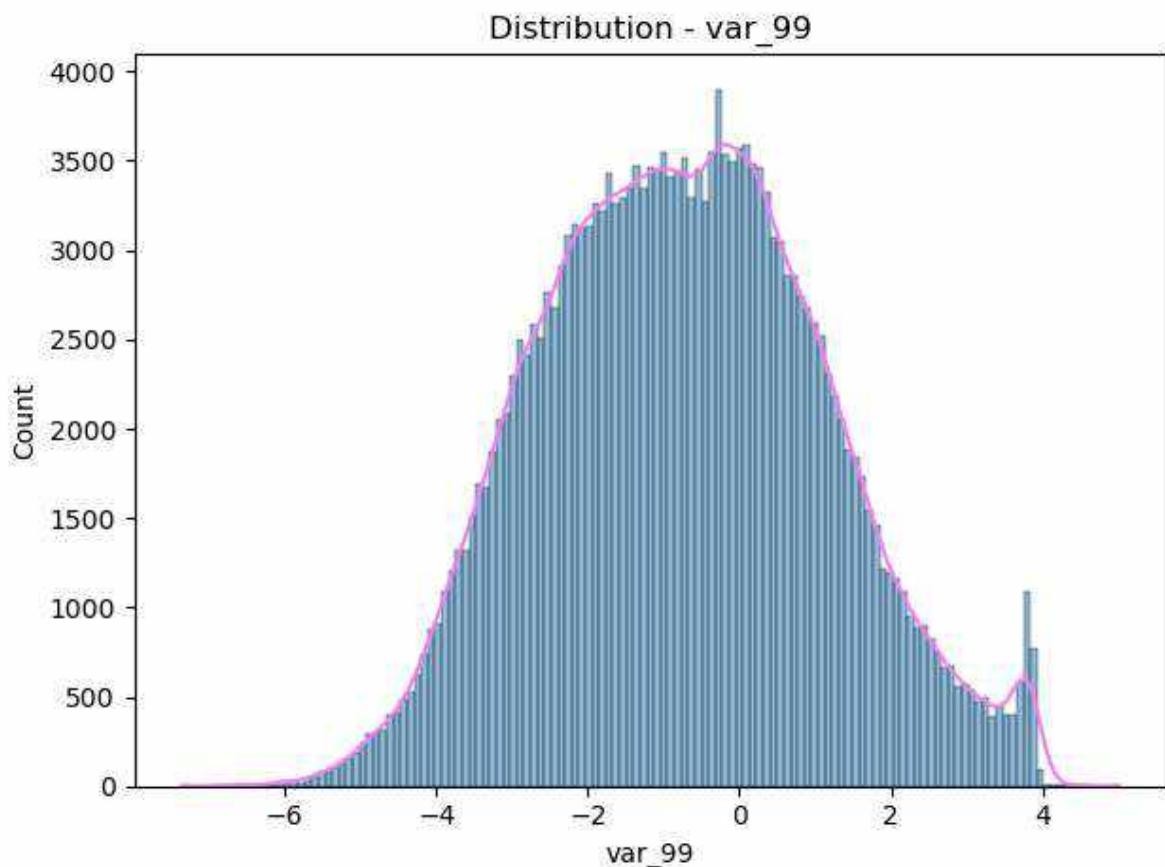


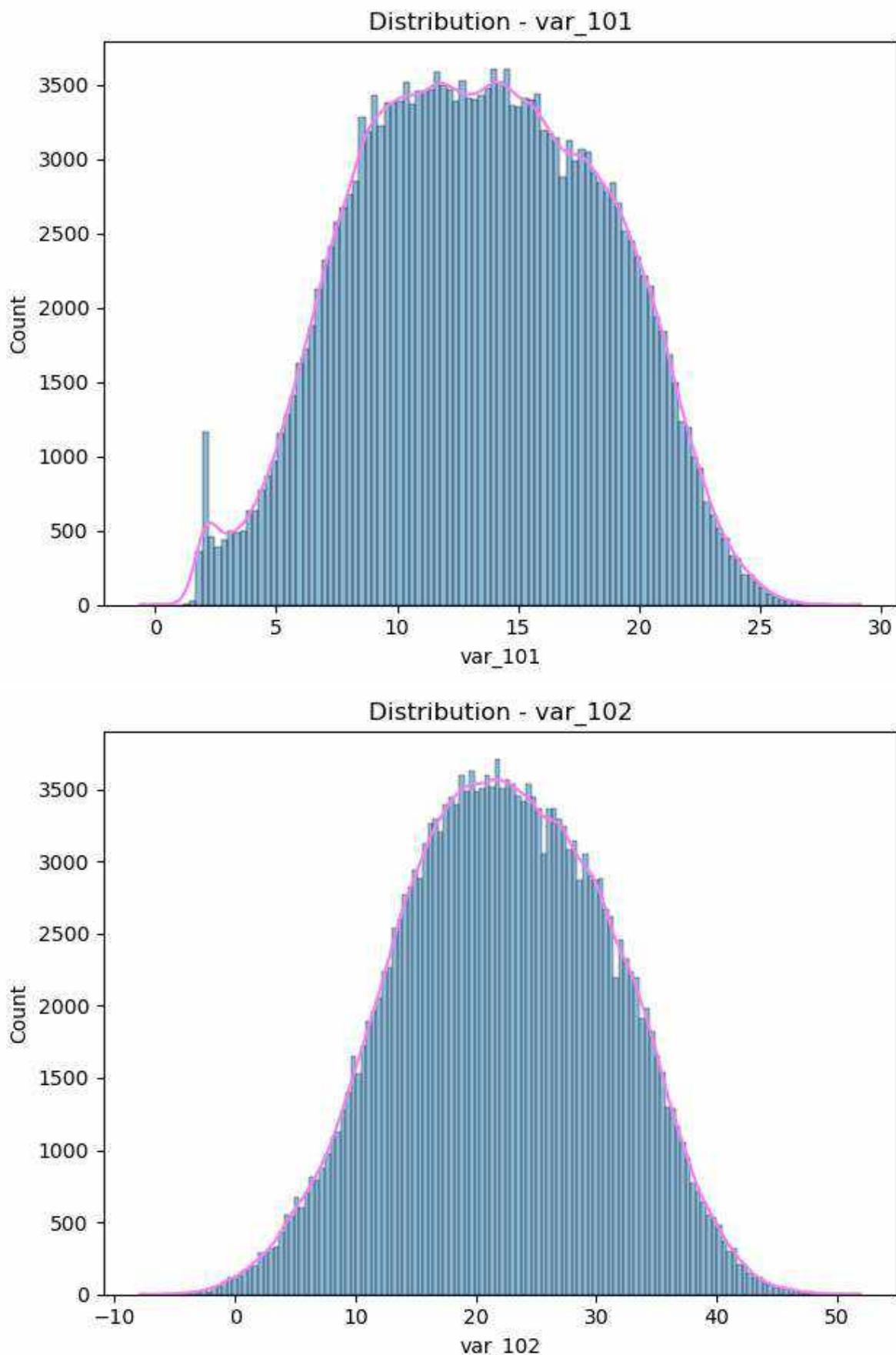


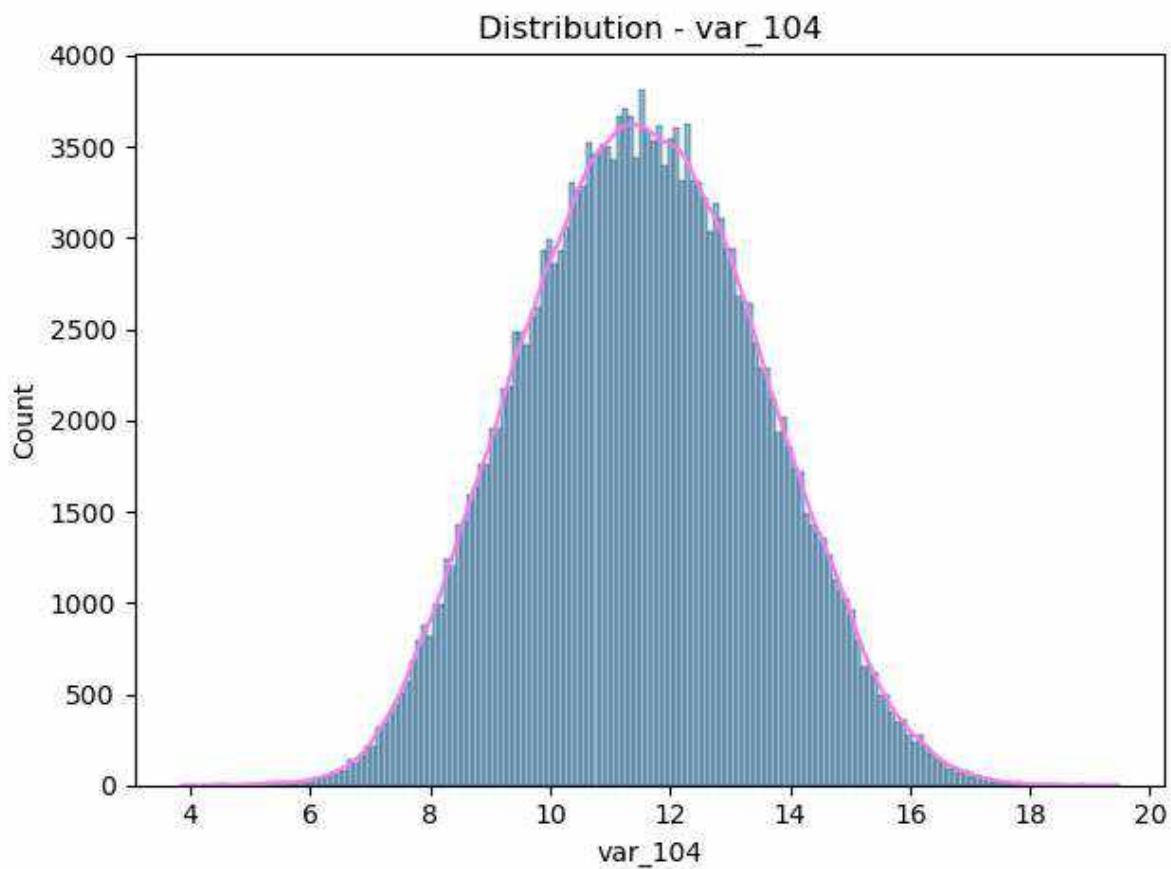
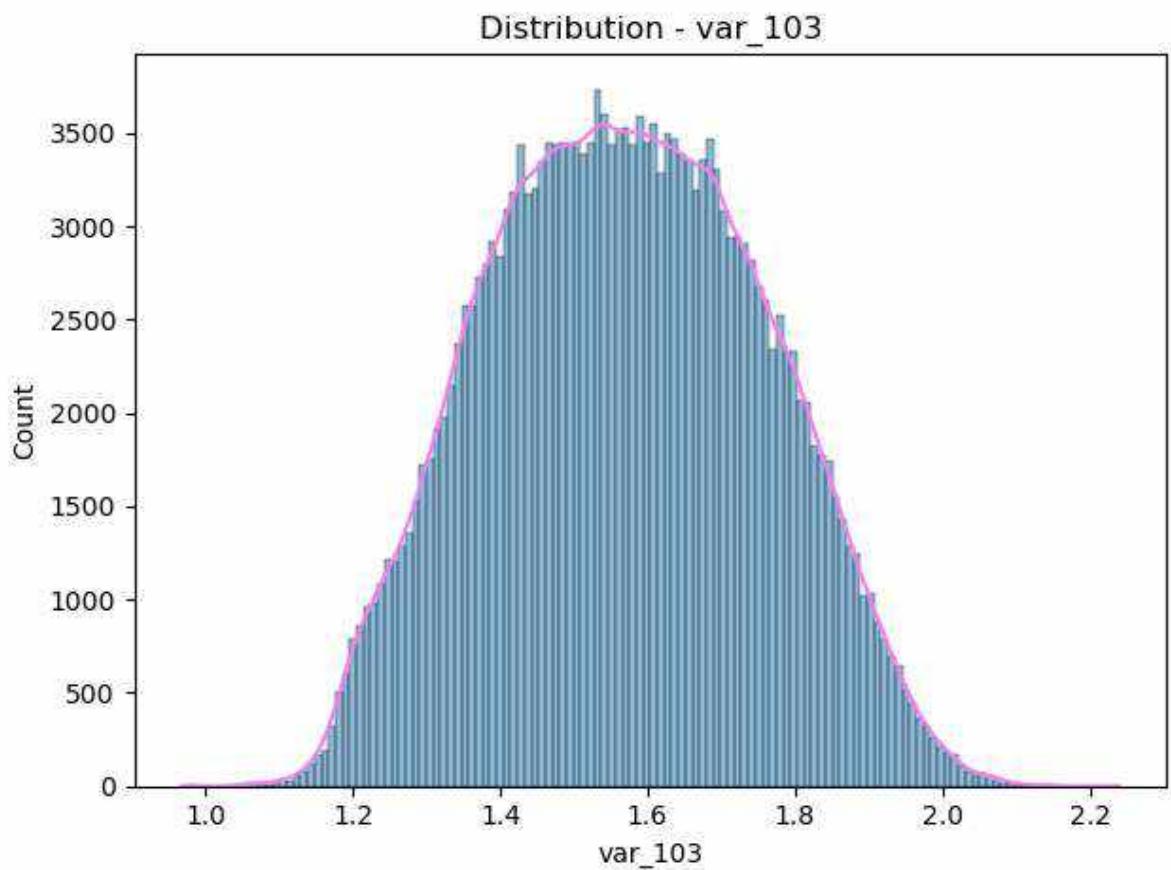


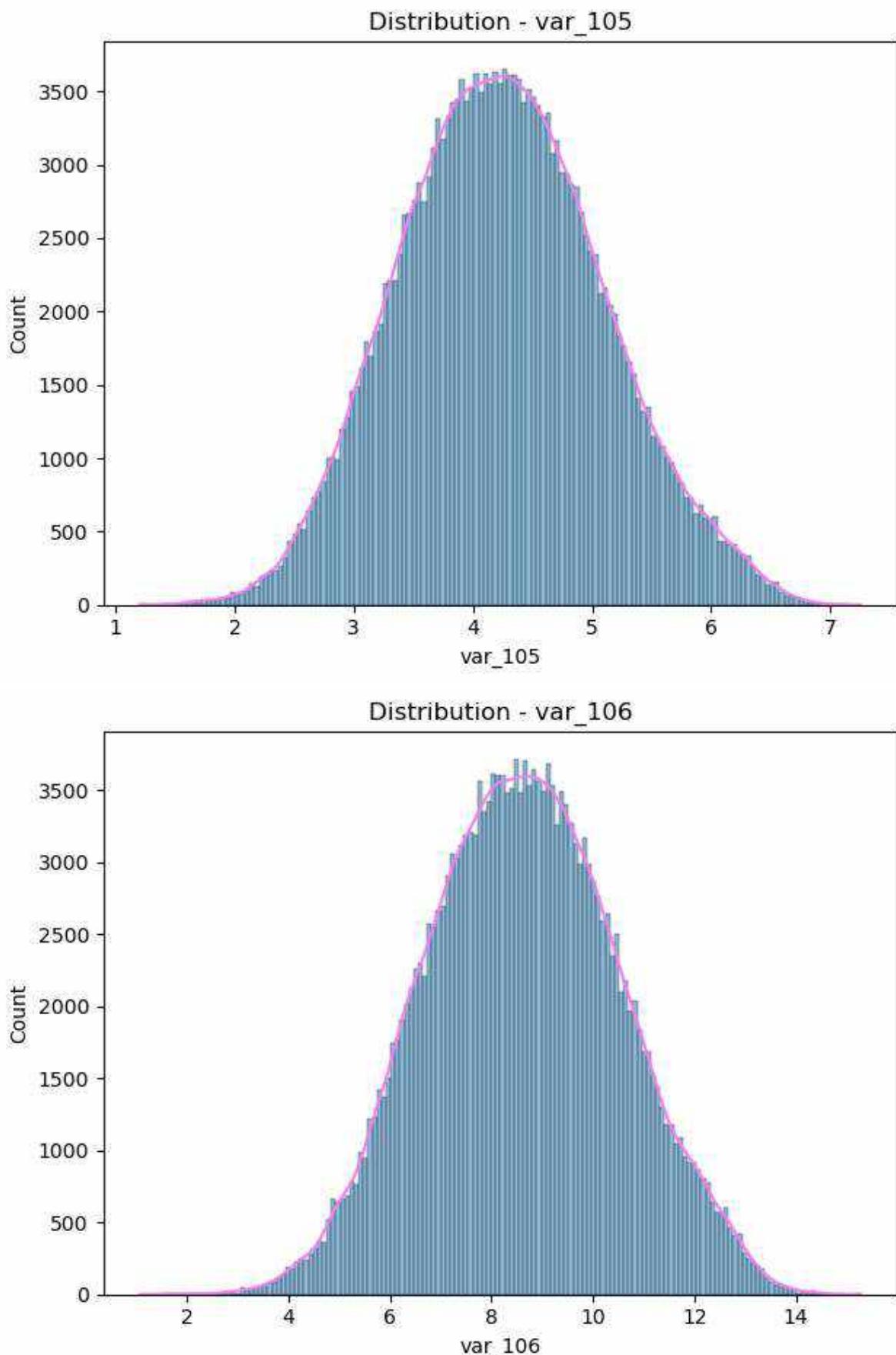


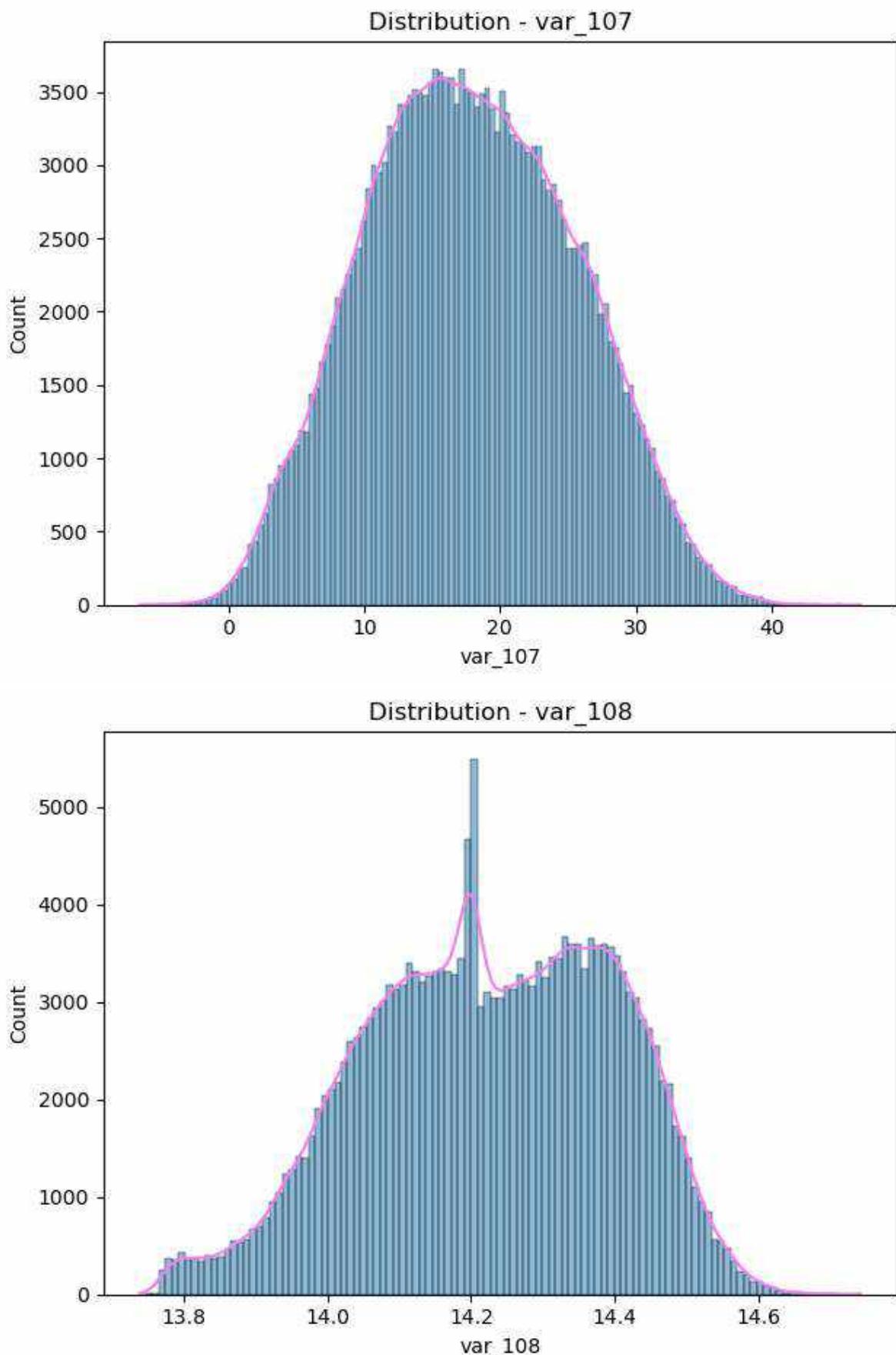


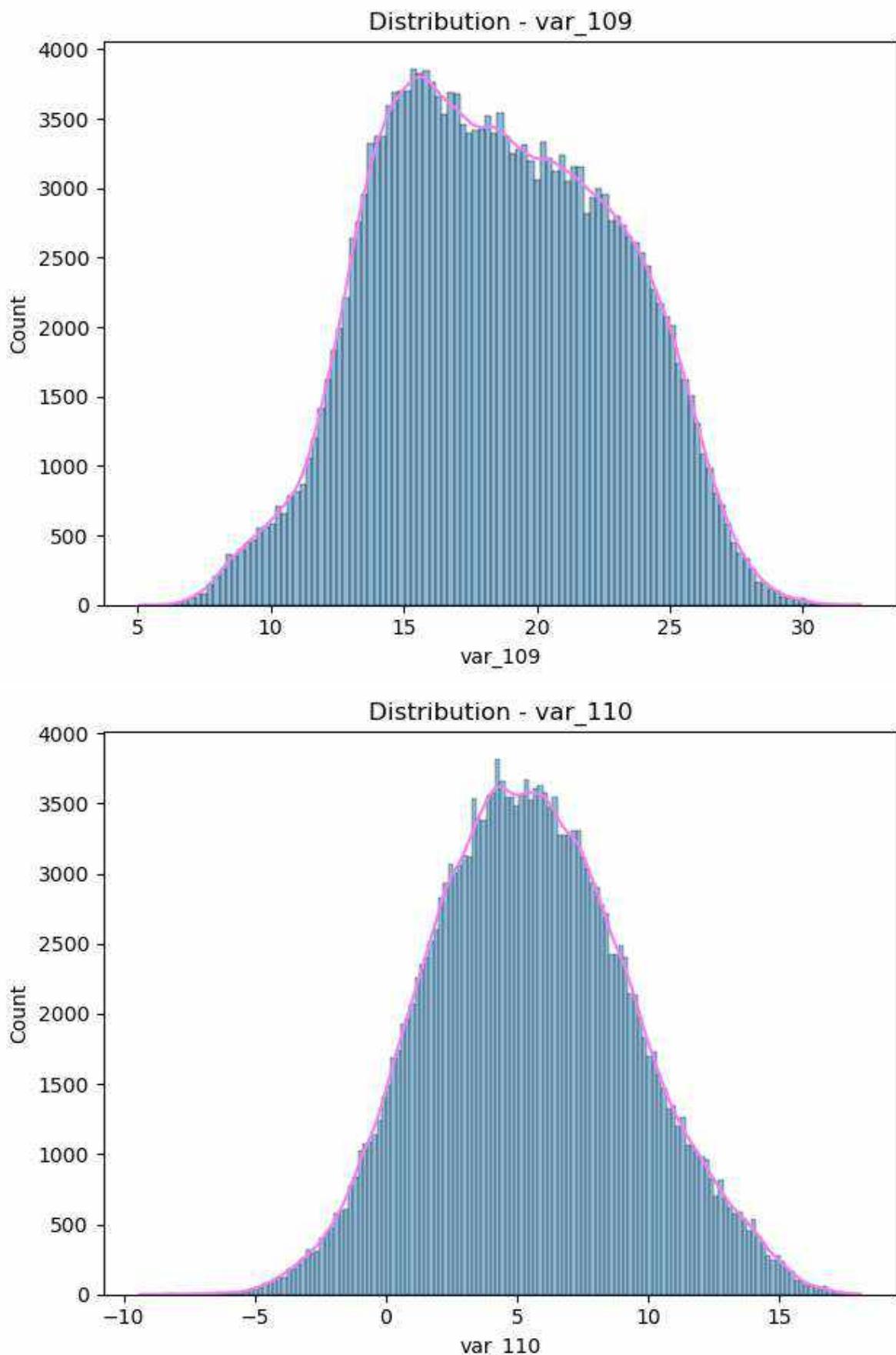


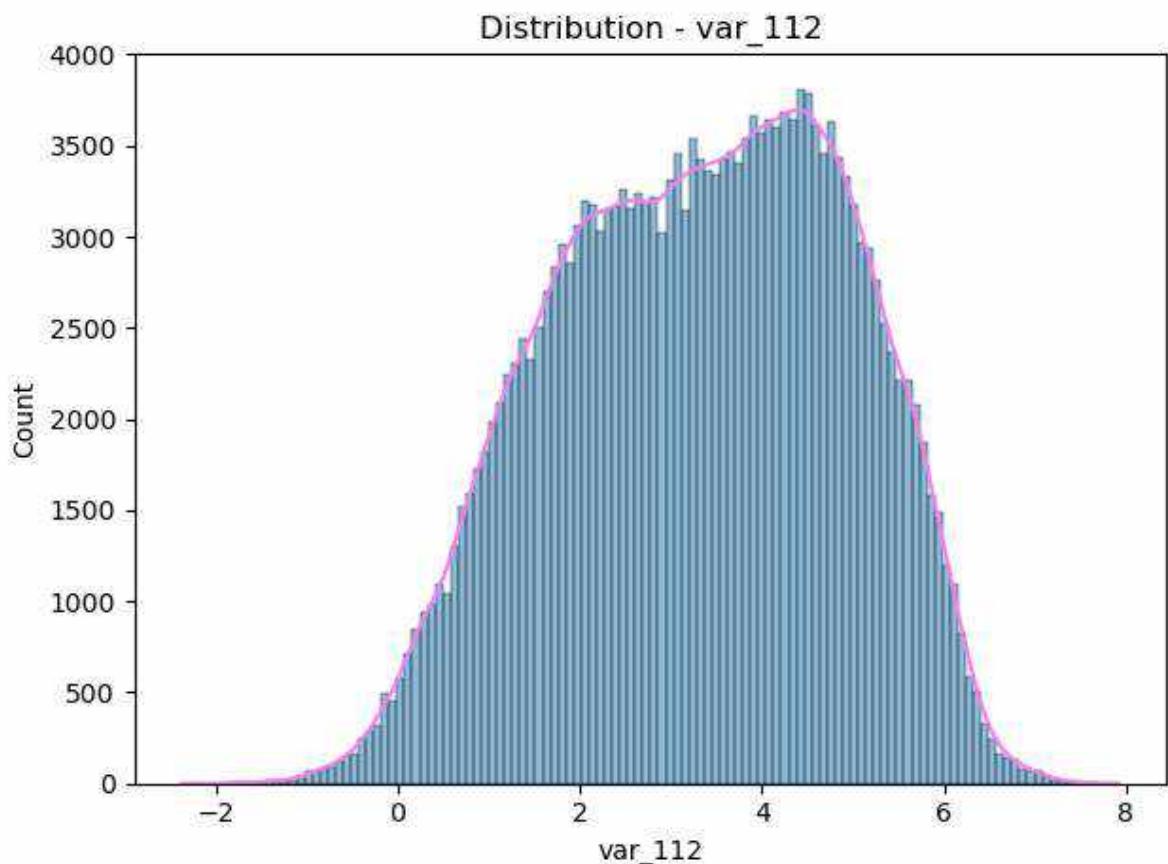
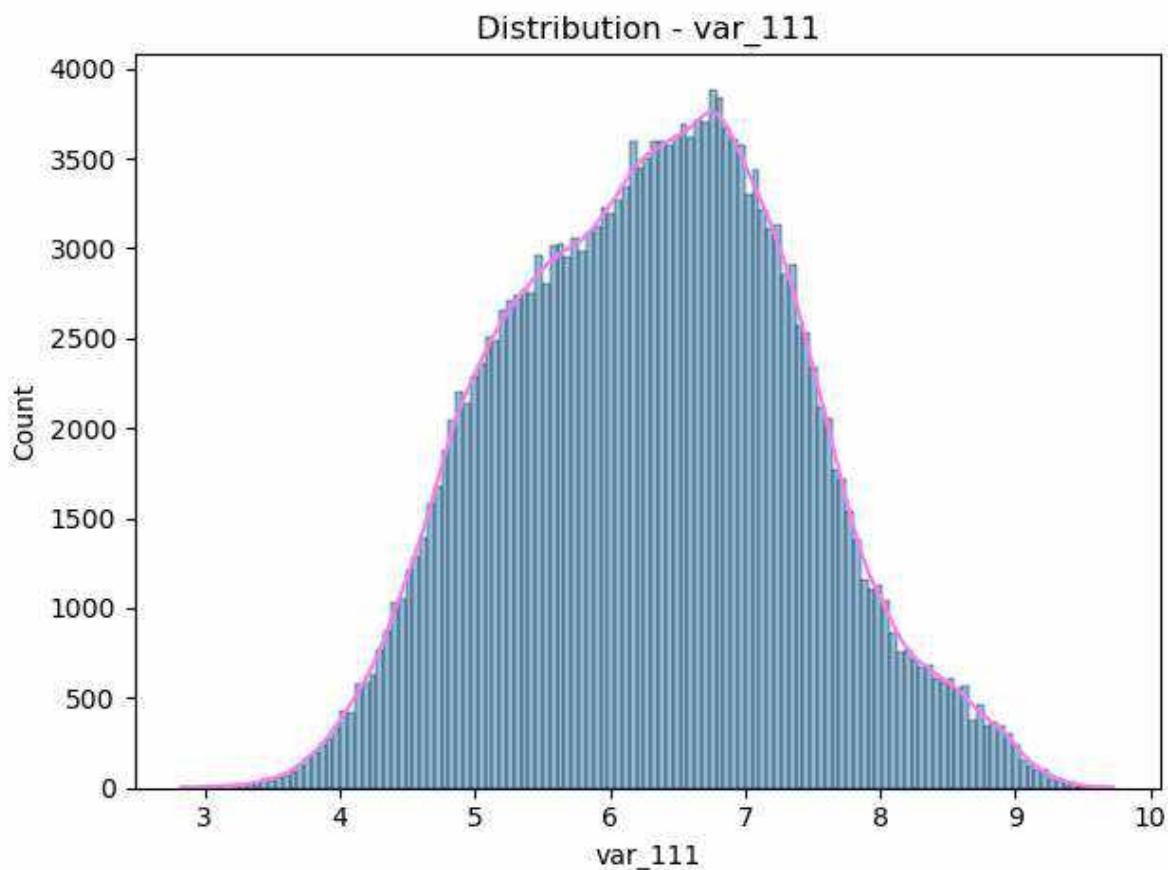


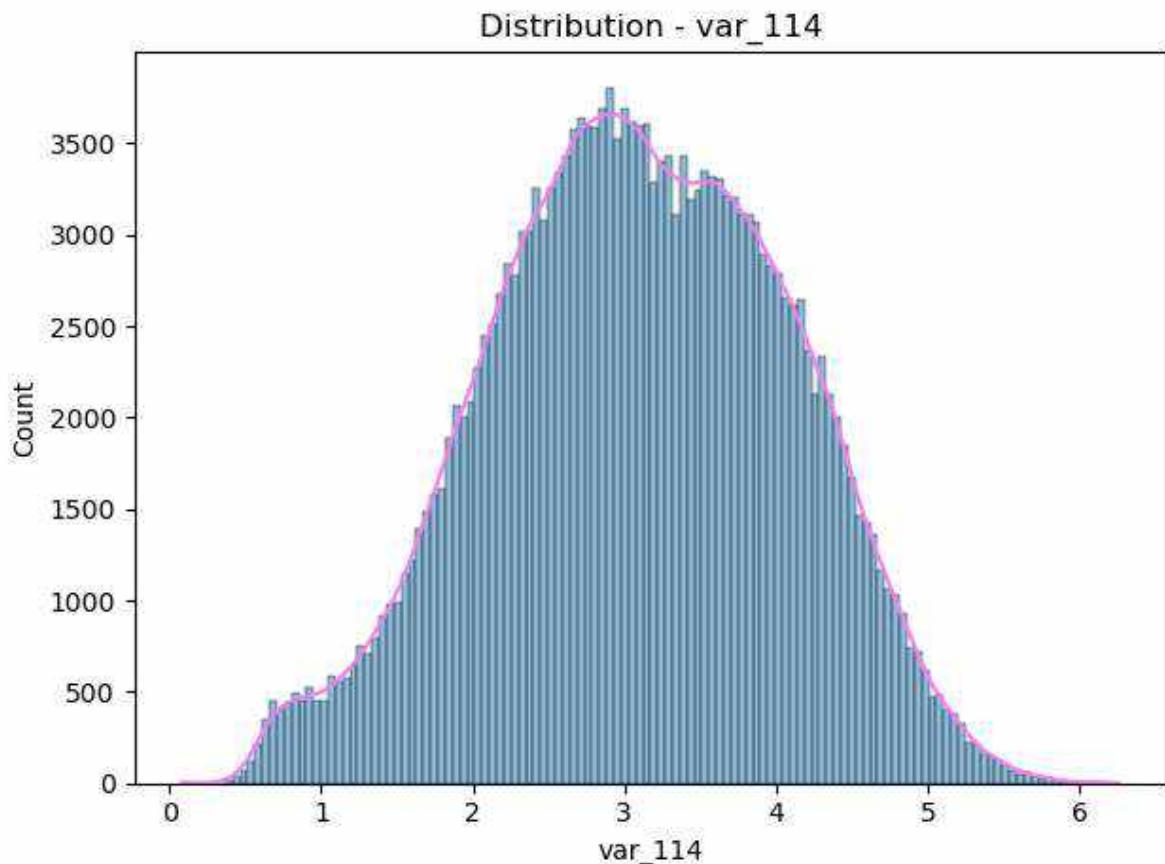
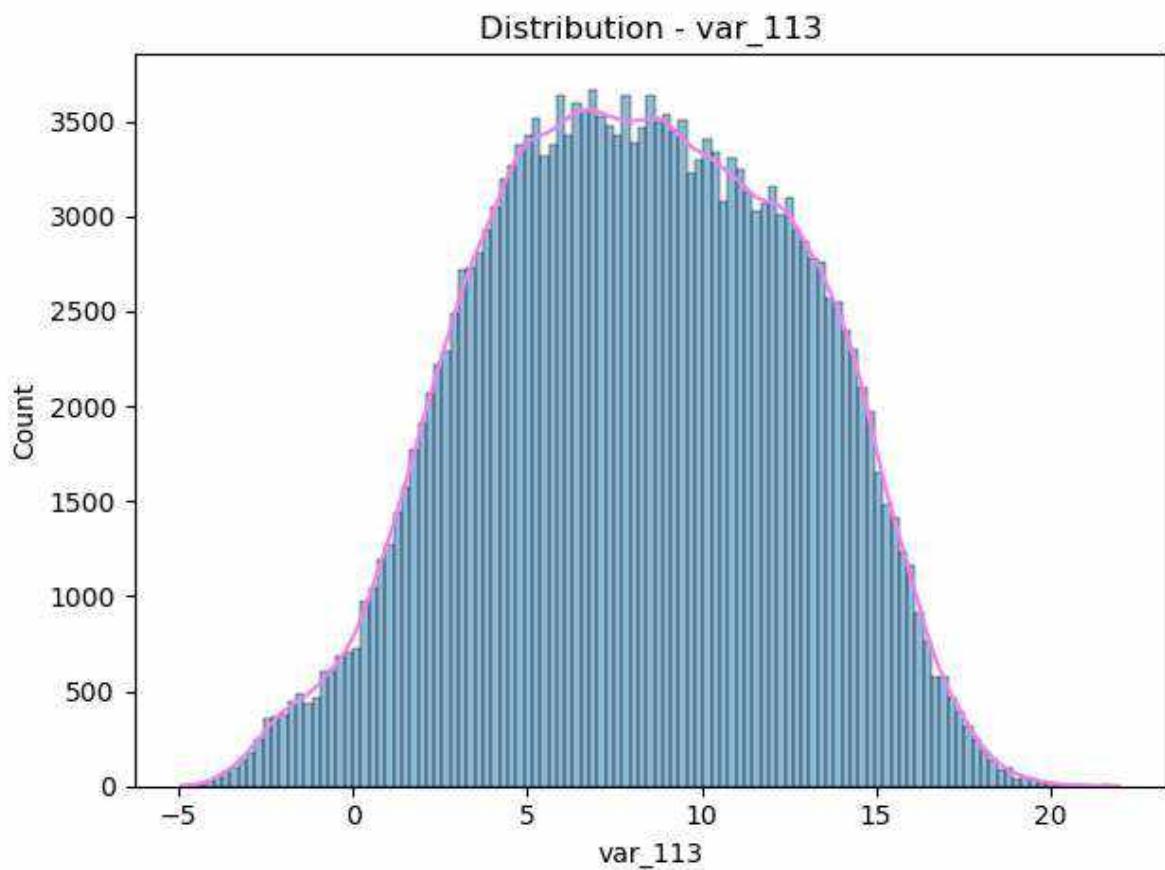


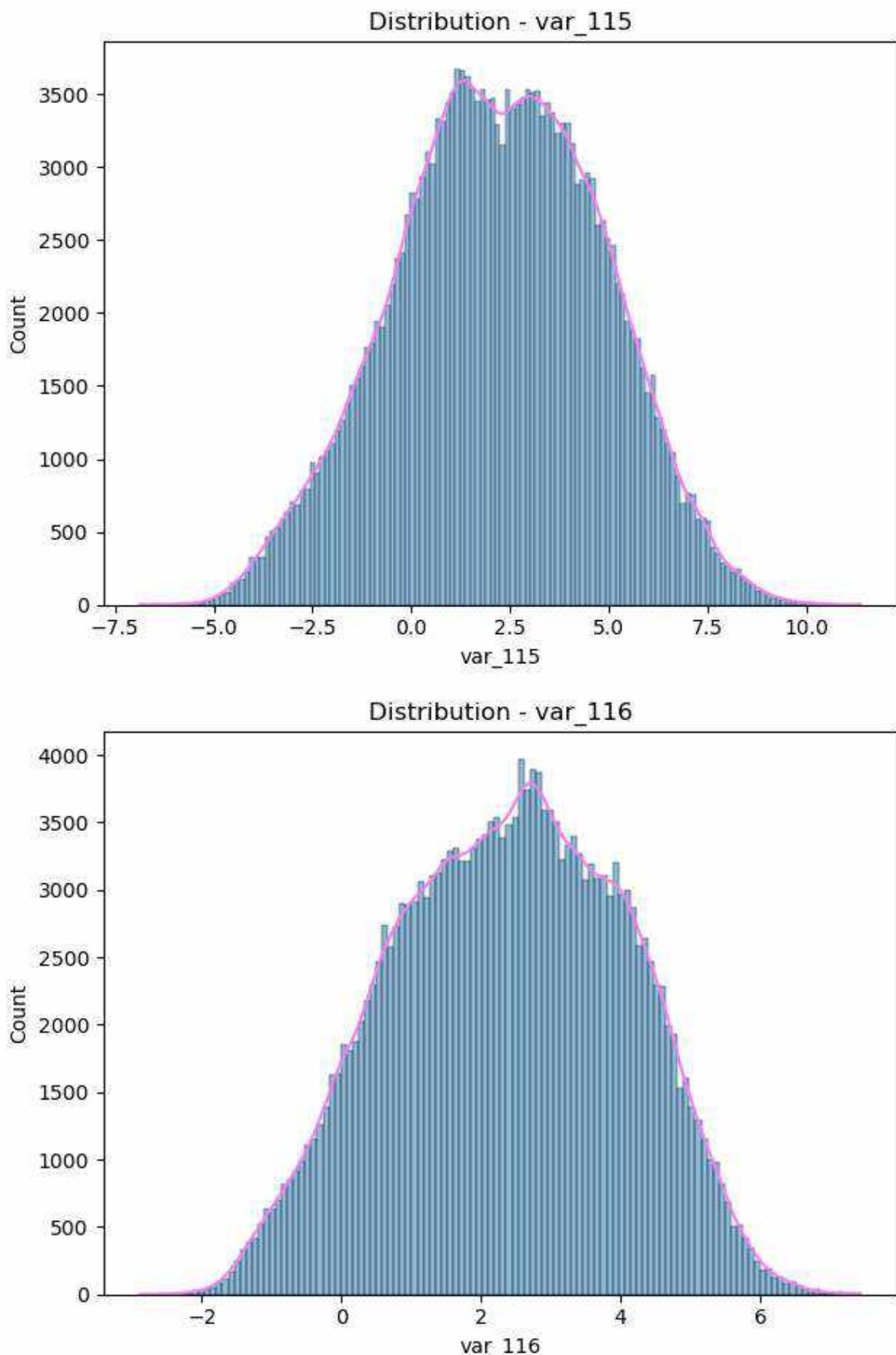


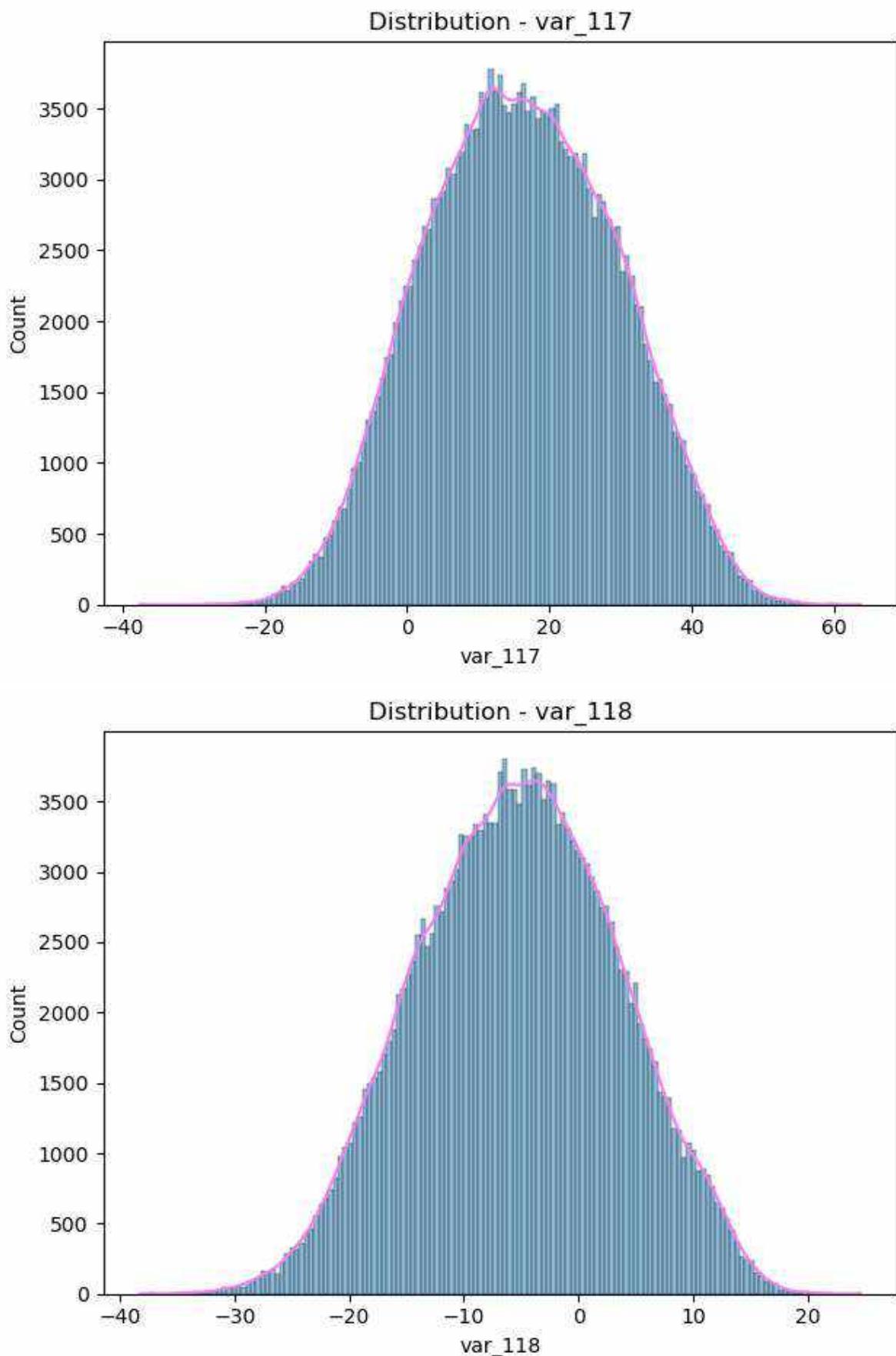


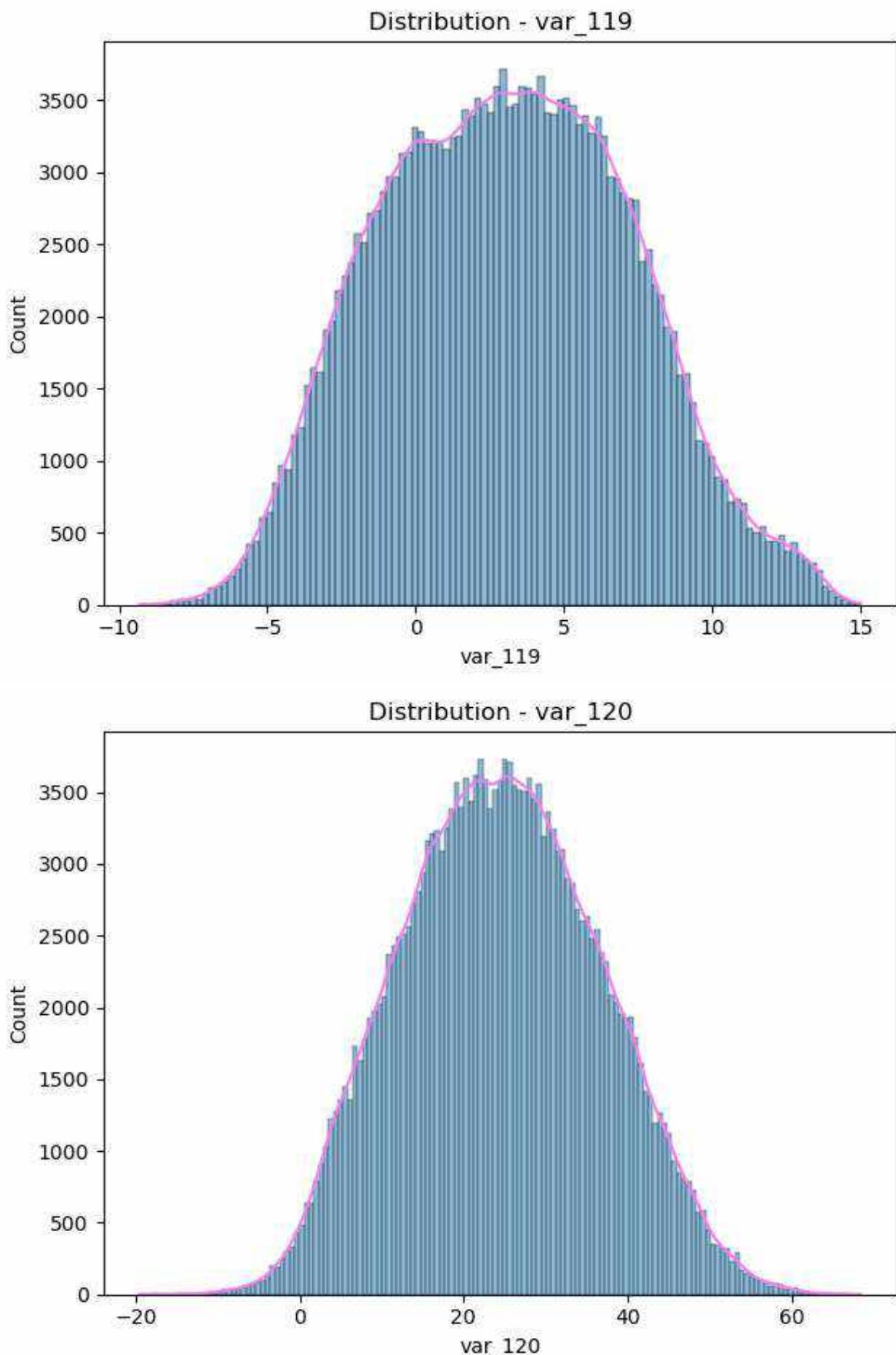


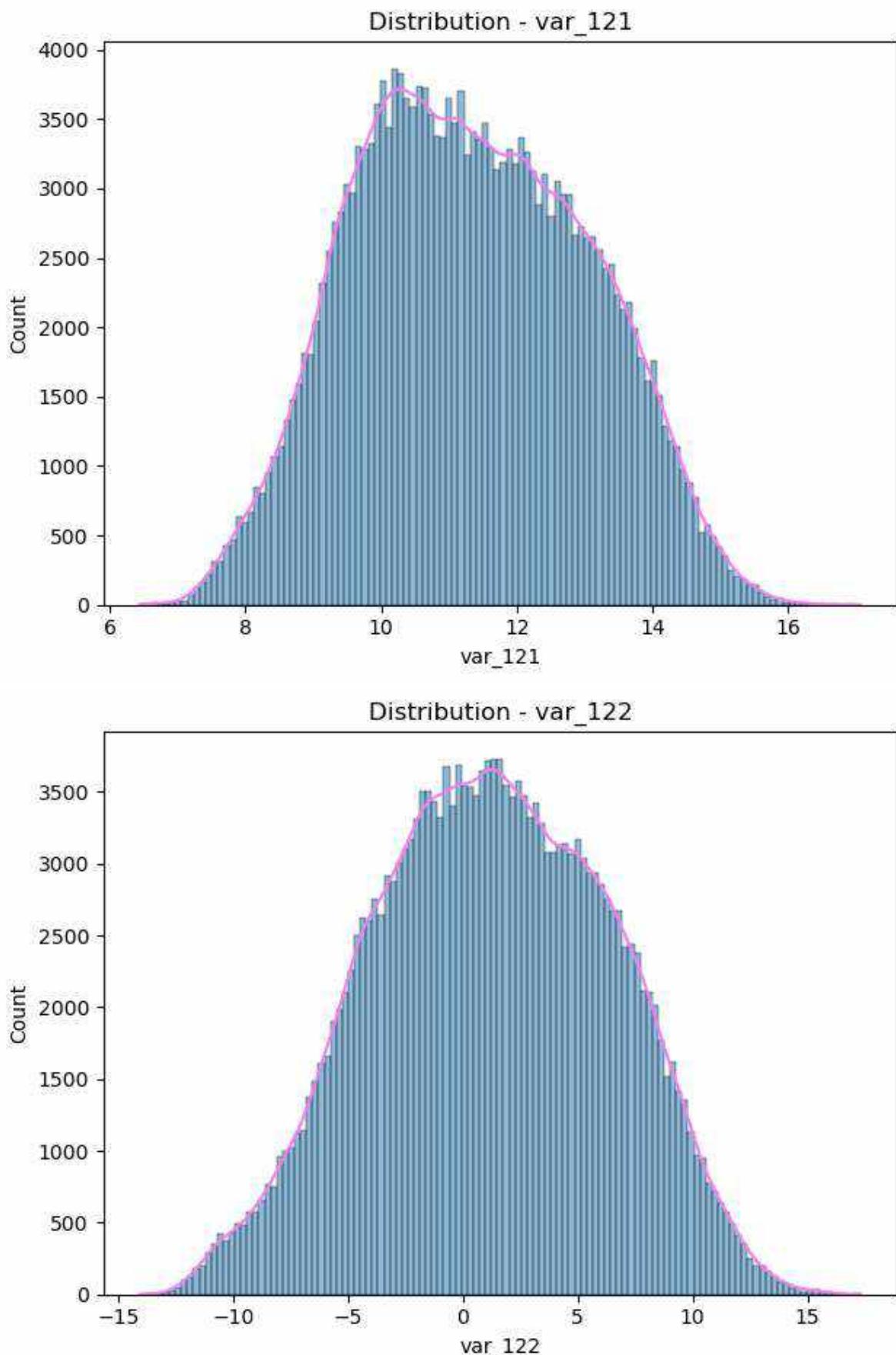


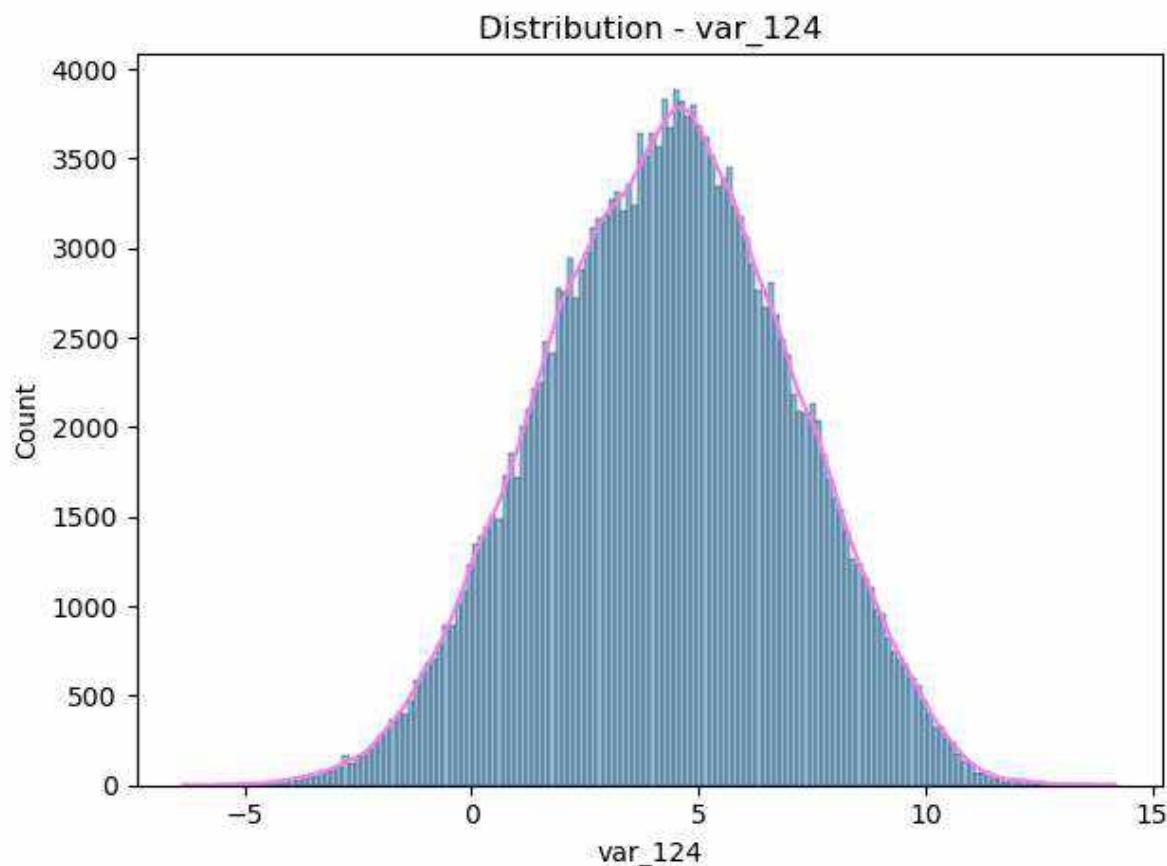
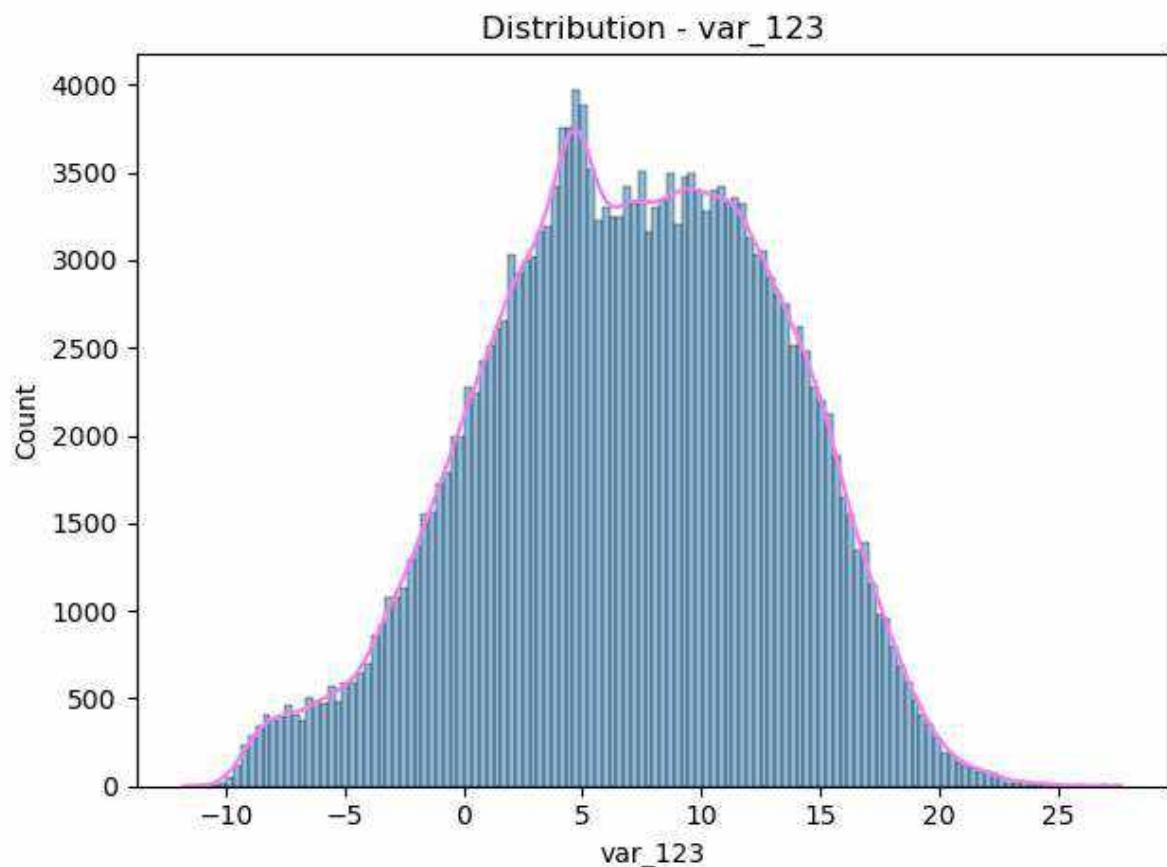


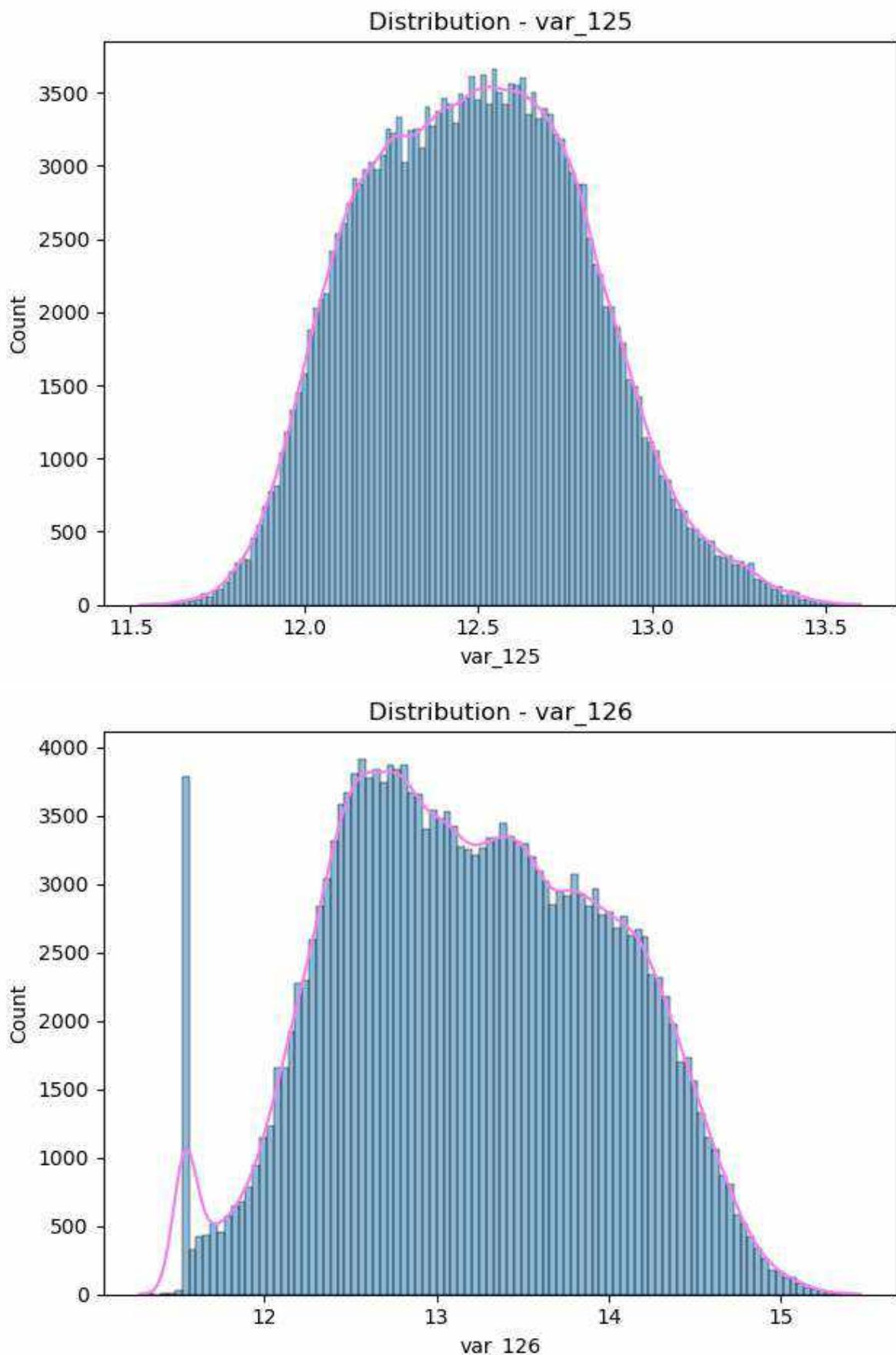


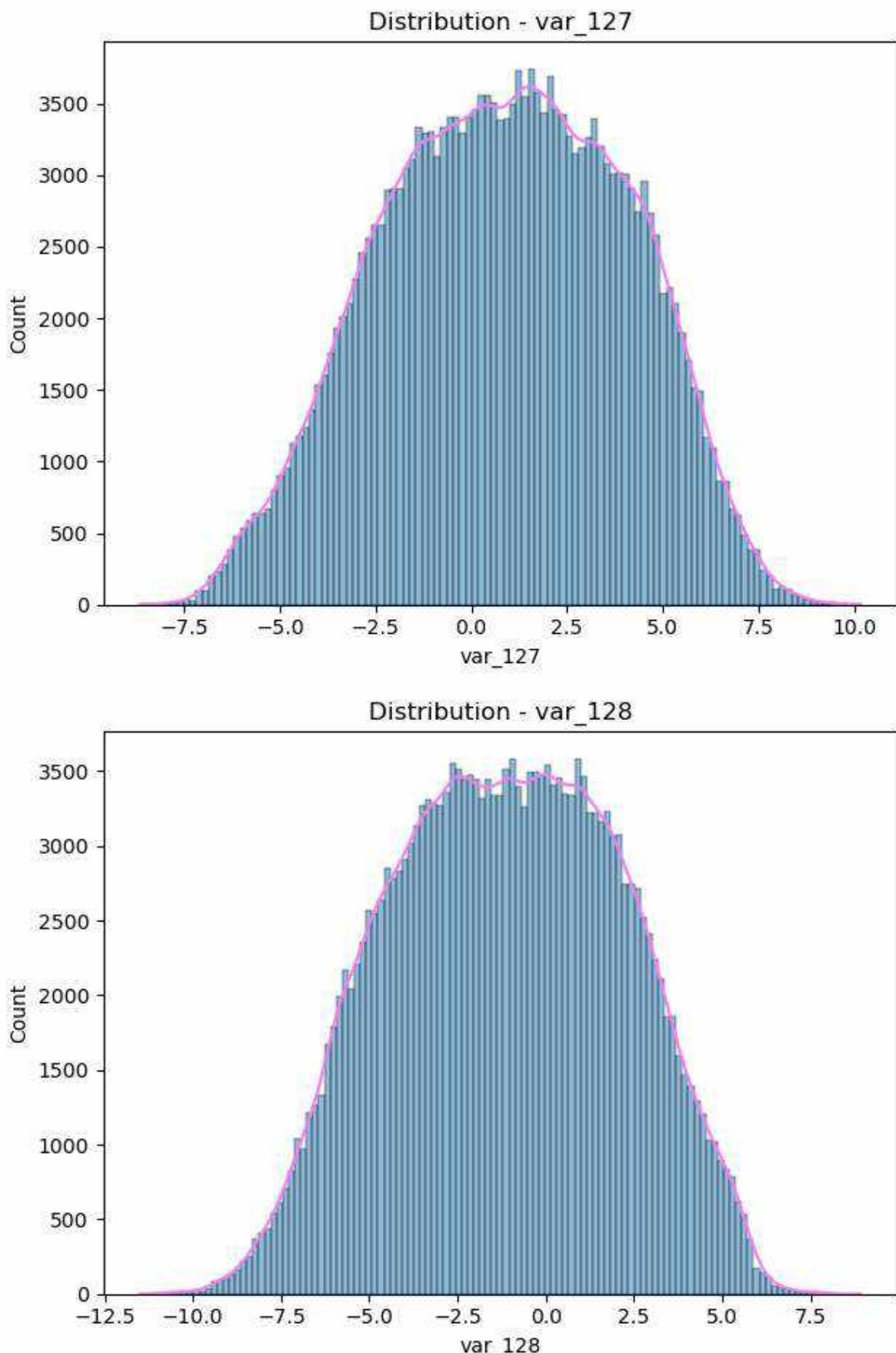


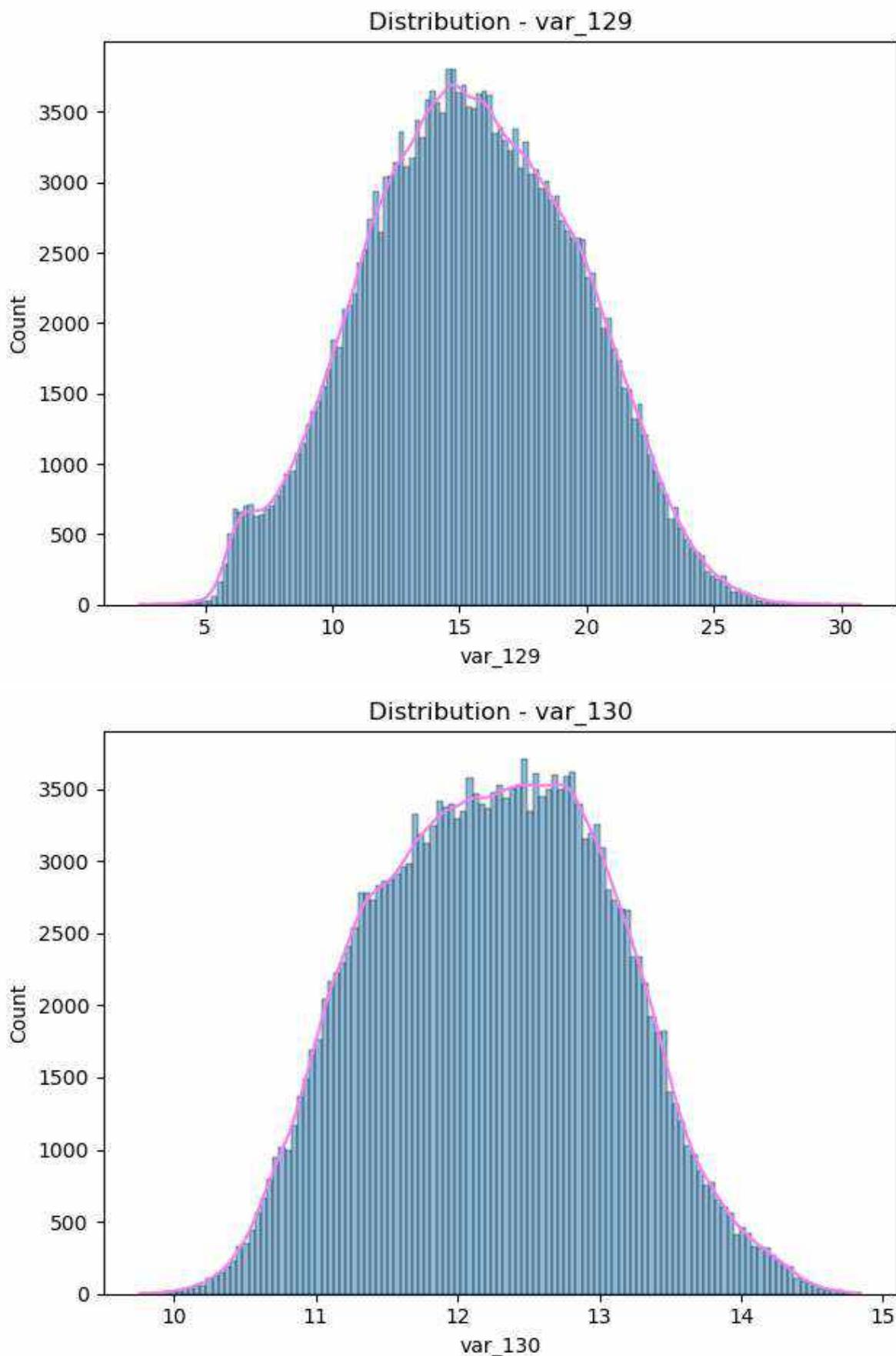


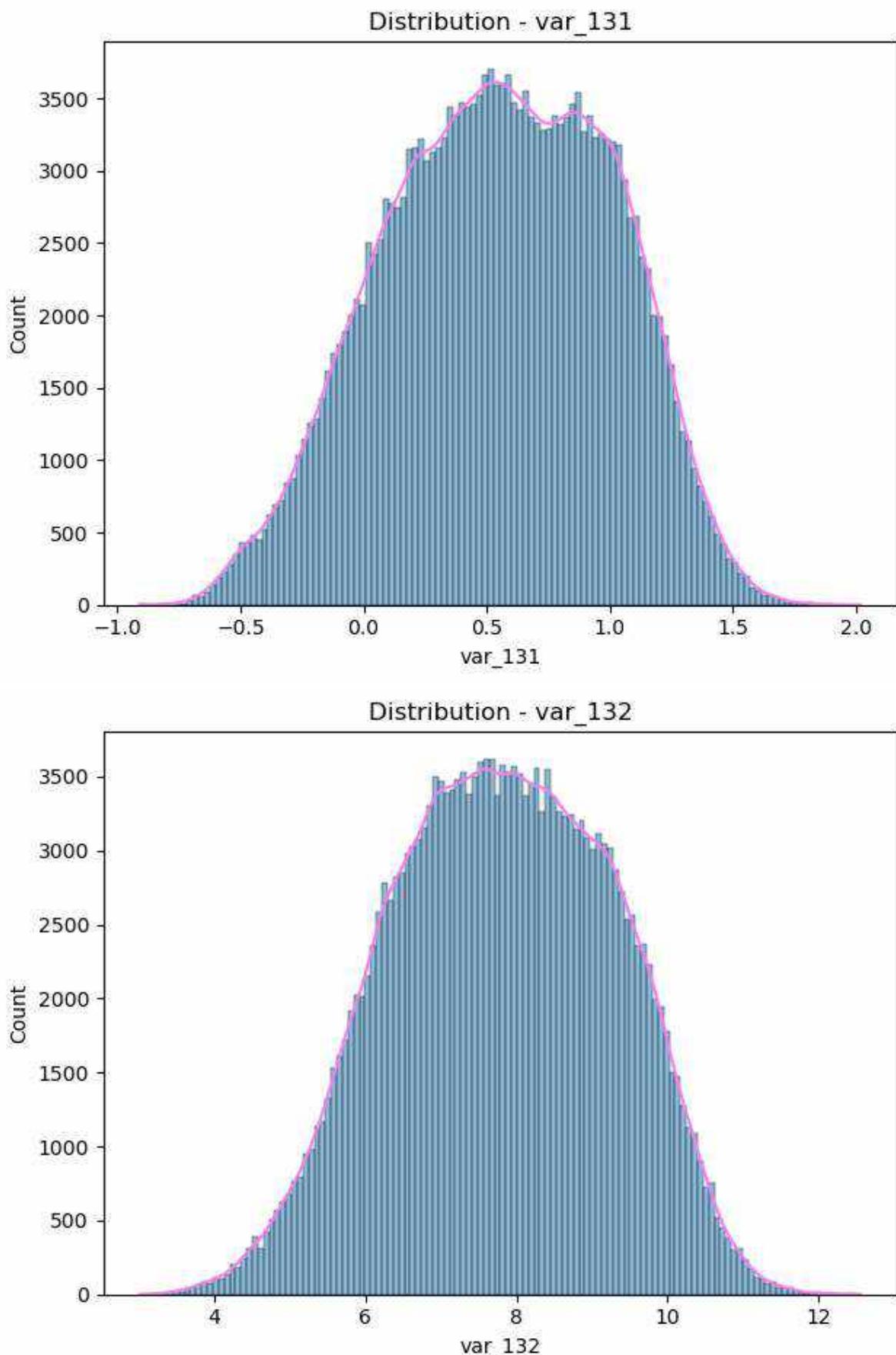


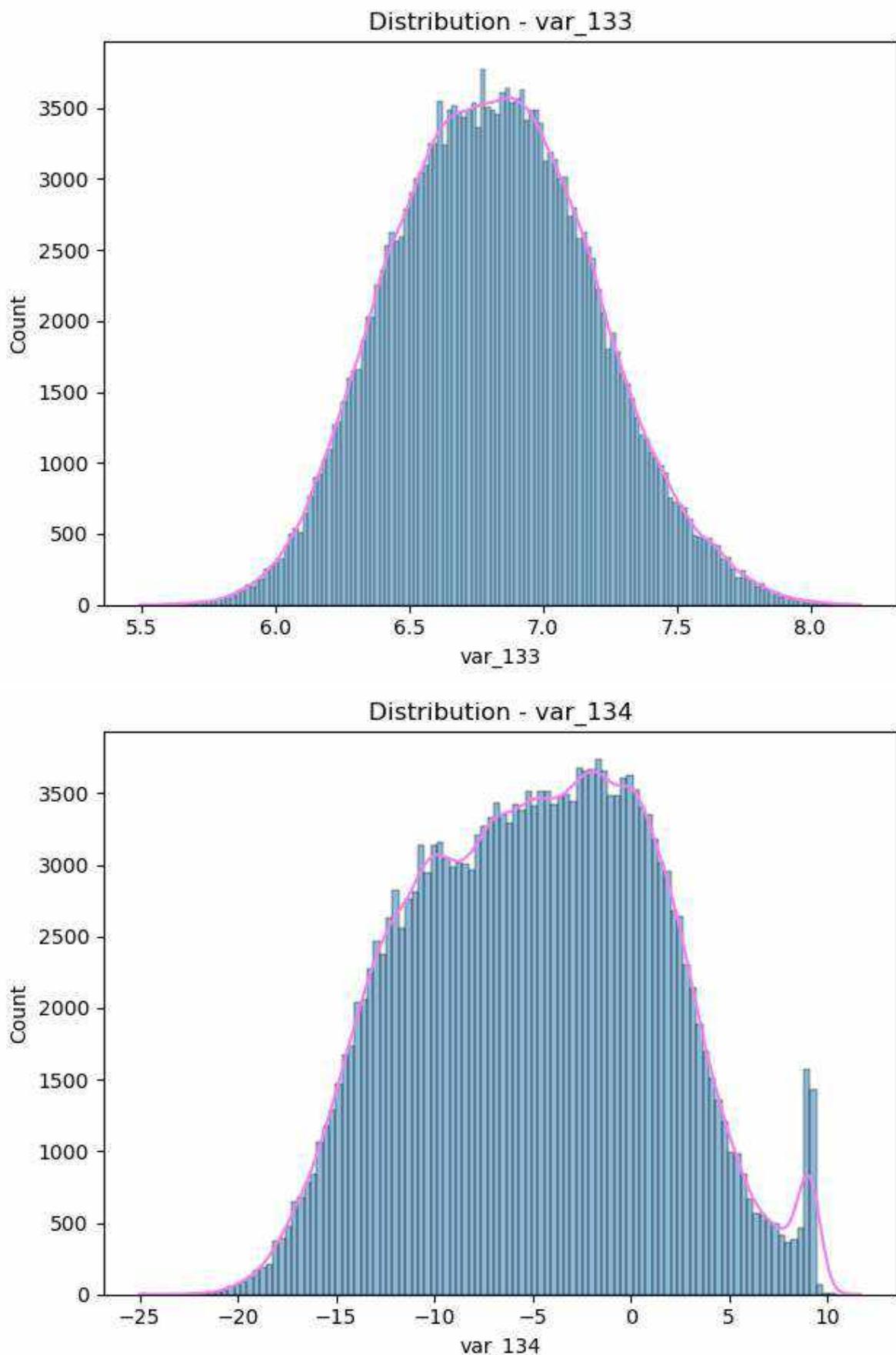


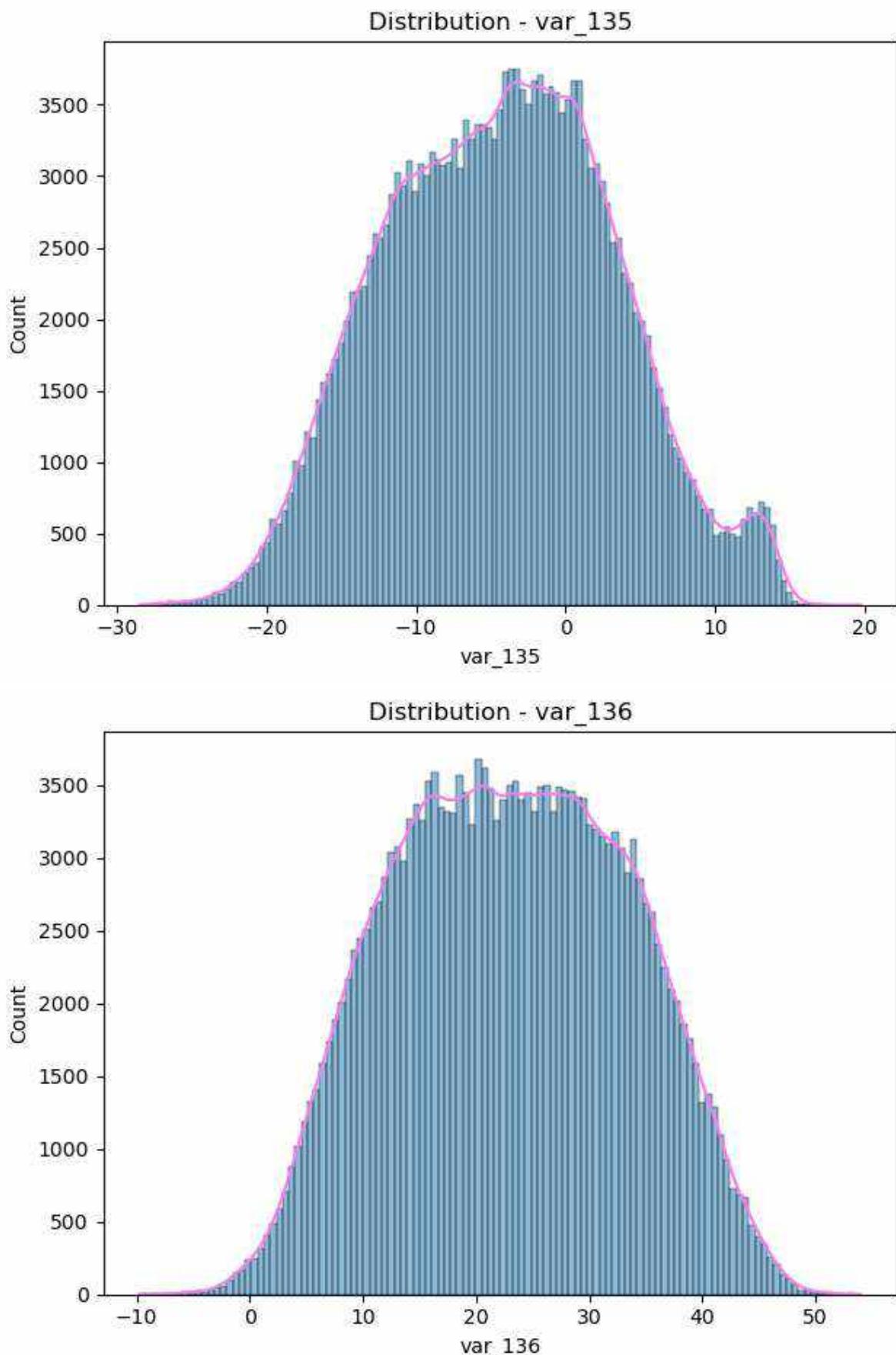


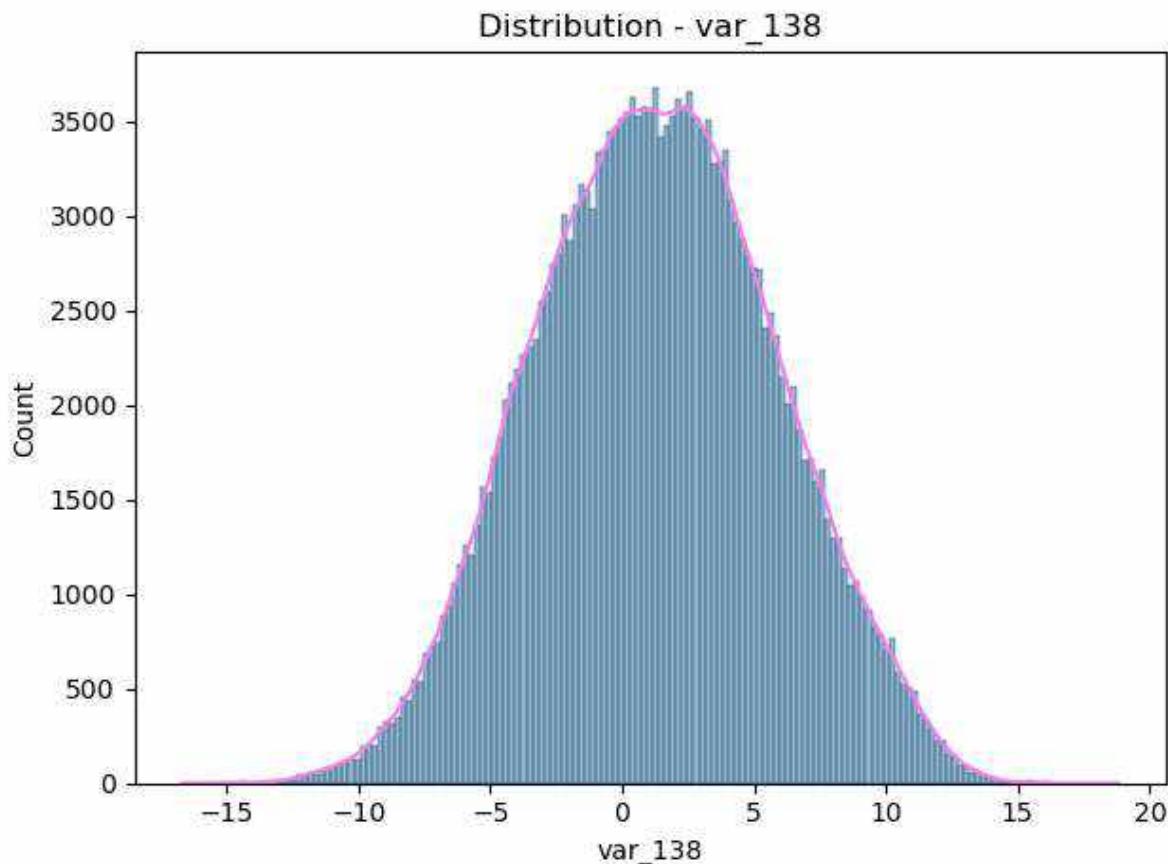
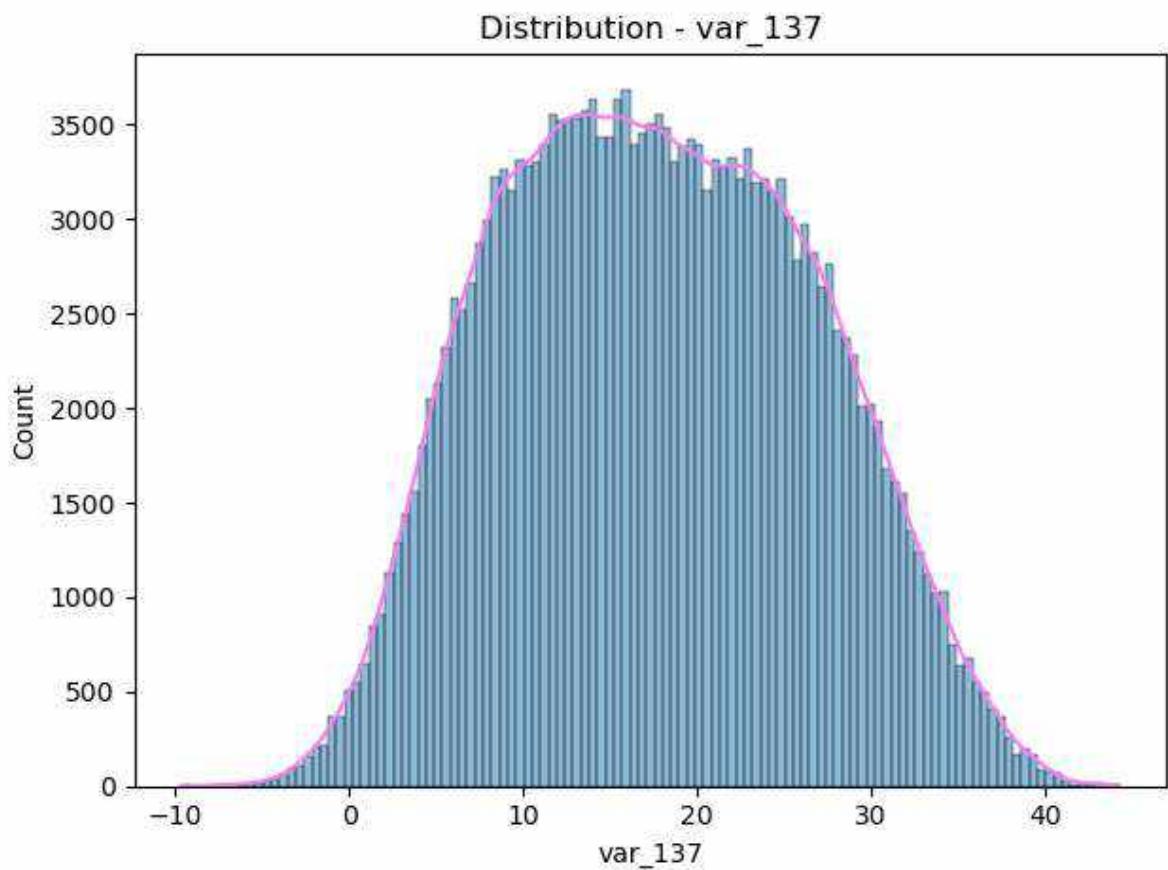


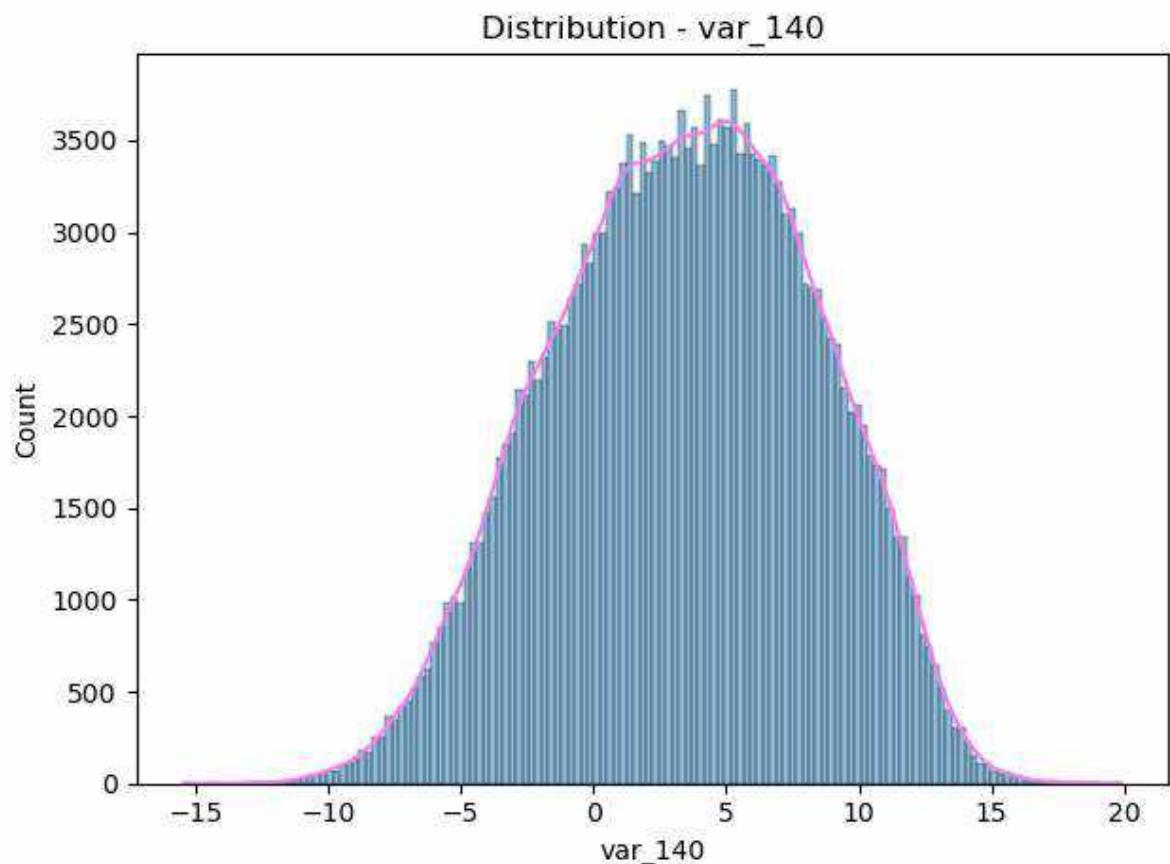
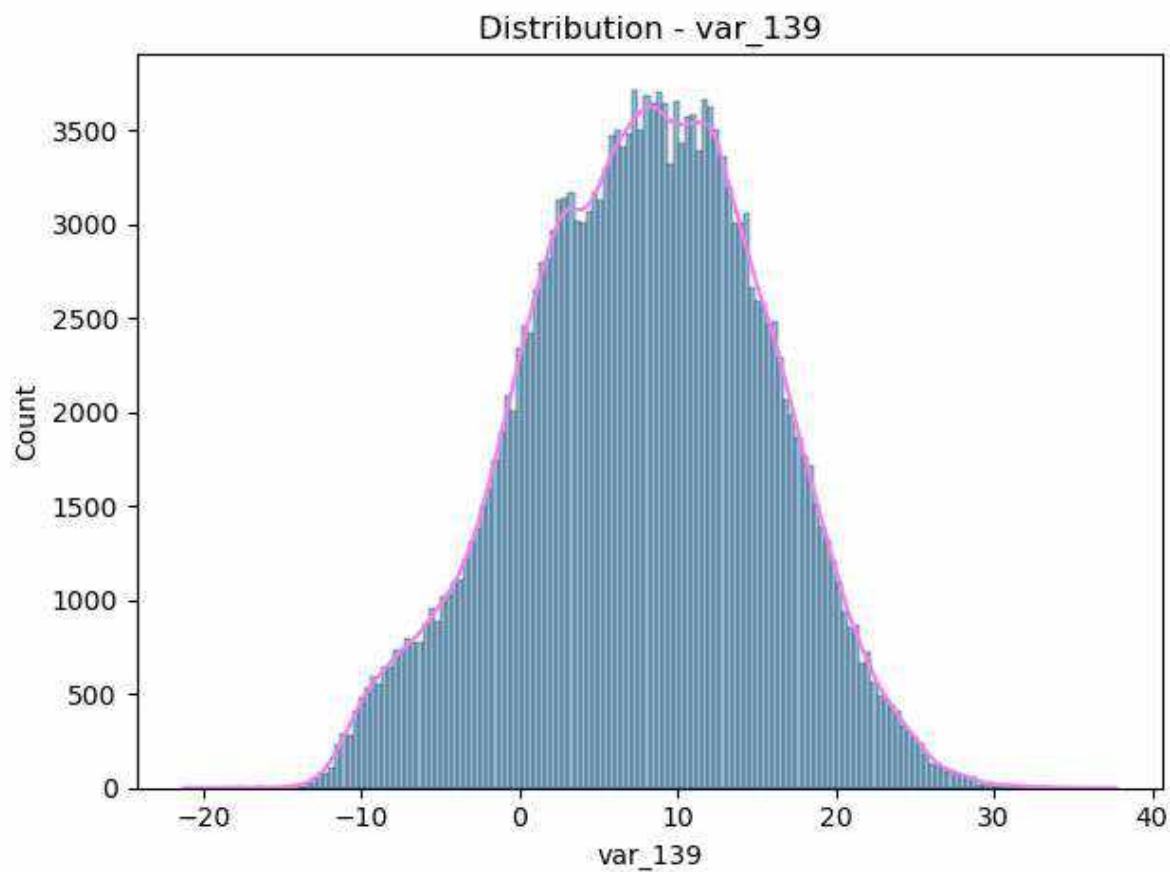




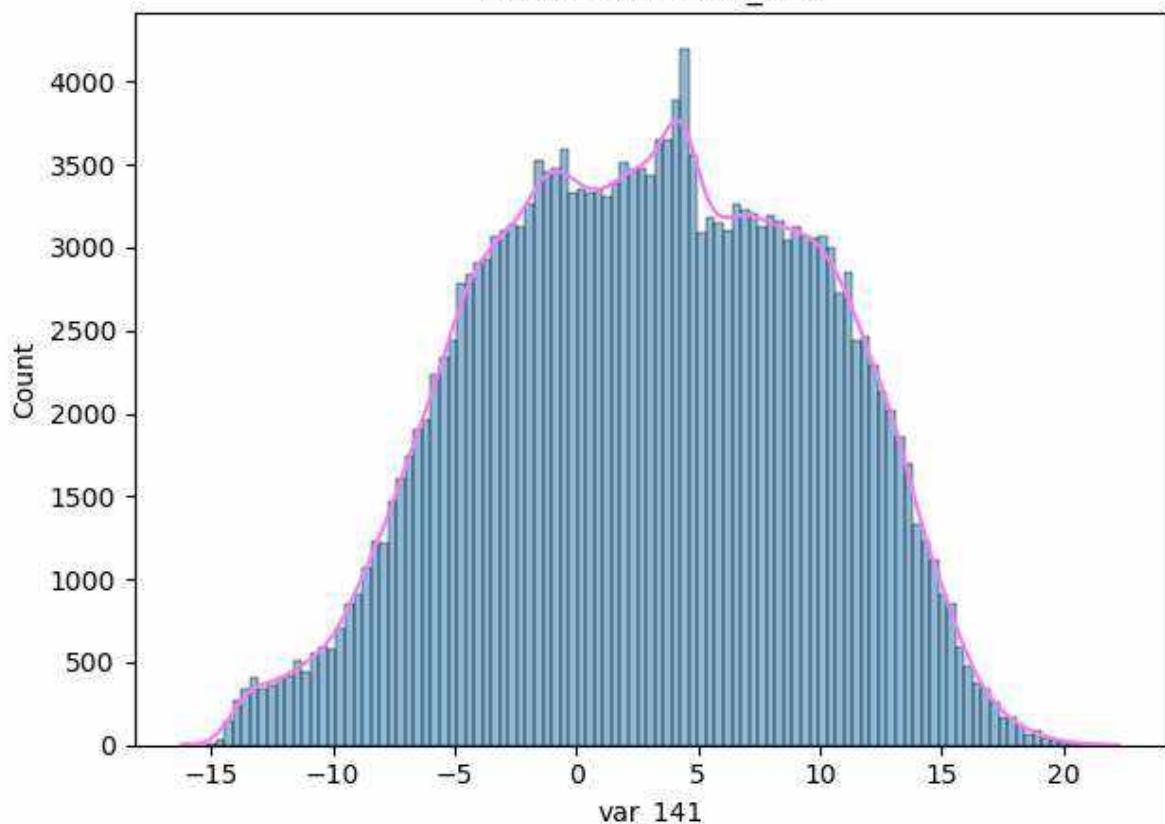




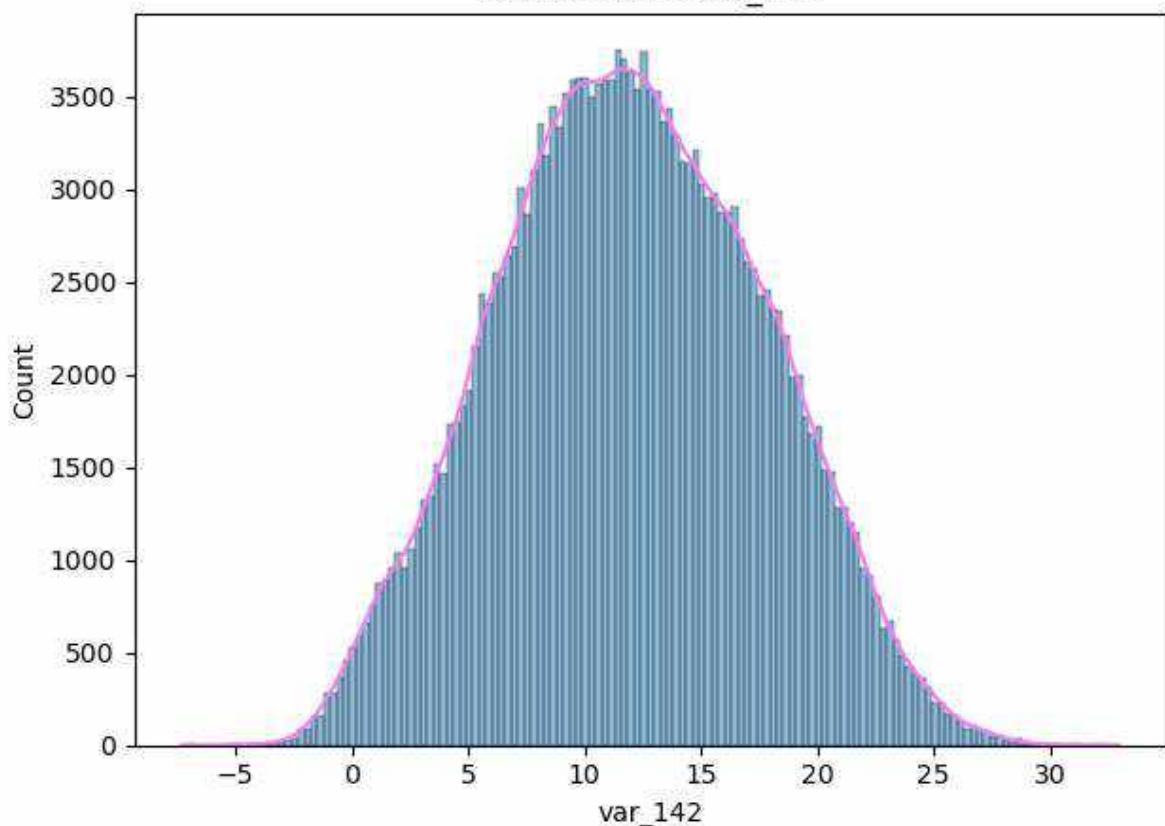




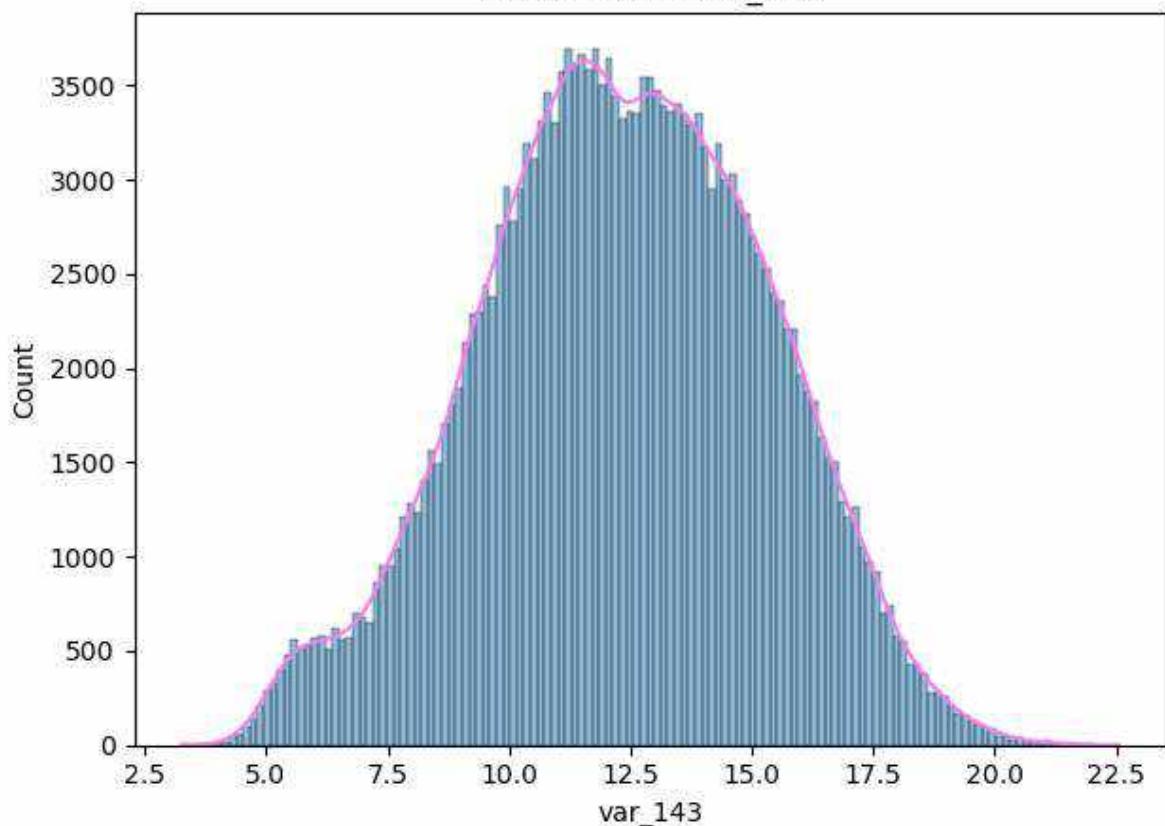
Distribution - var\_141



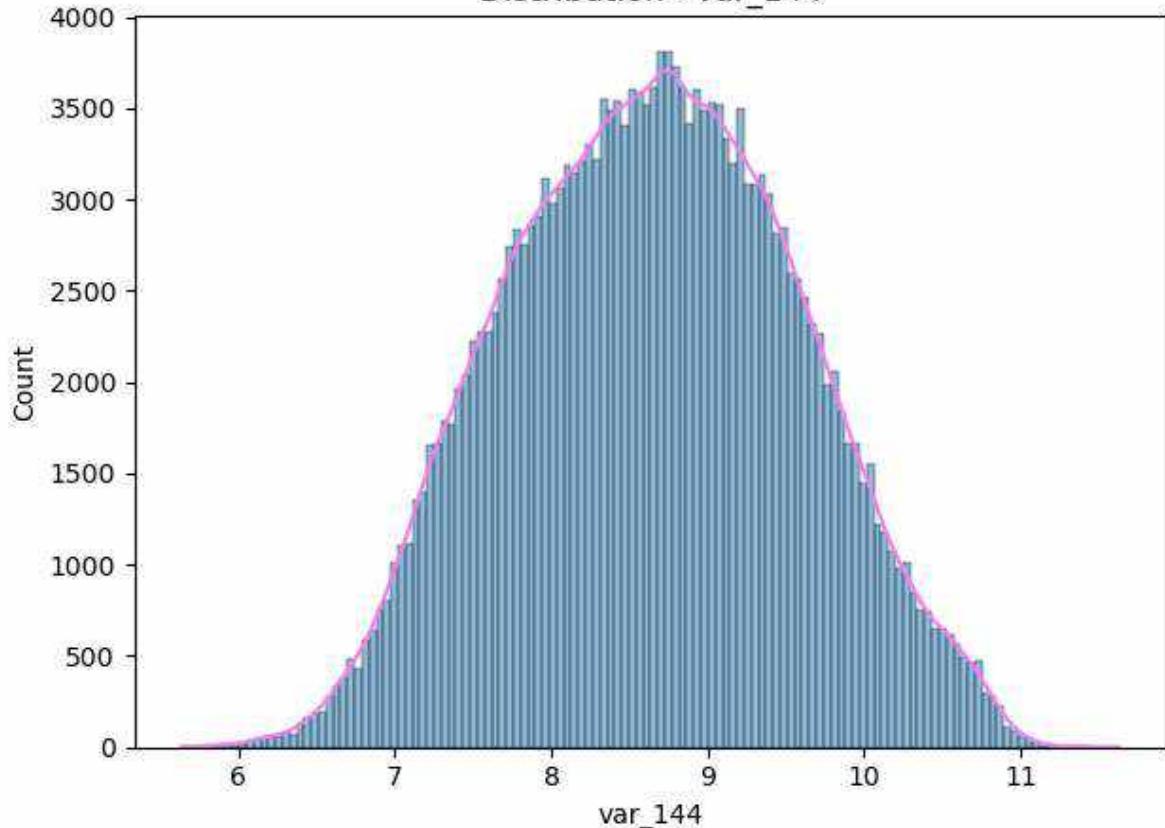
Distribution - var\_142

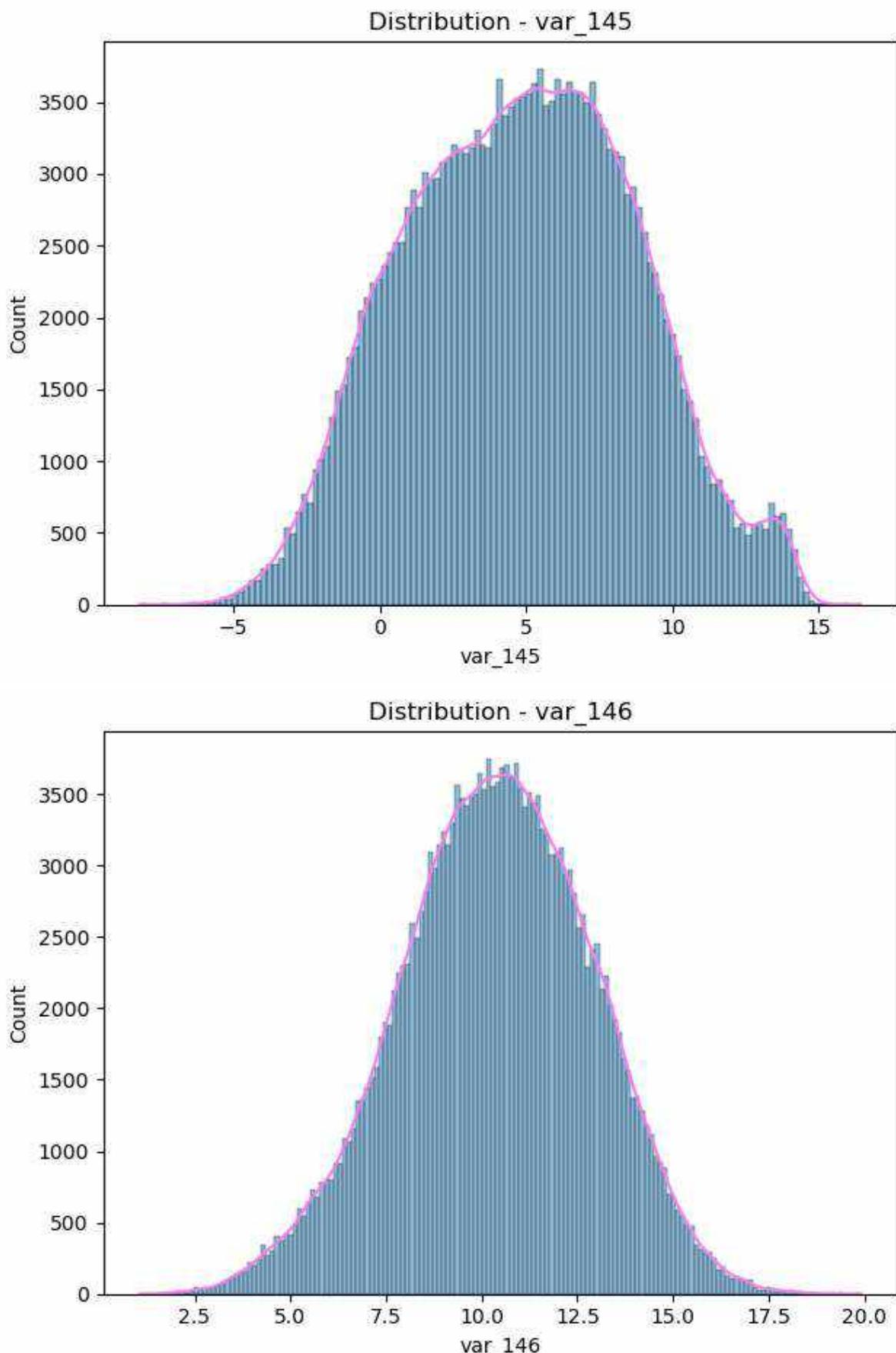


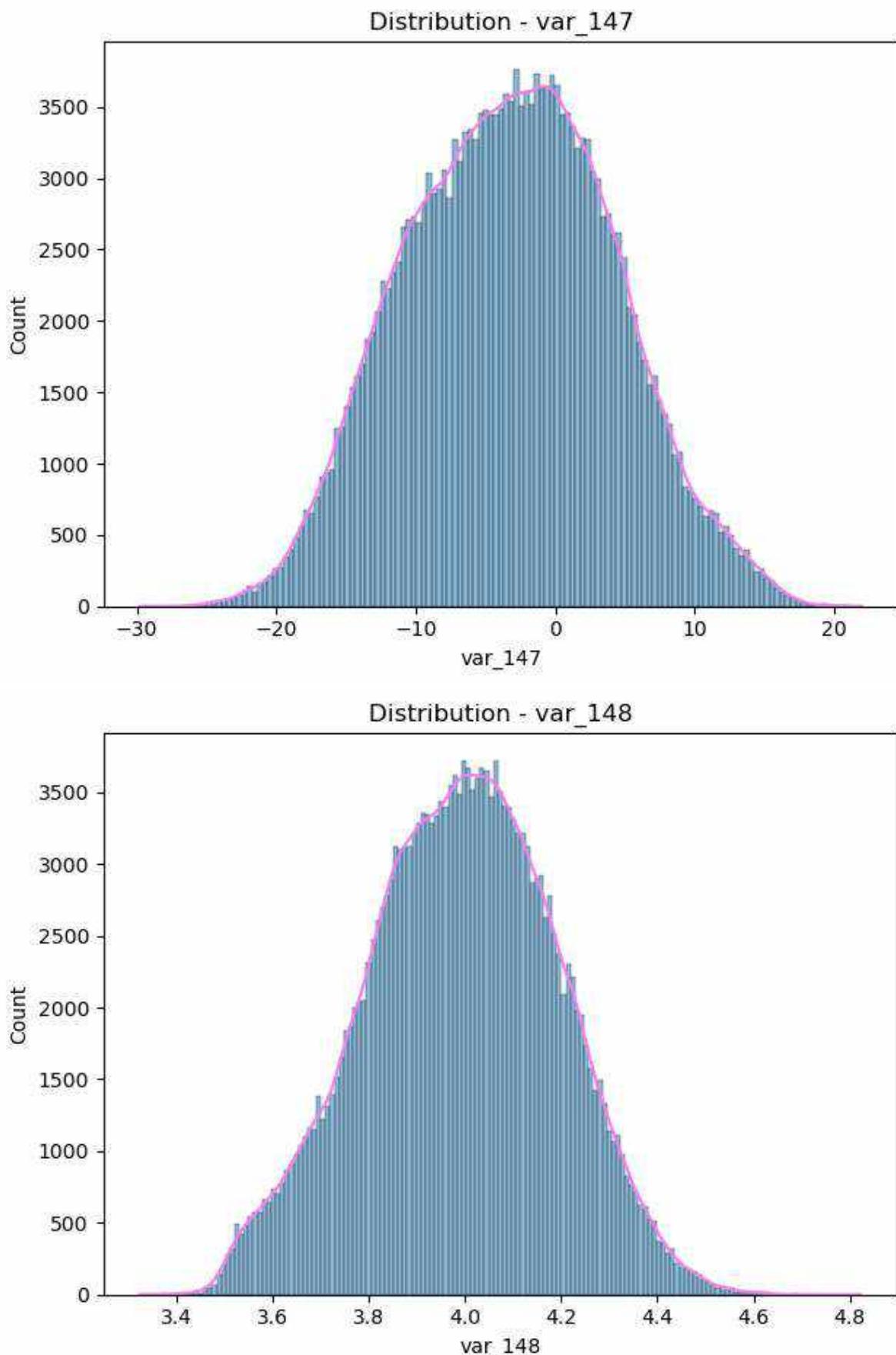
Distribution - var\_143

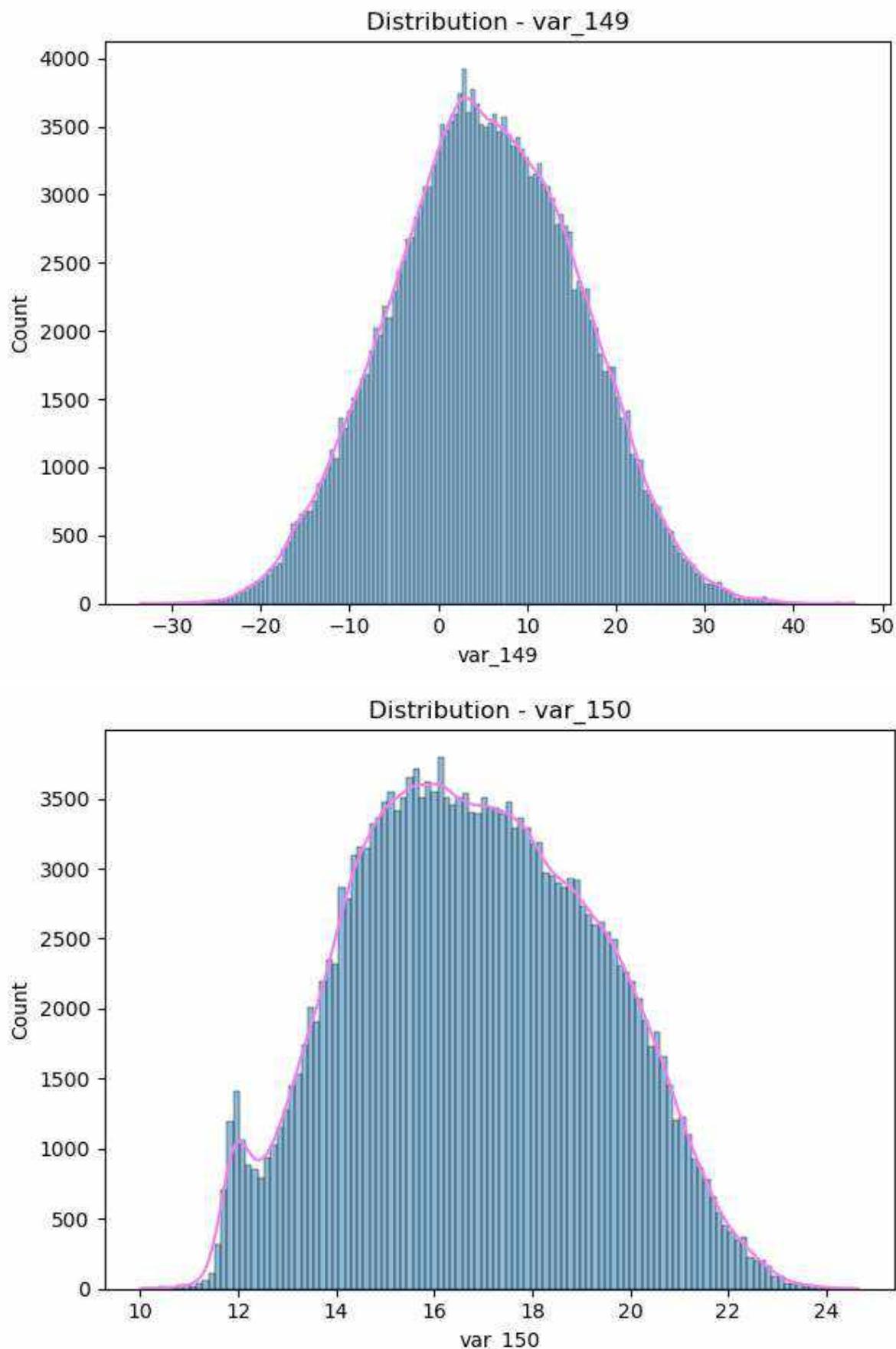


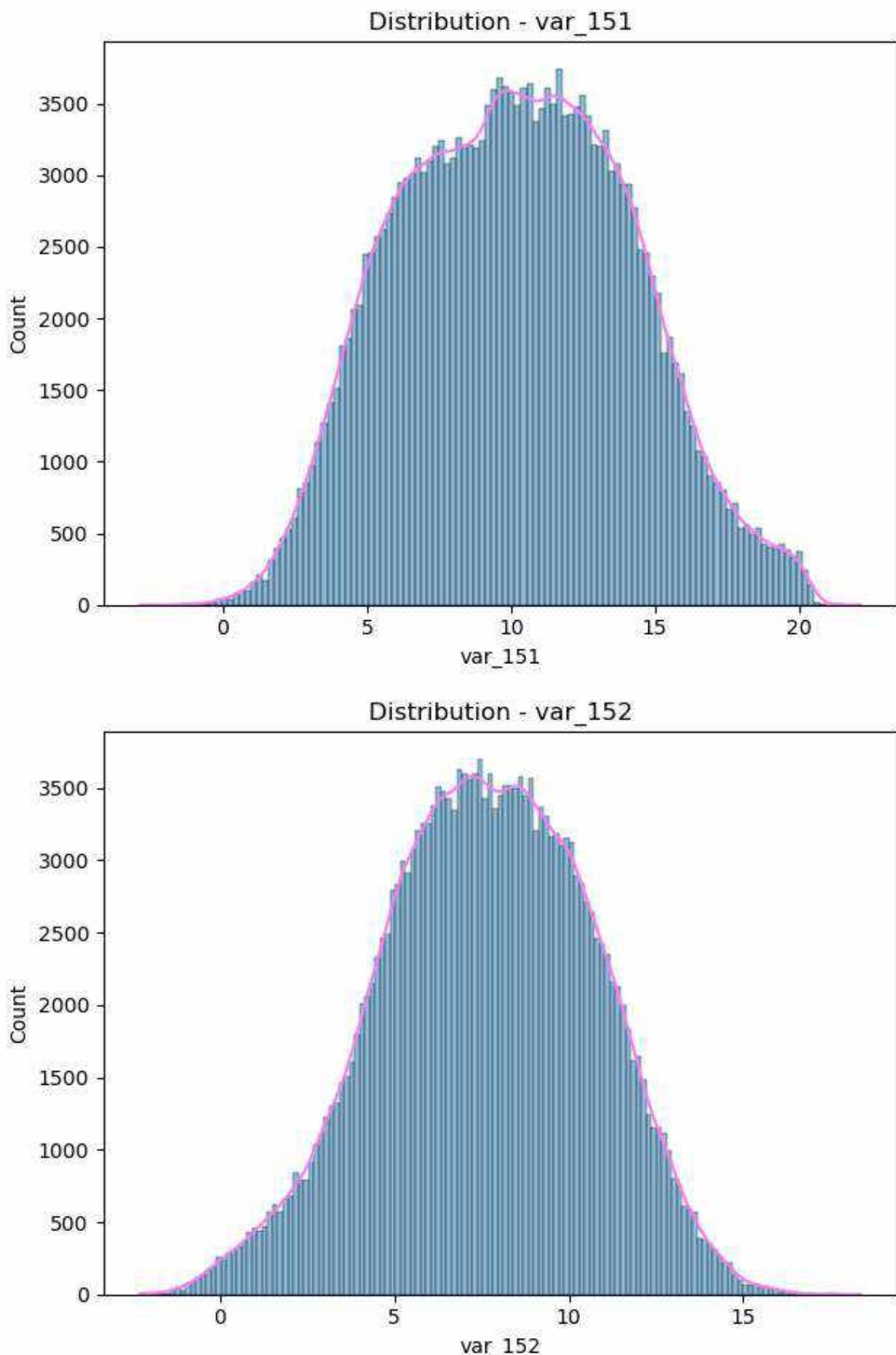
Distribution - var\_144

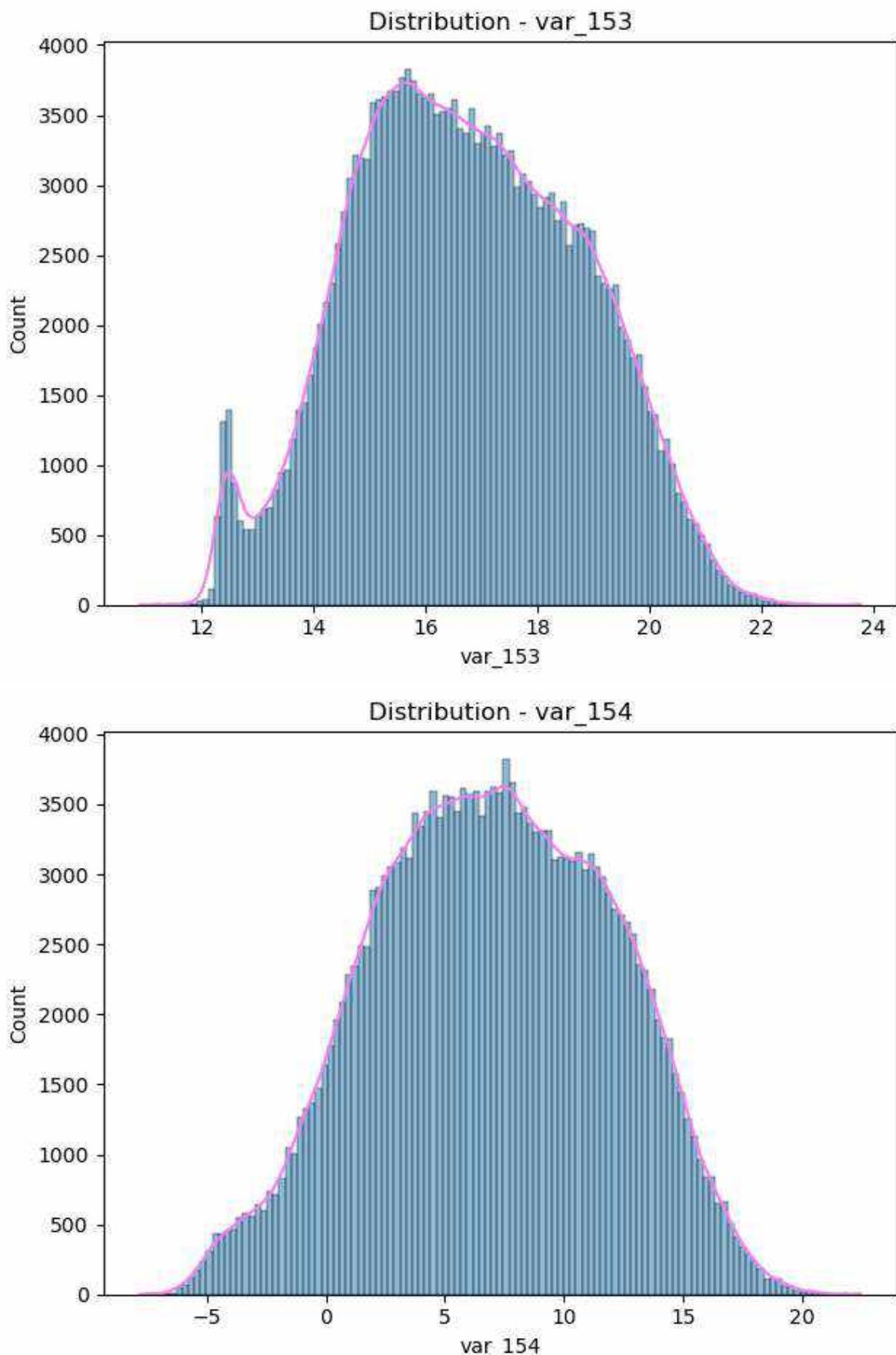


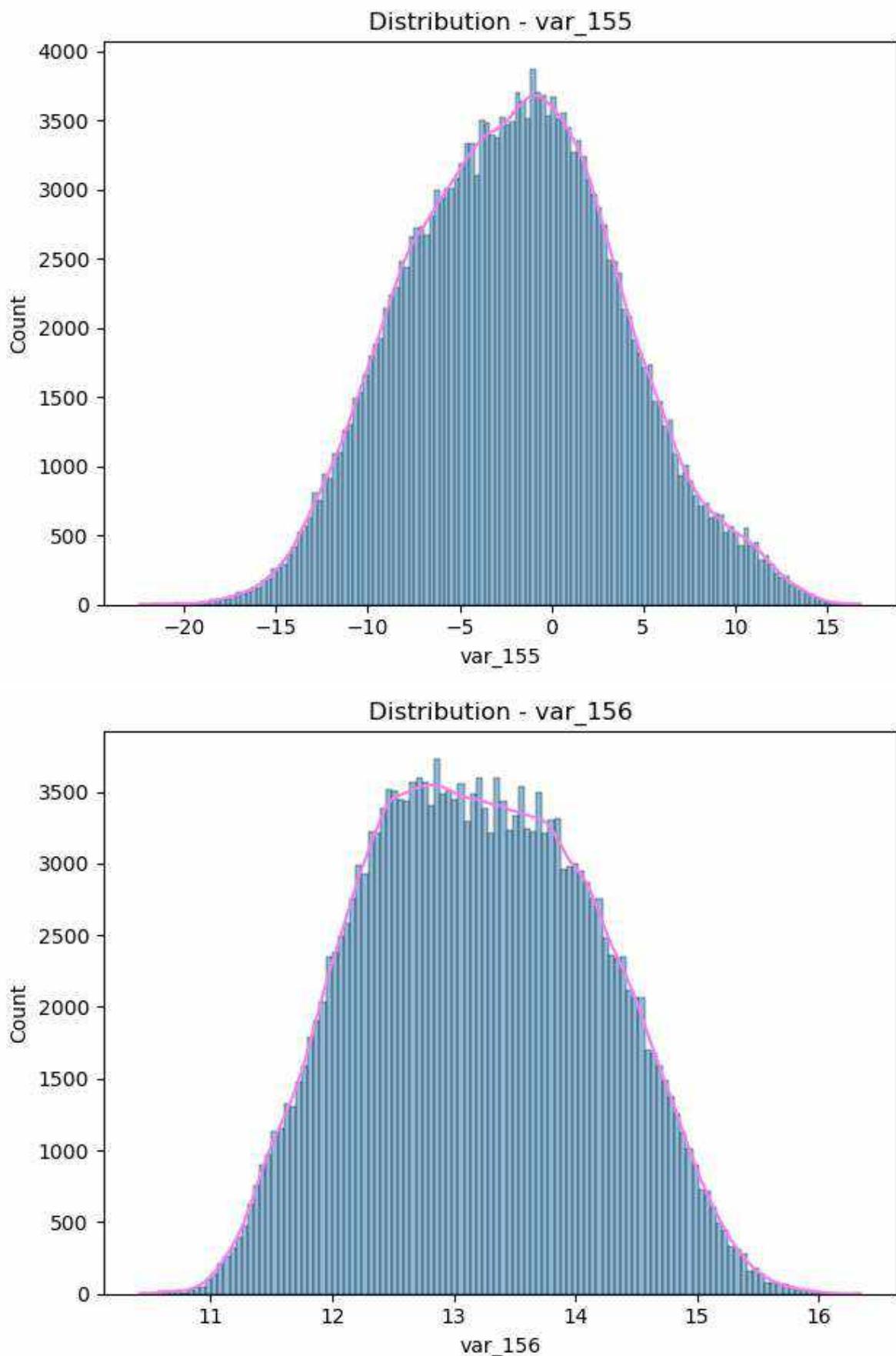




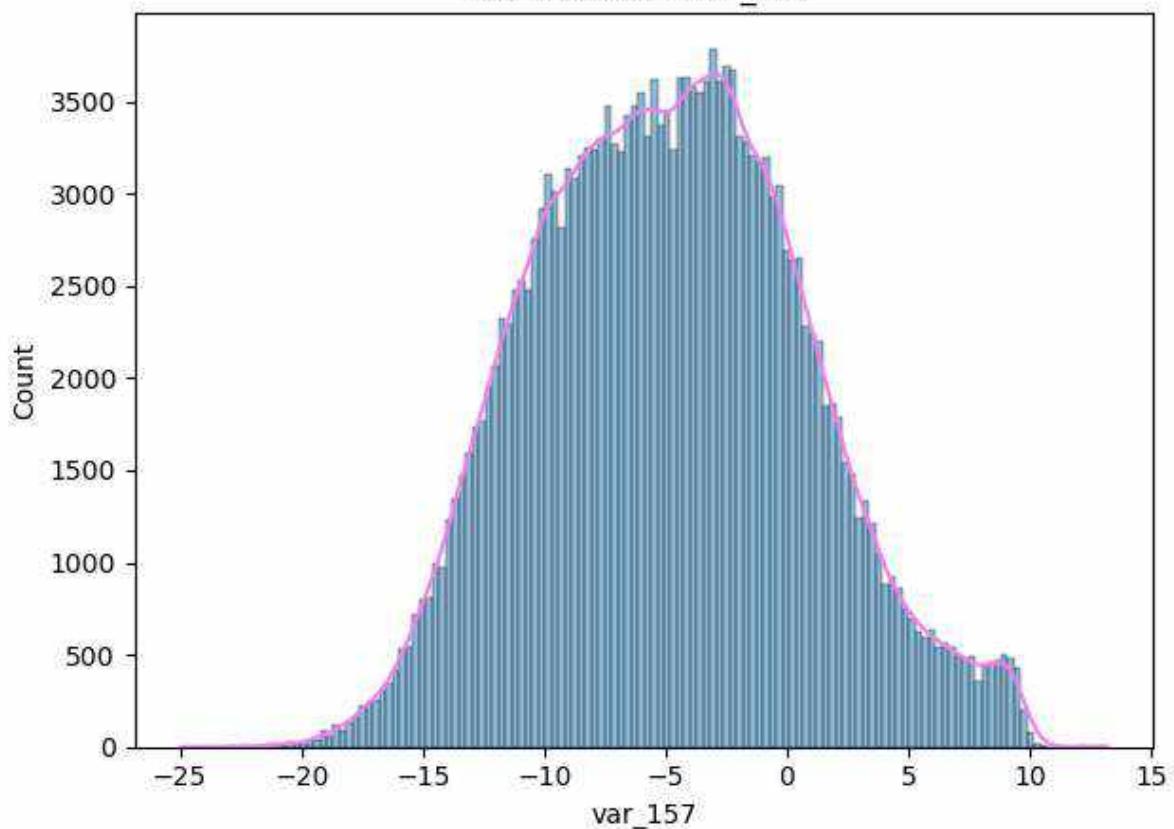




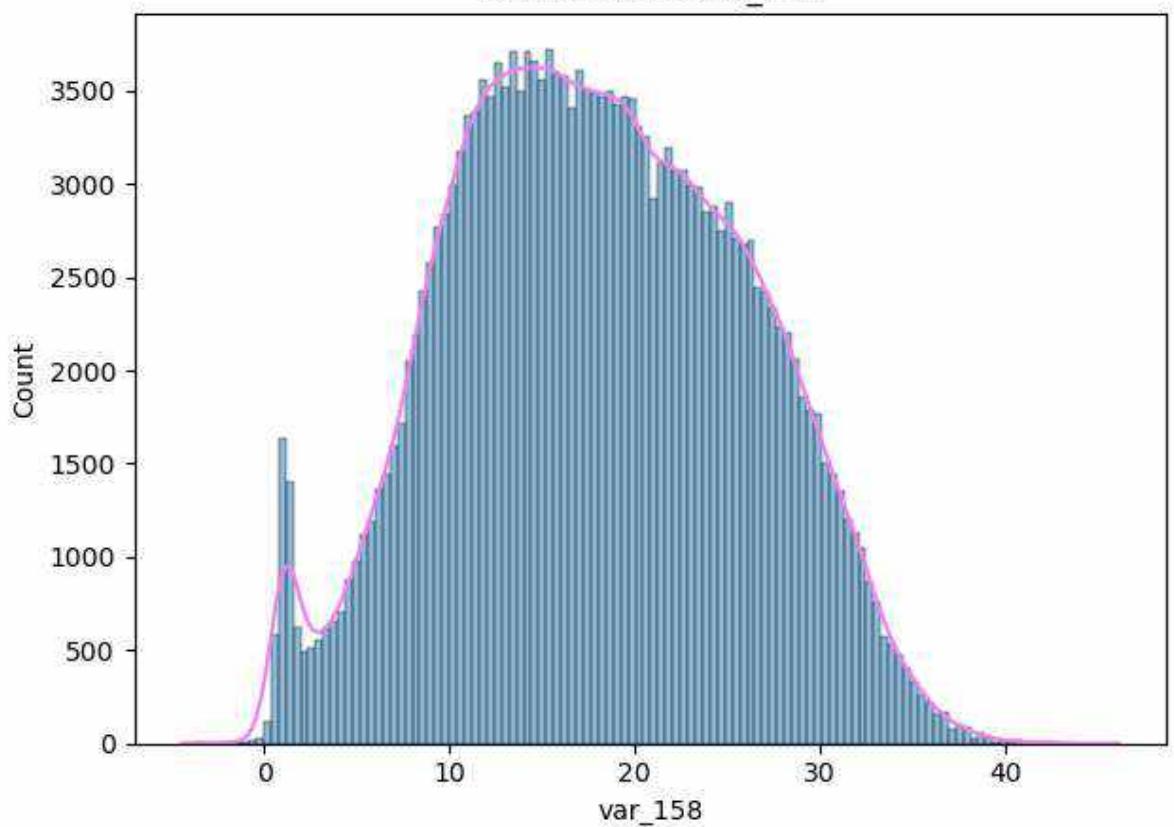


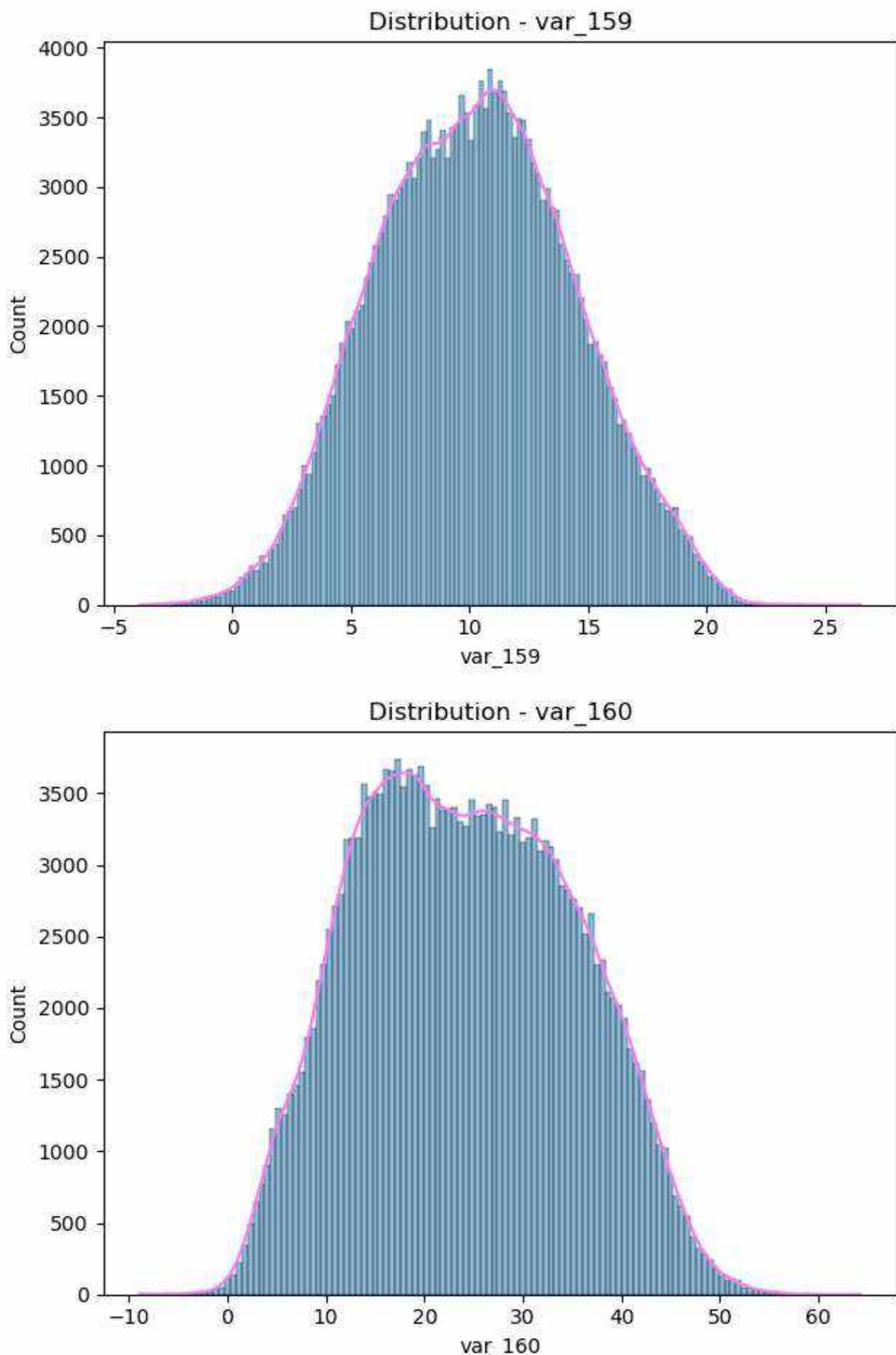


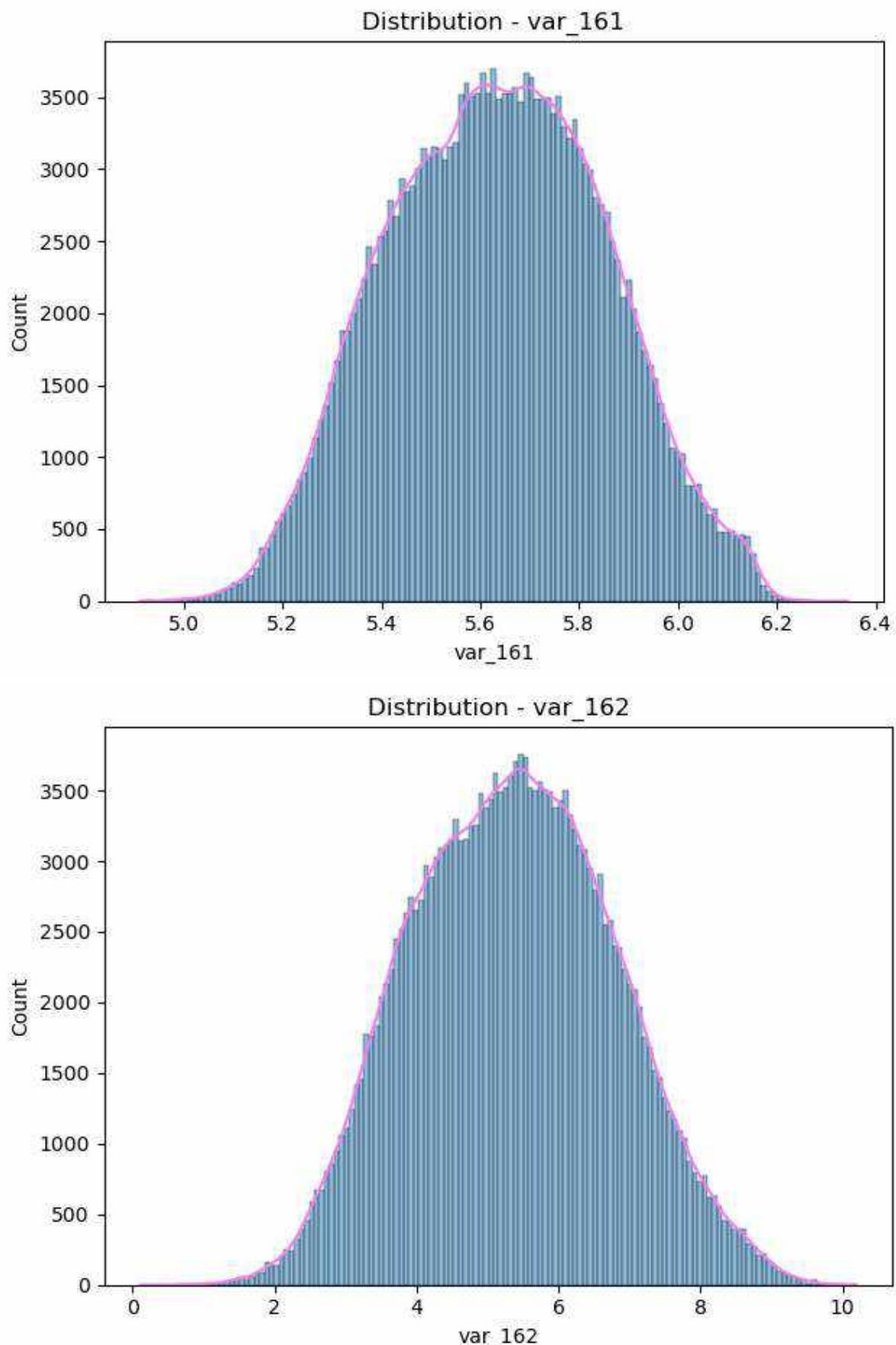
Distribution - var\_157

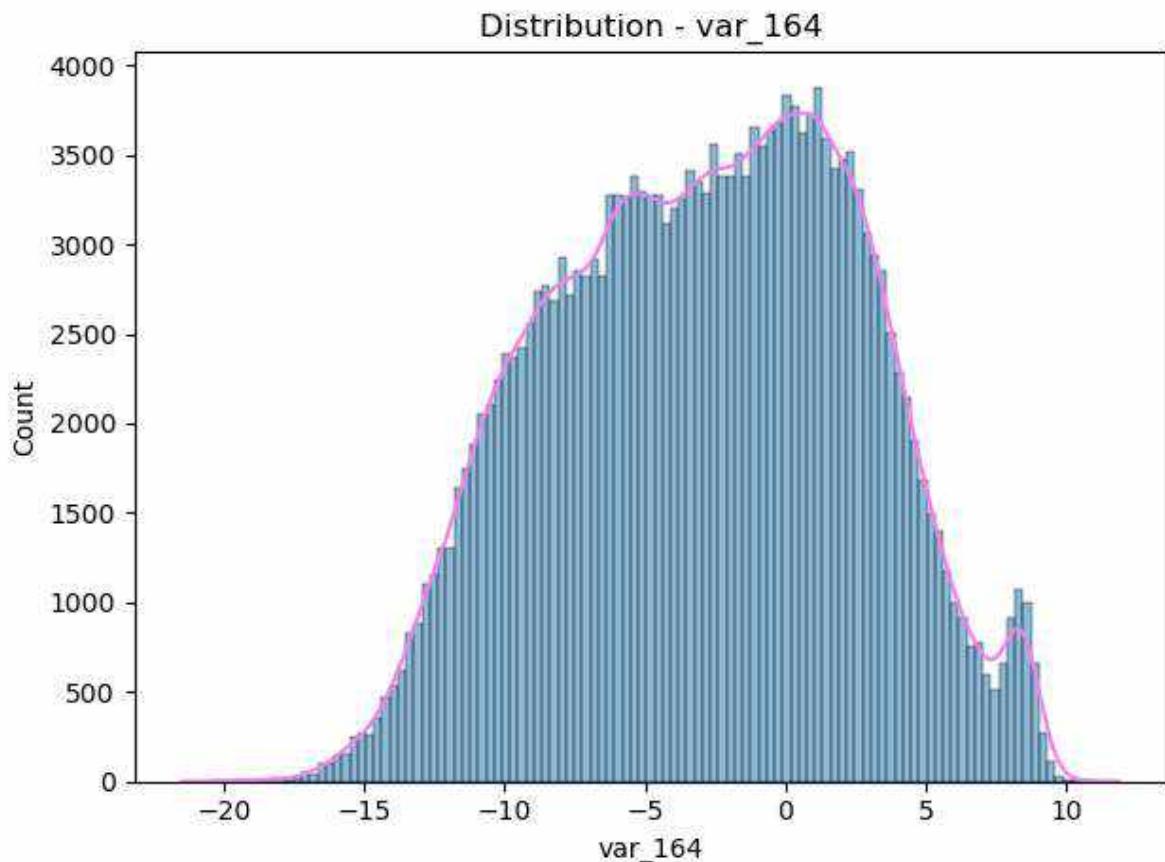
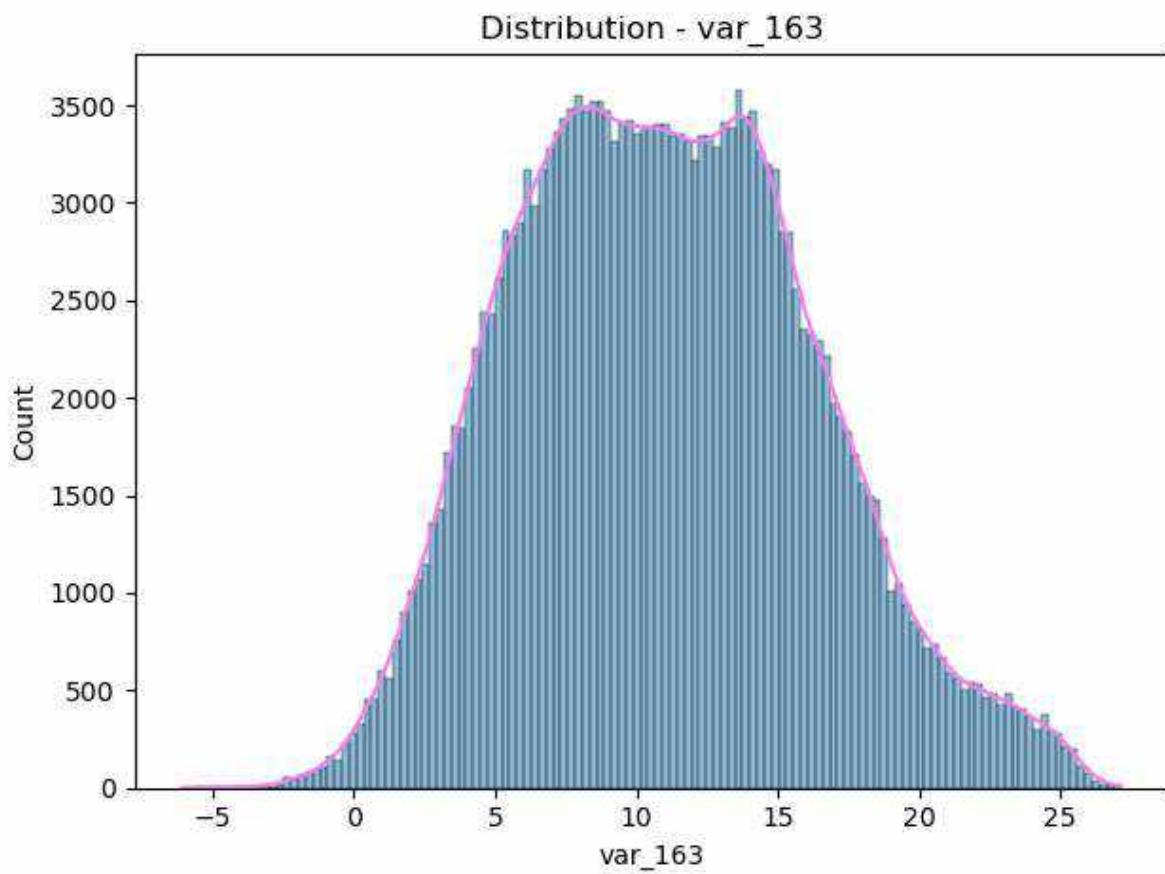


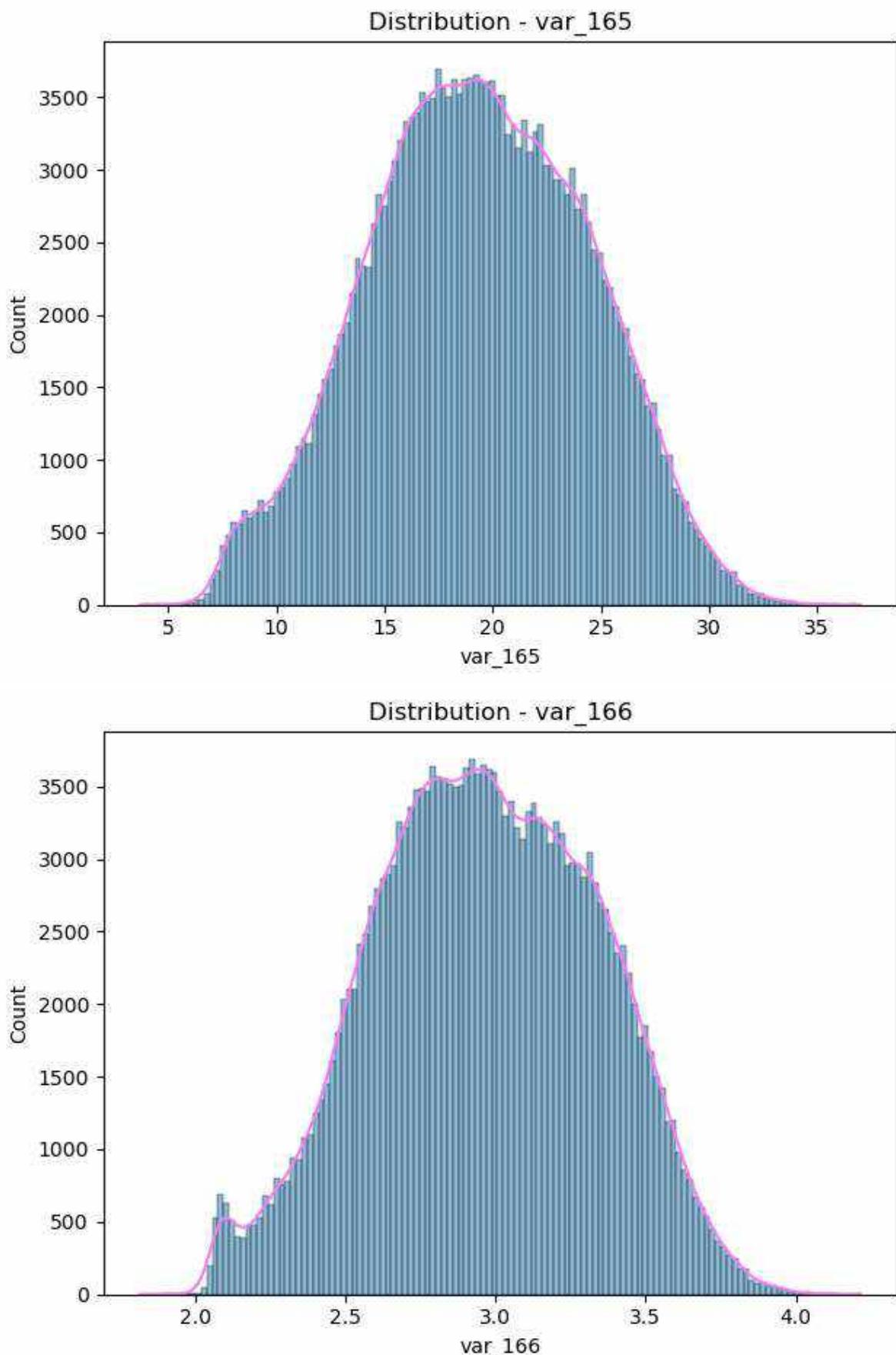
Distribution - var\_158

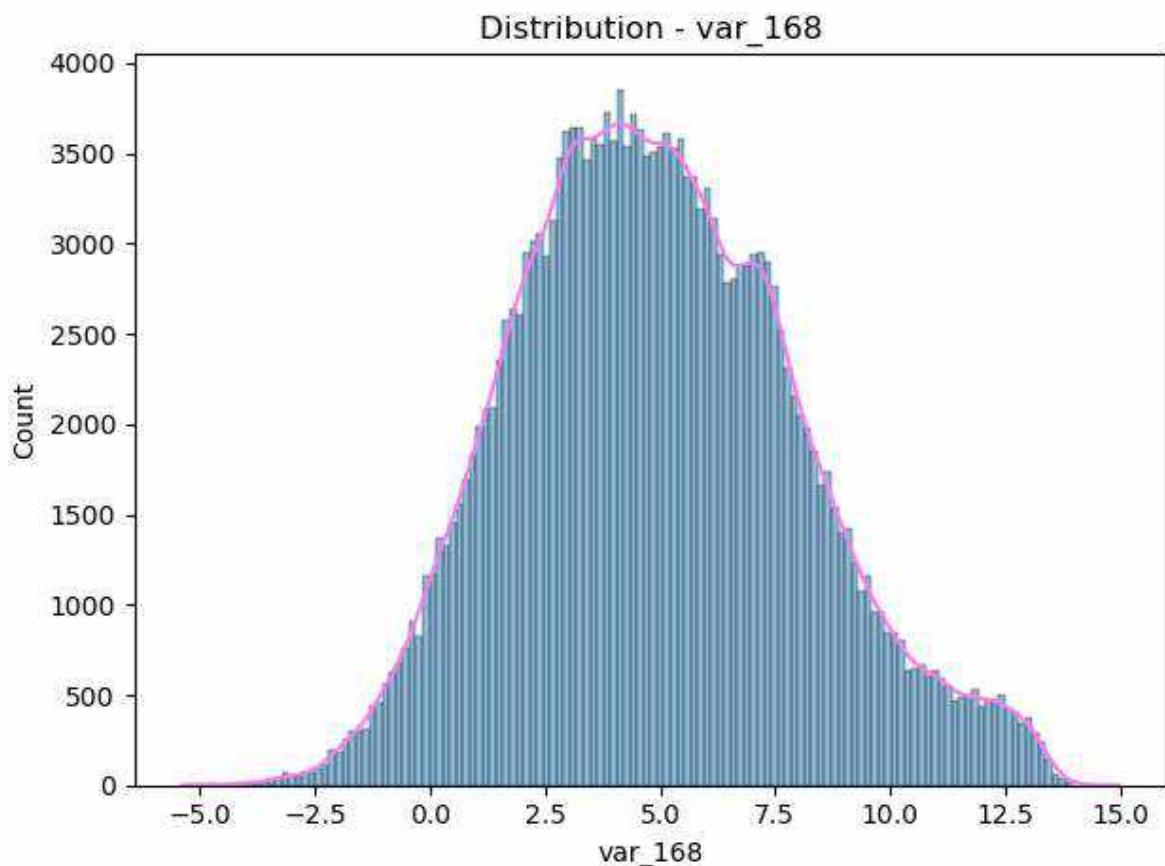
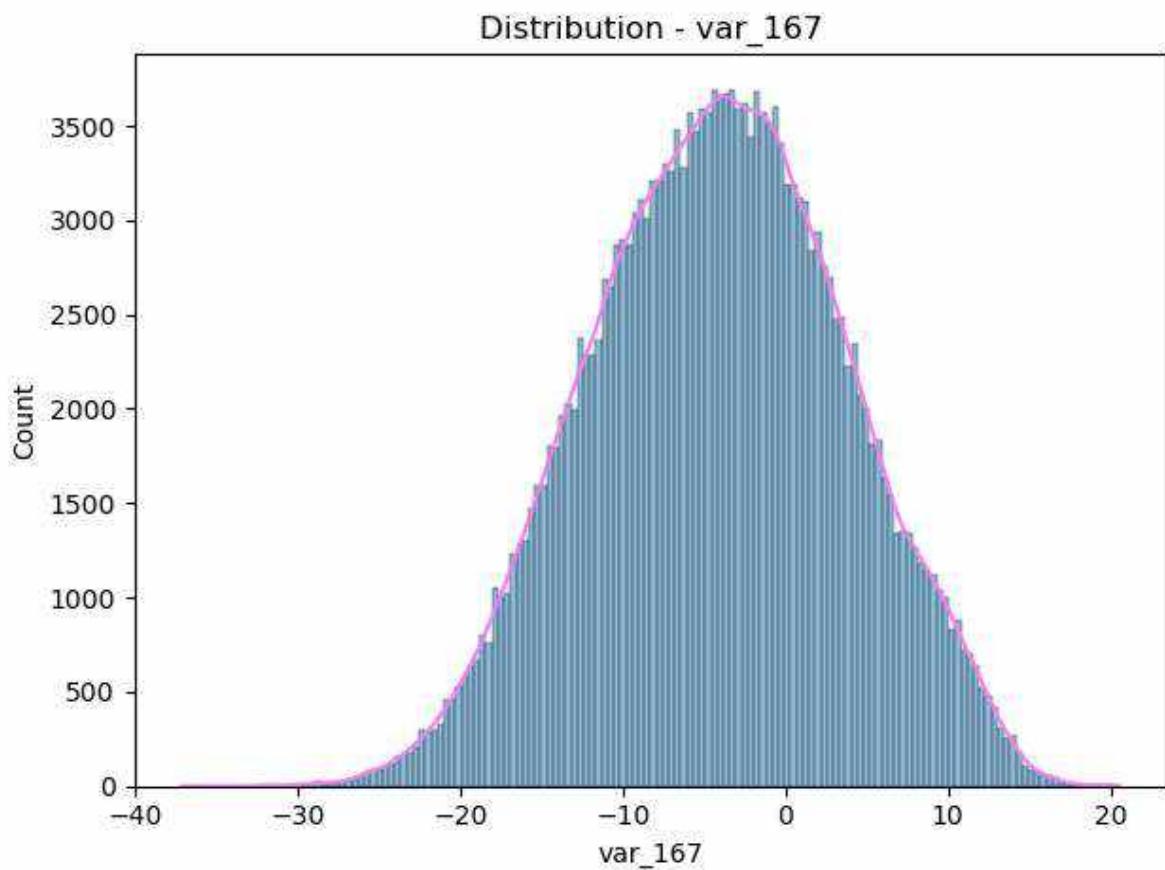


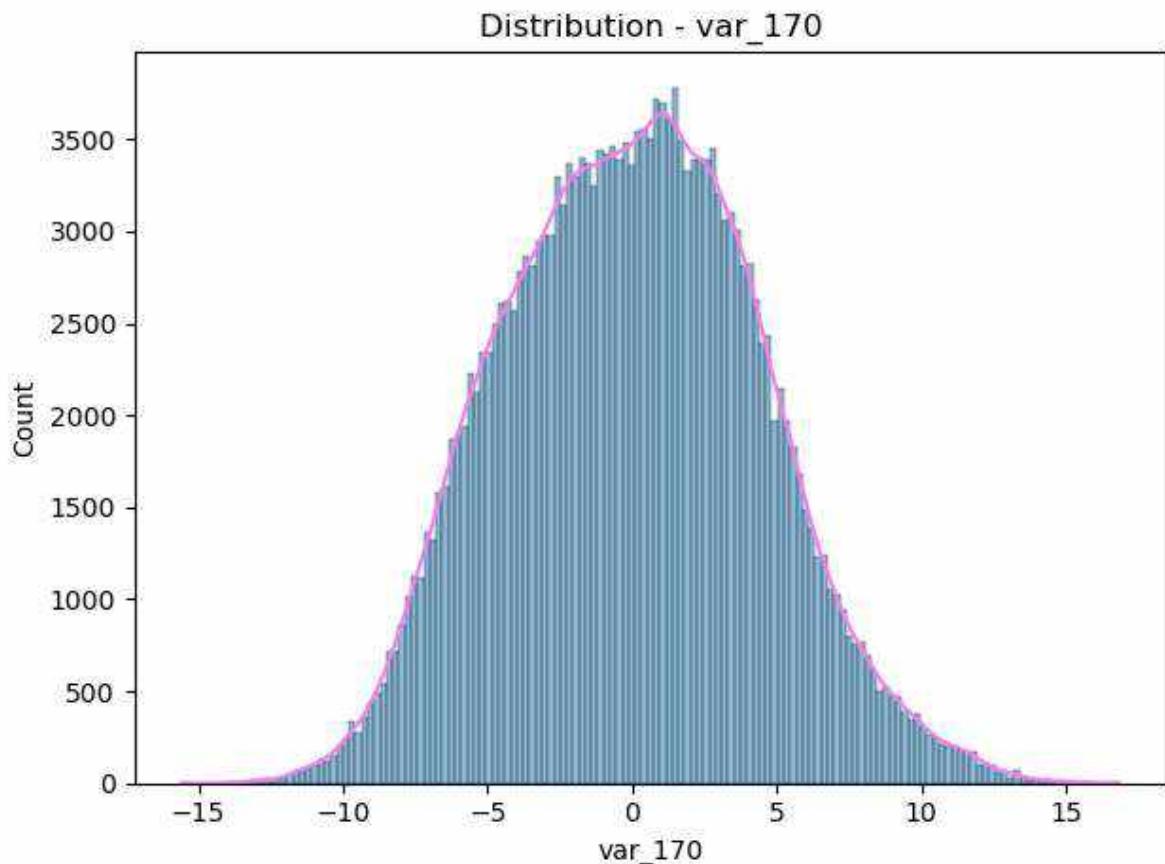
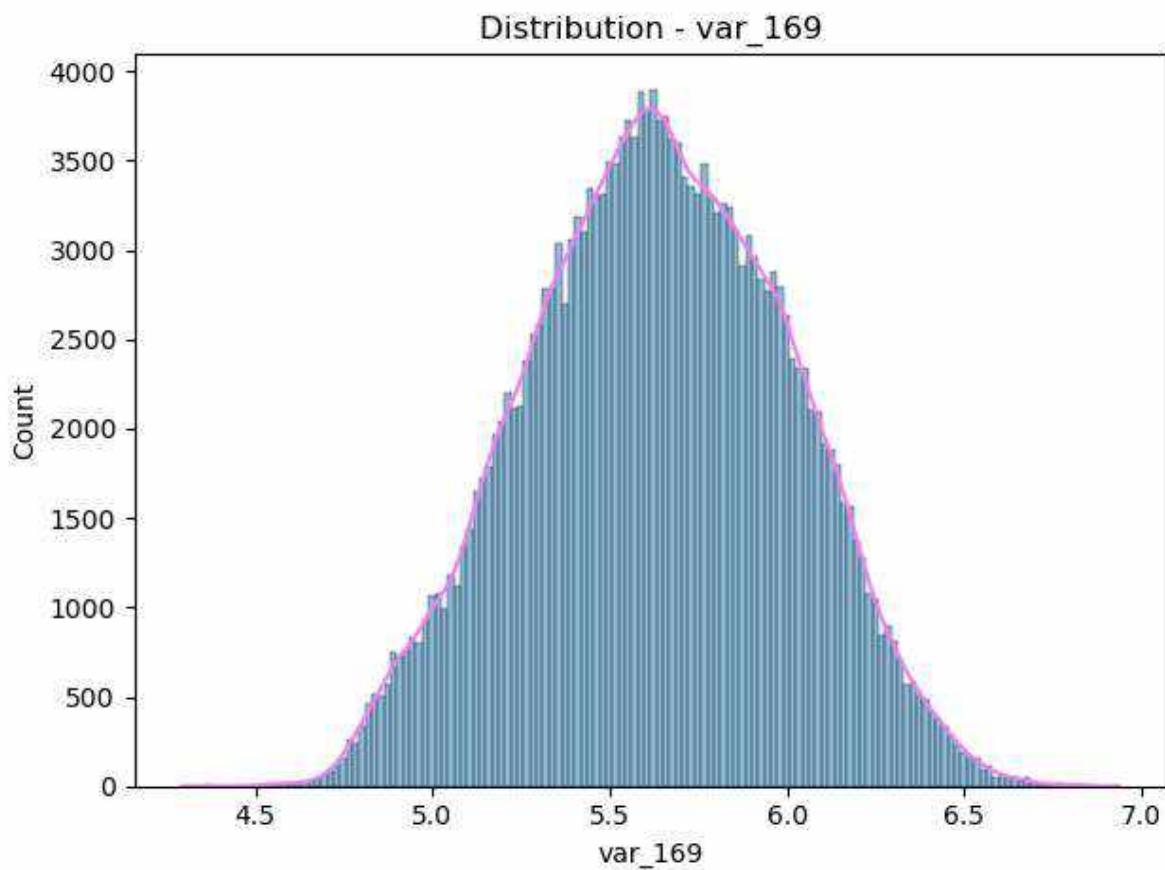


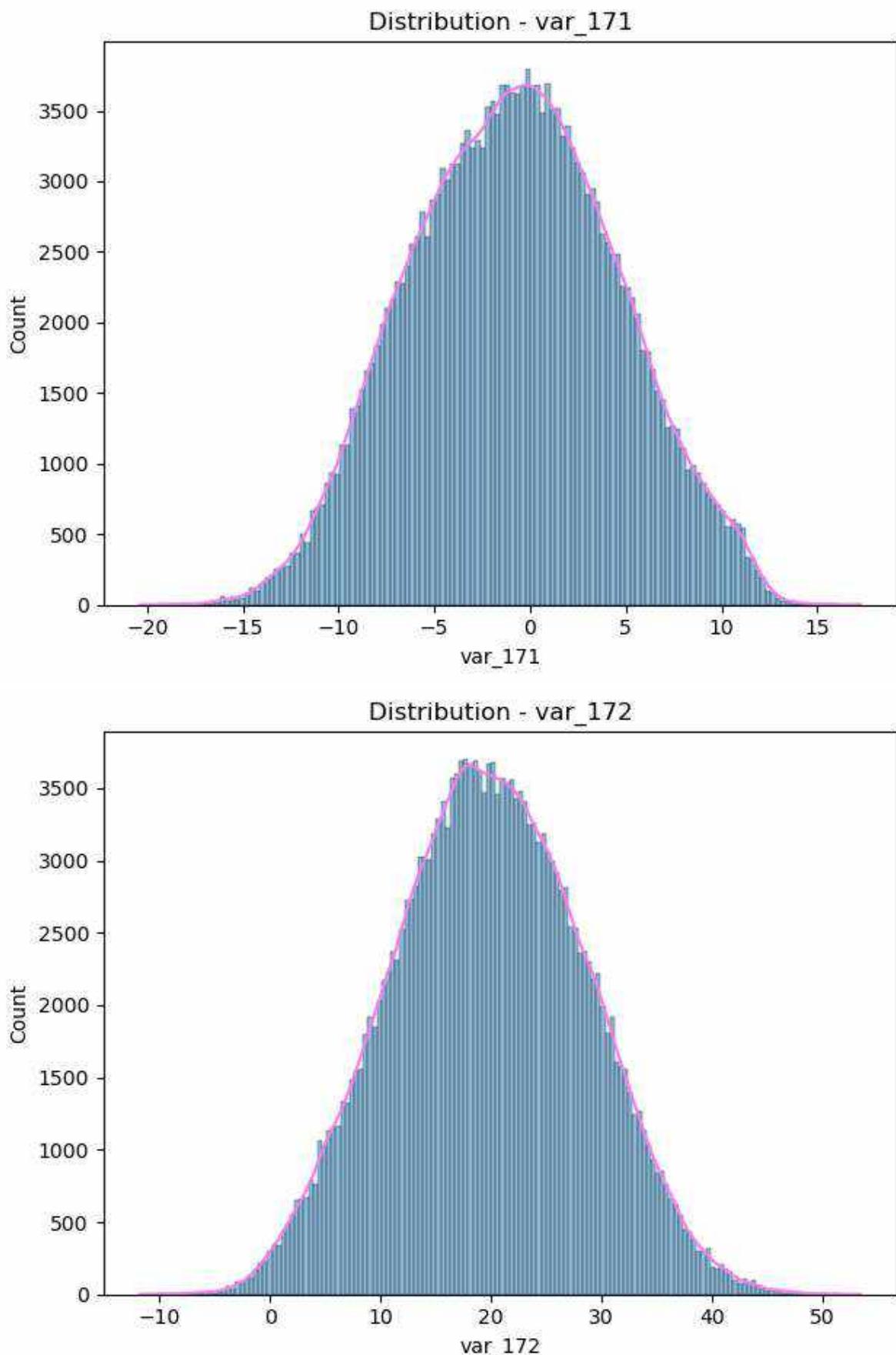


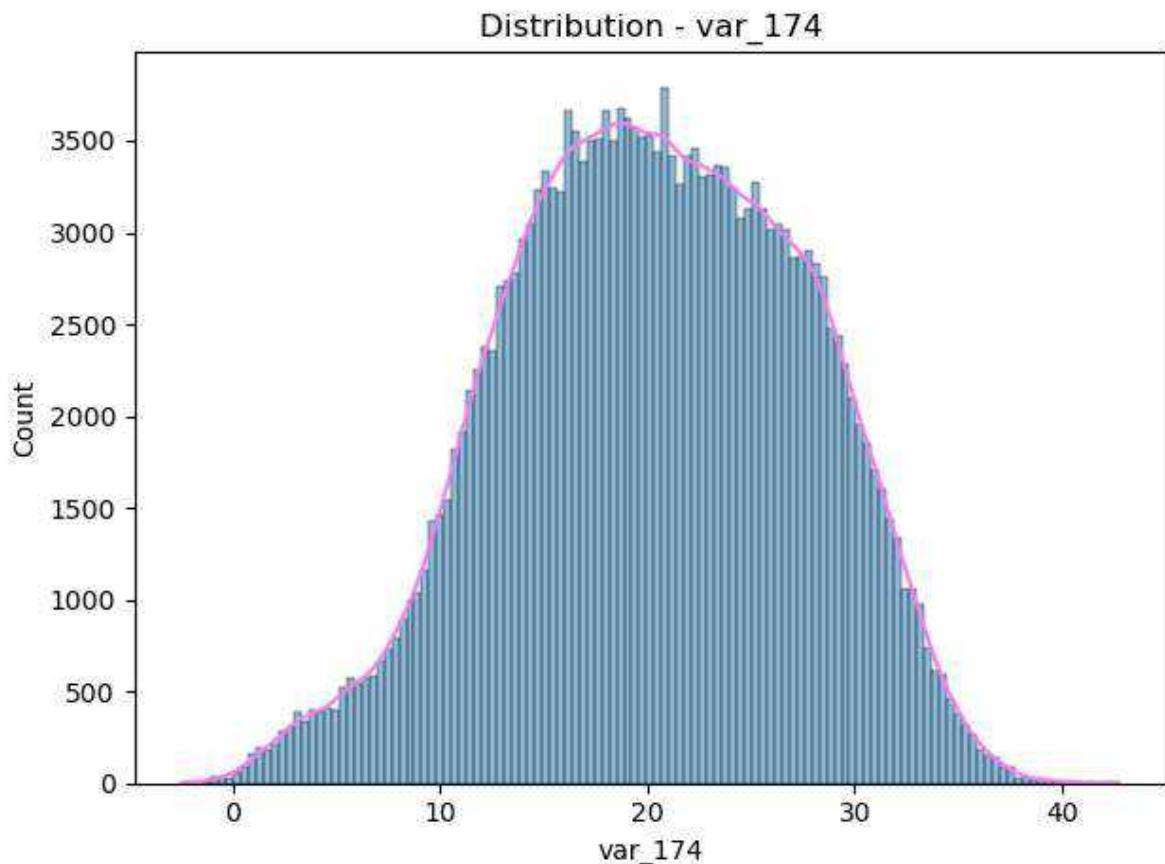
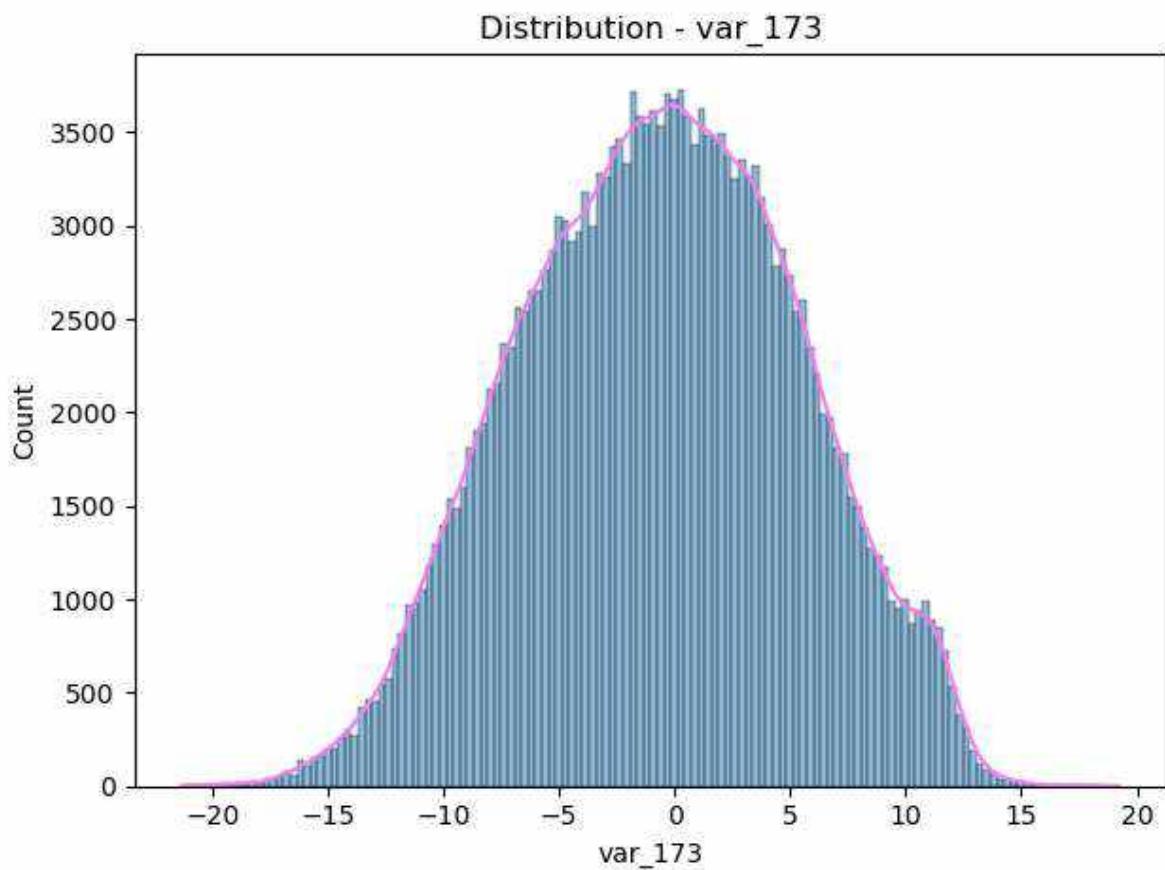




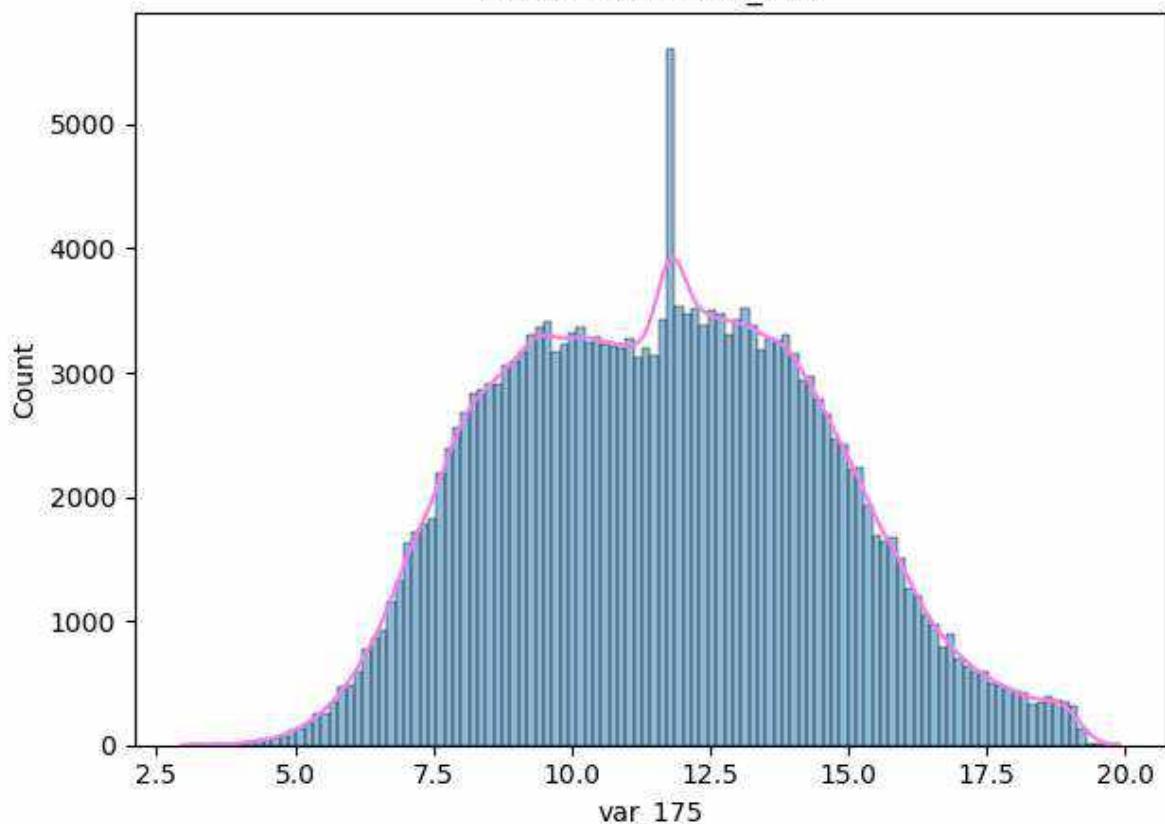




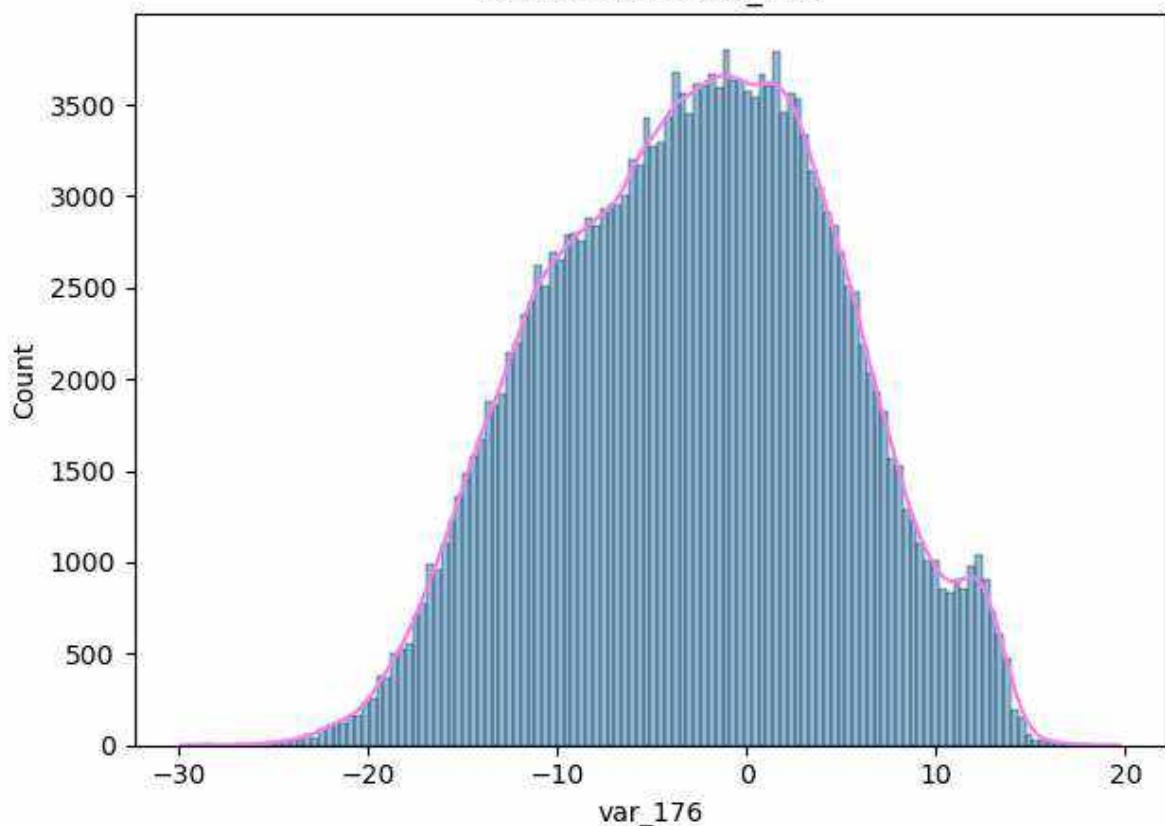


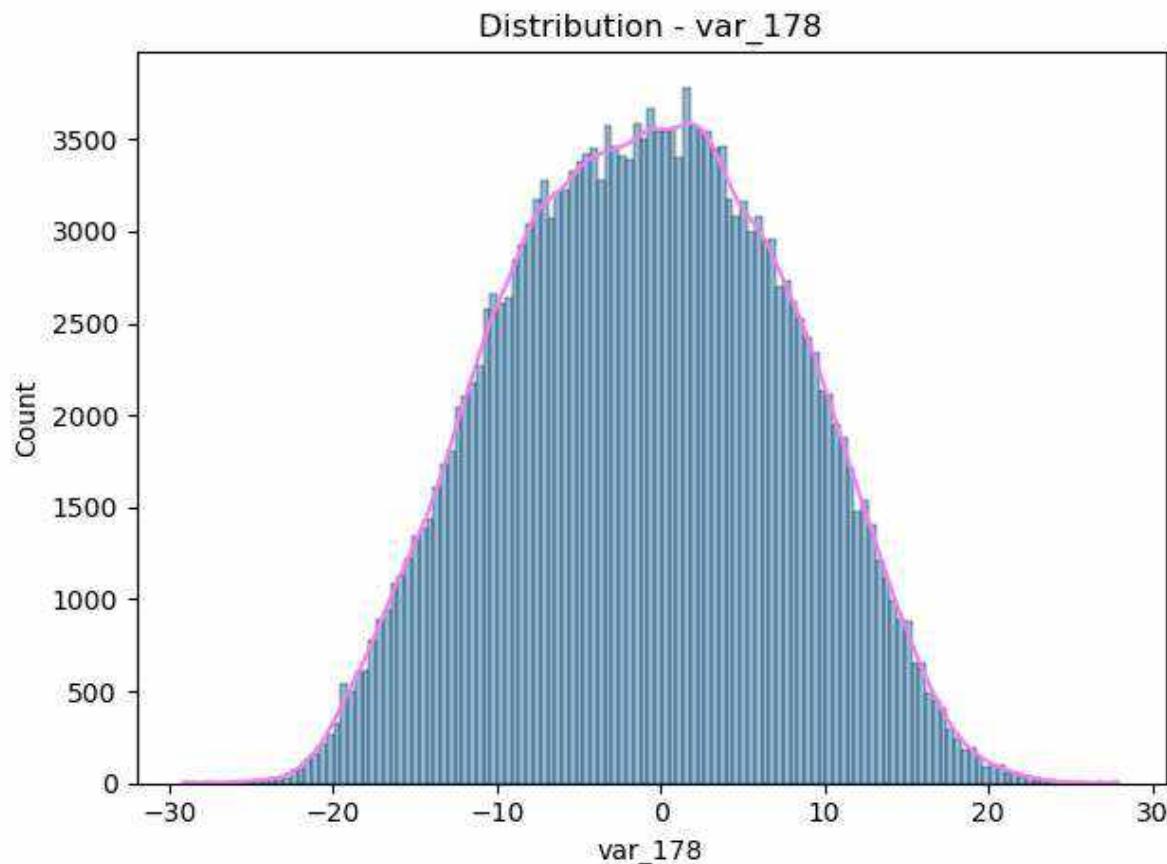
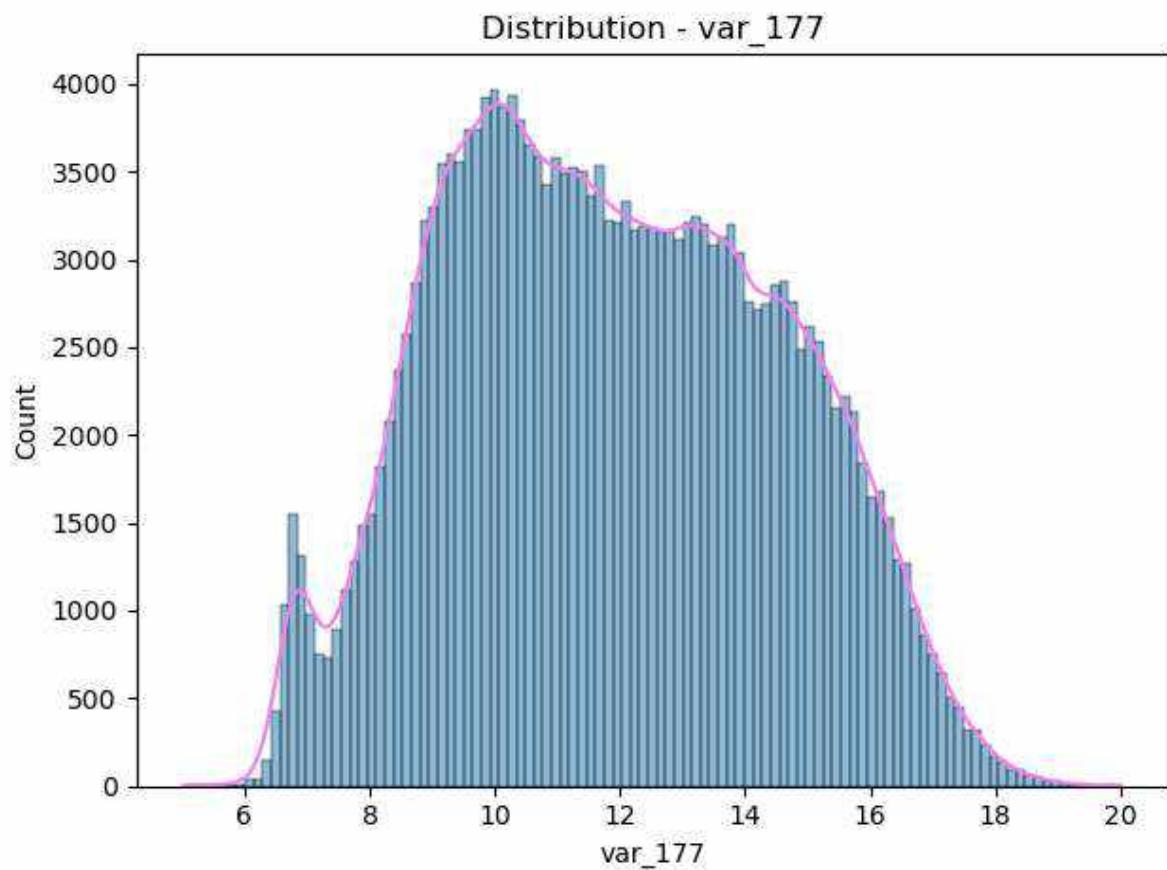


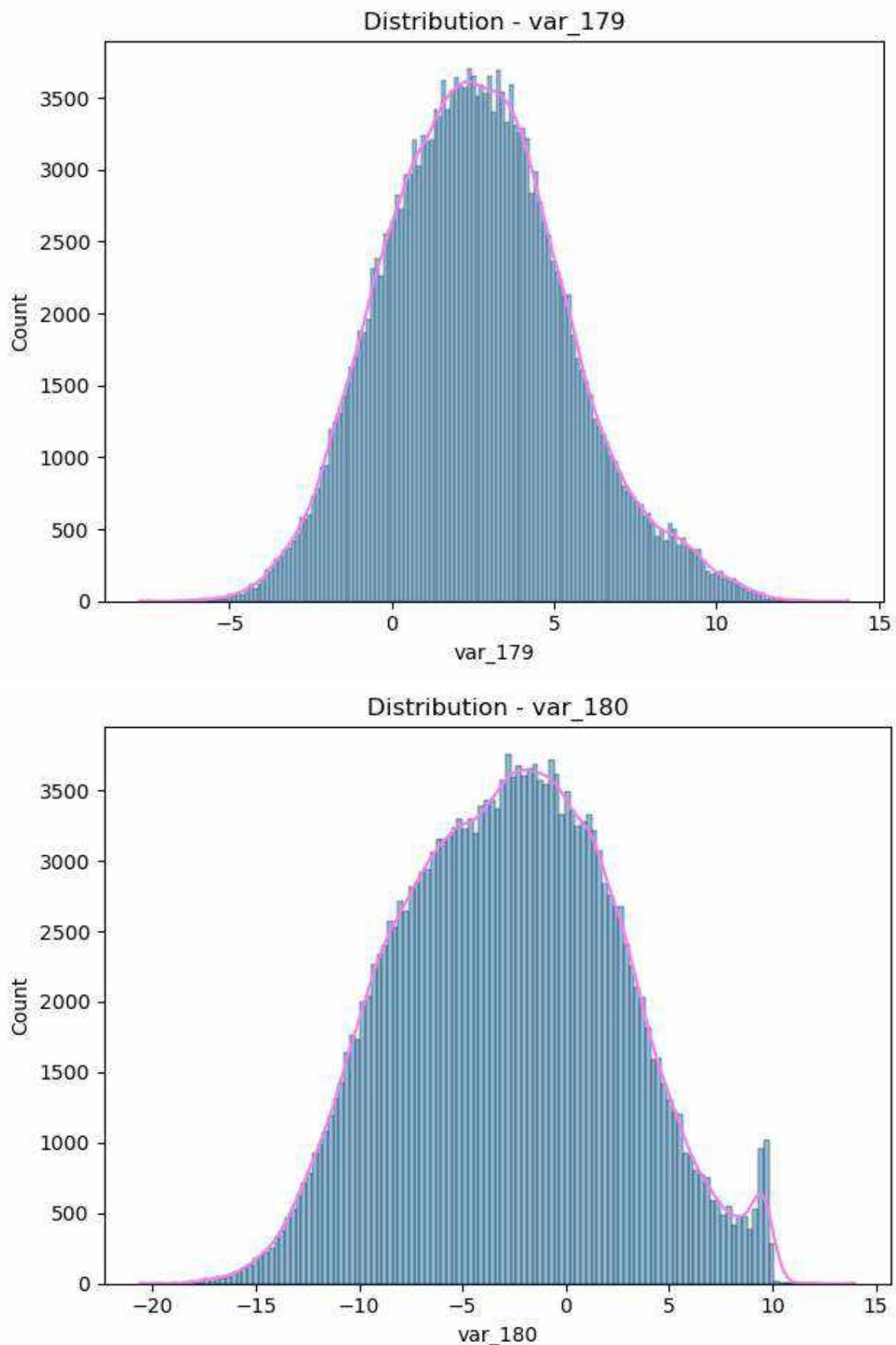
Distribution - var\_175

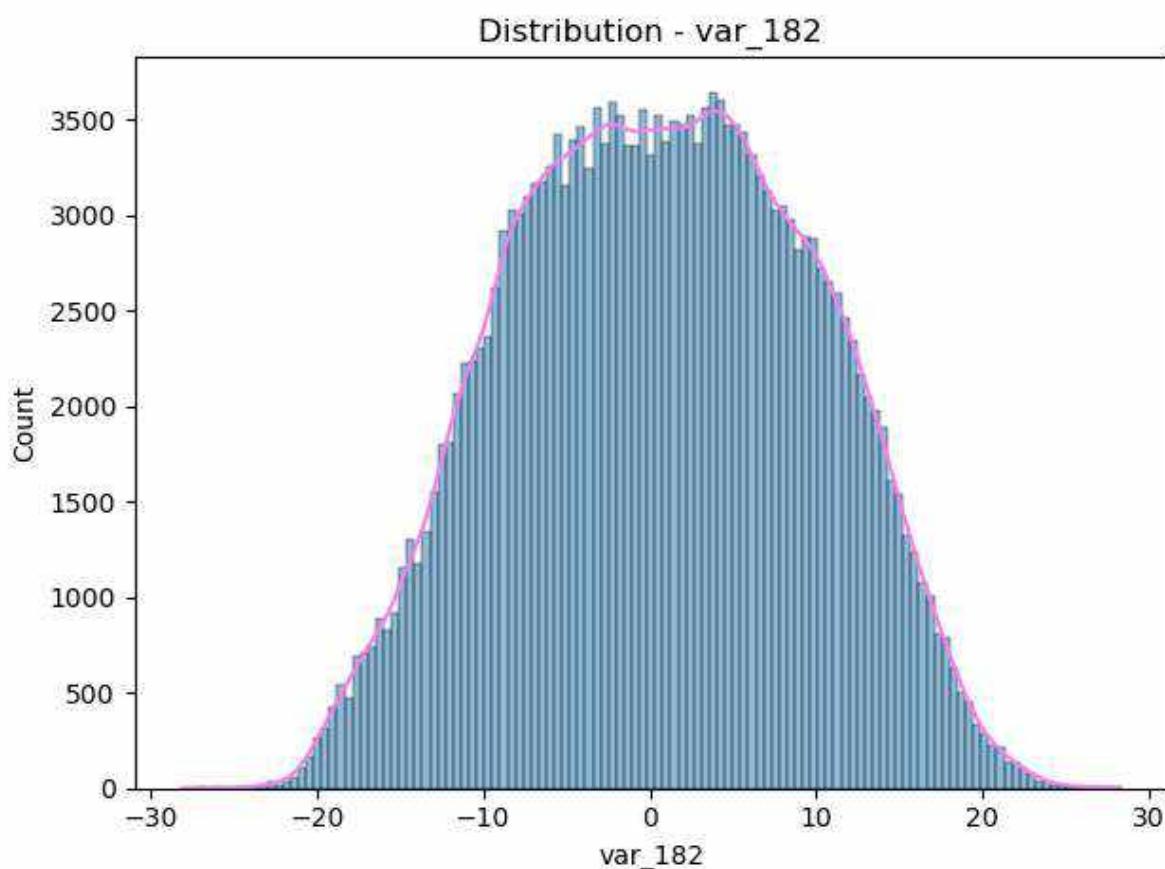
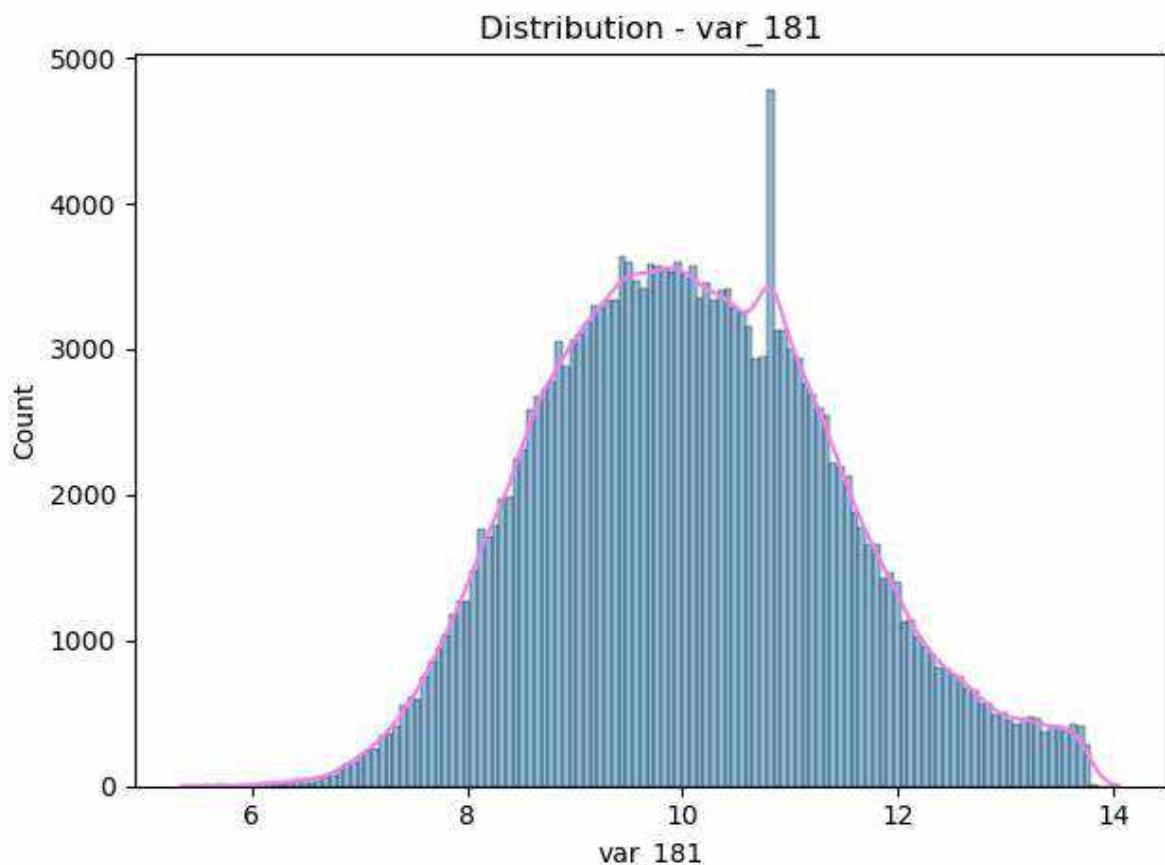


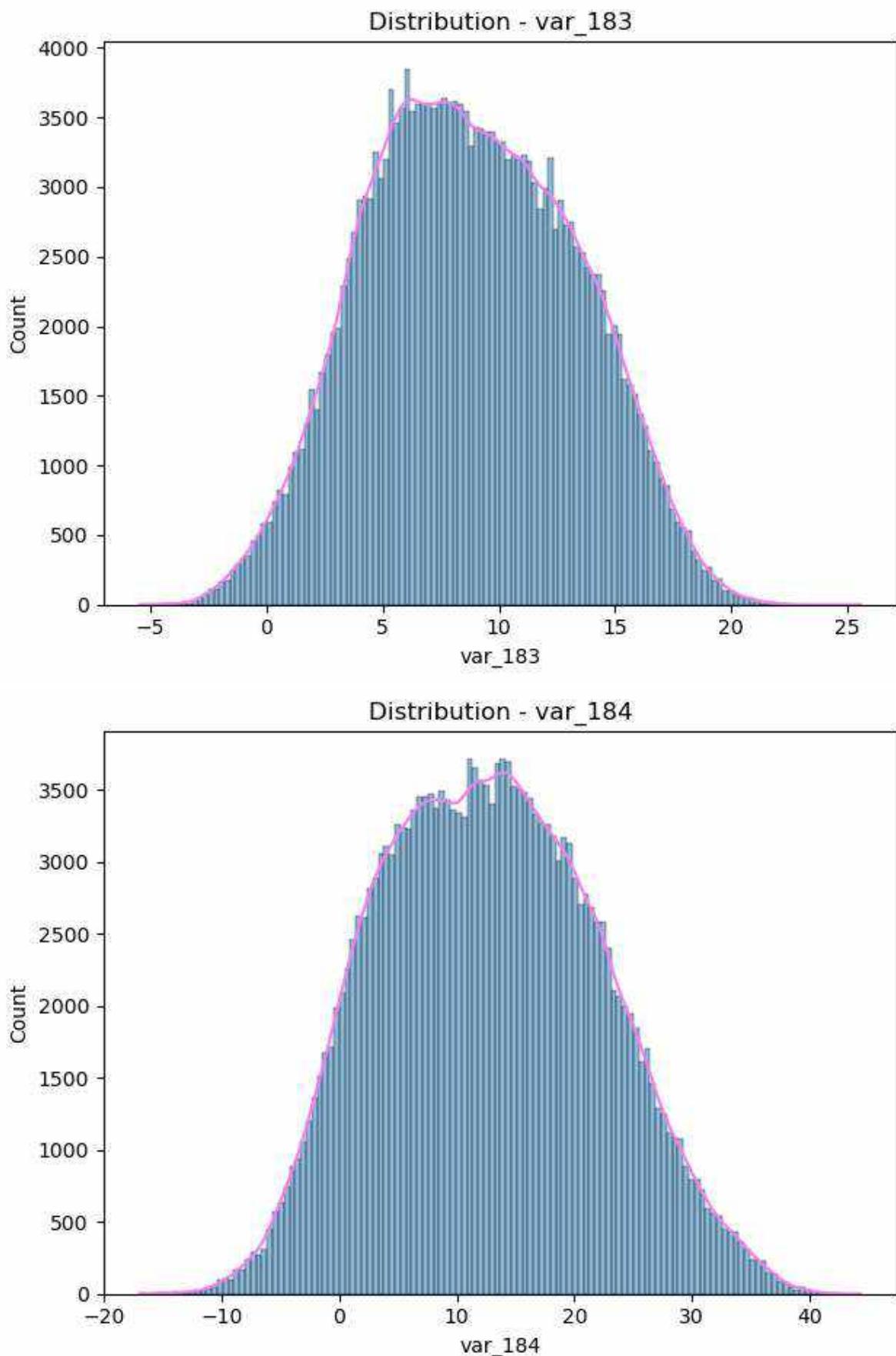
Distribution - var\_176

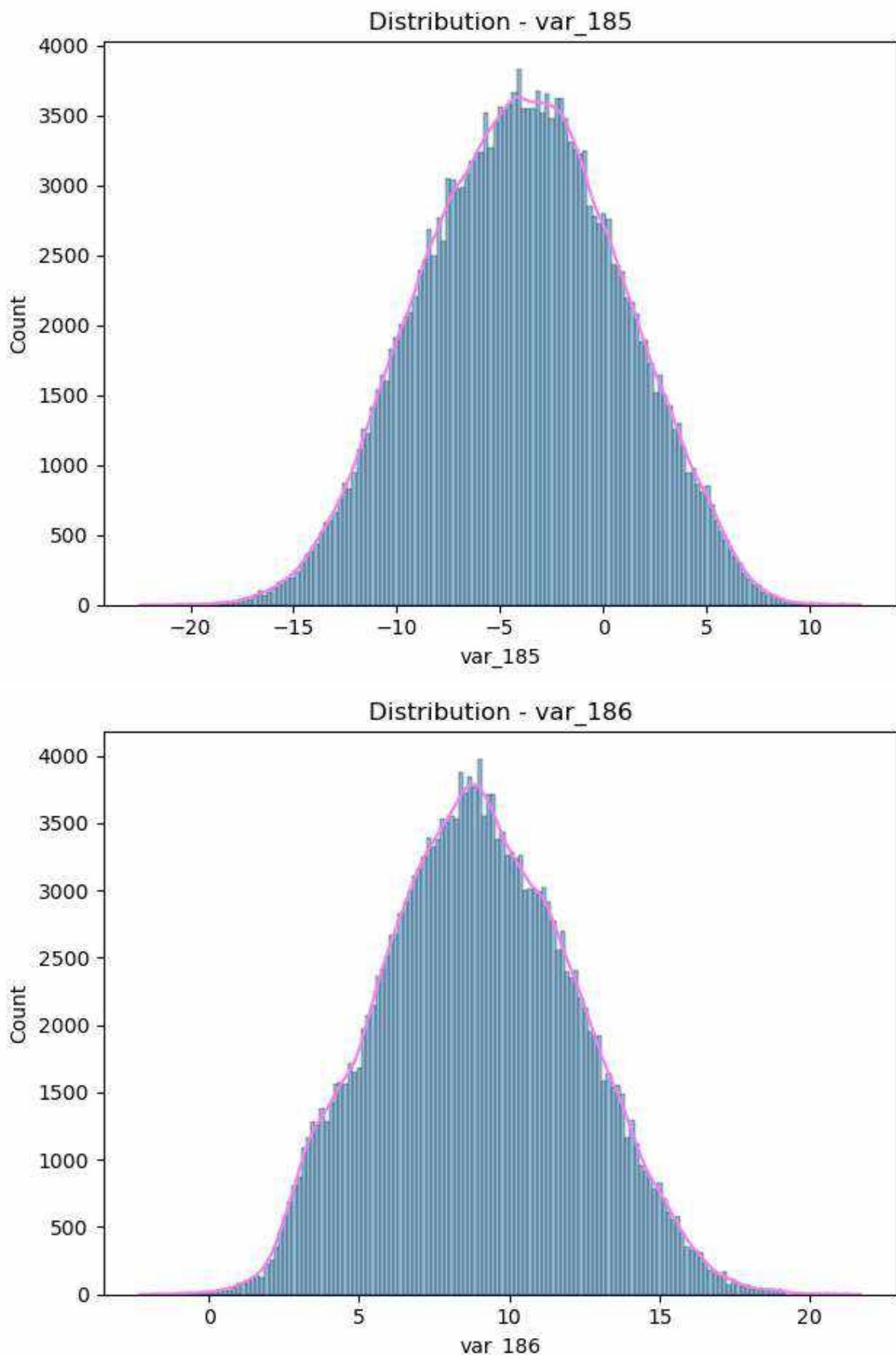


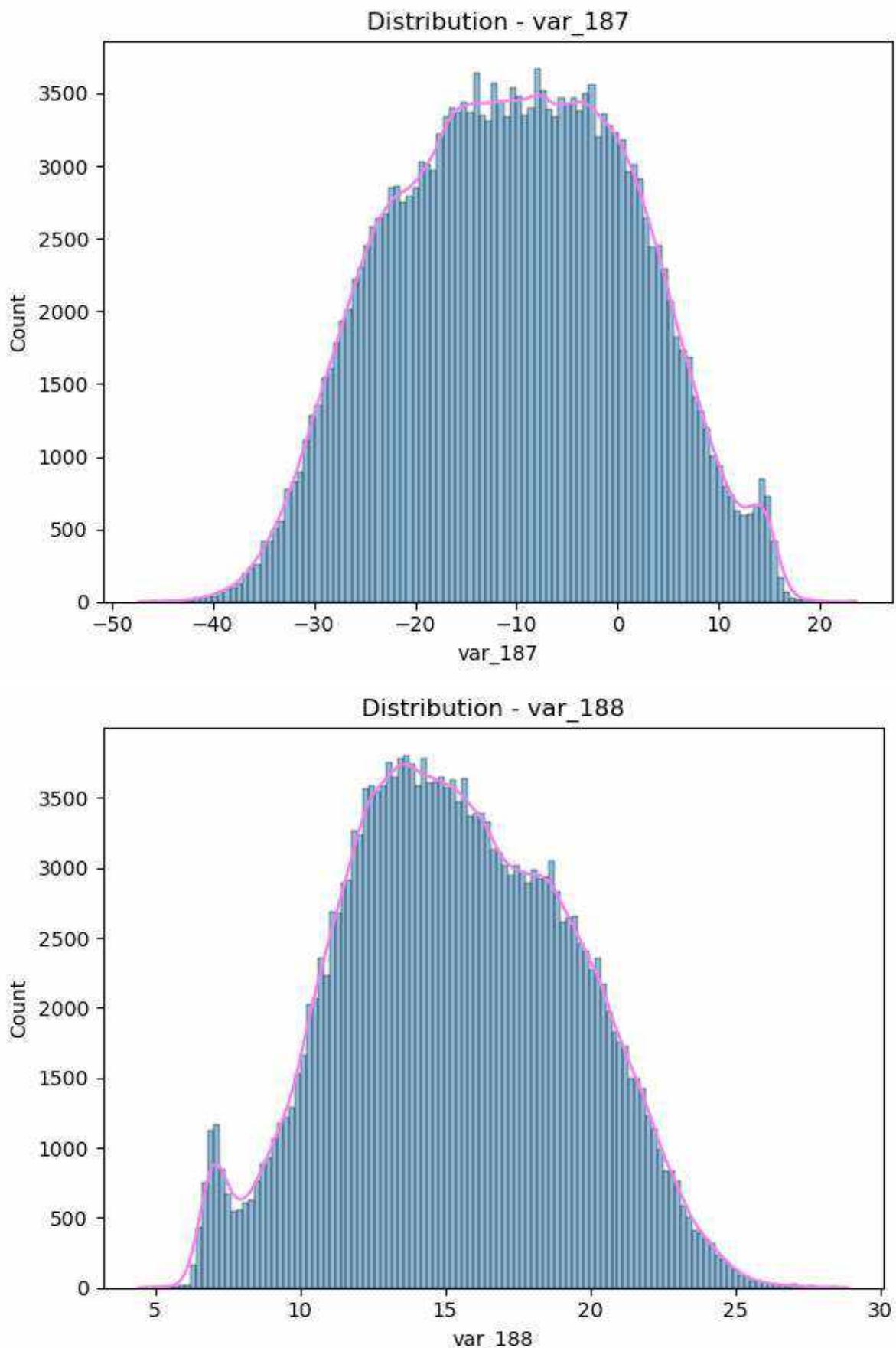


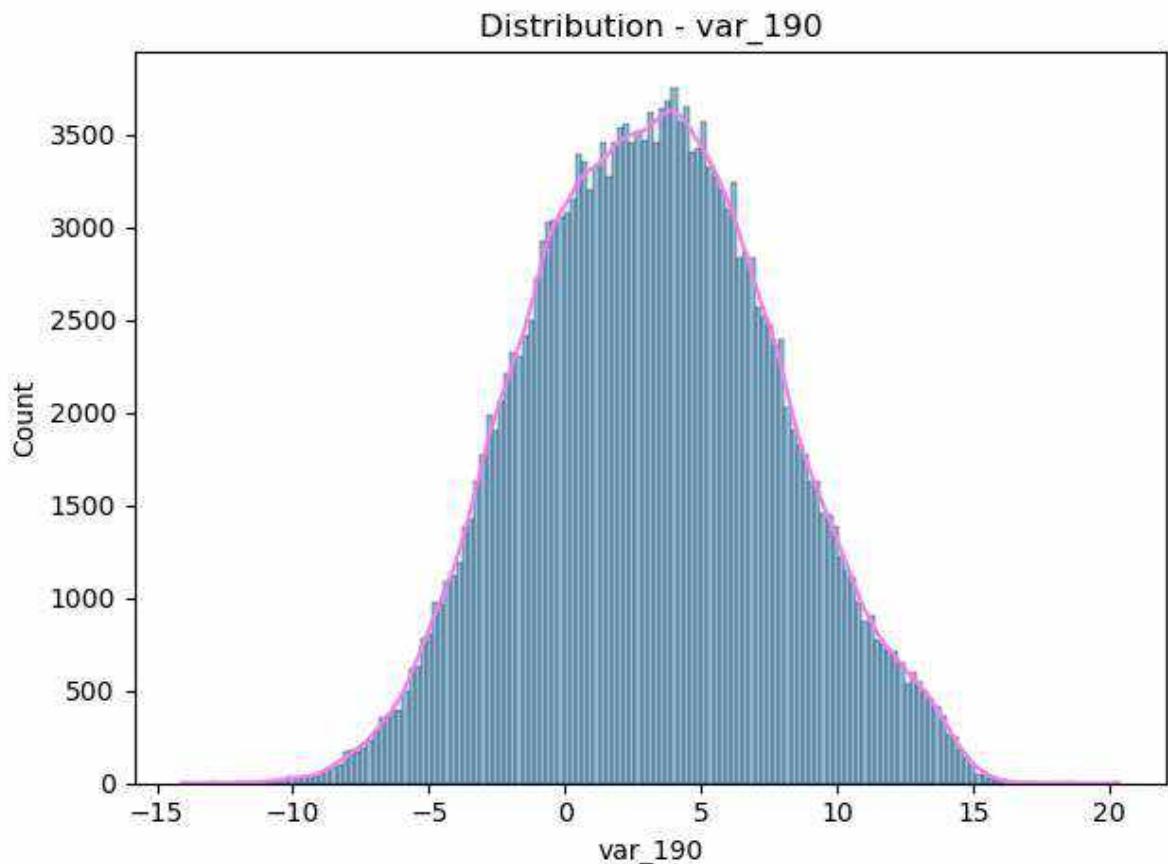
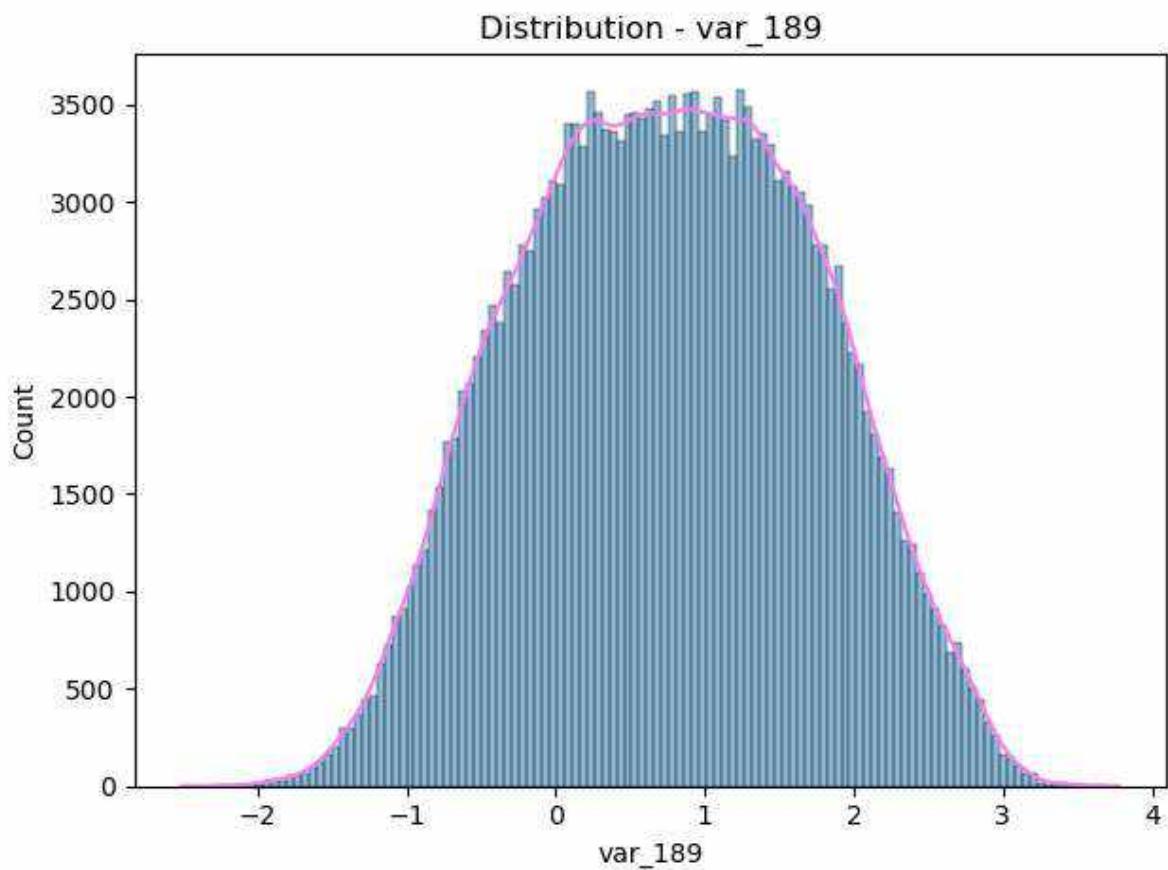


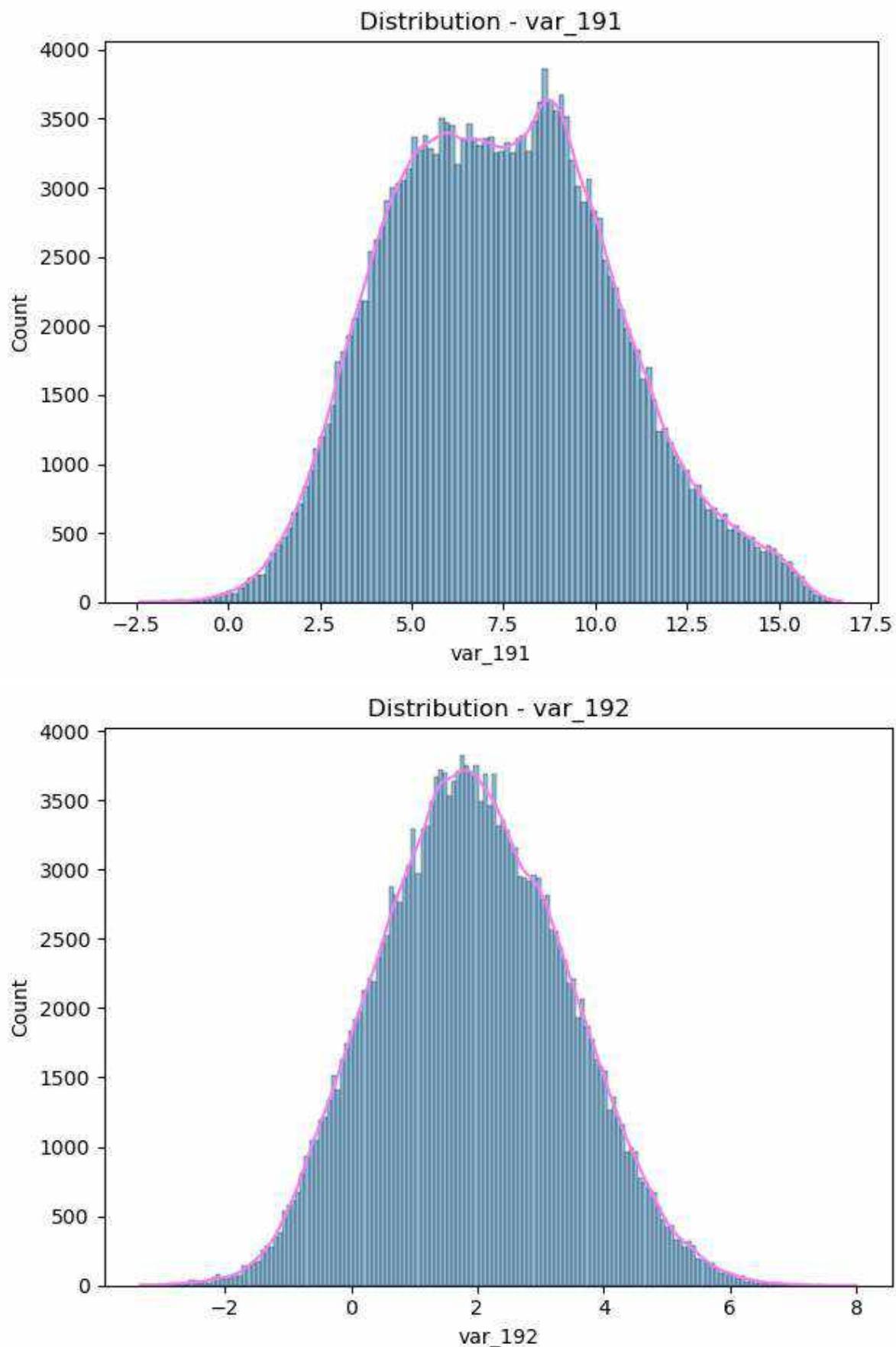


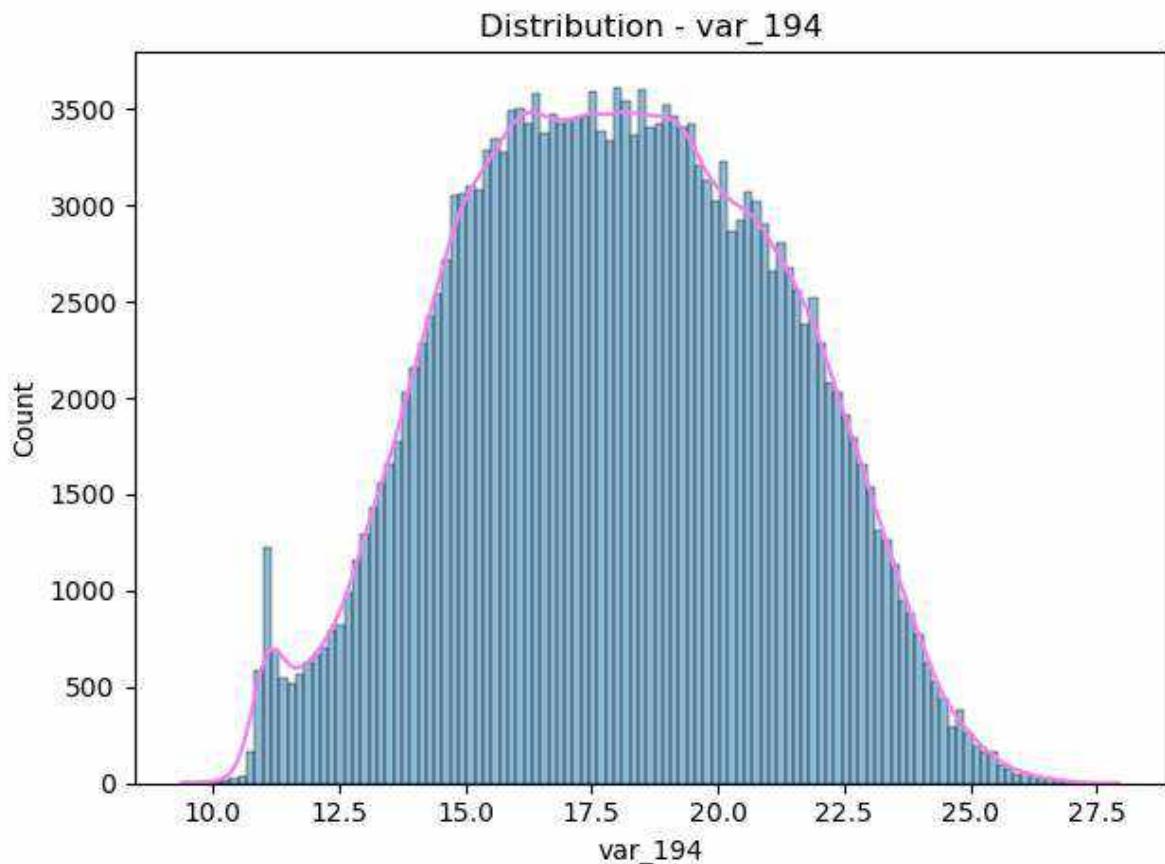
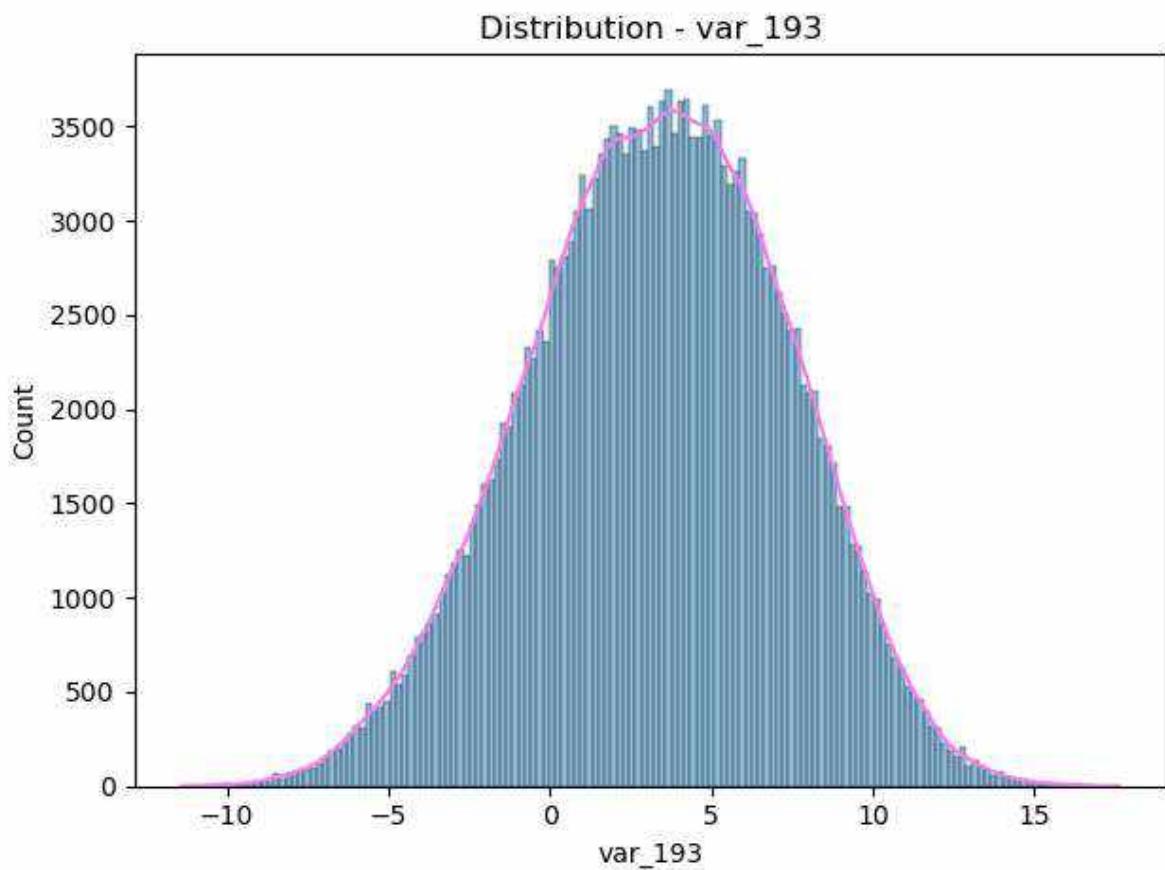


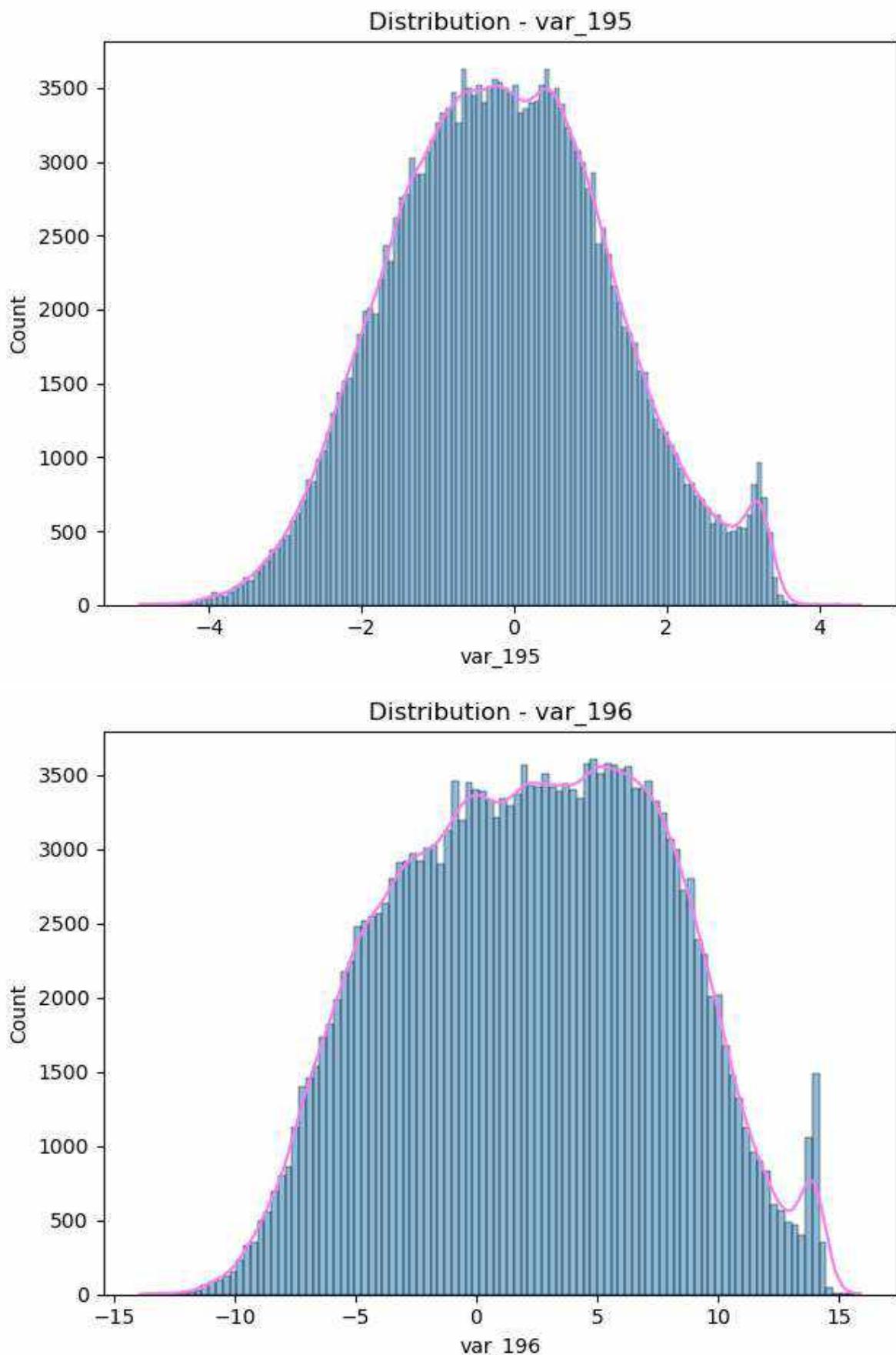


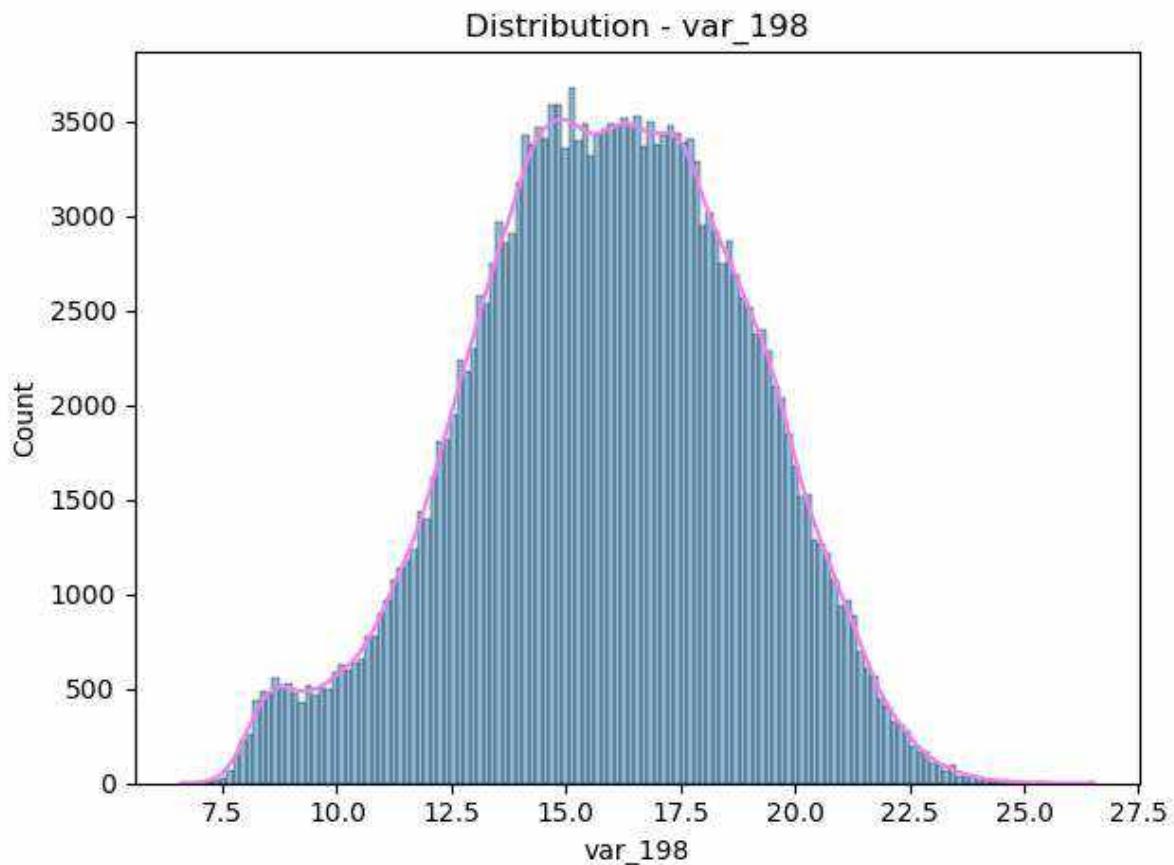
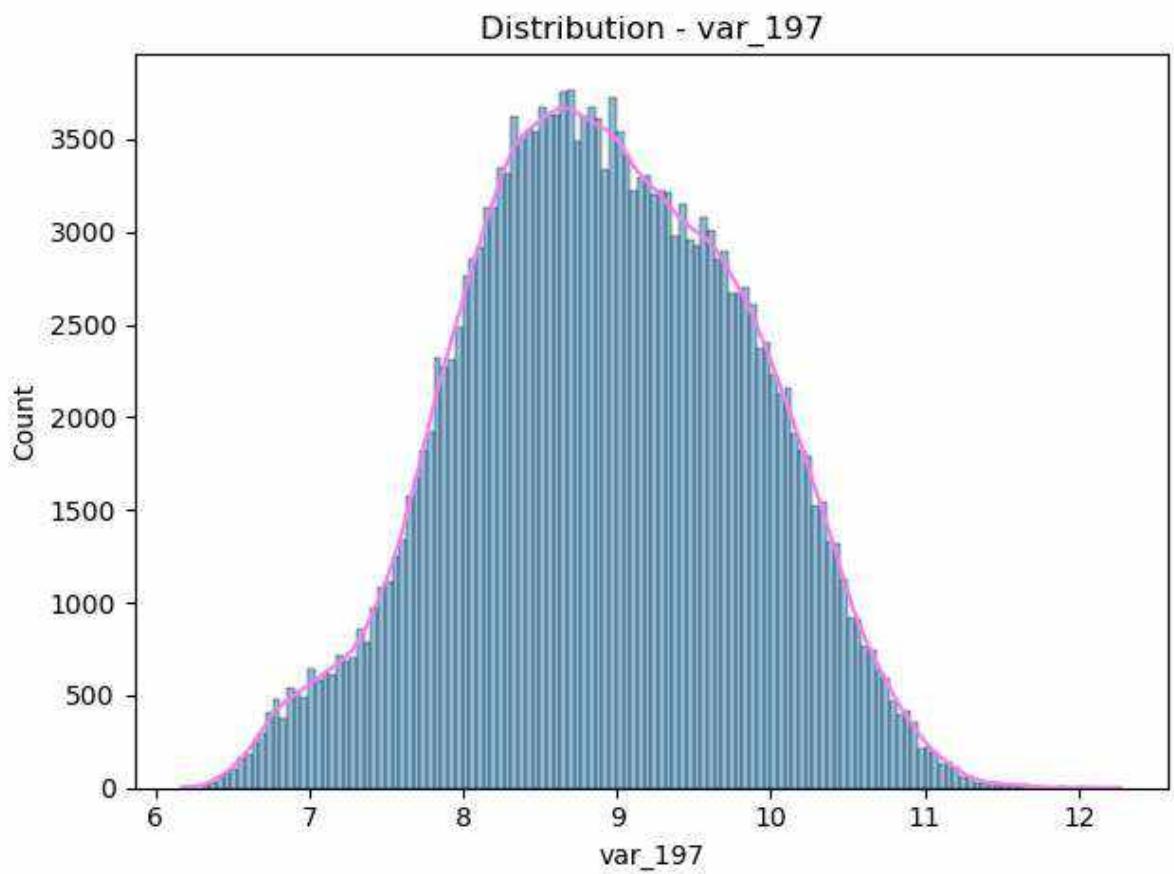


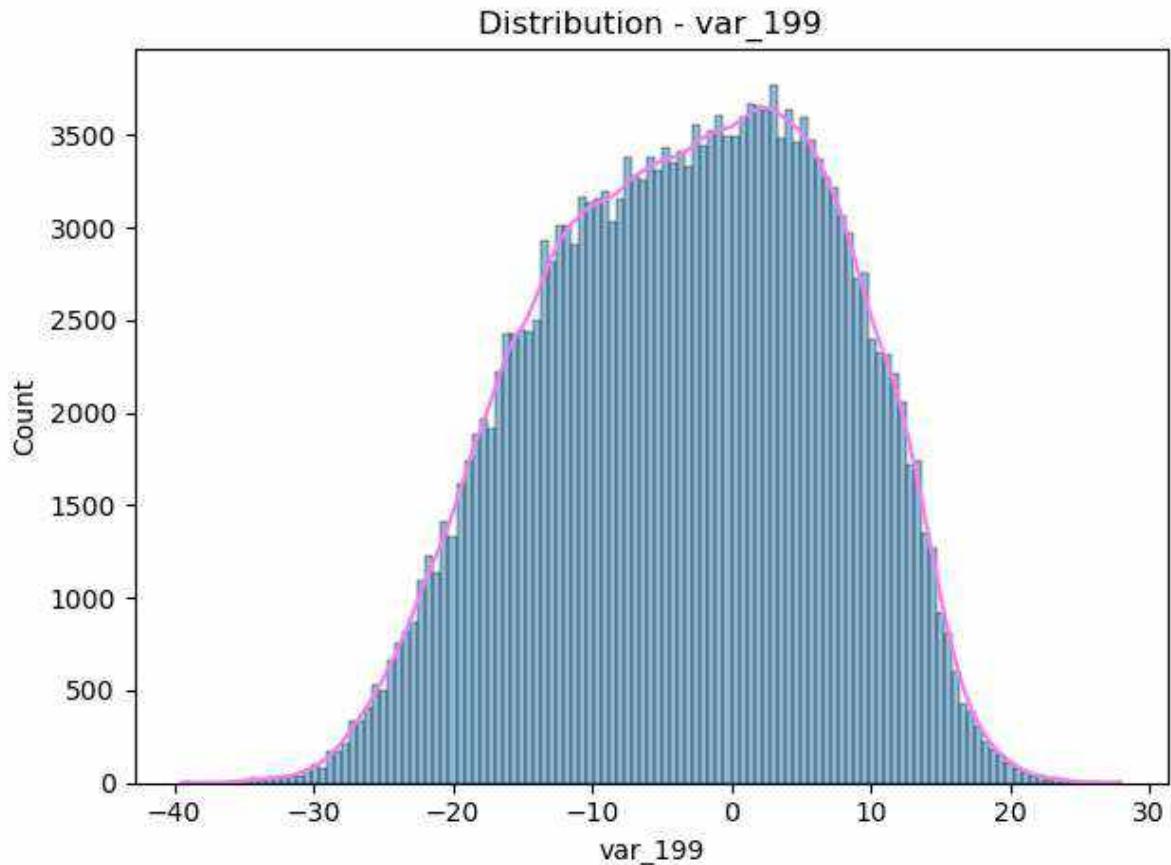












### test\_set: Distribution Summary based on columns

```
In [22]: # Exclude ID_code and target columns if present
numeric_columns = [column for column in data_test_set.columns if column not in ['ID_code', 'target']]

distribution_summary = {'Normal': 0, 'Right-skewed': 0, 'Left-skewed': 0, 'Outliers': 0}
outlier_columns = []

for column in numeric_columns:
    skewness = data_test_set[column].skew()
    if skewness > 1:
        distribution_summary['Right-skewed'] += 1
    elif skewness < -1:
        distribution_summary['Left-skewed'] += 1
    else:
        distribution_summary['Normal'] += 1

    q1 = data_test_set[column].quantile(0.25)
    q3 = data_test_set[column].quantile(0.75)
    iqr = q3 - q1
    lower_bound = q1 - 1.5 * iqr
    upper_bound = q3 + 1.5 * iqr
    outliers = data_test_set[(data_test_set[column] < lower_bound) | (data_test_set[column] > upper_bound)]

    if not outliers.empty:
        distribution_summary['Outliers'] += 1
        outlier_columns.append(column)

print("Distribution Summary based on columns:")
for shape, count in distribution_summary.items():
    print(f"{shape}: {count}")
```

Distribution Summary based on columns:

Normal: 200  
Right-skewed: 0  
Left-skewed: 0  
Outliers: 180

For test\_set, there are 200 columns that exhibit a normal distribution. So, the majority of the columns exhibit a normal distribution, while 180 columns have outliers. None of the columns are right-skewed or Left-skewed.

## Identifying potential outliers for test\_set

In [23]:

```

num_cols = 3
num_rows = (len(numeric_columns) + num_cols - 1) // num_cols

fig, axes = plt.subplots(num_rows, num_cols, figsize=(15, 5*num_rows))
fig.tight_layout(pad=3.0)

outliers = []
for i, column in enumerate(numeric_columns):
    row = i // num_cols
    col = i % num_cols
    ax = axes[row, col] if num_rows > 1 else axes[col]

    sns.boxplot(data=data_test_set[column], ax=ax, color='skyblue')

    # Identify potential outliers
    q1 = data_test_set[column].quantile(0.25)
    q3 = data_test_set[column].quantile(0.75)
    iqr = q3 - q1
    lower_bound = q1 - 1.5 * iqr
    upper_bound = q3 + 1.5 * iqr
    column_outliers = data_test_set[(data_test_set[column] < lower_bound) | (data_
        outliers.extend(column_outliers.index))

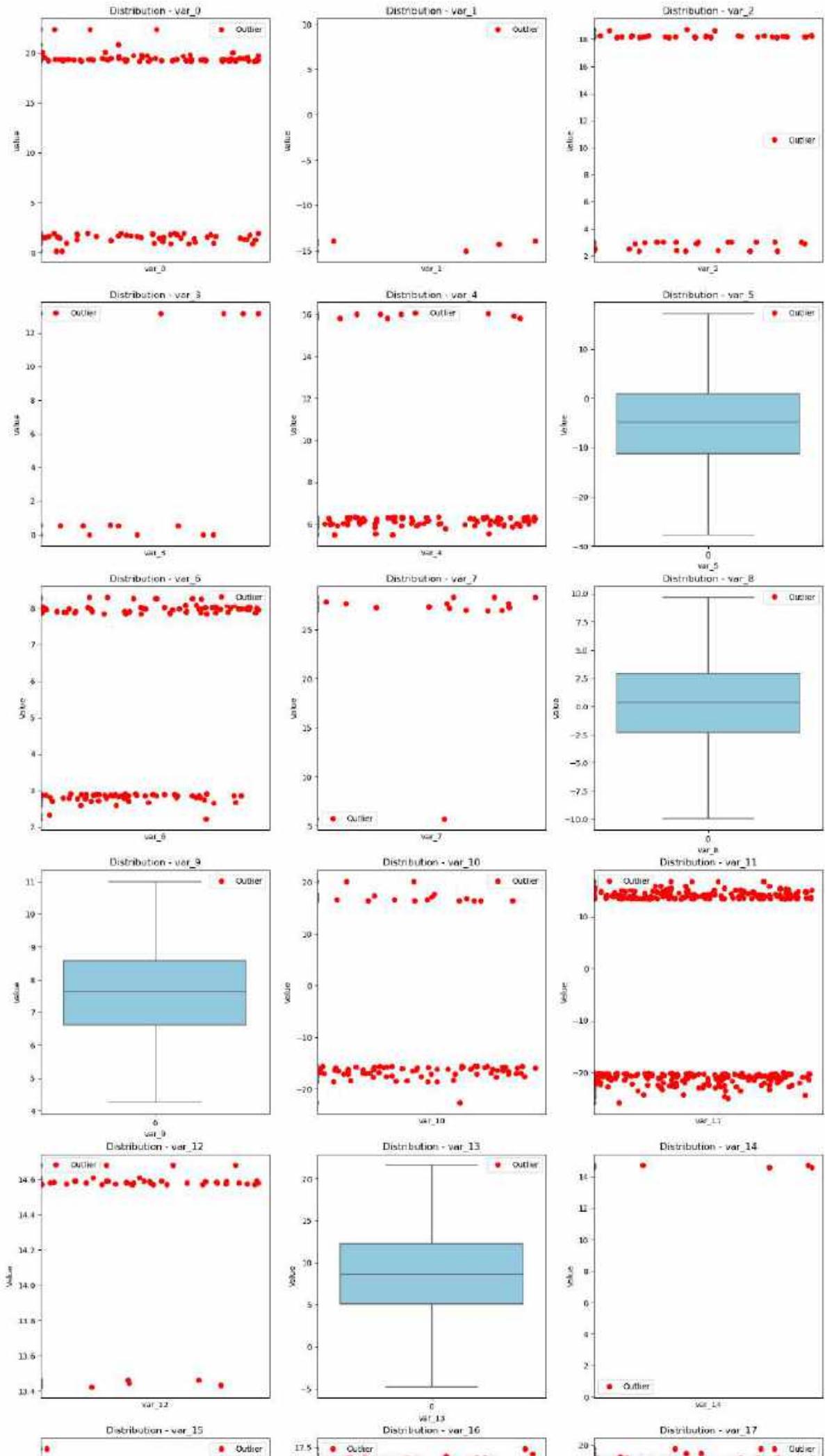
    ax.plot(column_outliers.index, column_outliers[column], 'ro', label='Outlier')
    ax.legend()

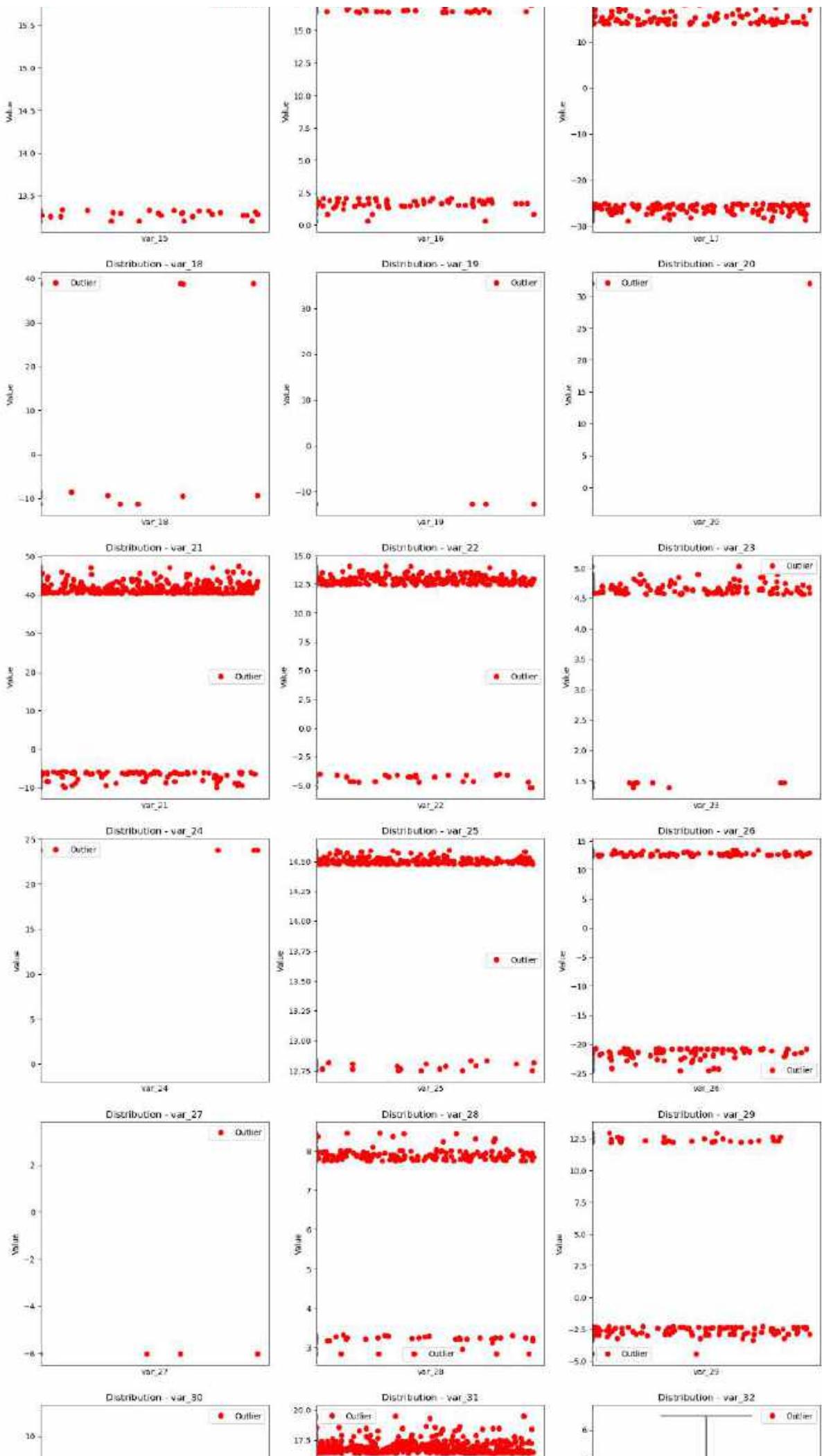
    ax.set_title(f'Distribution - {column}')
    ax.set_xlabel(column)
    ax.set_ylabel('Value')

if len(numeric_columns) % num_cols != 0:
    empty_cols = num_cols - len(numeric_columns) % num_cols
    if num_rows > 1:
        for i in range(empty_cols):
            axes[num_rows-1, num_cols-i-1].axis('off')
    else:
        for i in range(empty_cols):
            axes[num_cols-i-1].axis('off')

plt.show()

```

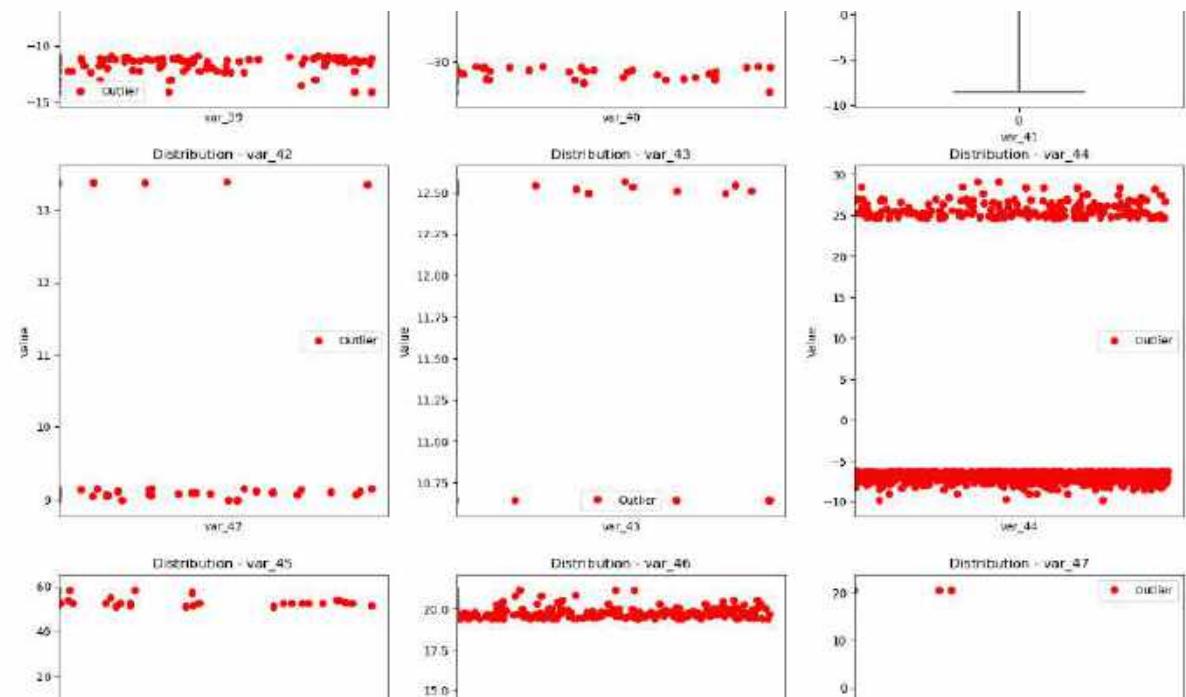
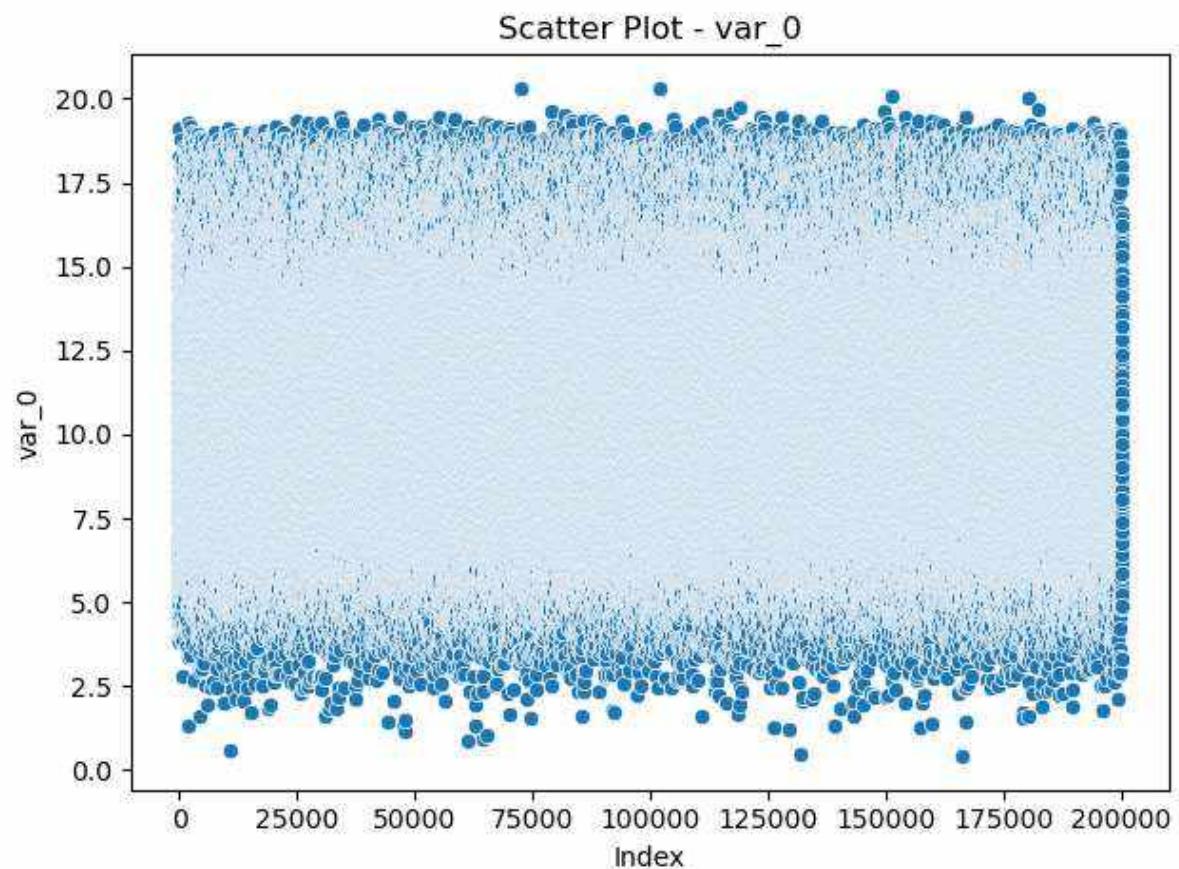


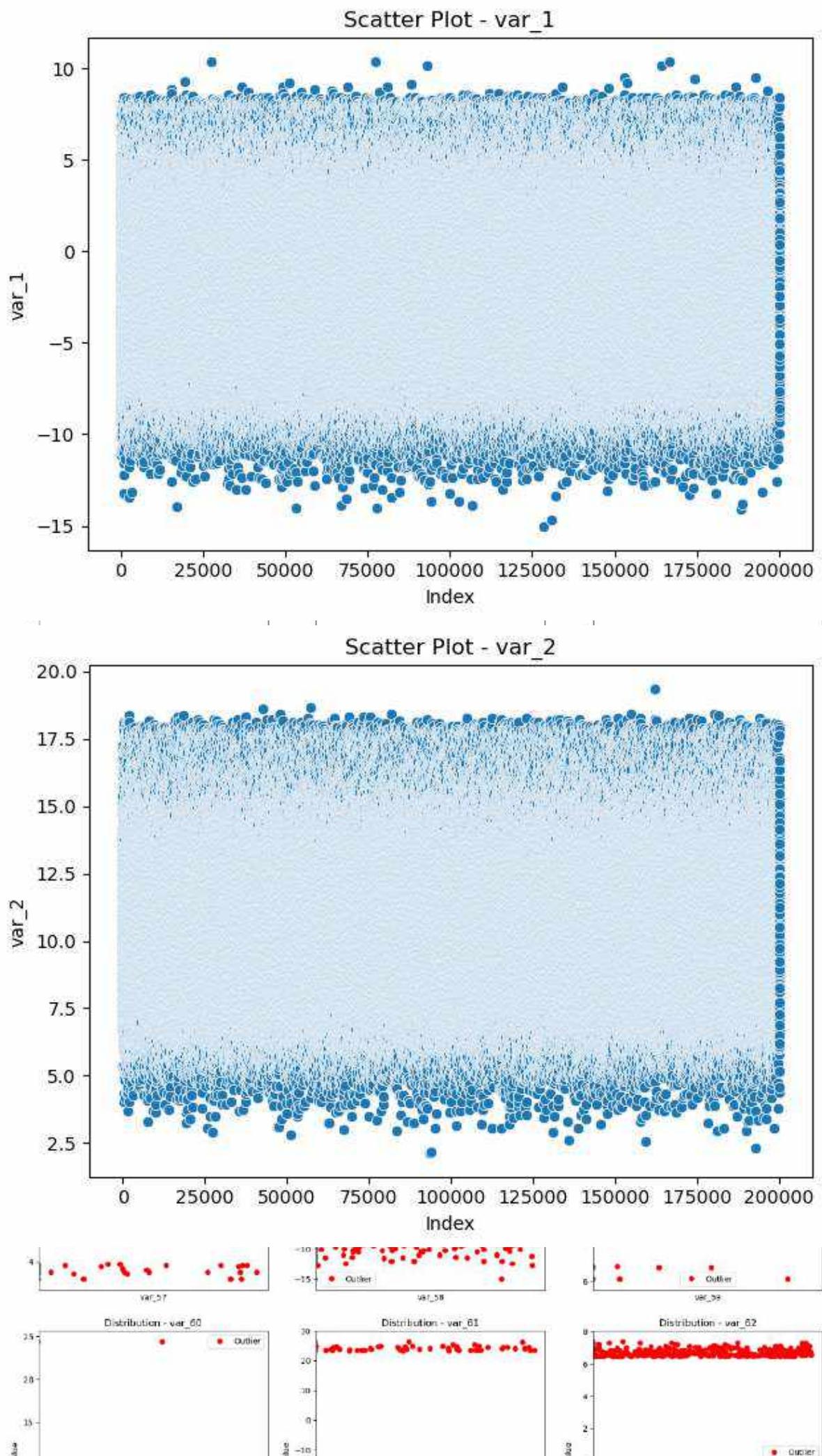


## Scatter plot of test and training set

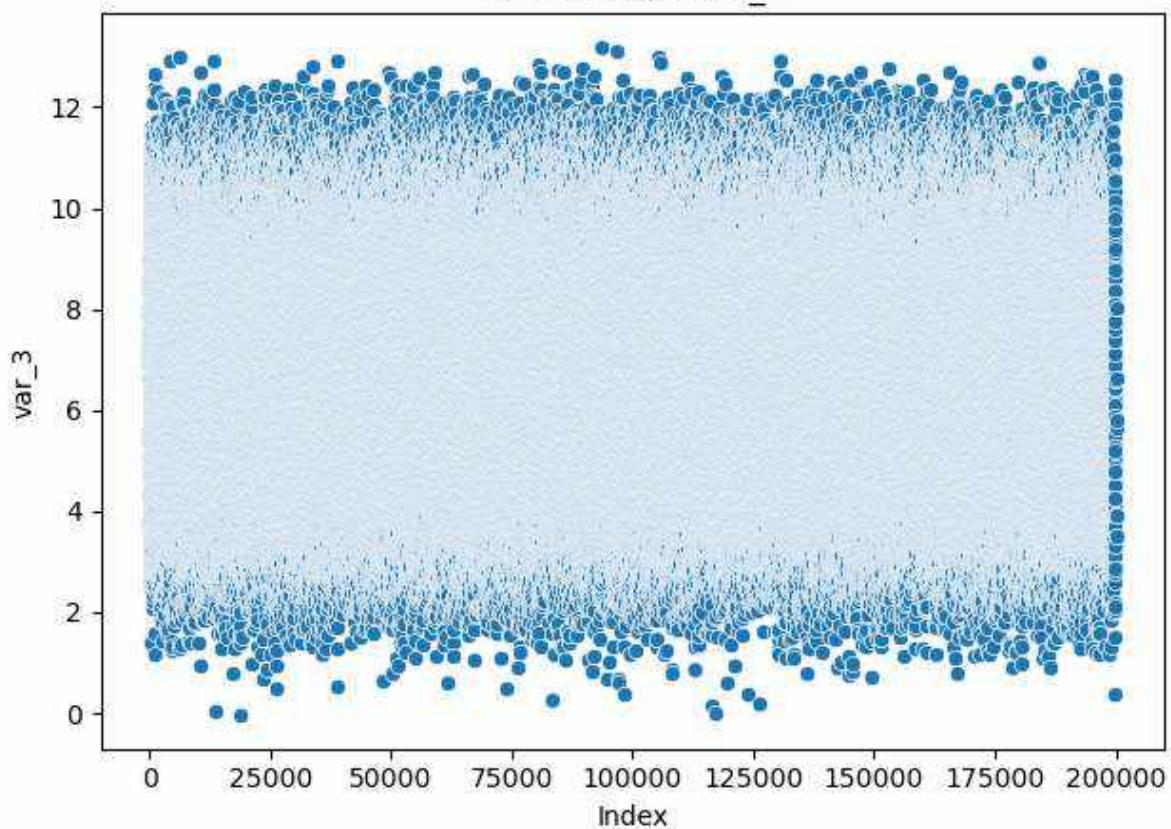
In [25]: # Scatter plot of training set

```
for column in data_training_set.columns:
    if column not in ['ID_code', 'target']:
        sns.scatterplot(x=range(len(data_training_set)), y=data_training_set[column])
        plt.title(f'Scatter Plot - {column}')
        plt.xlabel('Index')
        plt.ylabel(column)
        plt.tight_layout()
        plt.show()
```

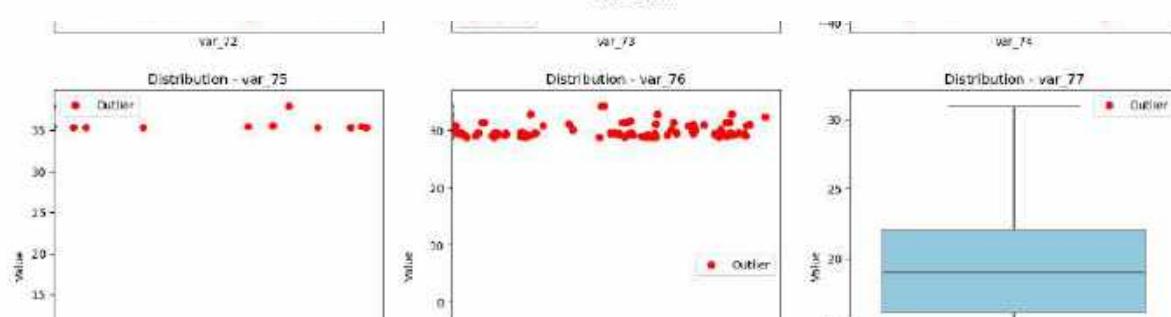
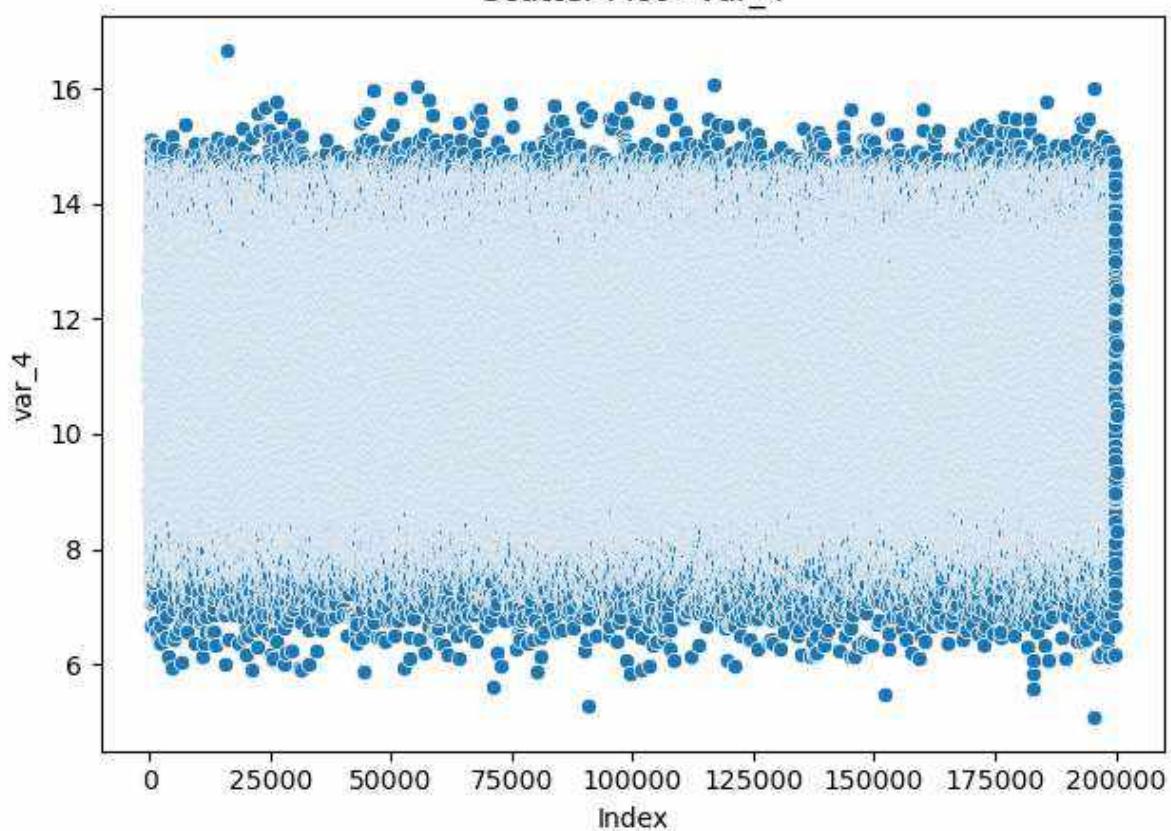




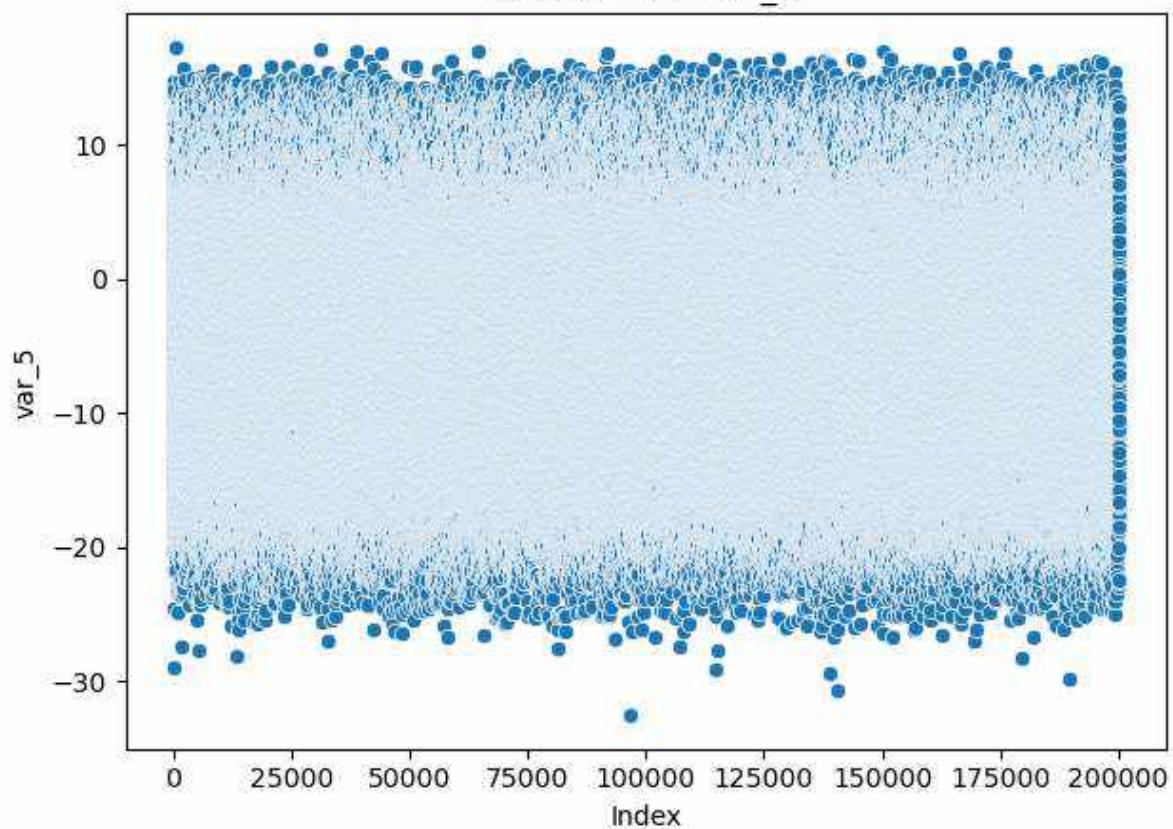
Scatter Plot - var\_3



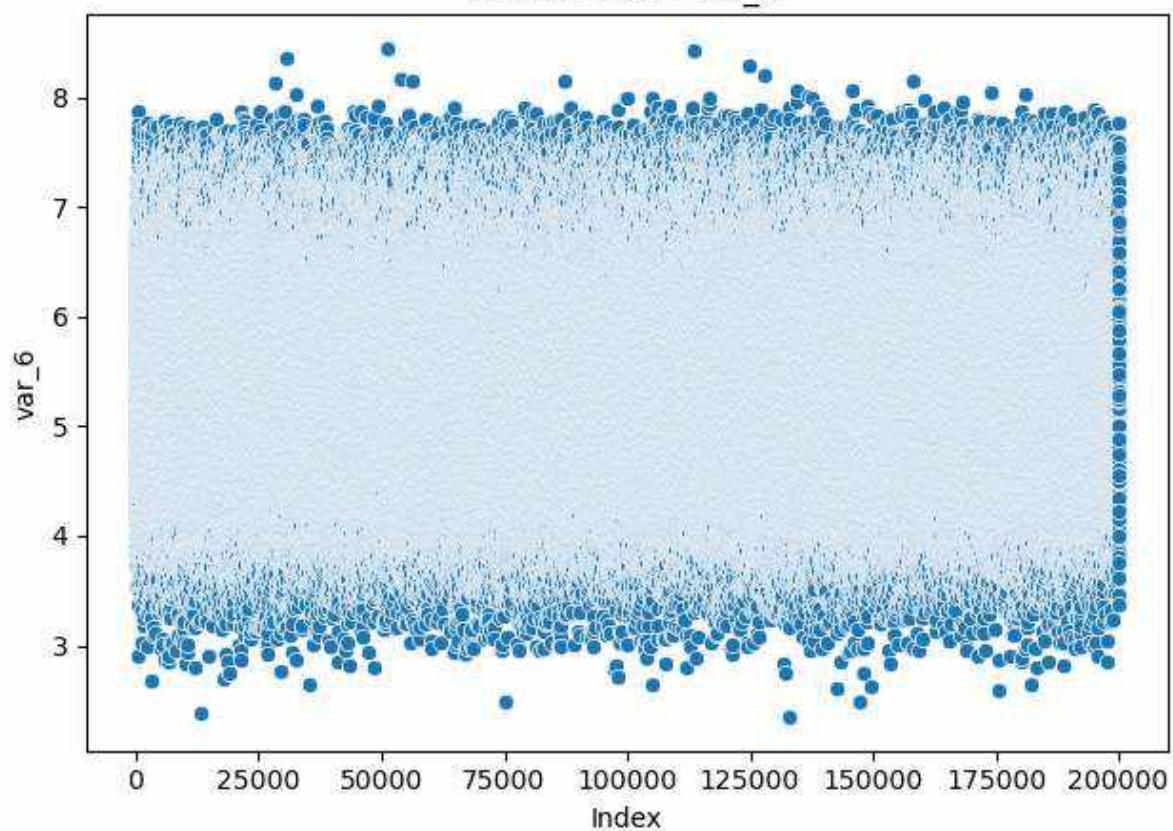
Scatter Plot - var\_4



Scatter Plot - var\_5



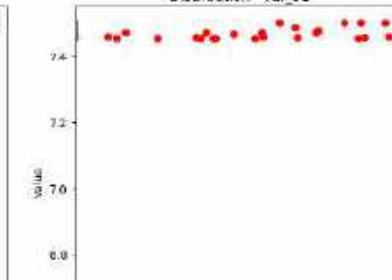
Scatter Plot - var\_6



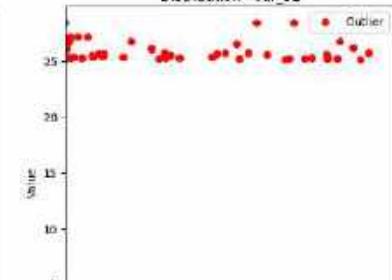
Distribution - var\_90



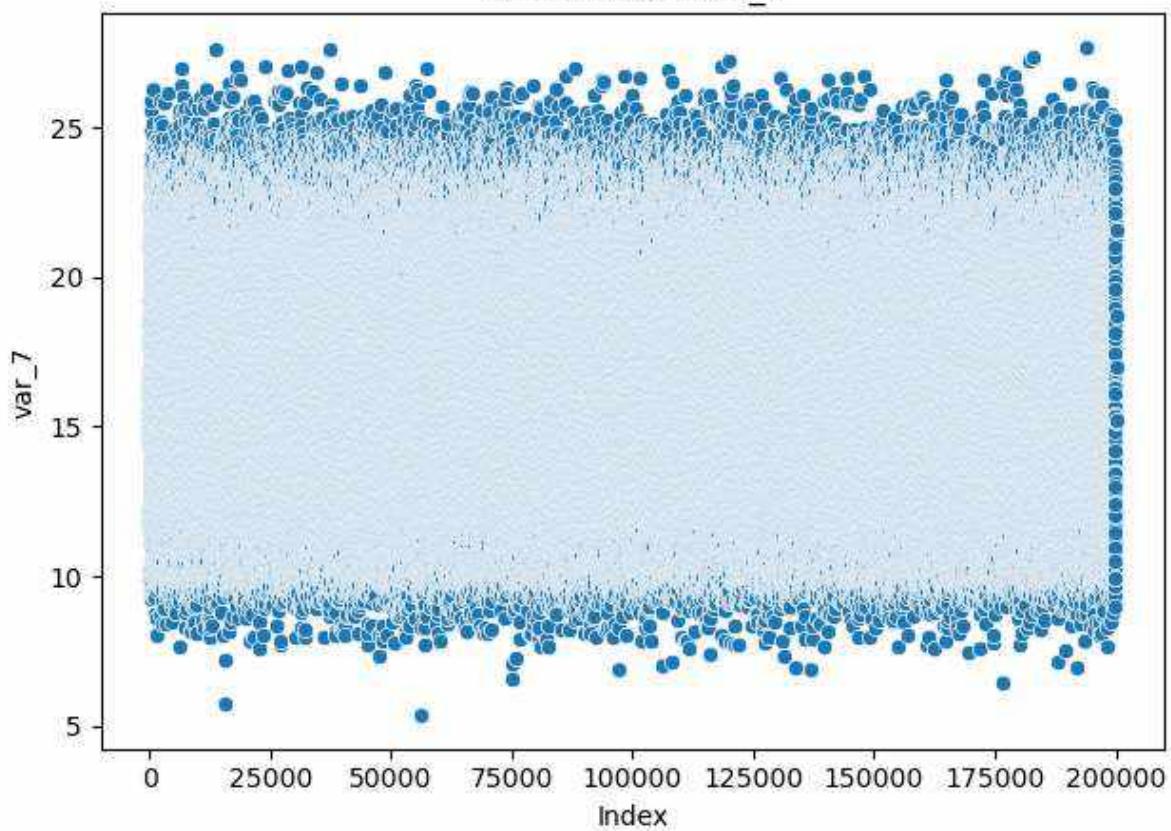
Distribution - var\_91



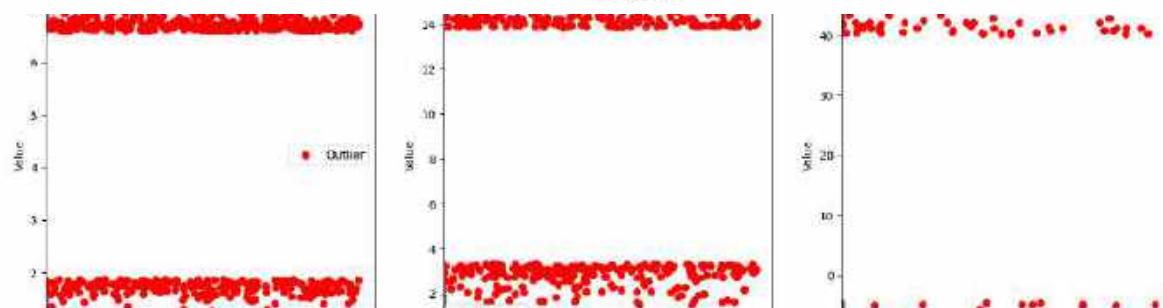
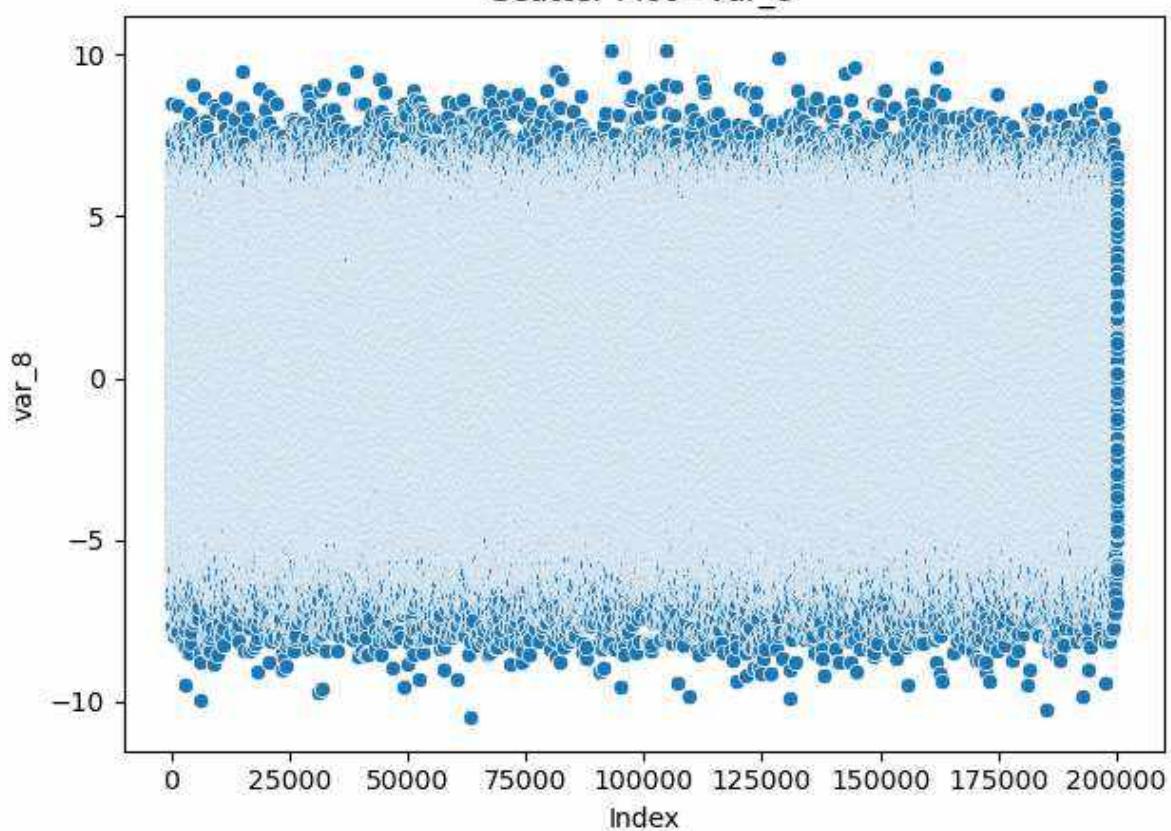
Distribution - var\_92

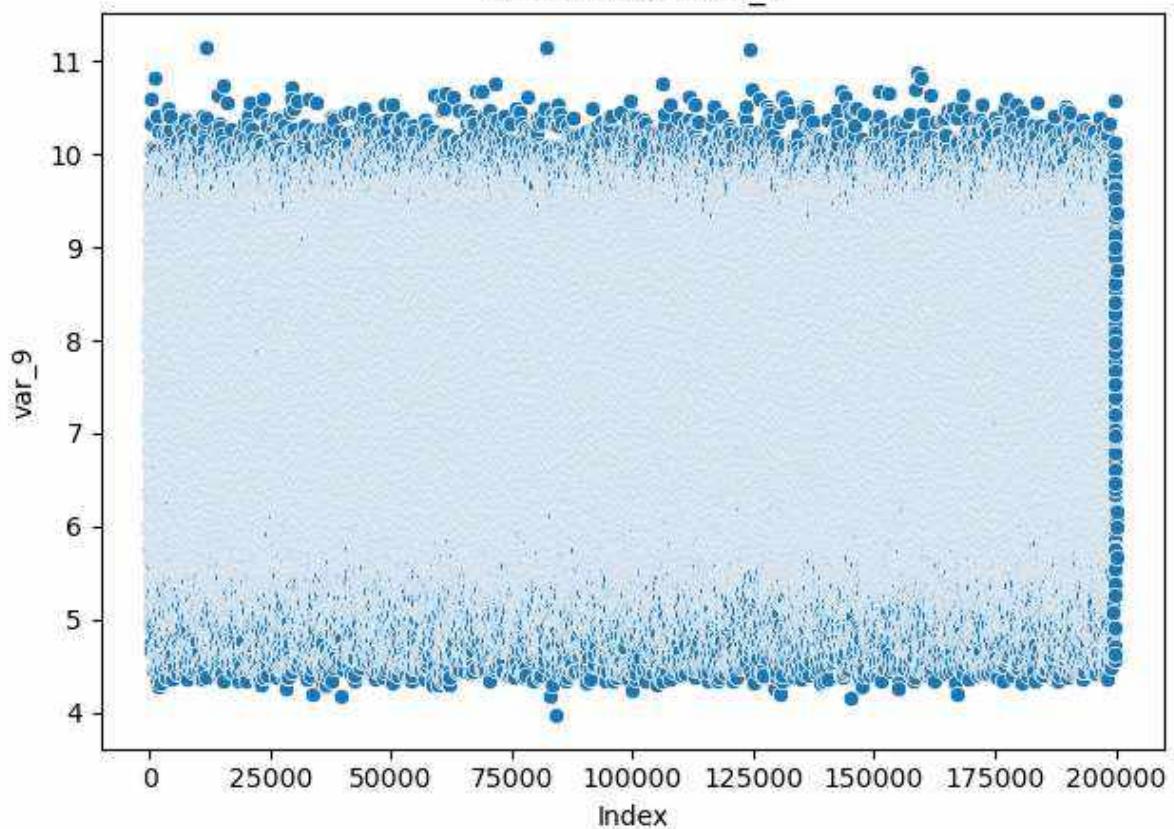
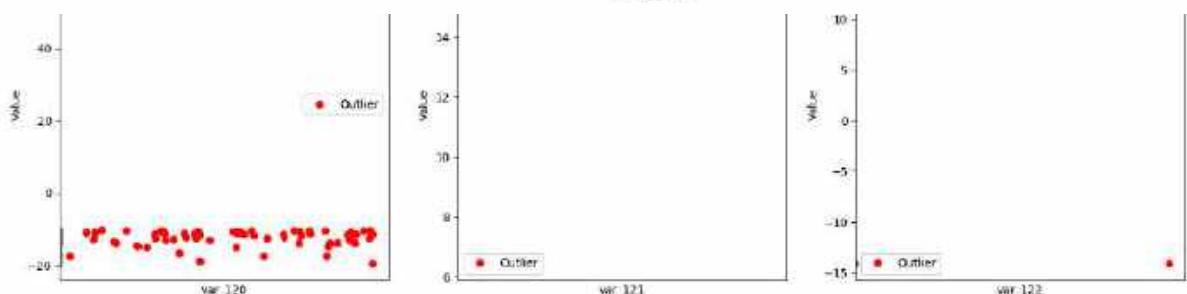
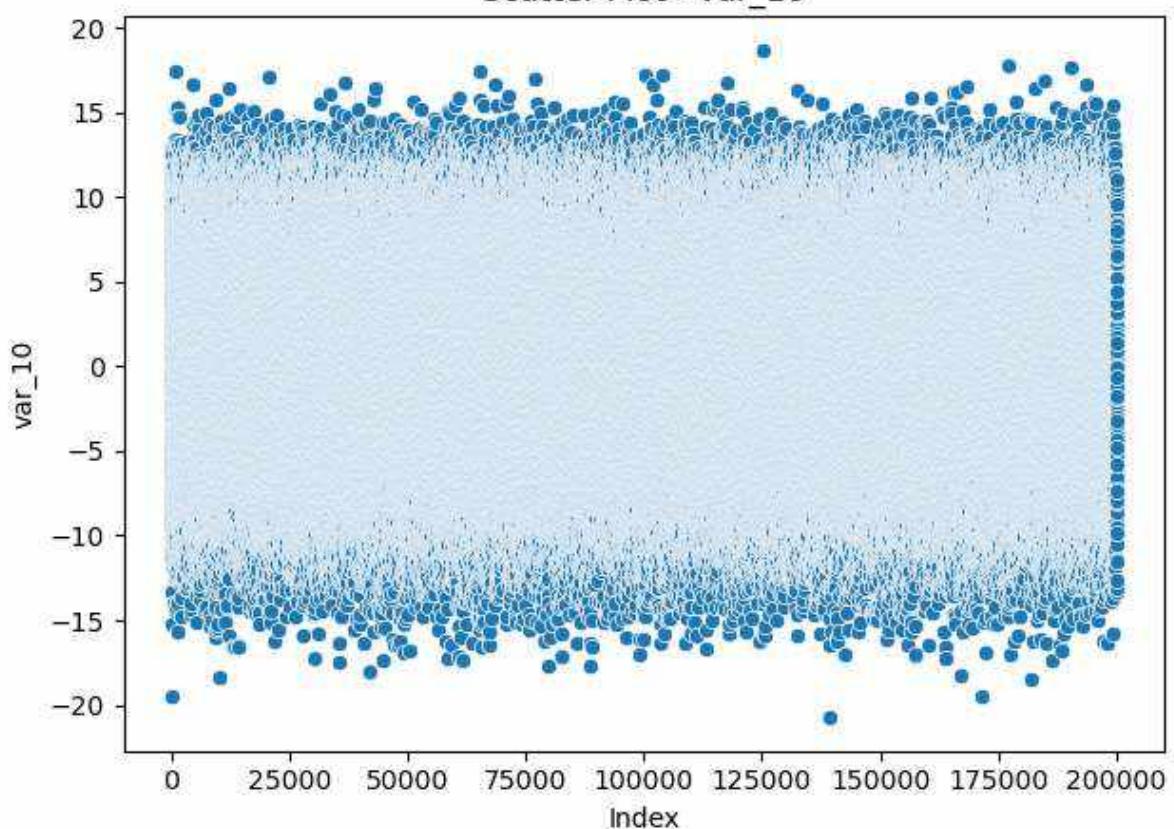


Scatter Plot - var\_7

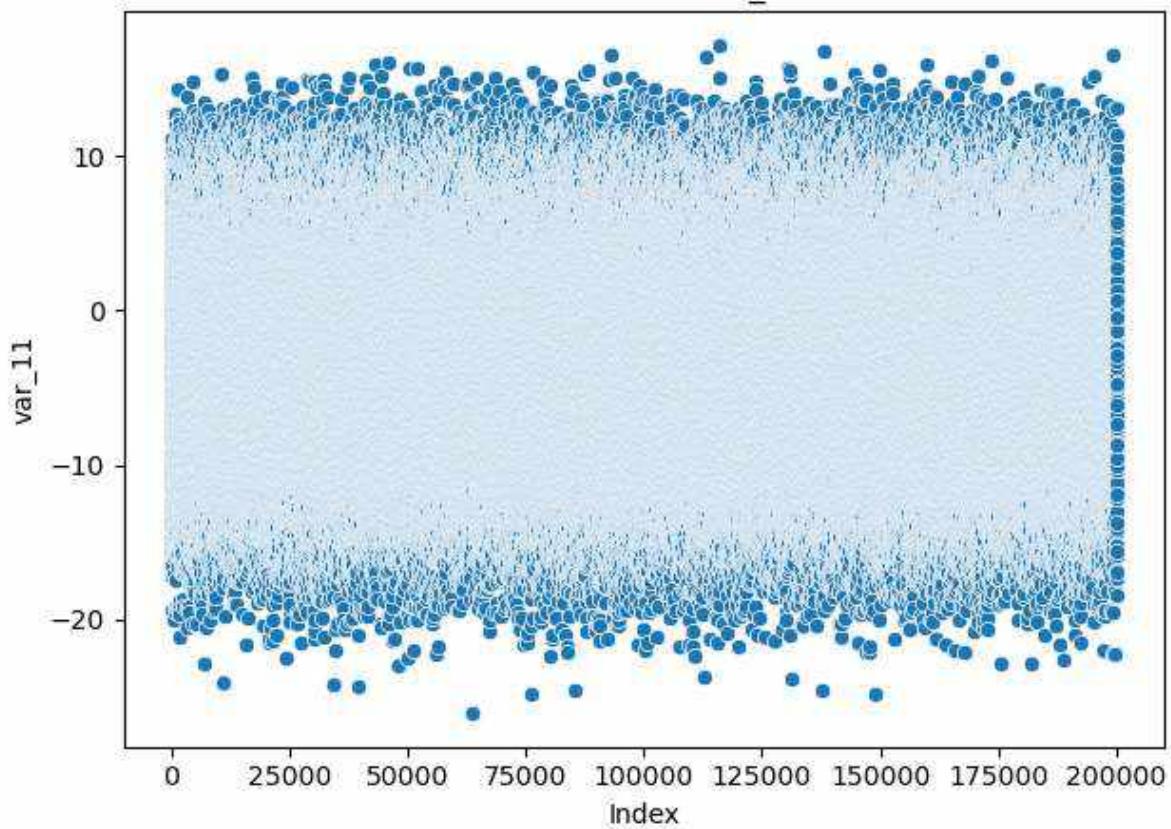


Scatter Plot - var\_8

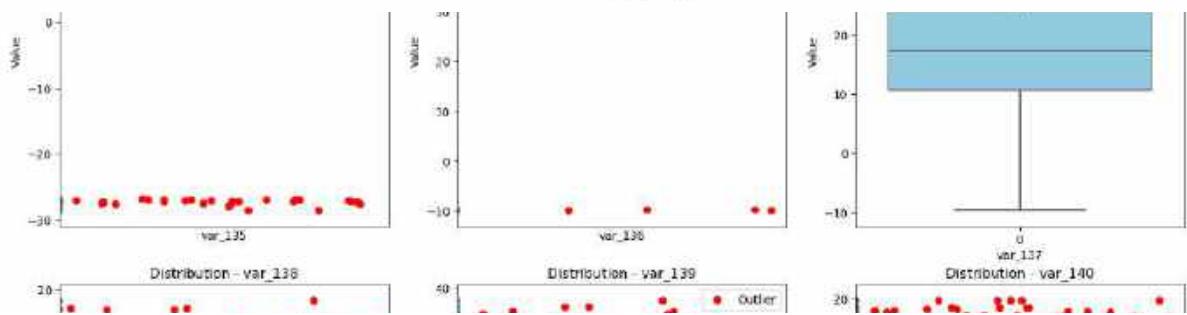
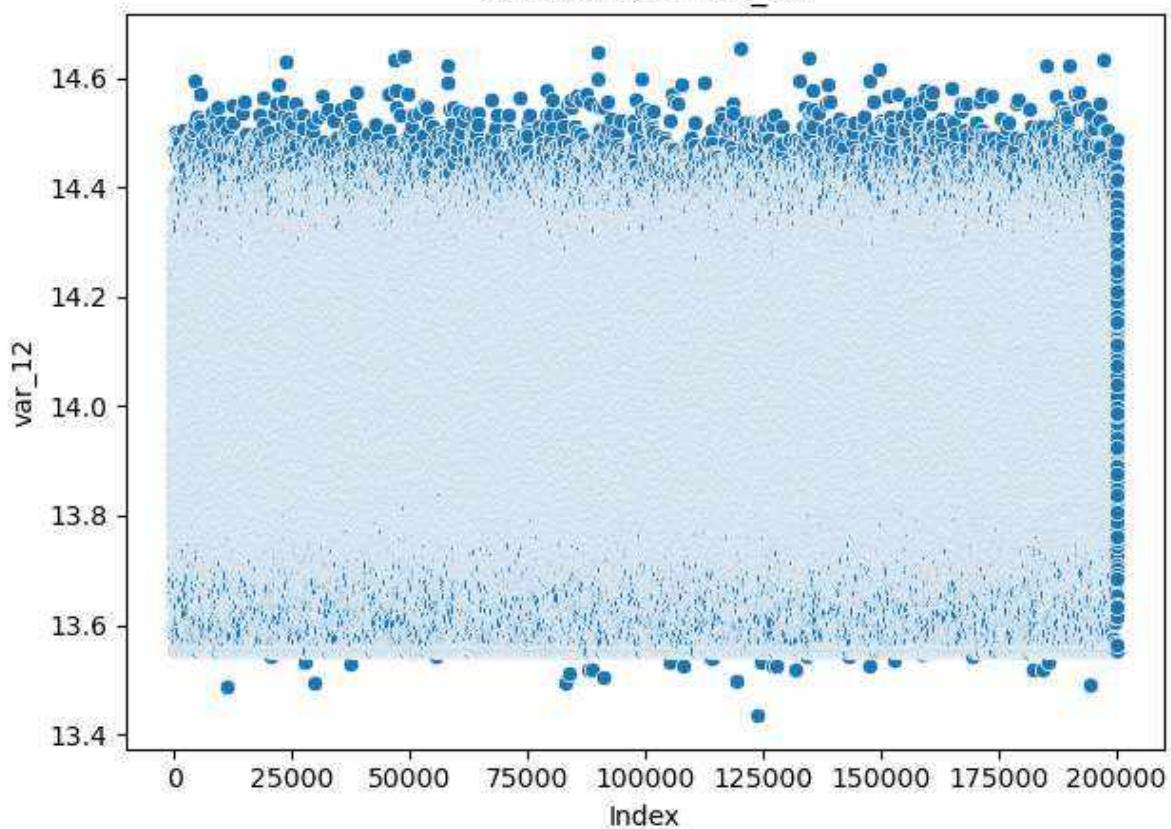


**Scatter Plot - var\_9****Scatter Plot - var\_10**

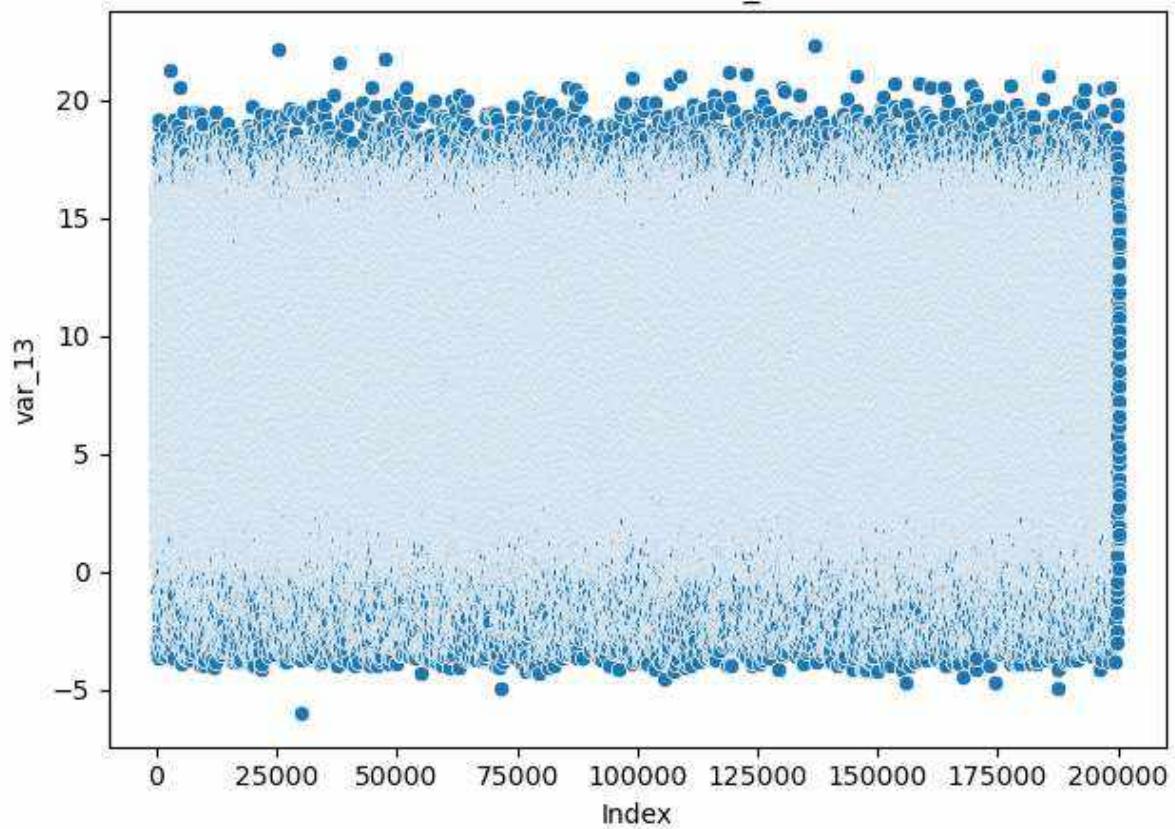
## Scatter Plot - var\_11



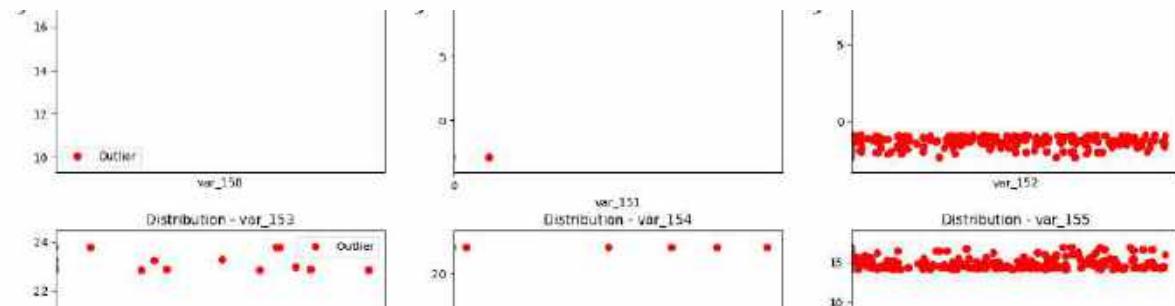
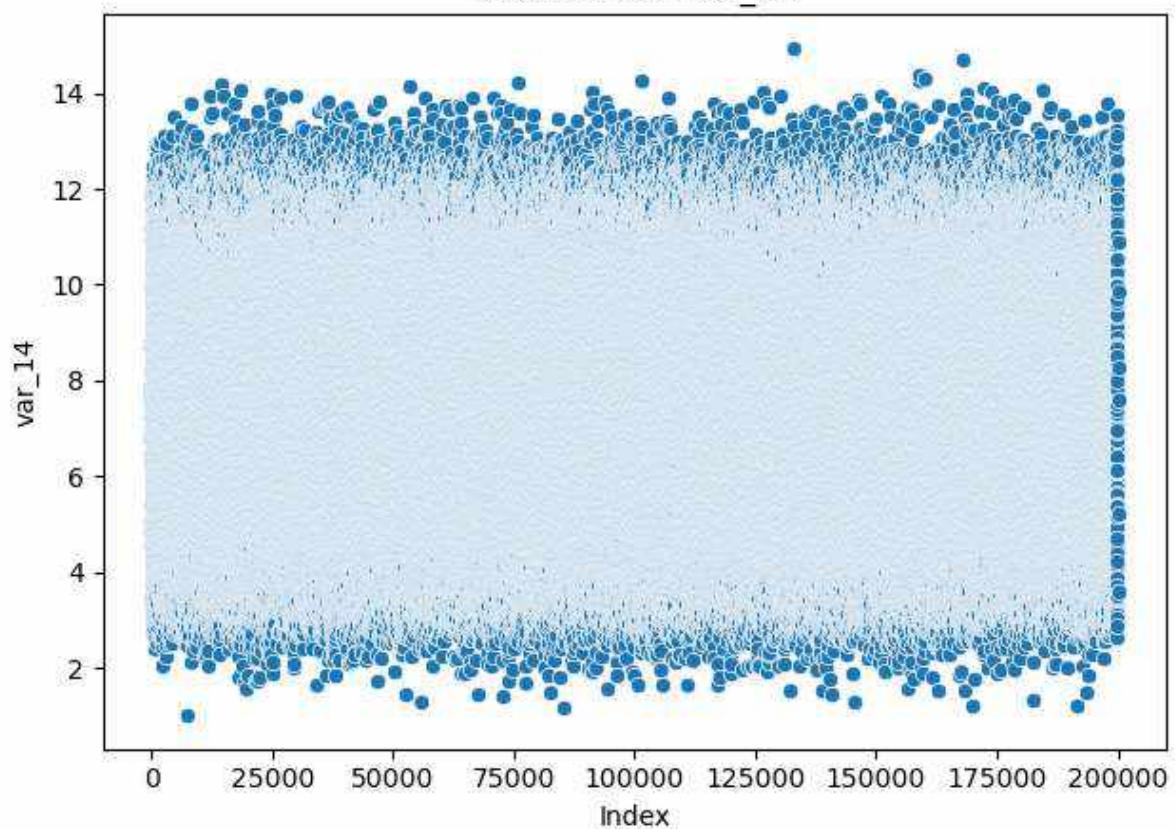
## Scatter Plot - var\_12

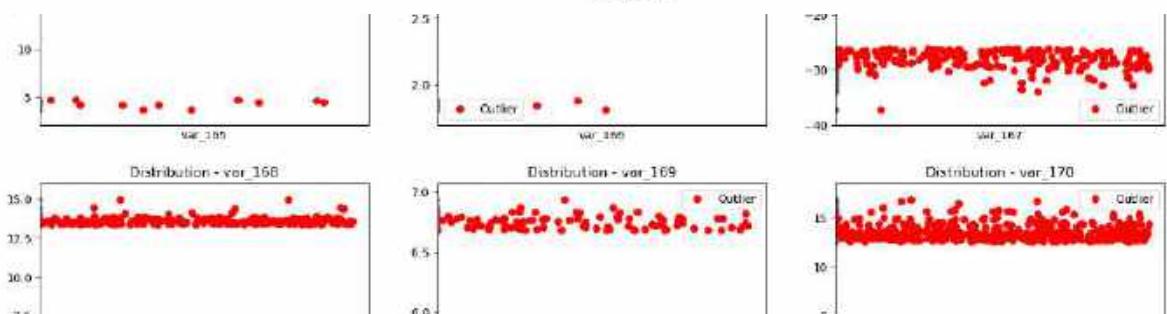
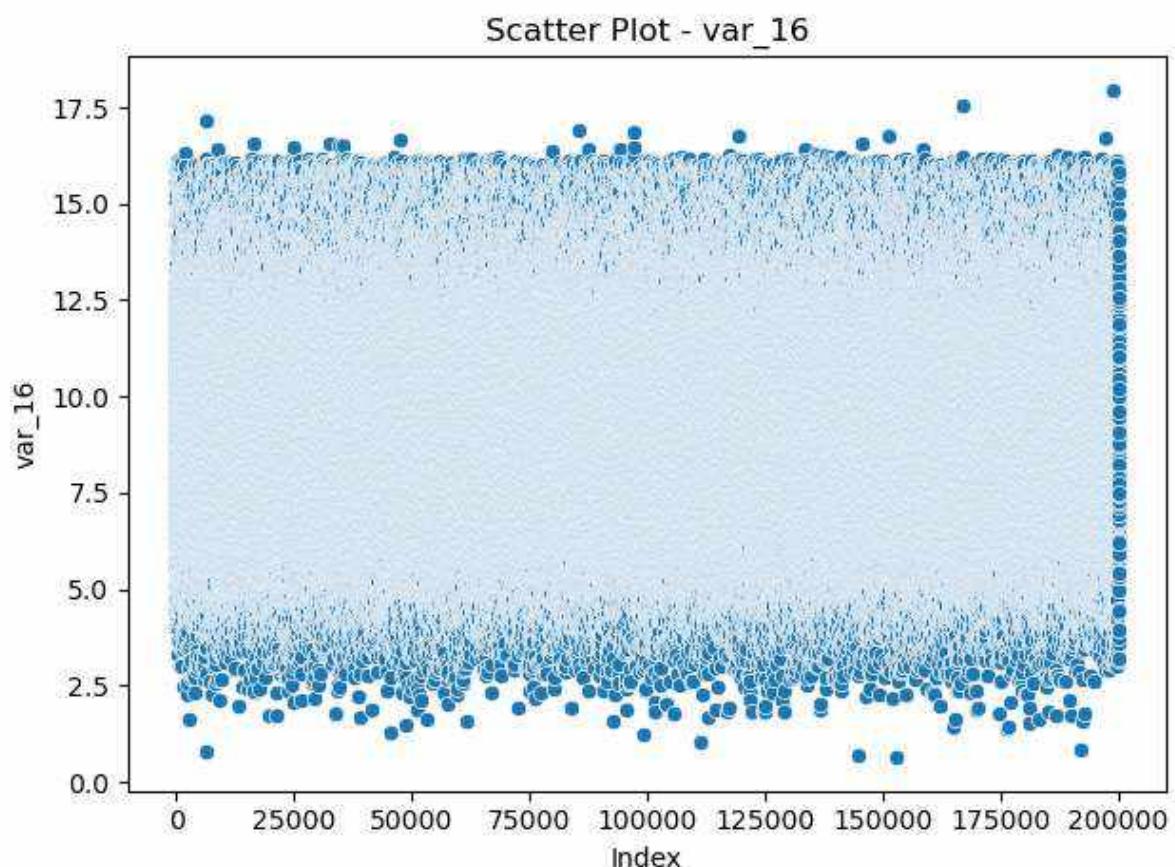
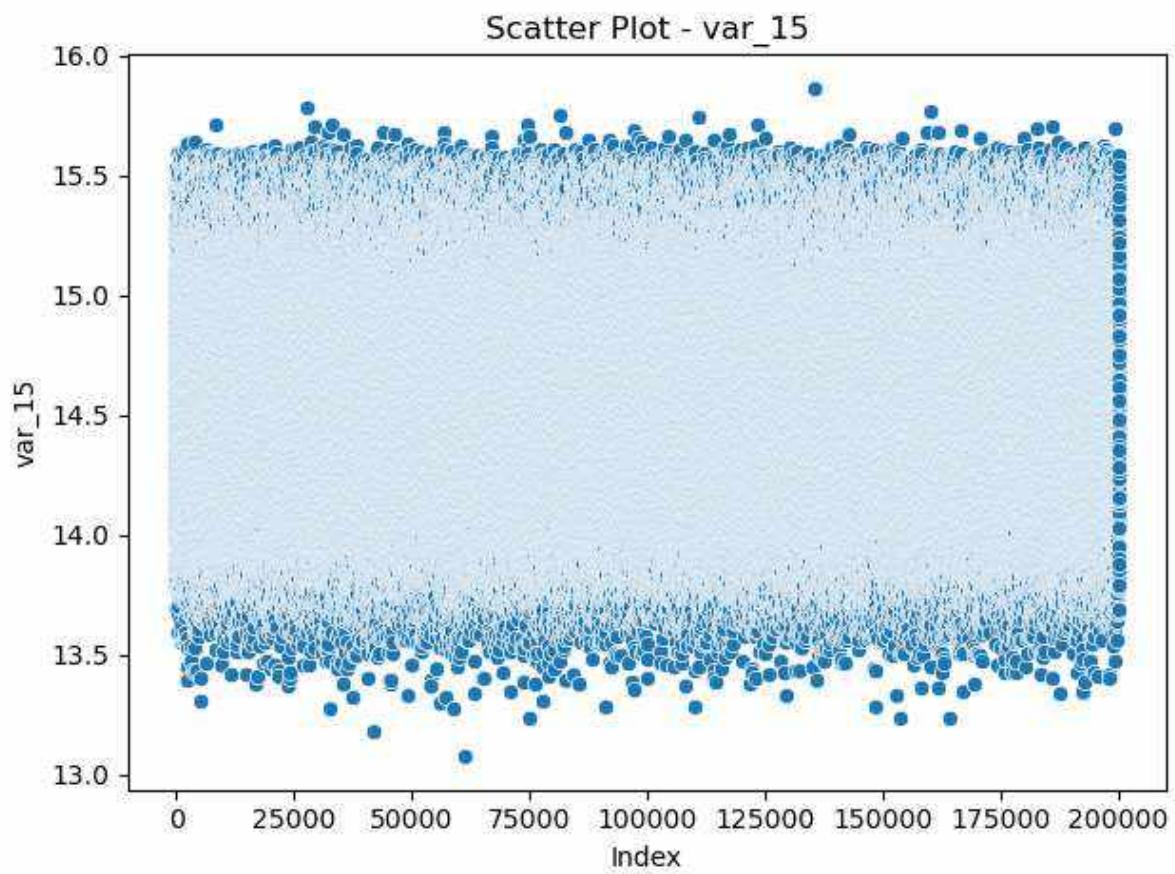


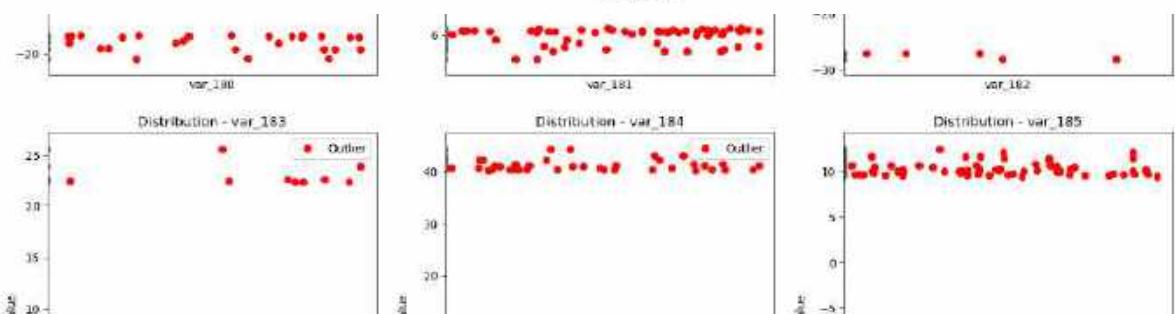
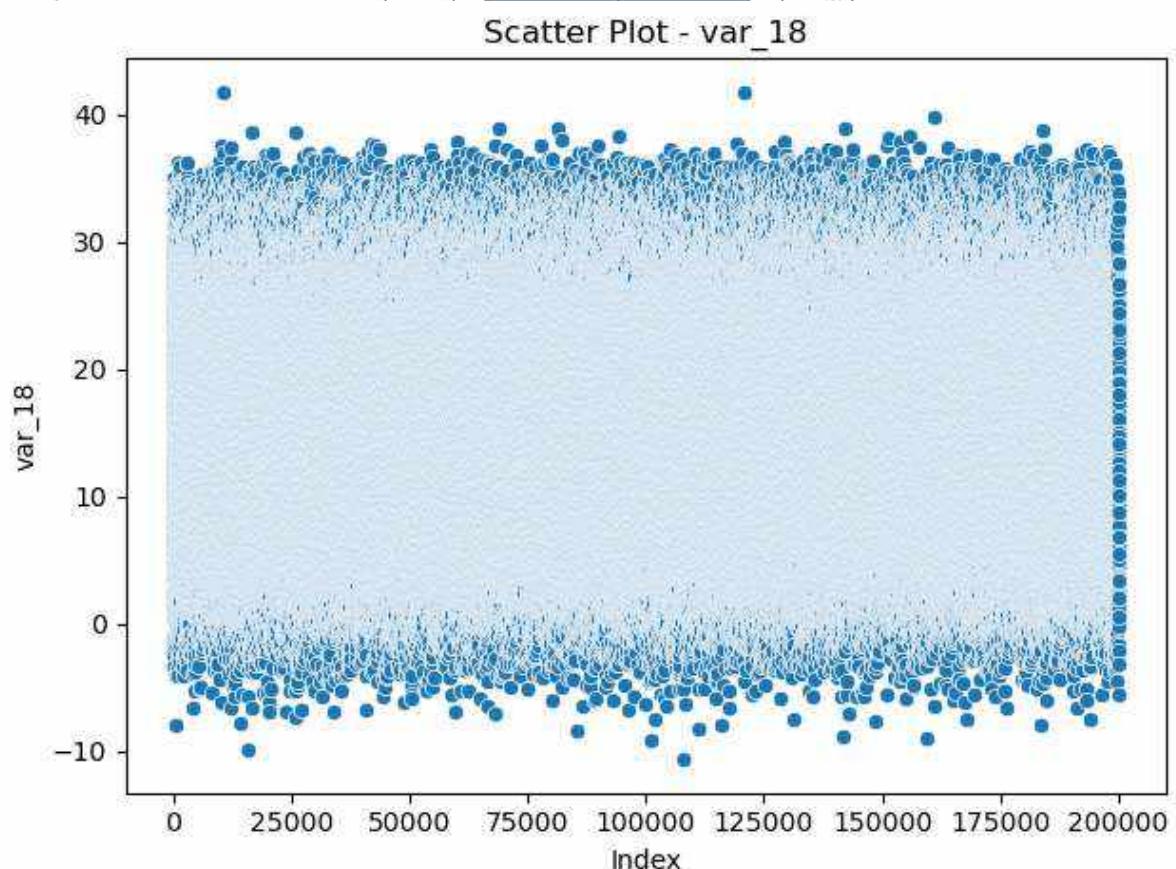
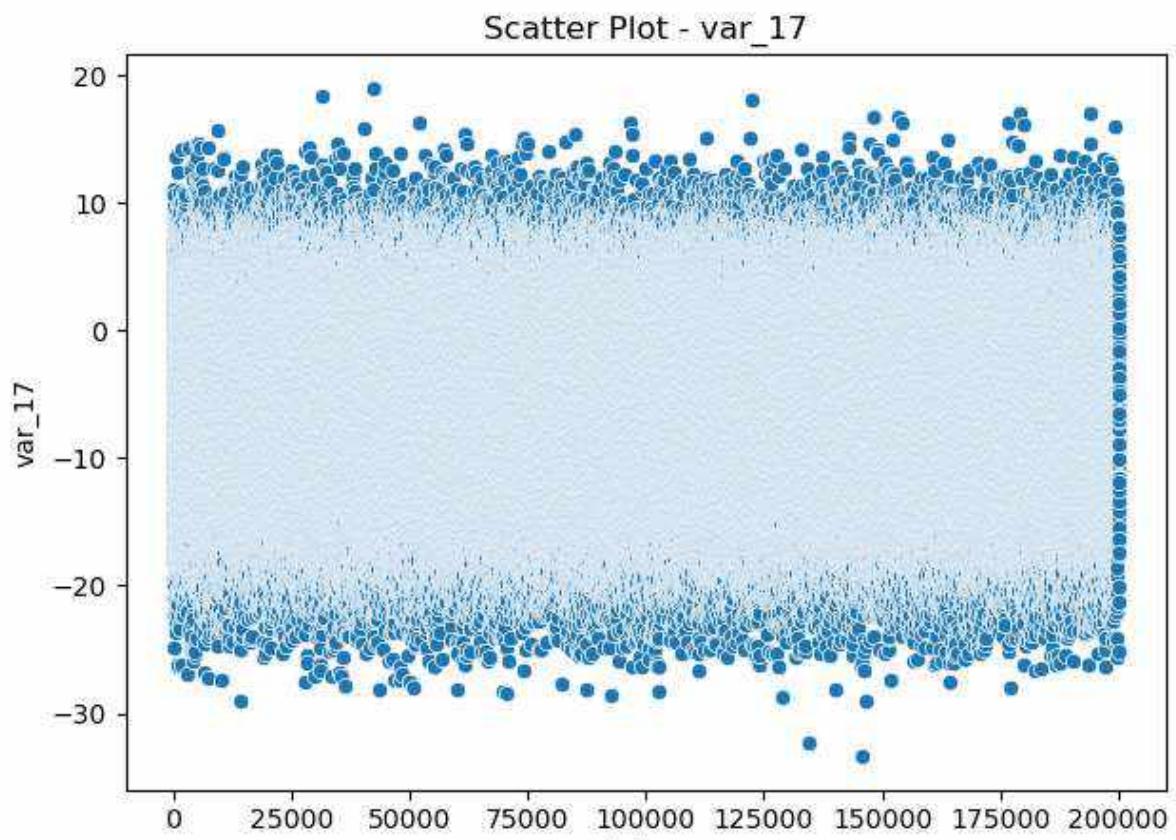
Scatter Plot - var\_13

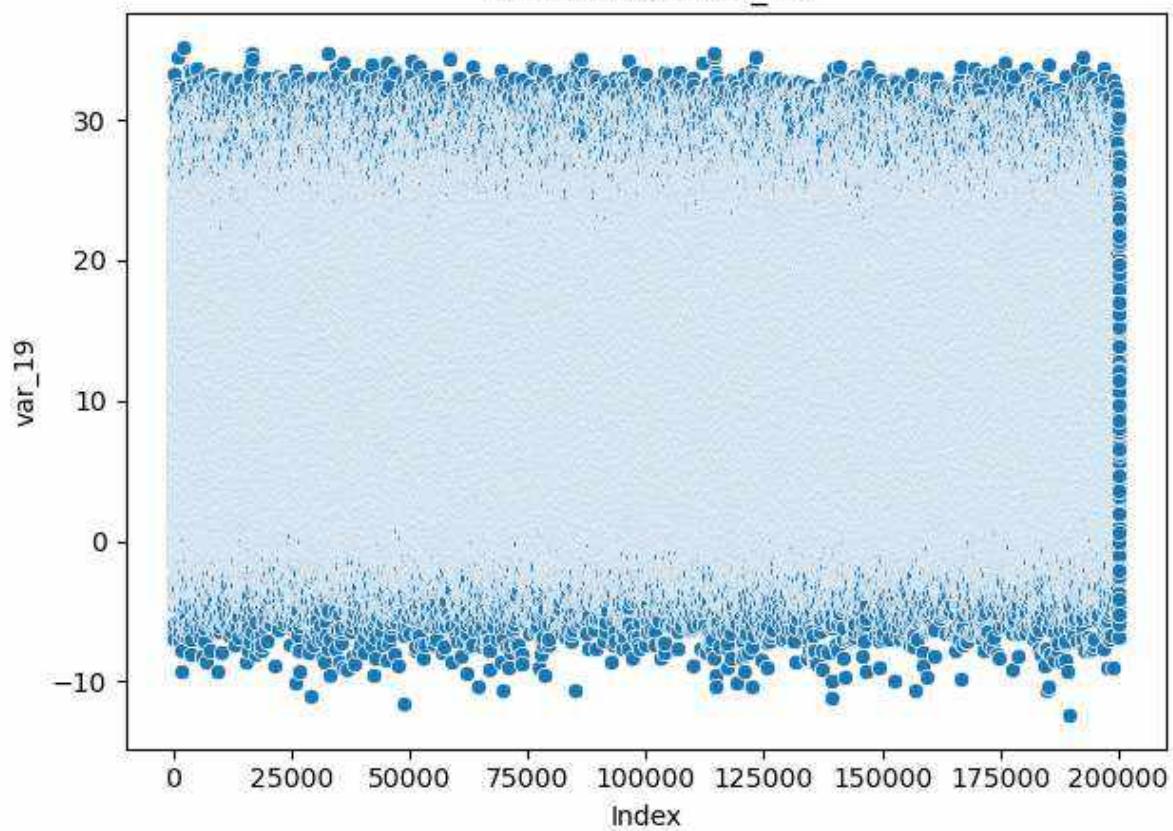
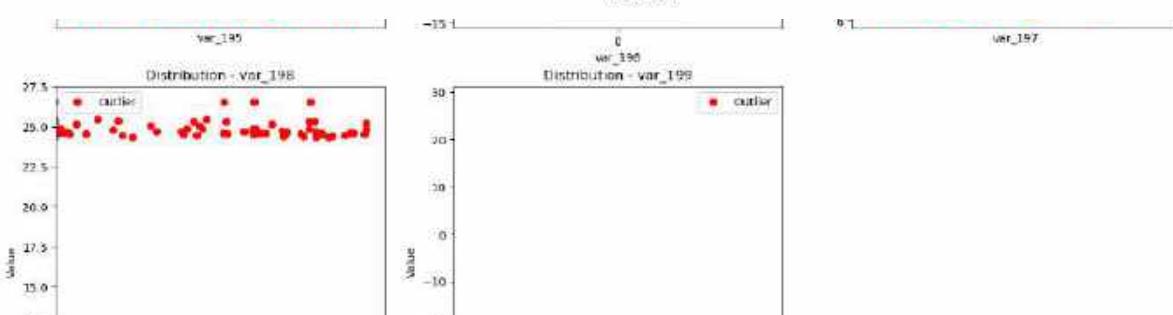
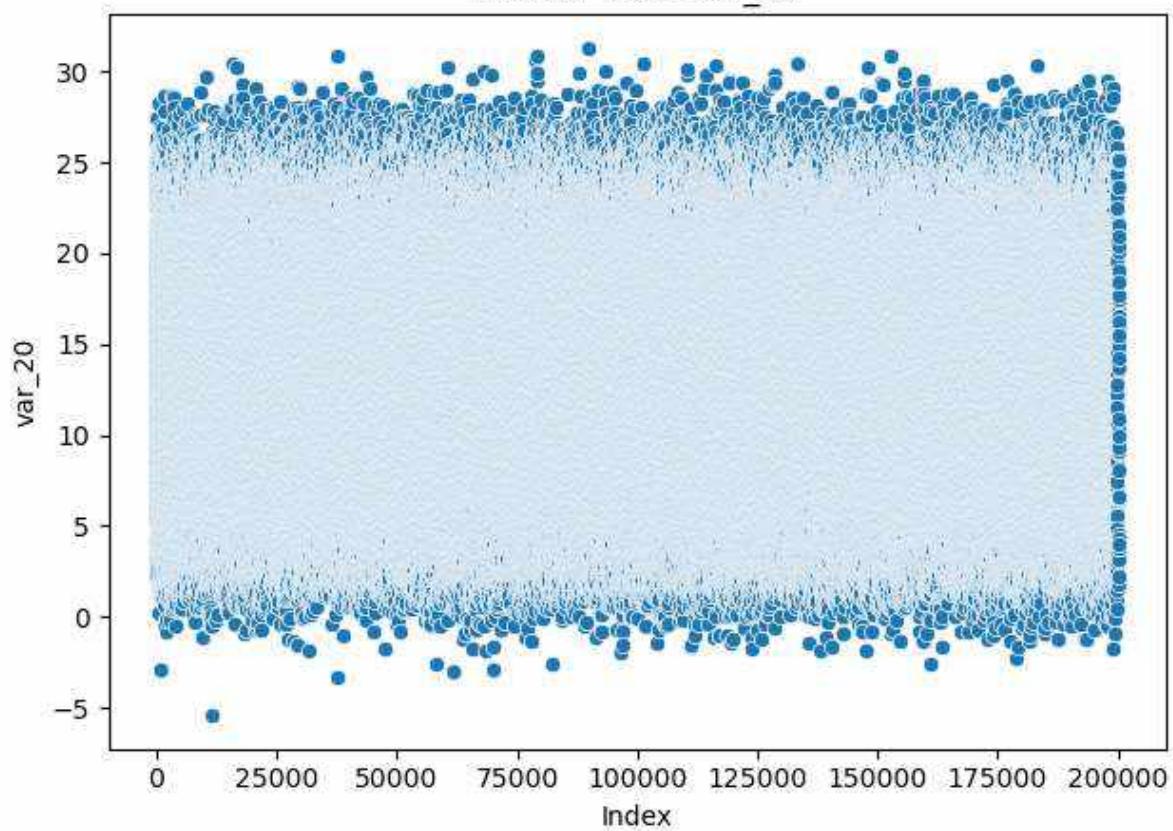


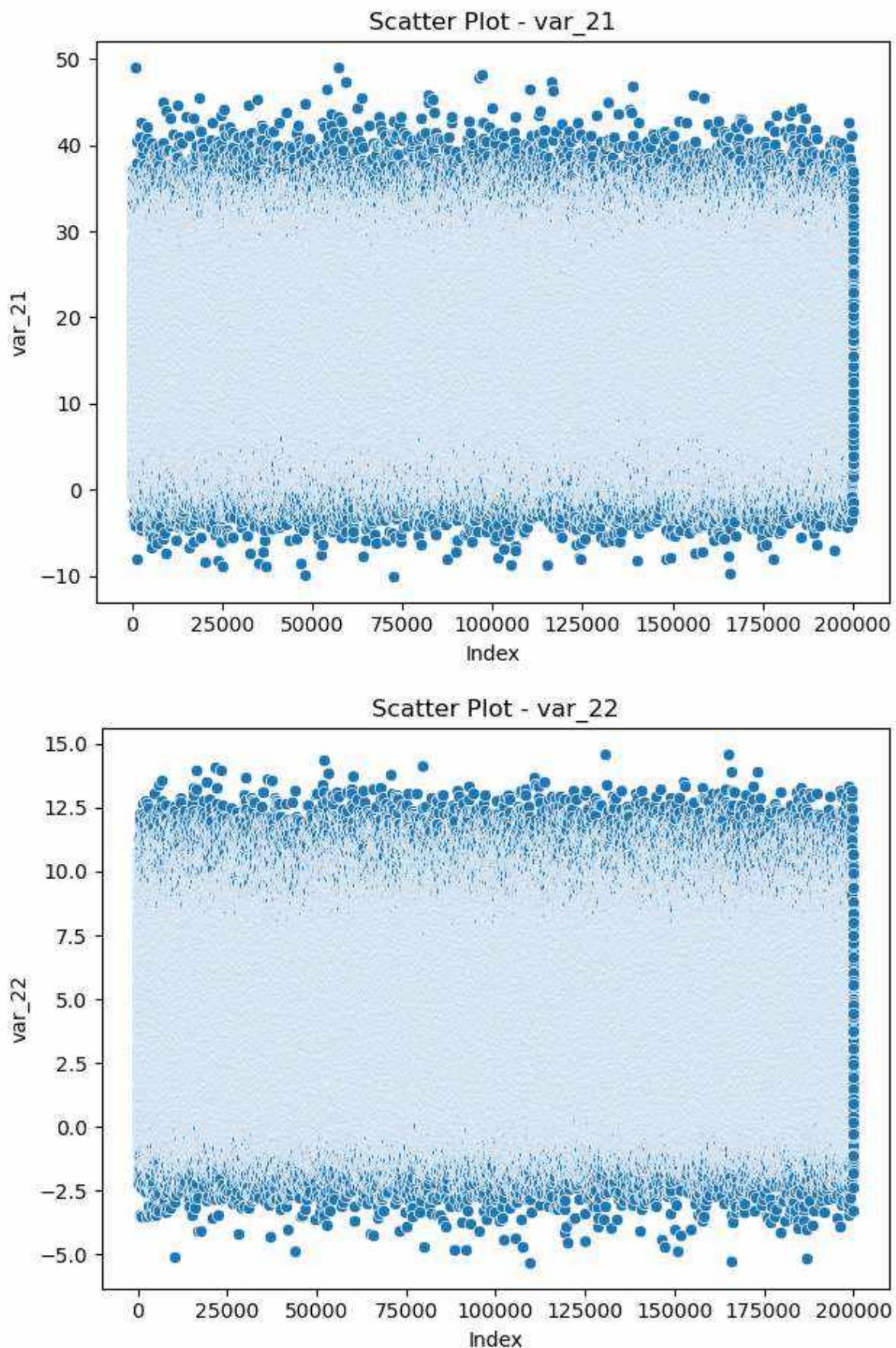
Scatter Plot - var\_14

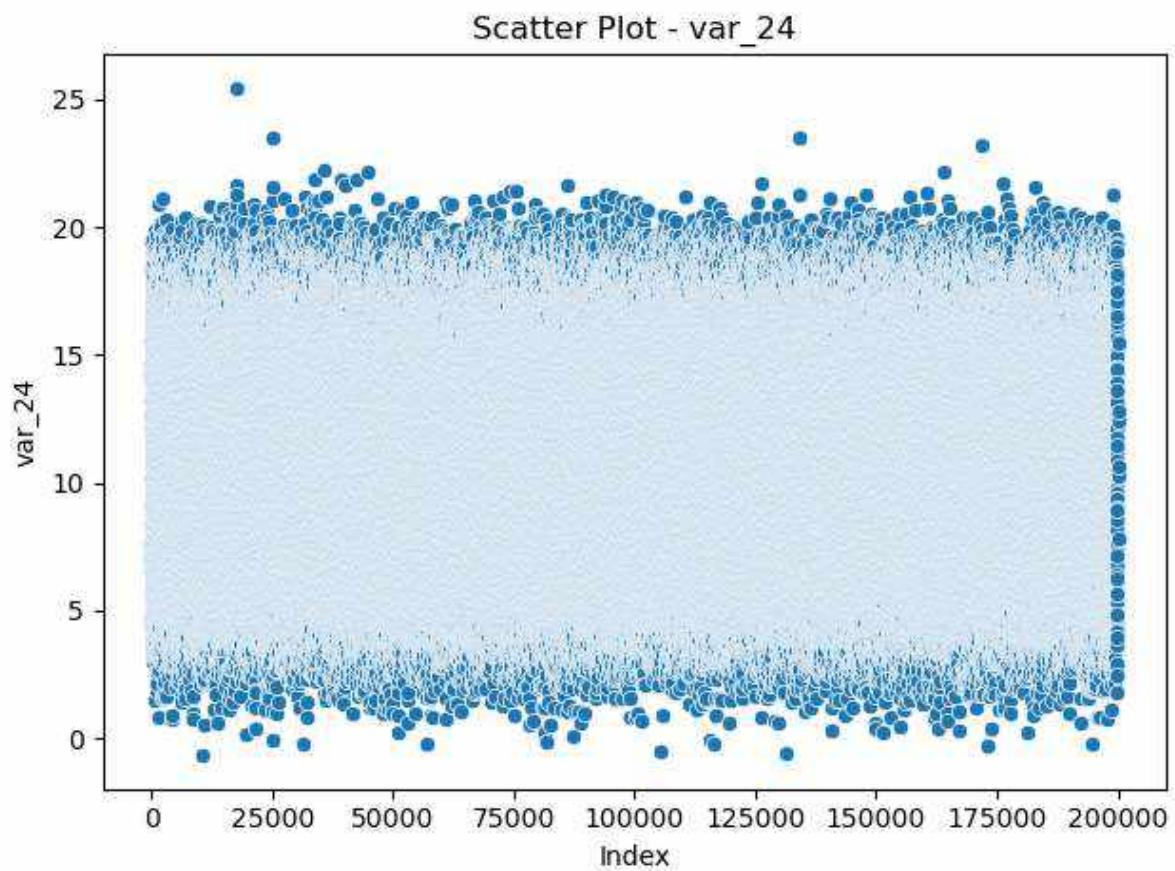
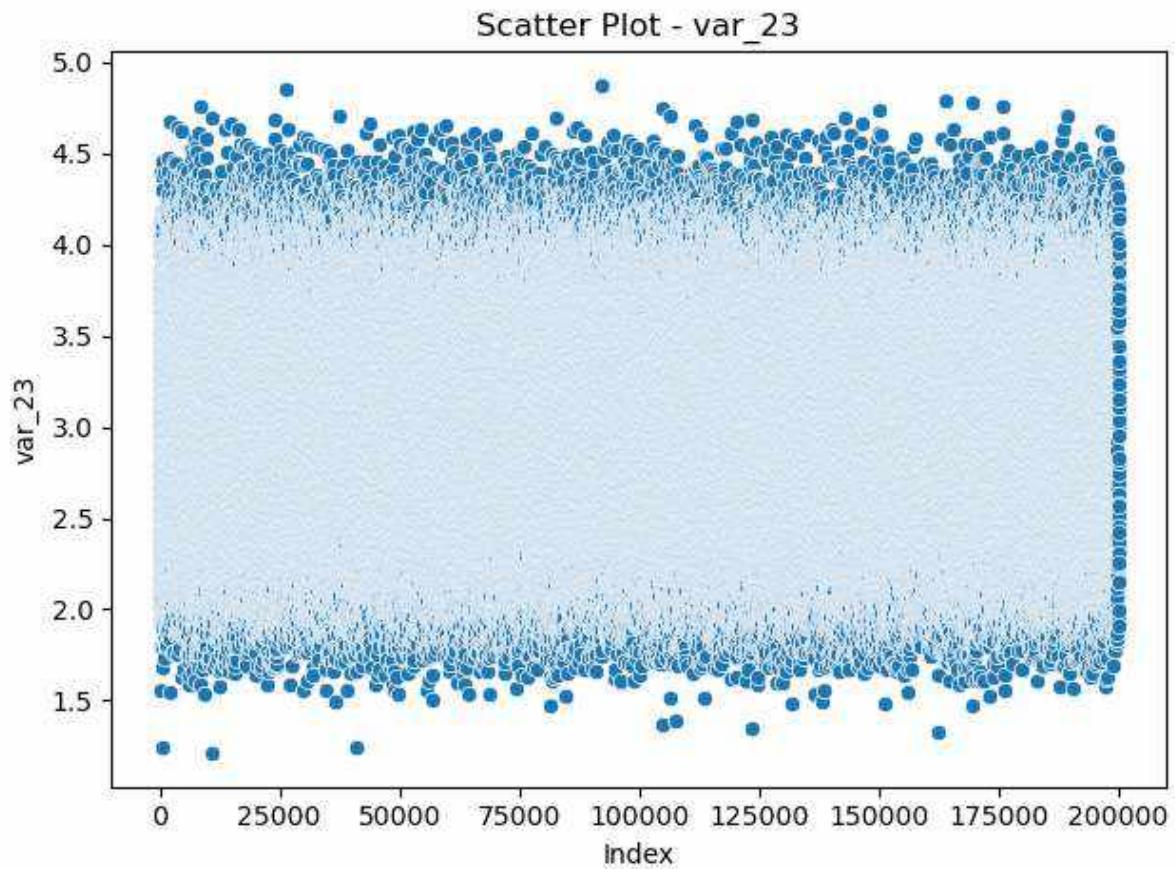


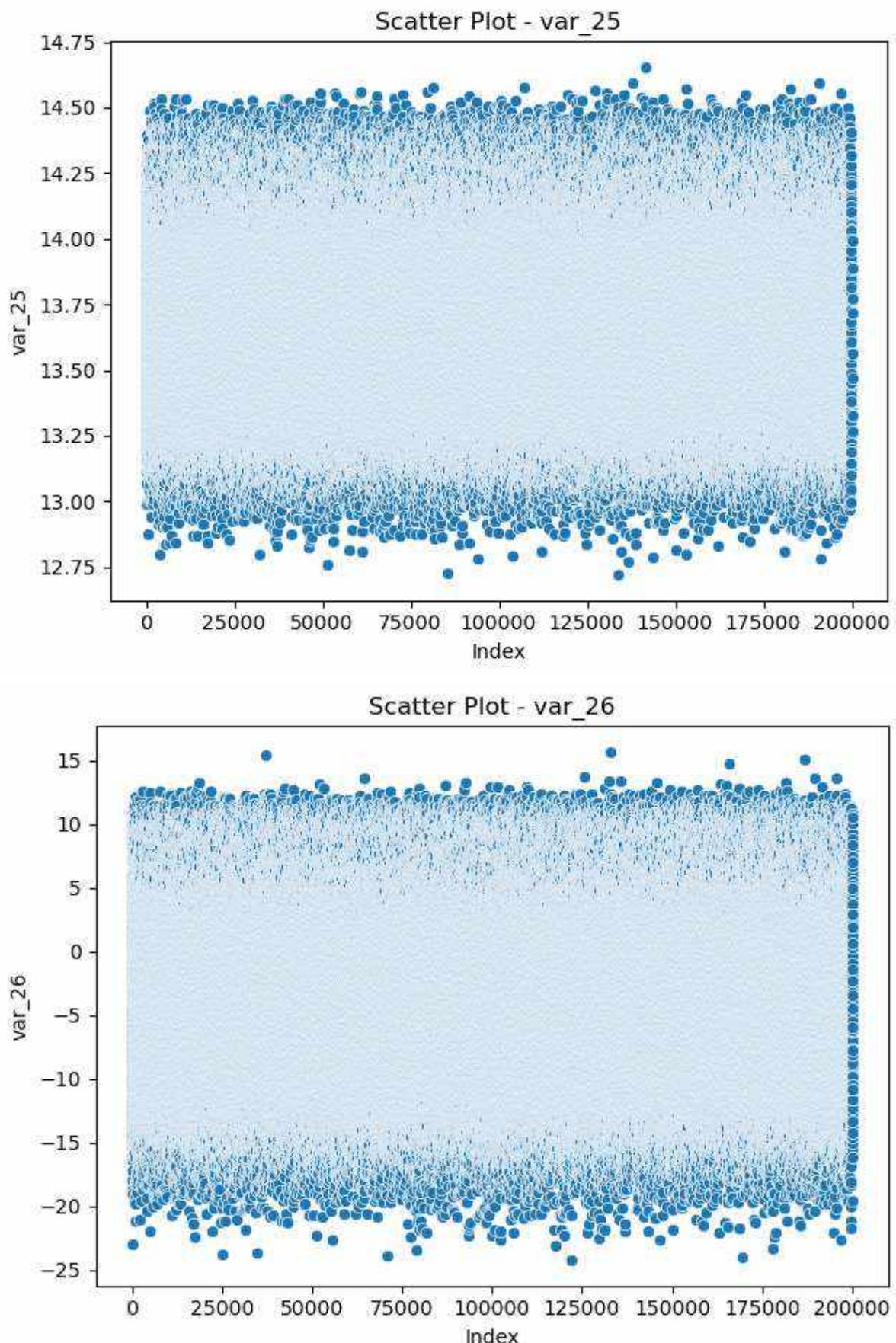




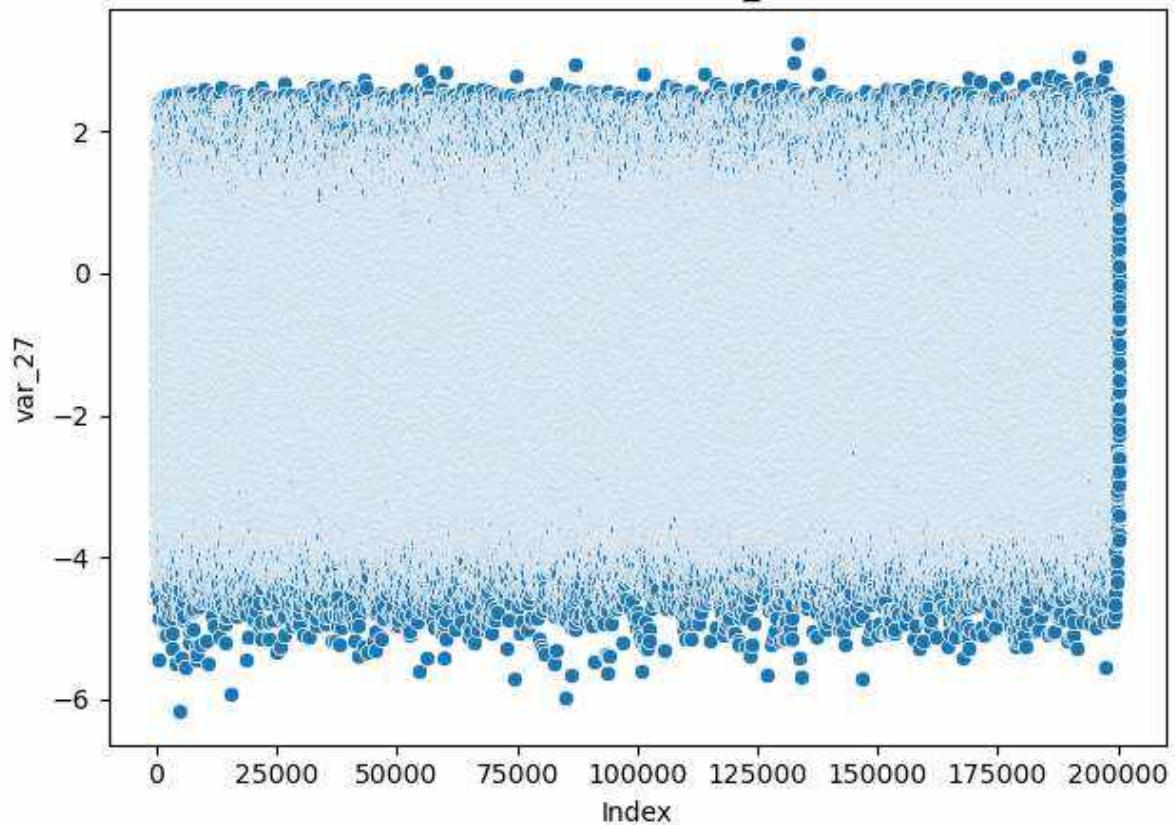
**Scatter Plot - var\_19****Scatter Plot - var\_20**



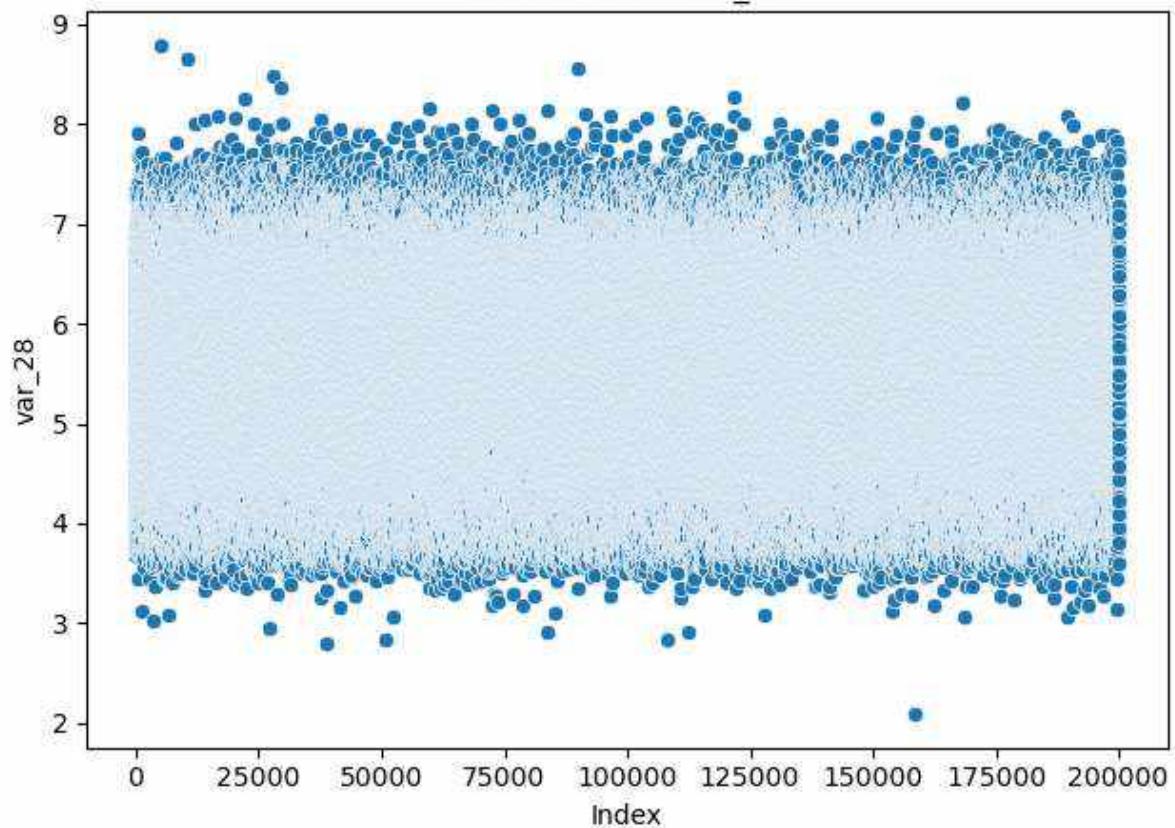


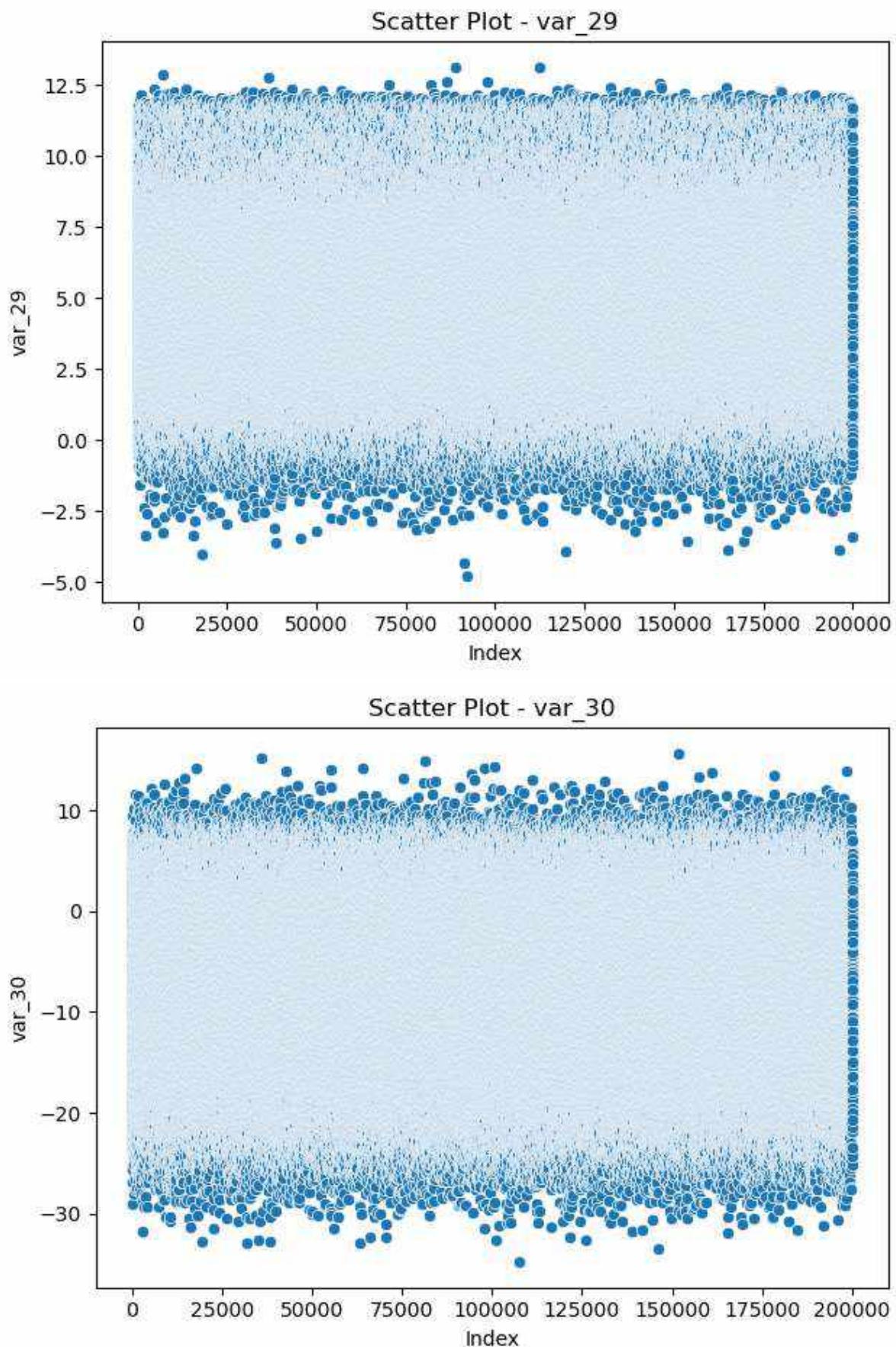


Scatter Plot - var\_27

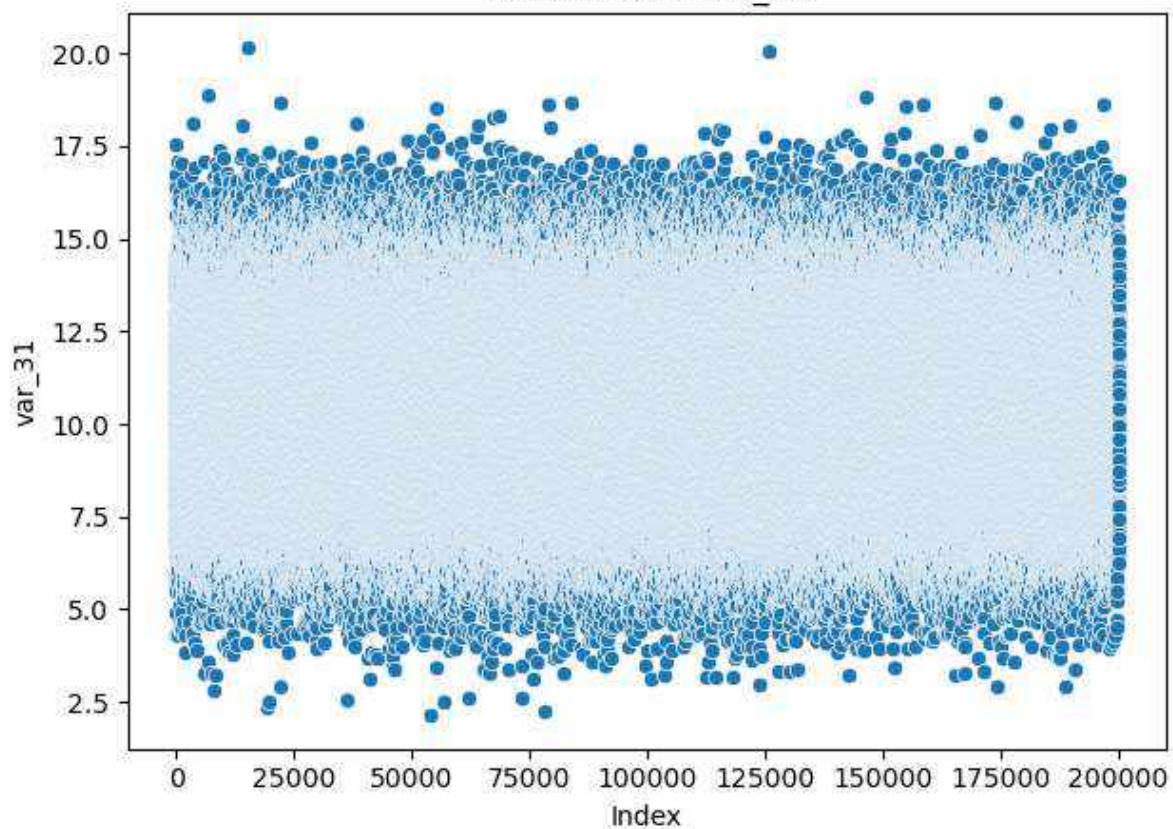


Scatter Plot - var\_28

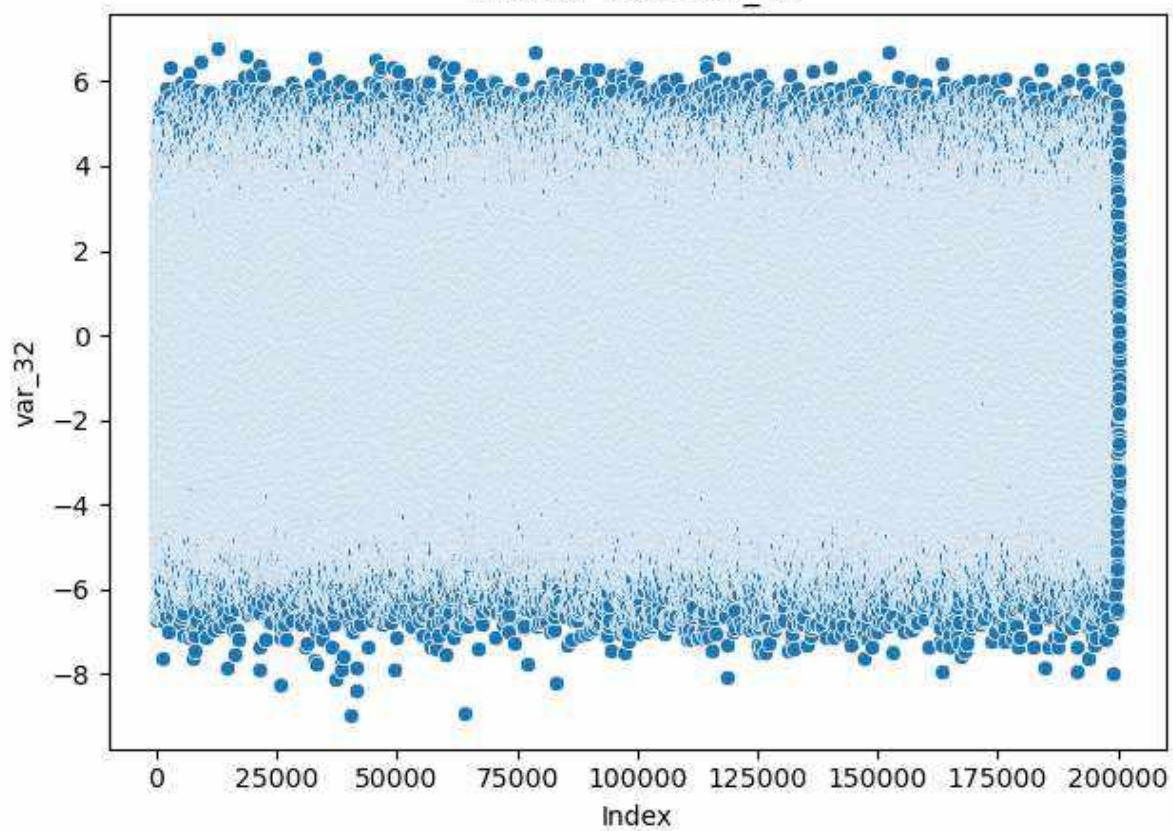




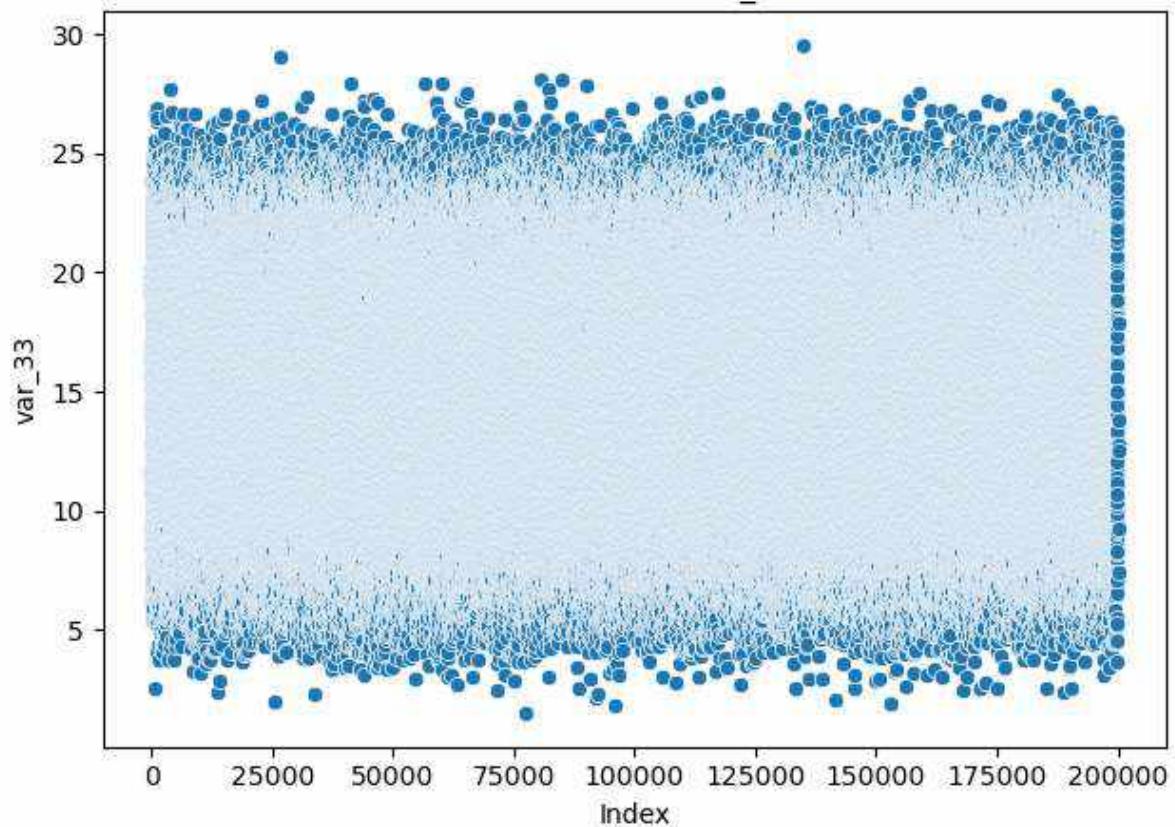
Scatter Plot - var\_31



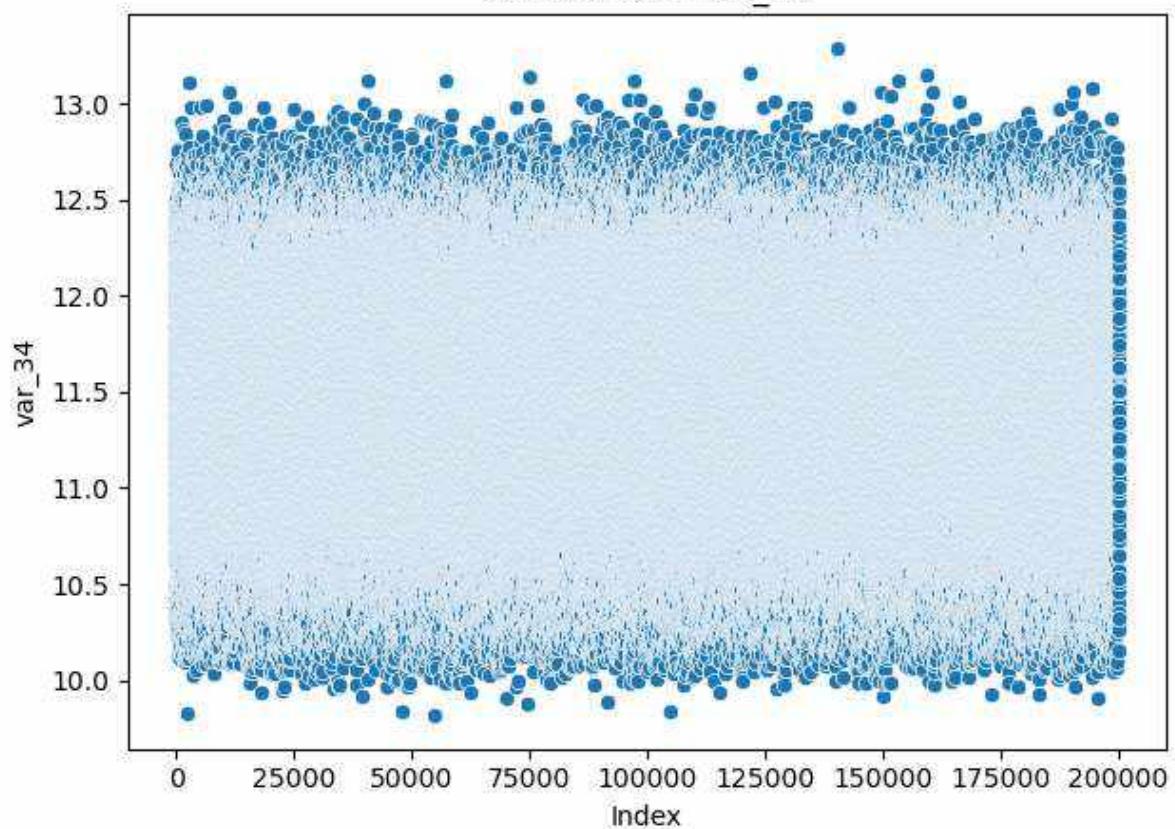
Scatter Plot - var\_32



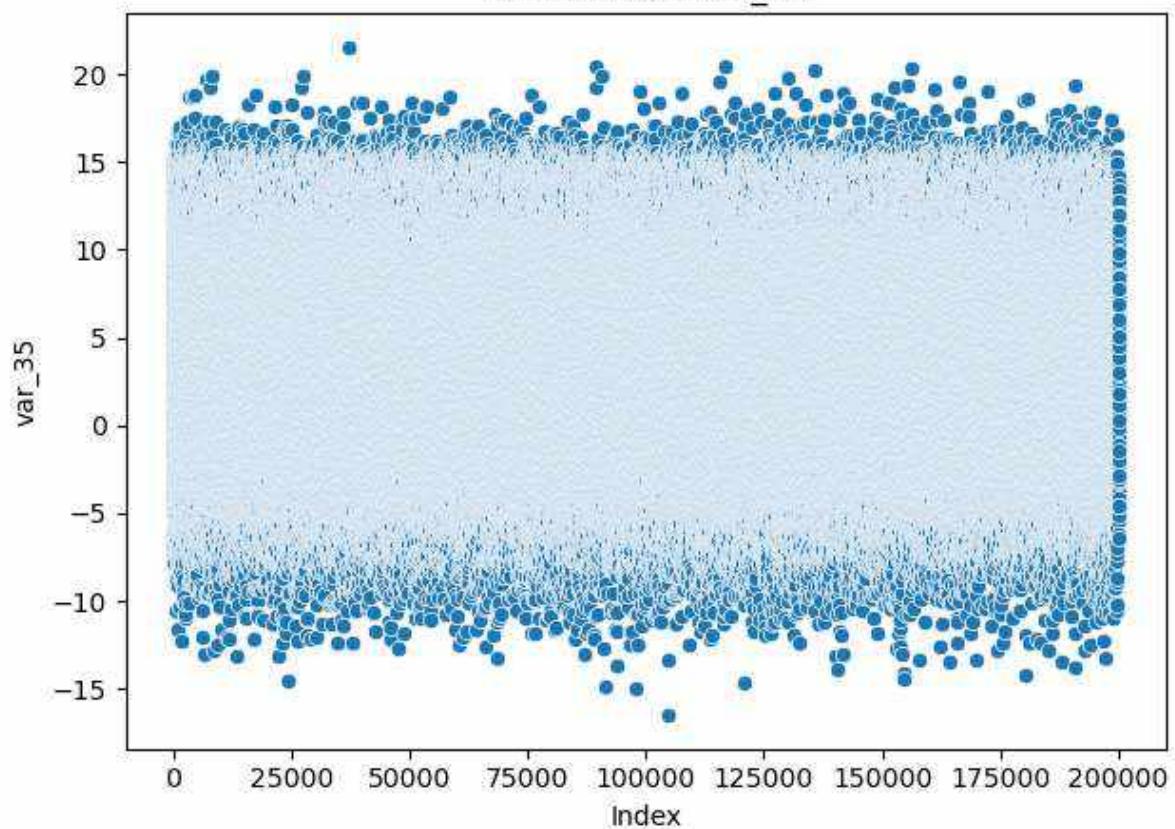
Scatter Plot - var\_33



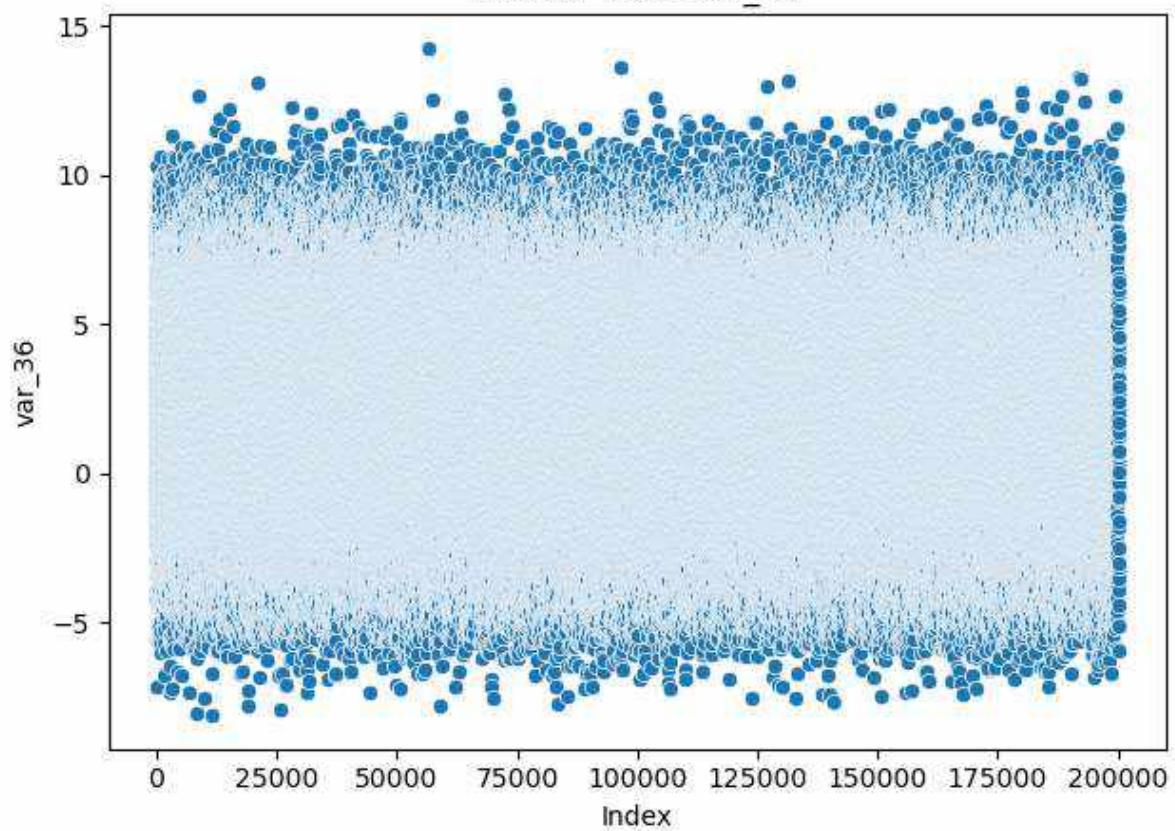
Scatter Plot - var\_34



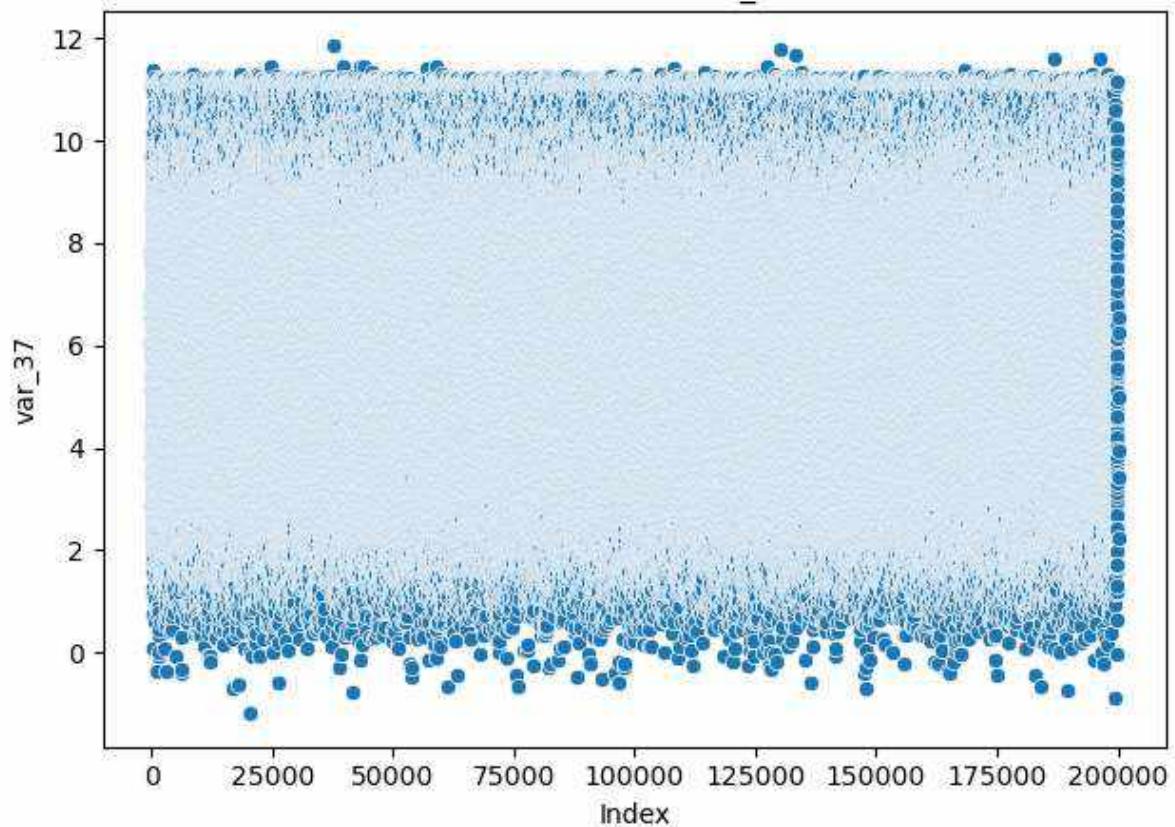
Scatter Plot - var\_35



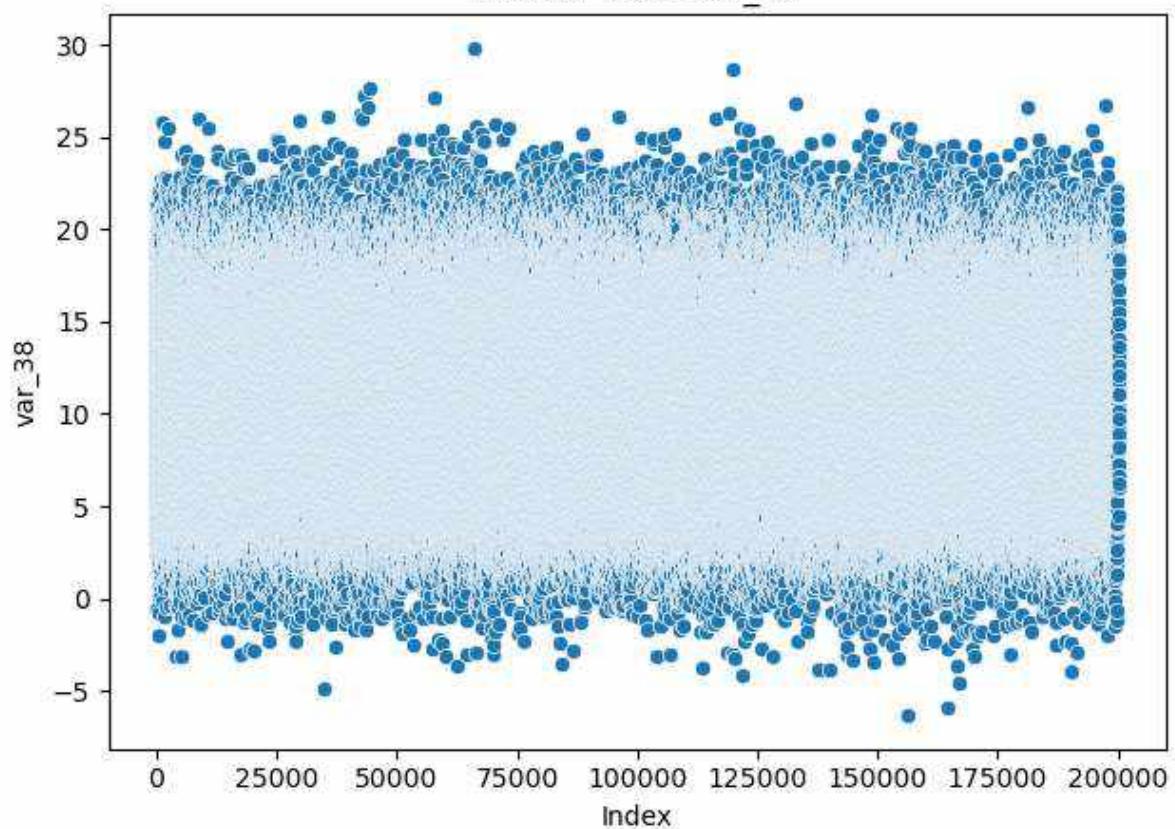
Scatter Plot - var\_36



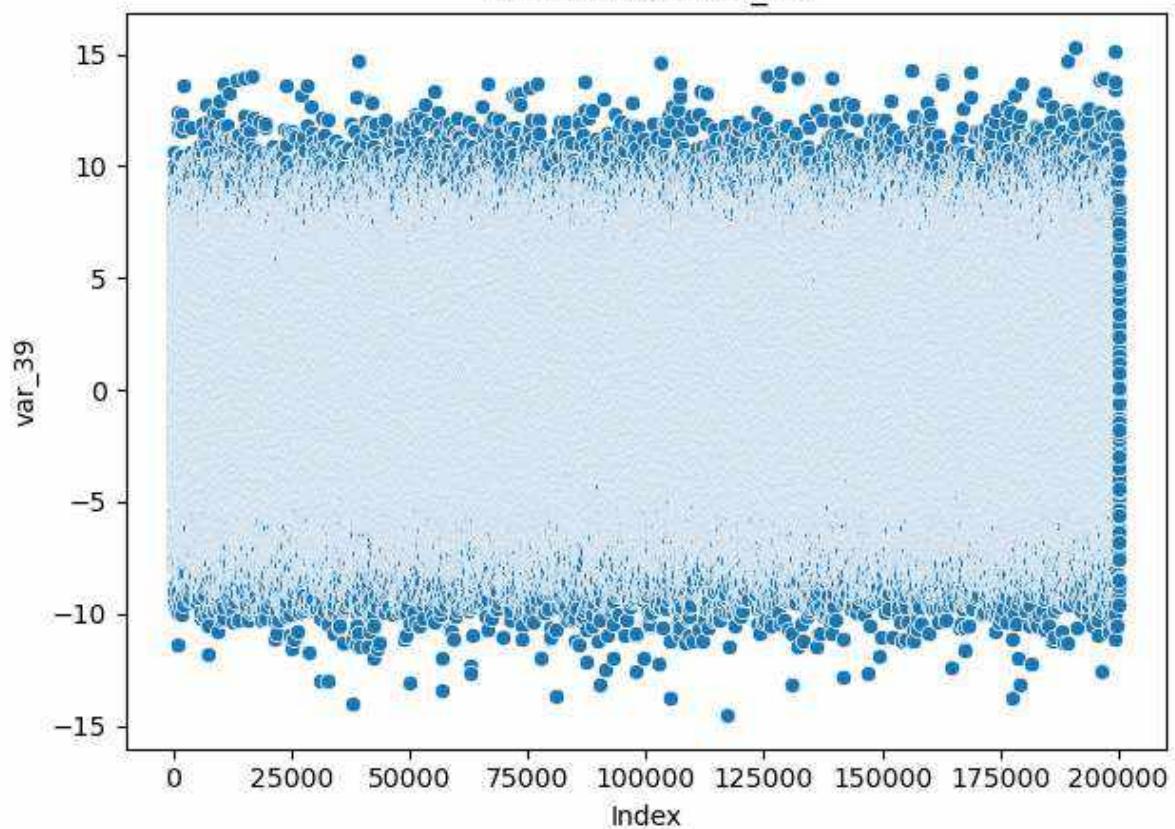
Scatter Plot - var\_37



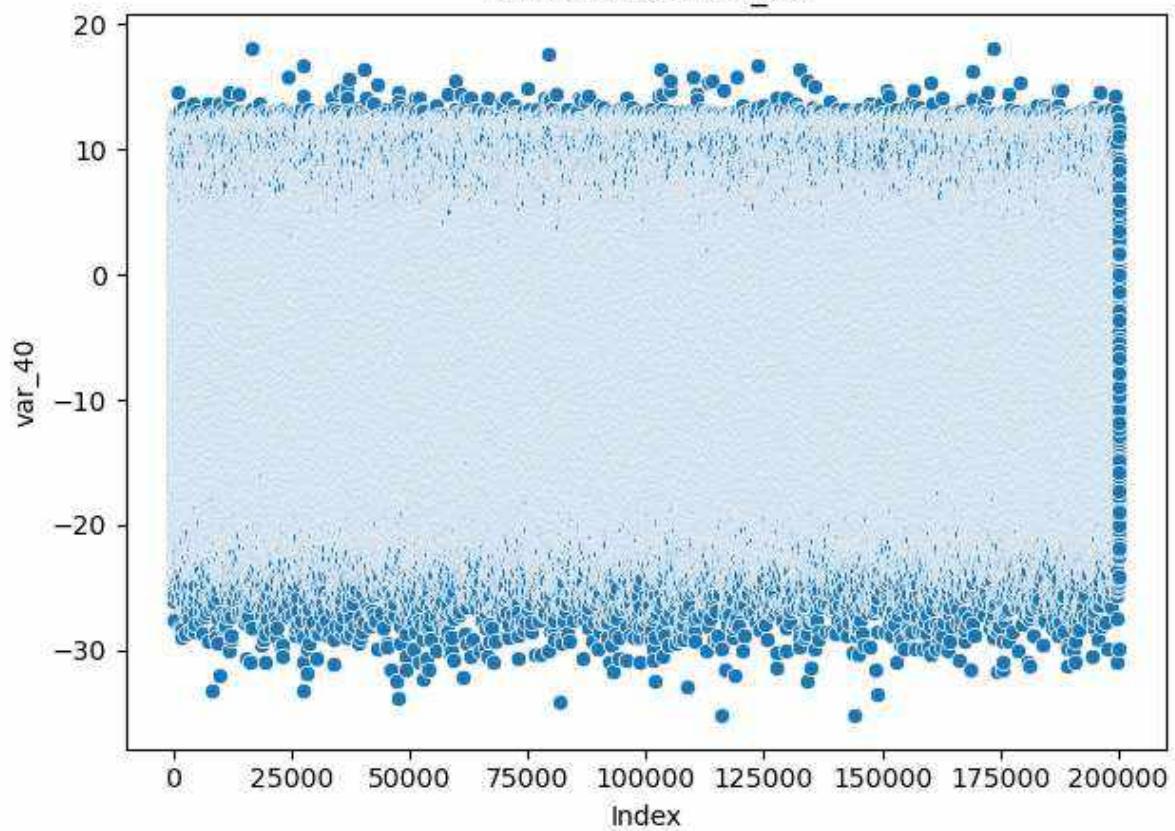
Scatter Plot - var\_38

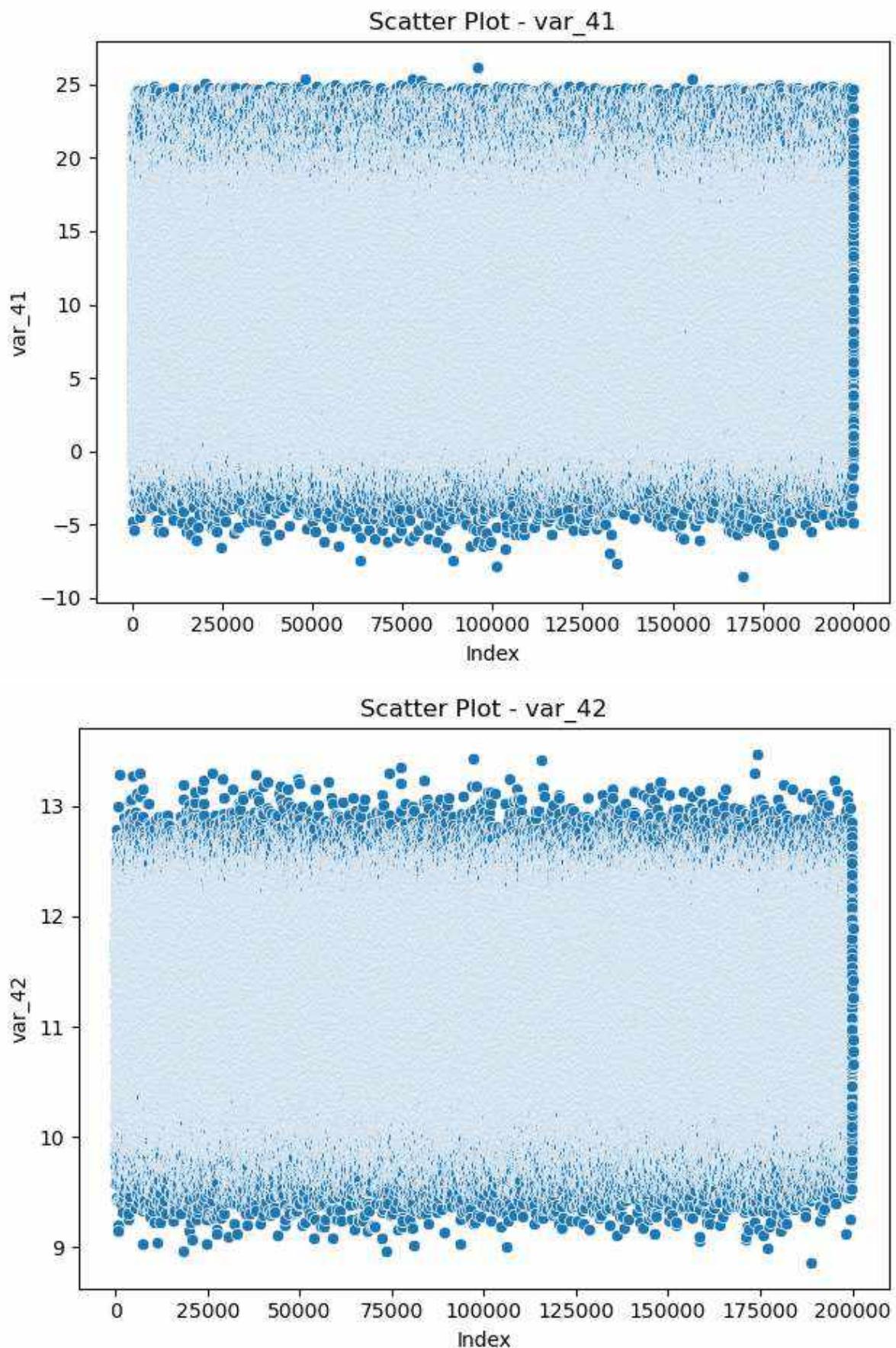


Scatter Plot - var\_39

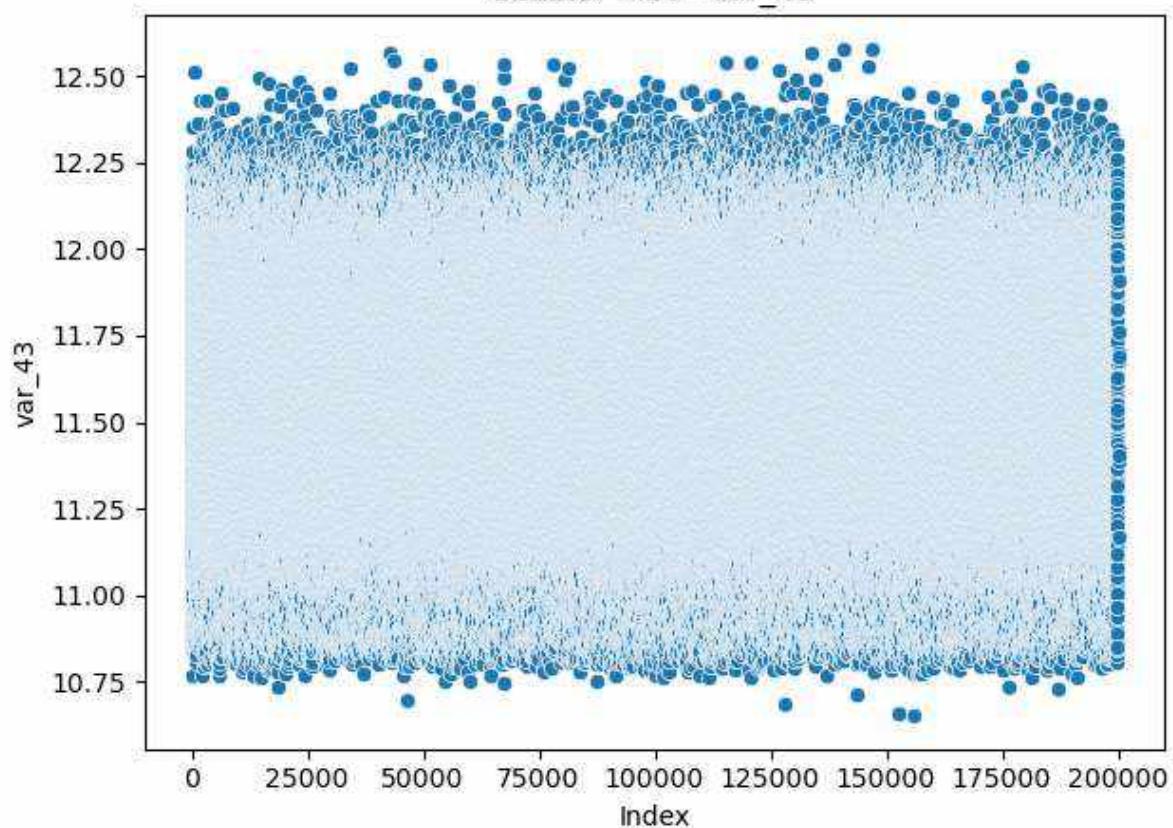


Scatter Plot - var\_40

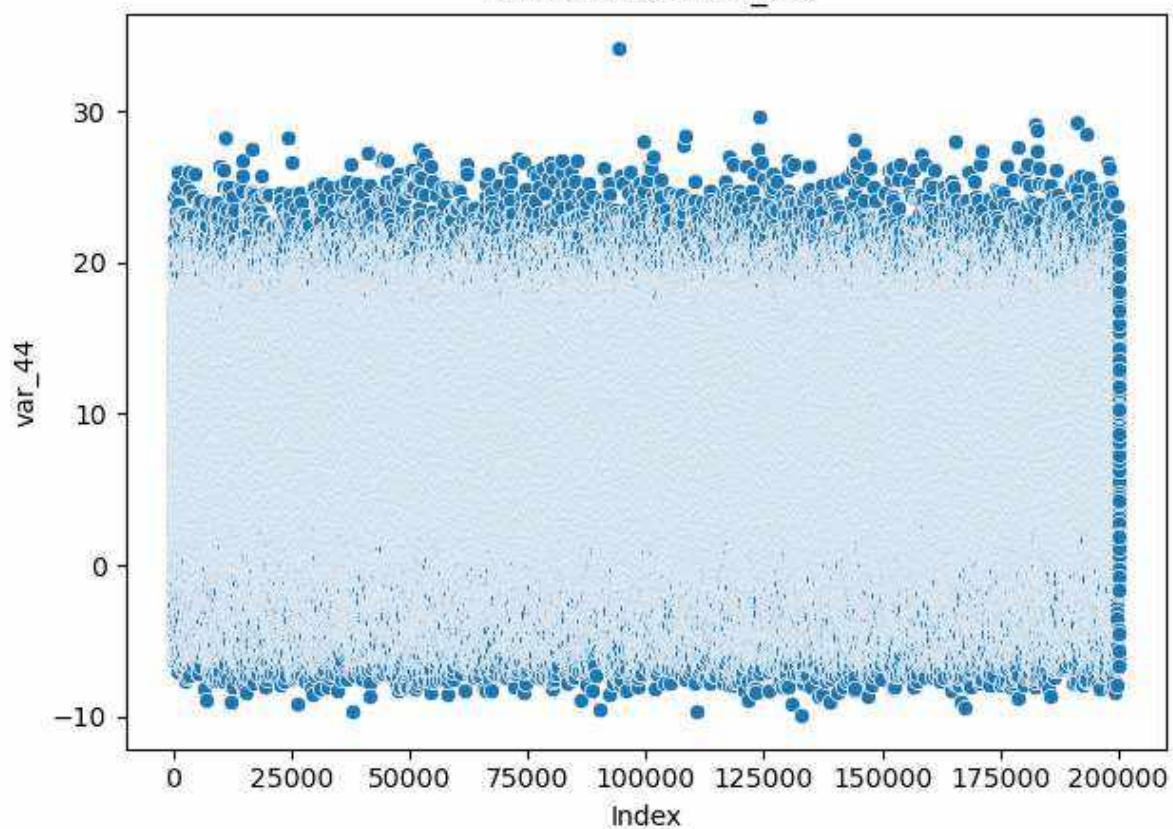




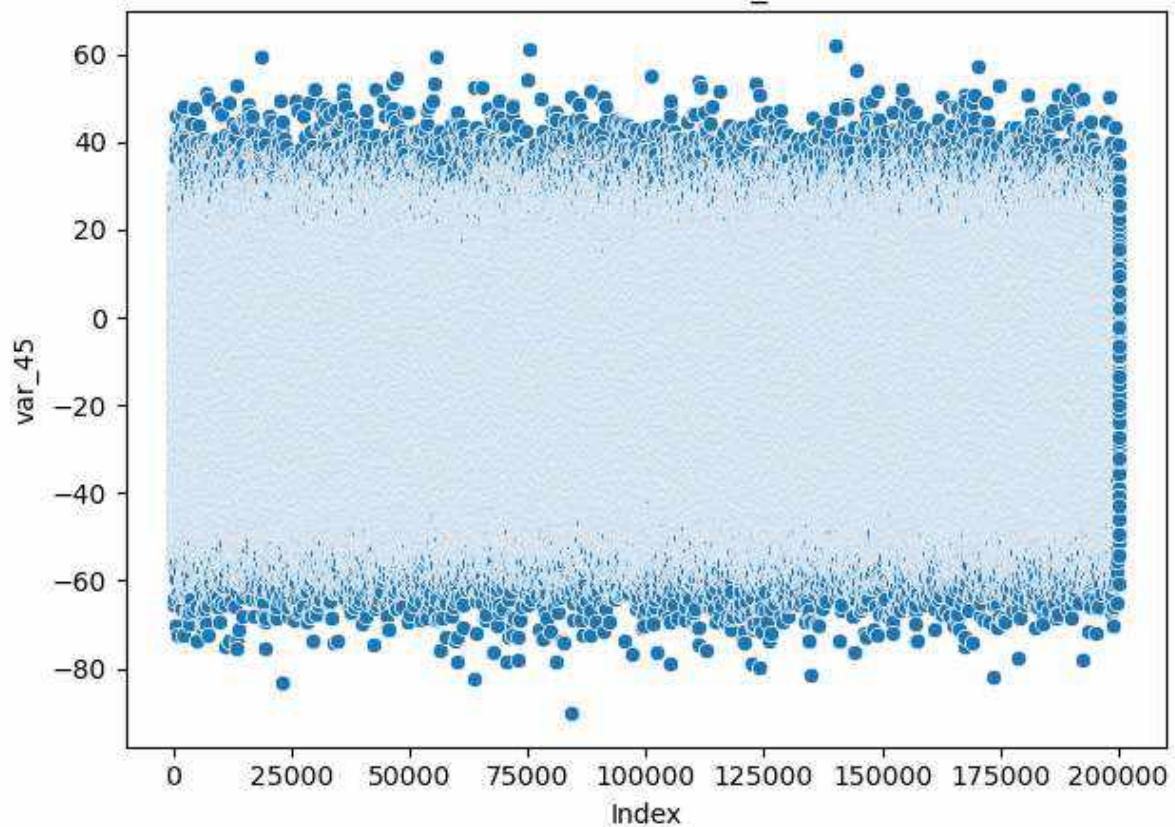
Scatter Plot - var\_43



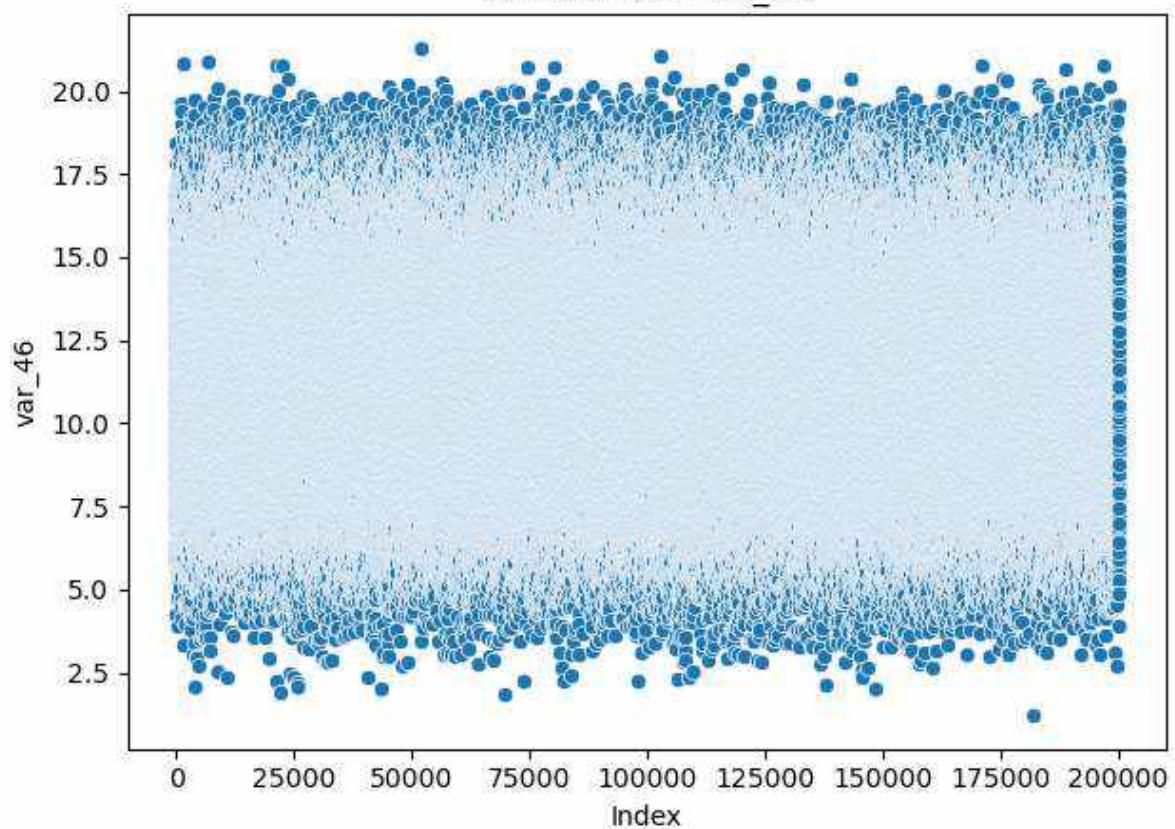
Scatter Plot - var\_44



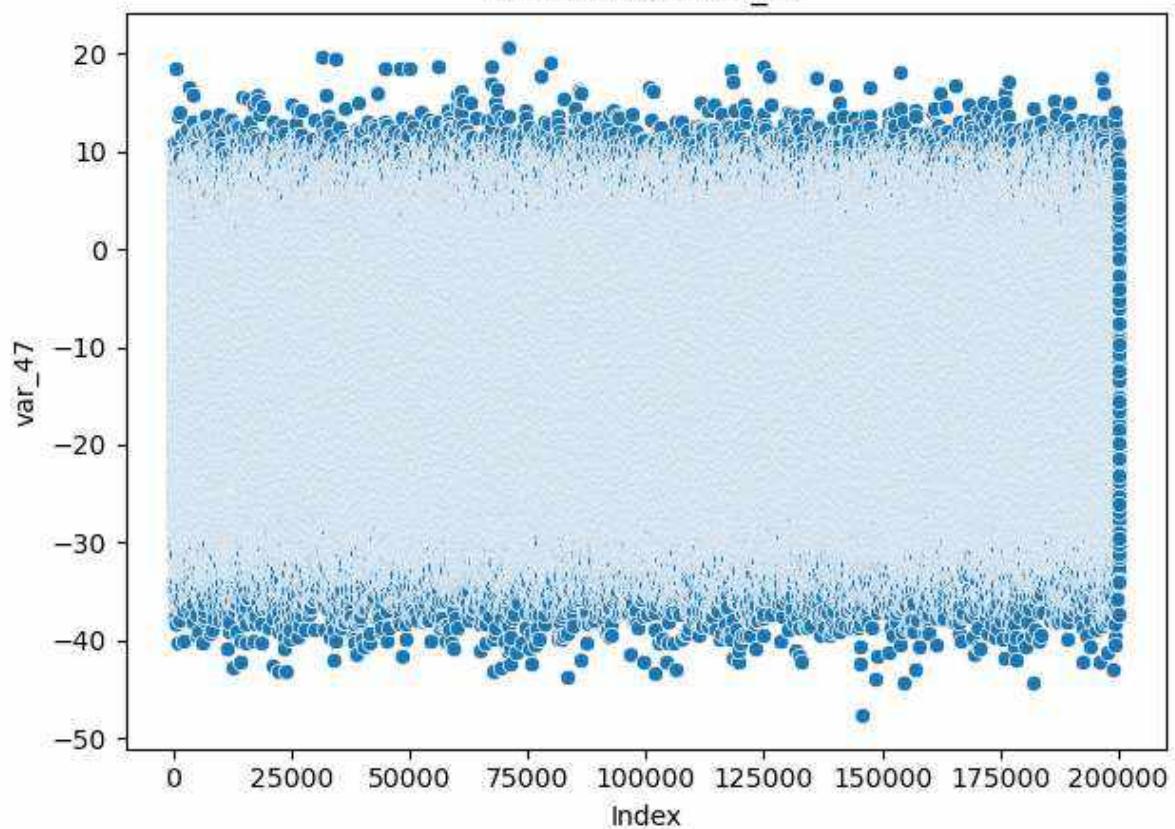
Scatter Plot - var\_45



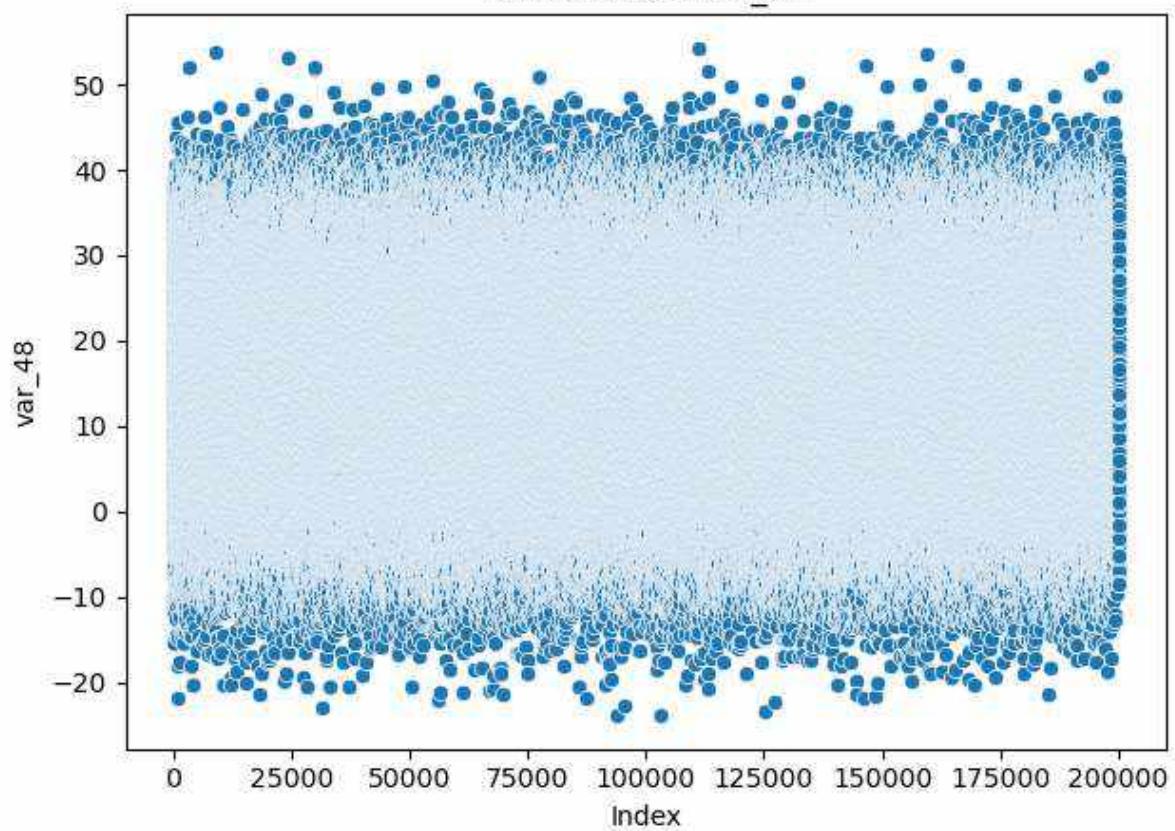
Scatter Plot - var\_46



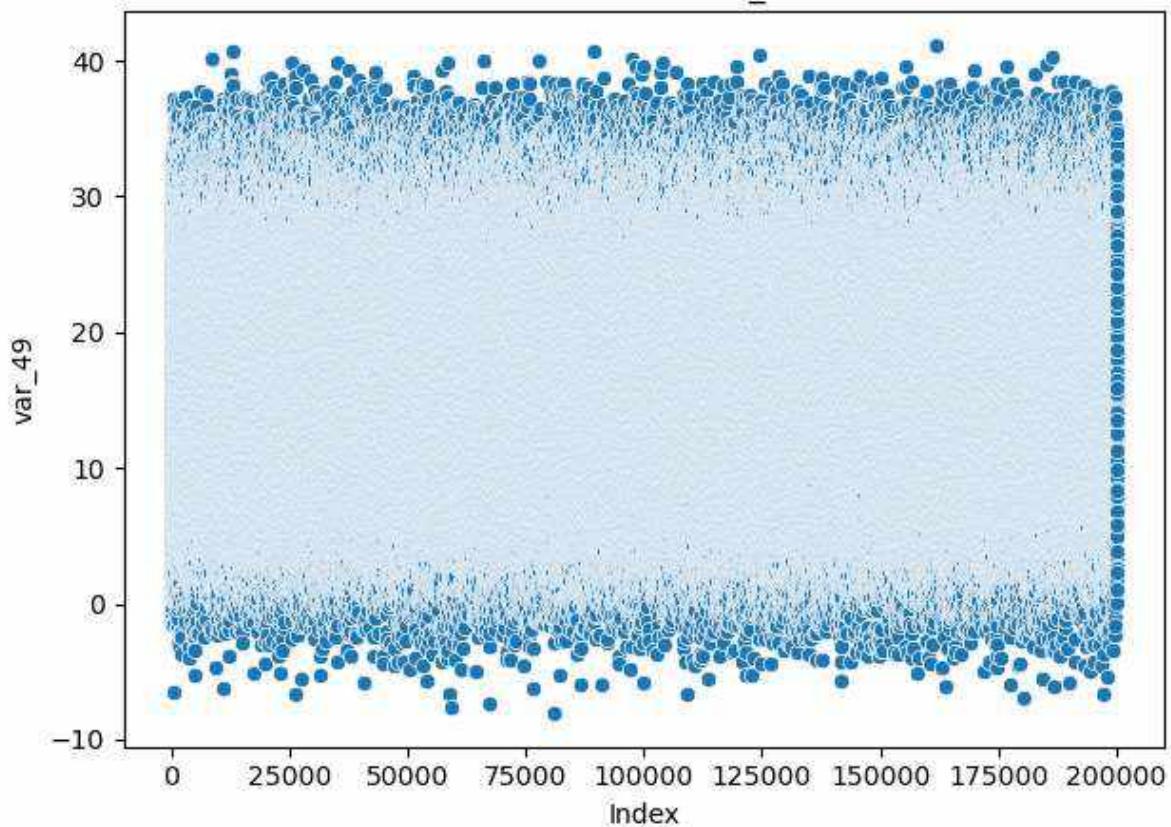
Scatter Plot - var\_47



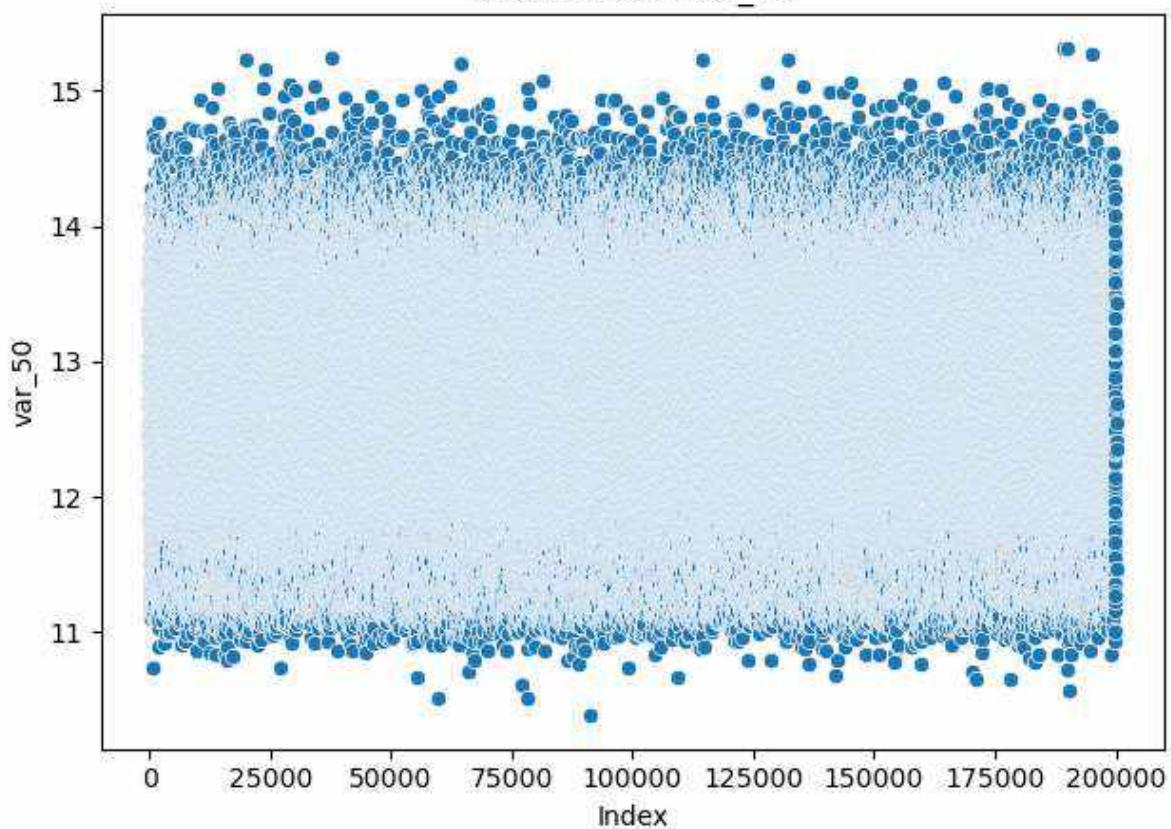
Scatter Plot - var\_48



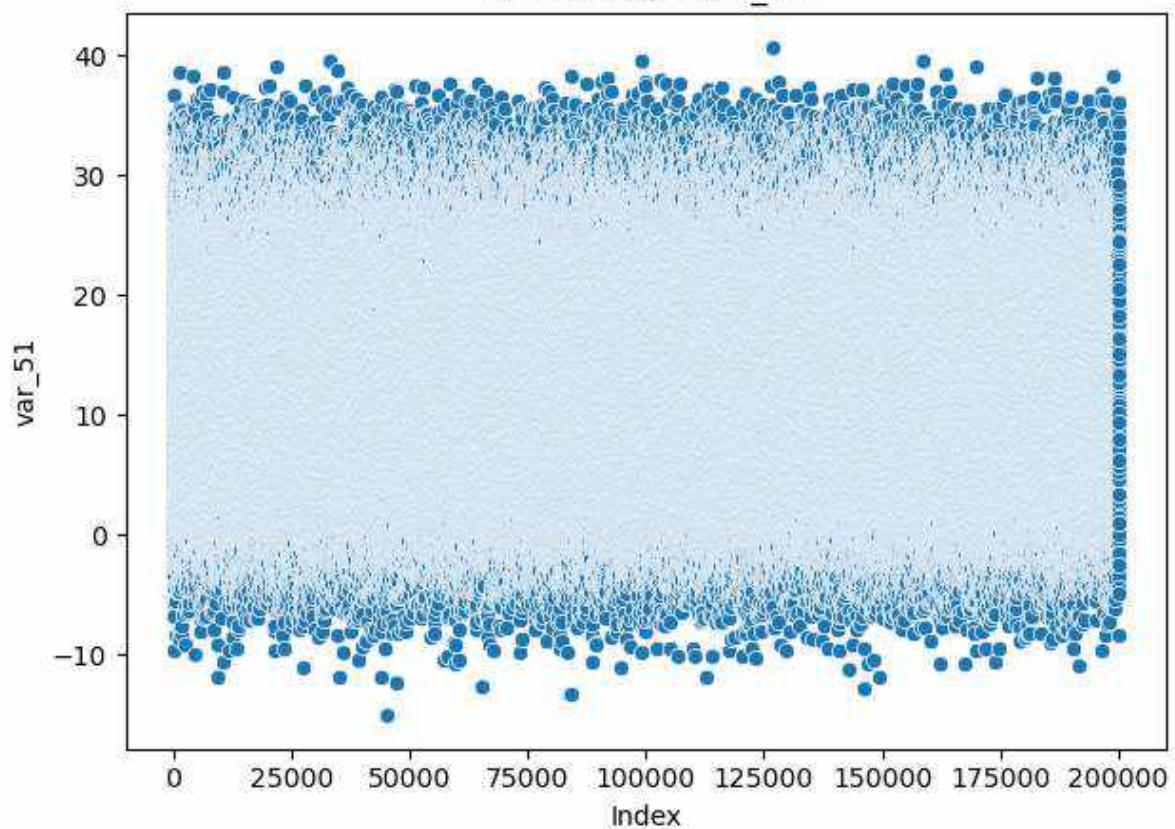
Scatter Plot - var\_49



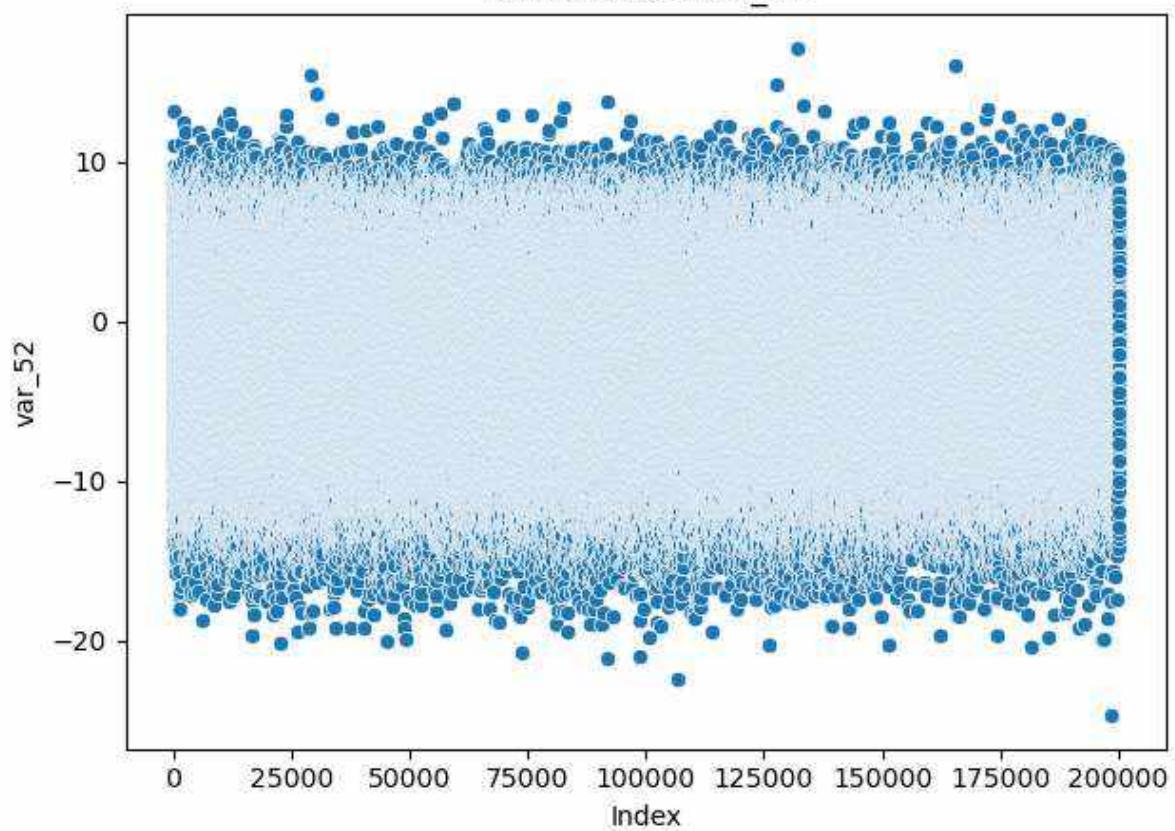
Scatter Plot - var\_50



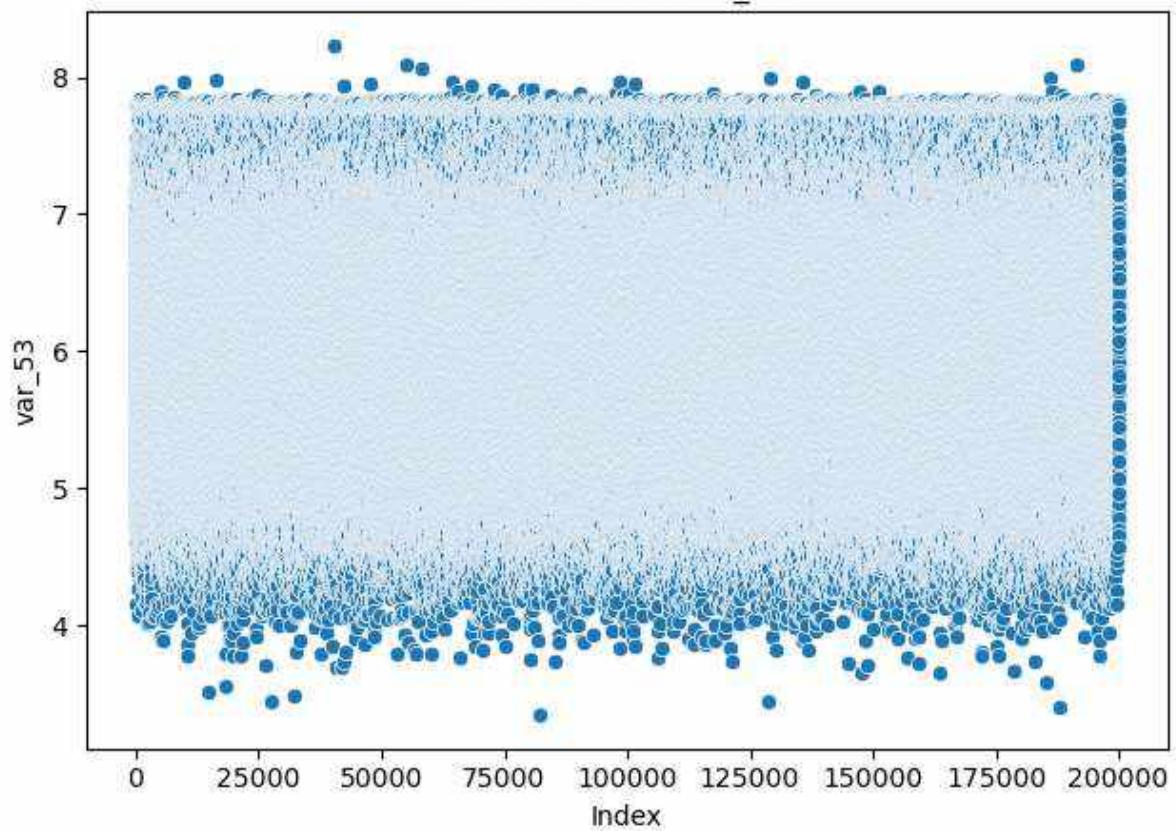
Scatter Plot - var\_51



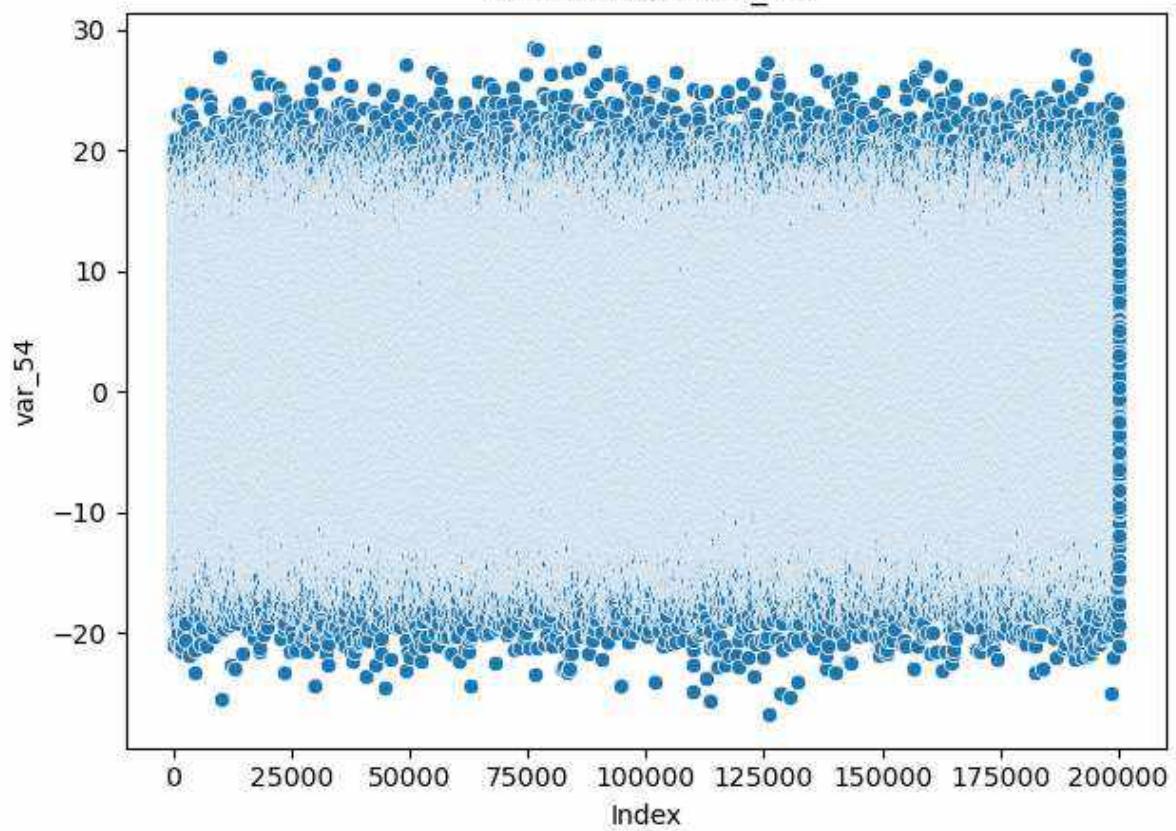
Scatter Plot - var\_52

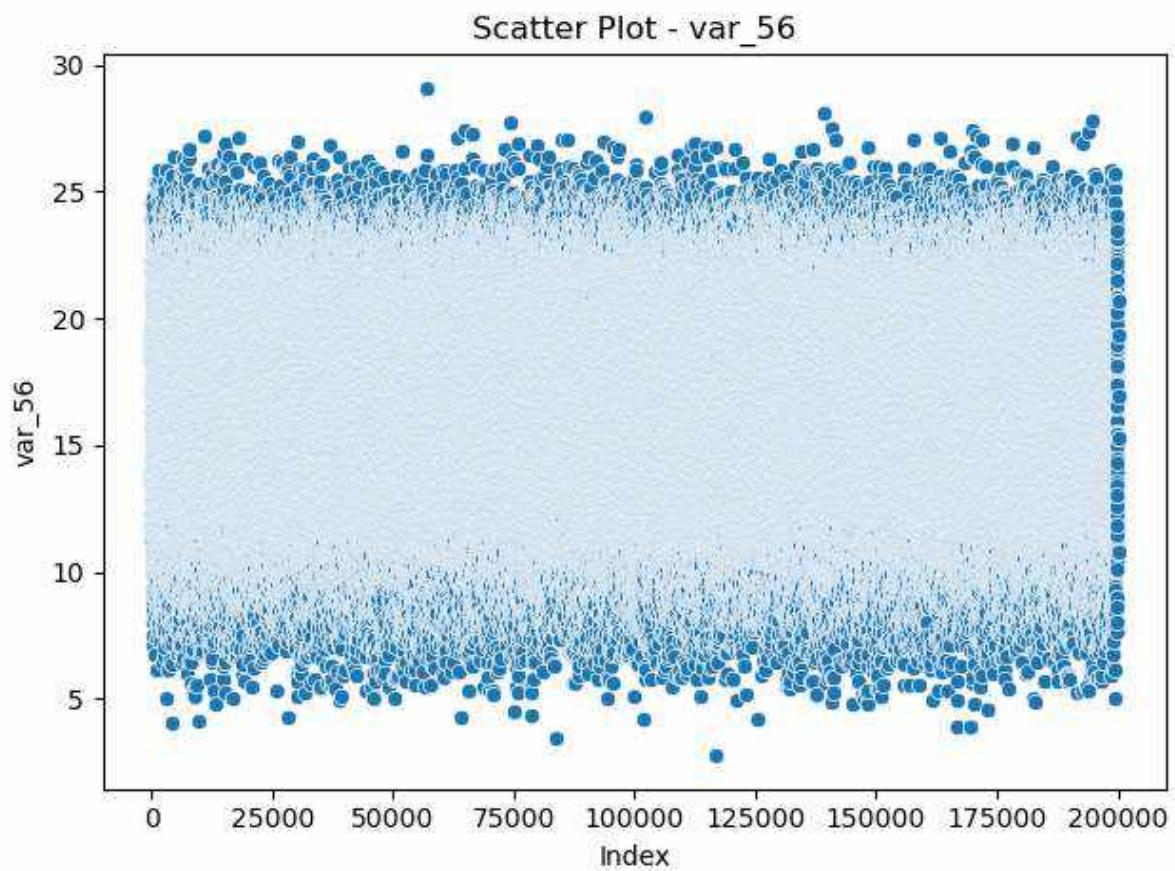
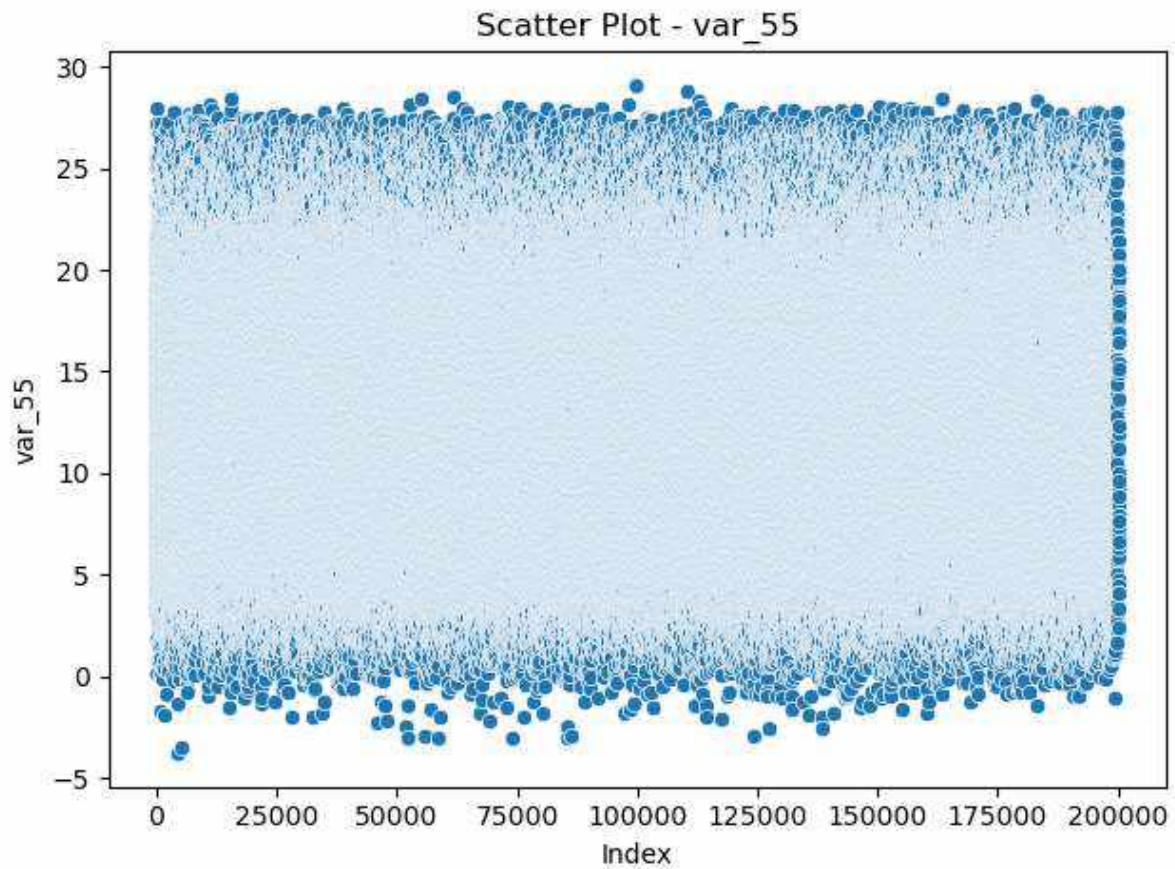


Scatter Plot - var\_53

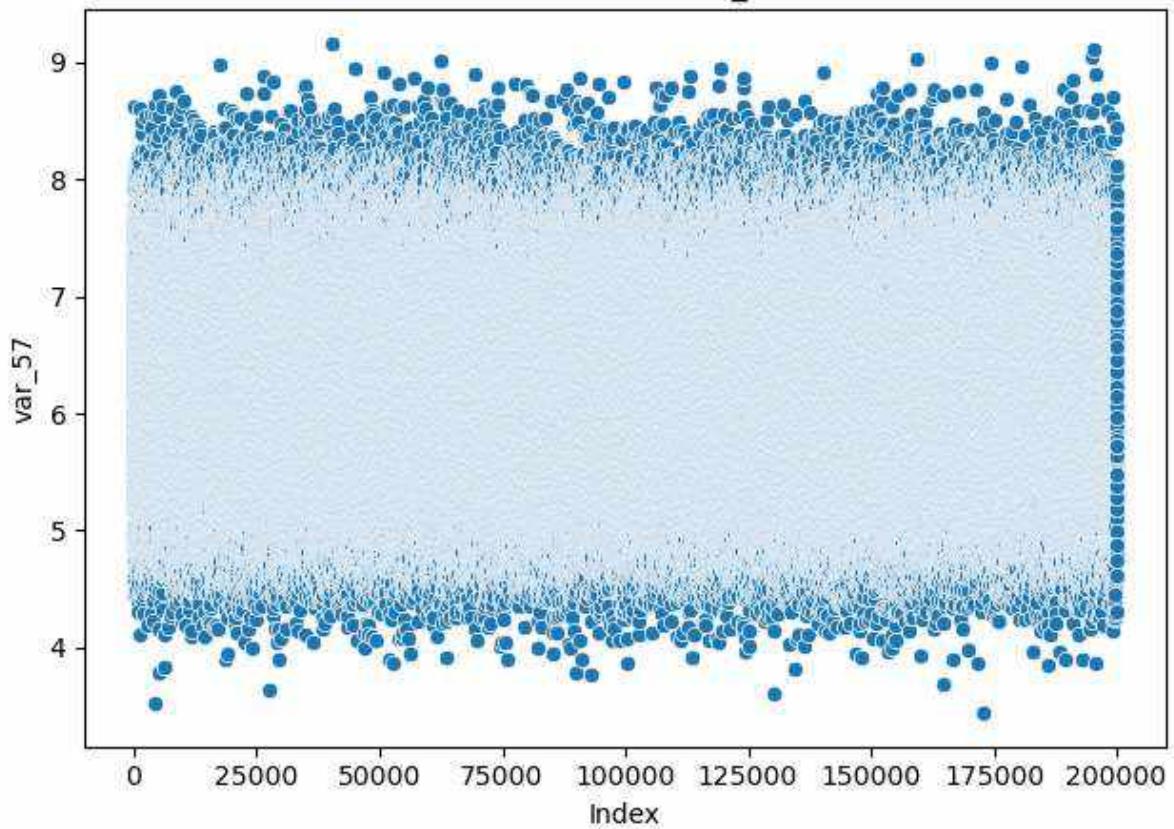


Scatter Plot - var\_54

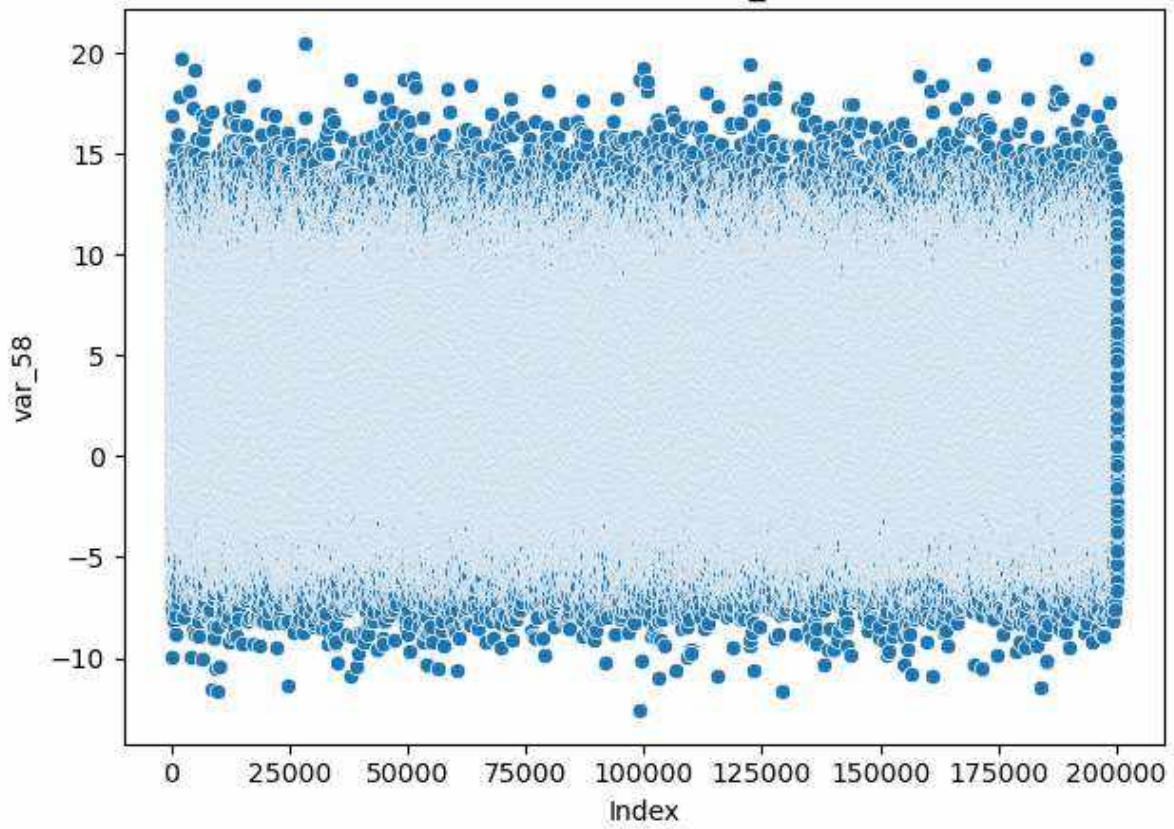




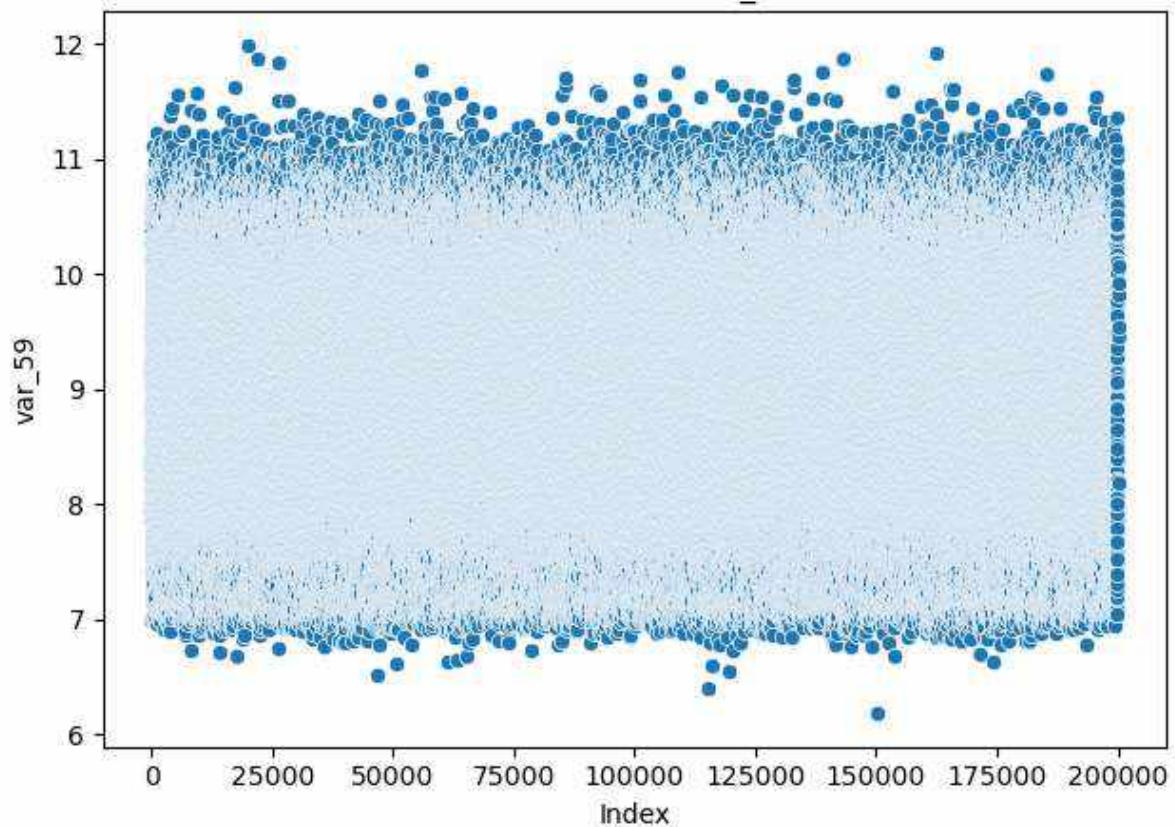
Scatter Plot - var\_57



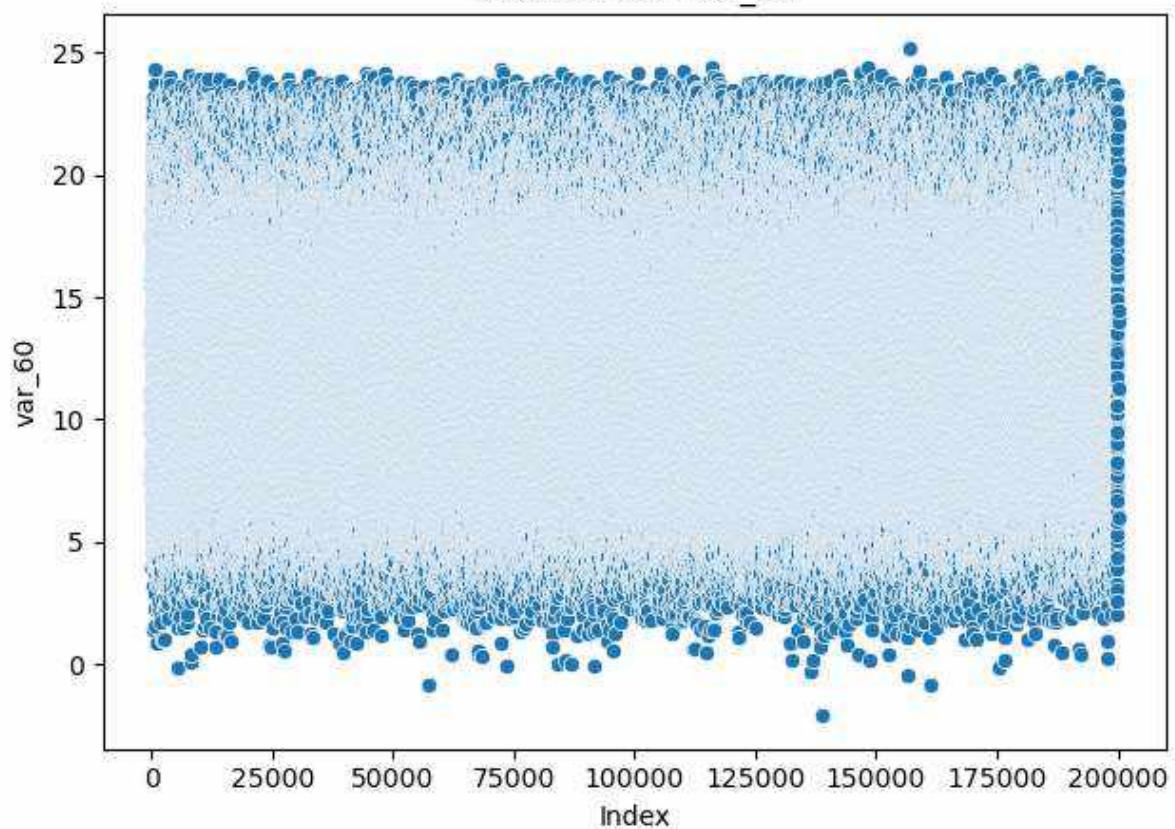
Scatter Plot - var\_58

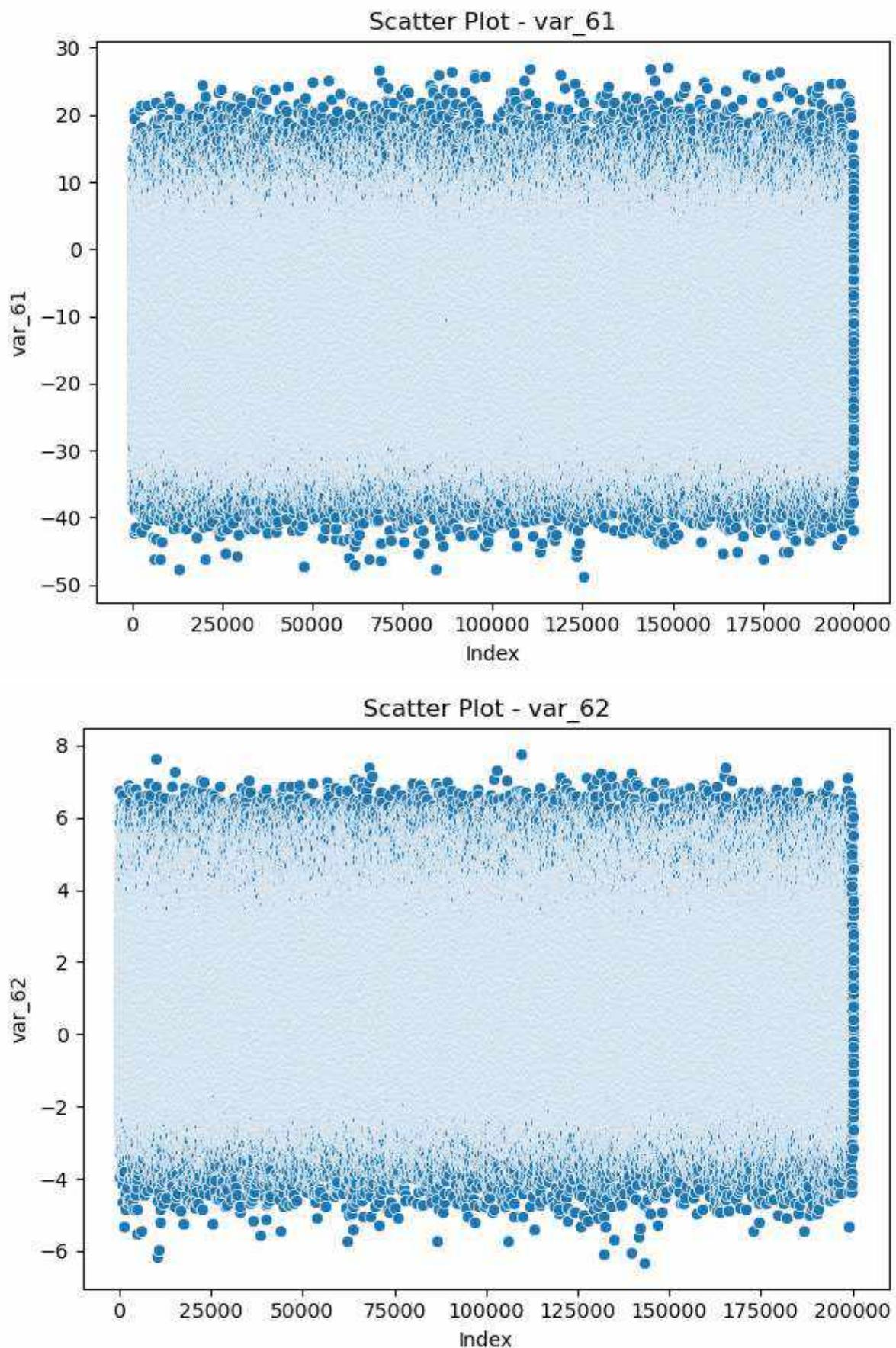


Scatter Plot - var\_59

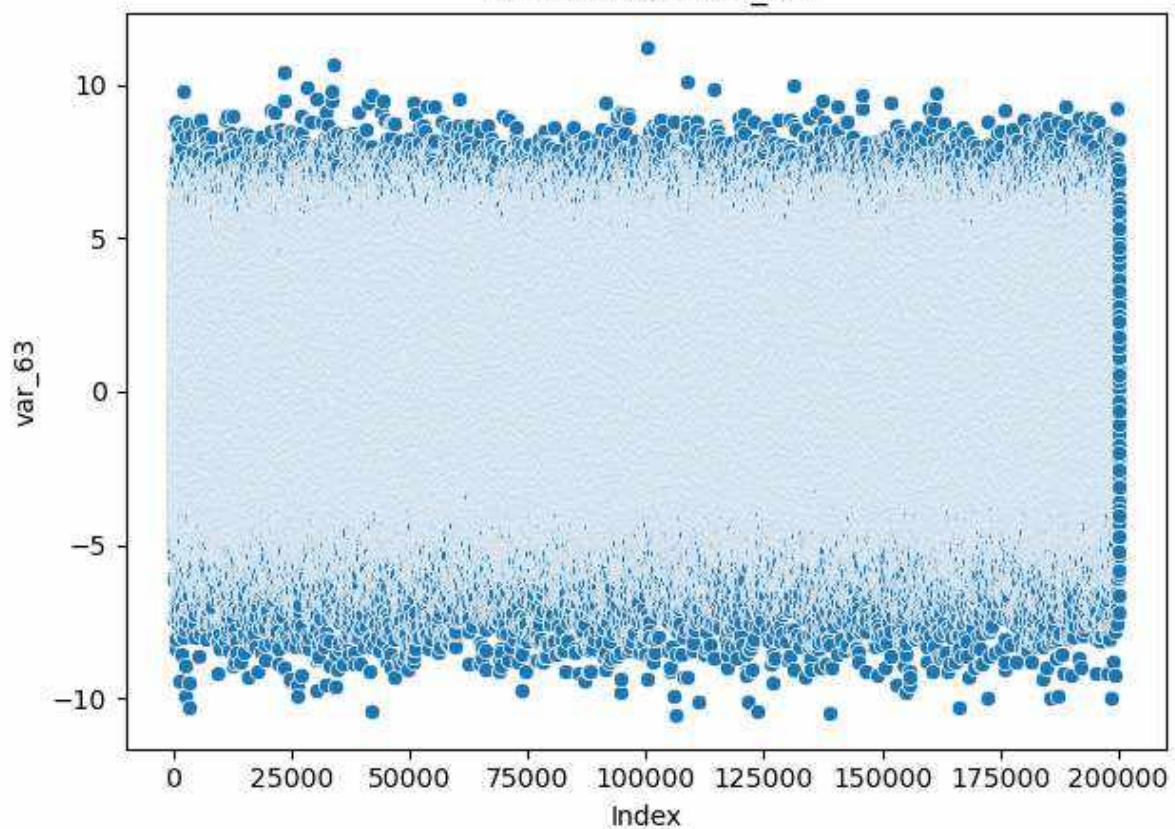


Scatter Plot - var\_60

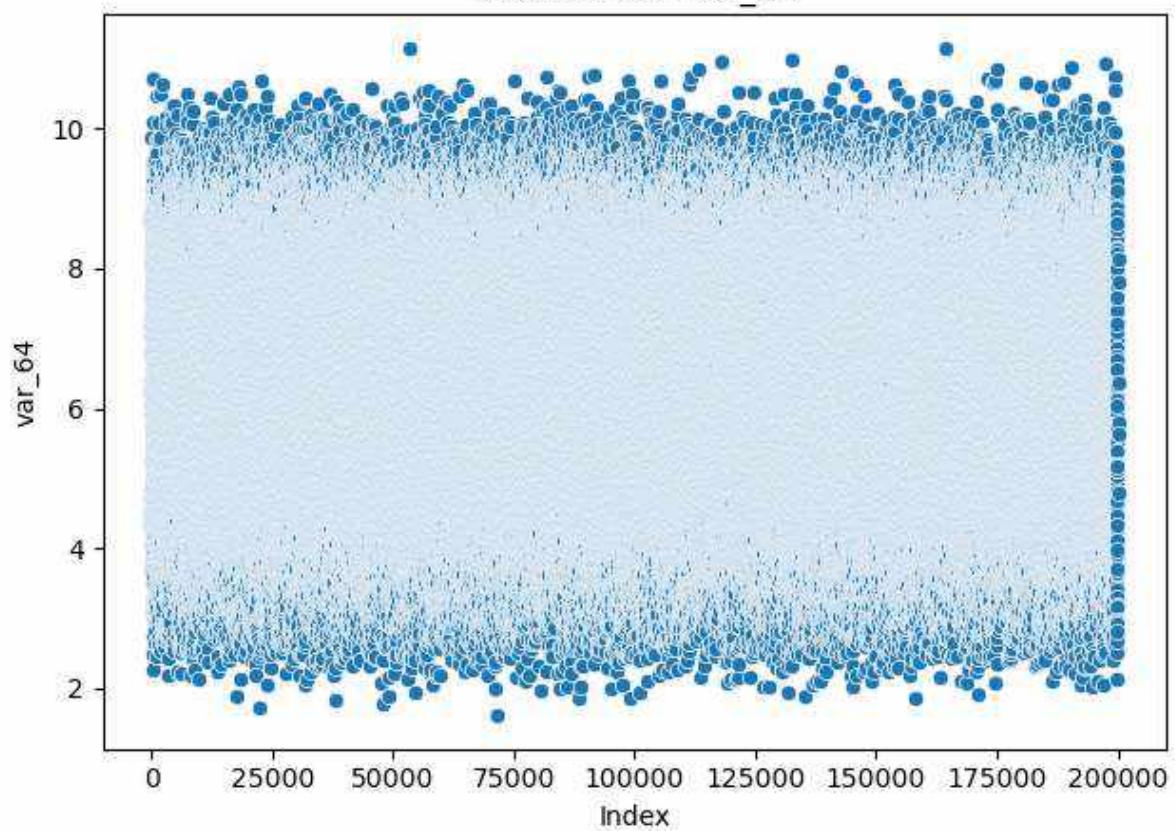


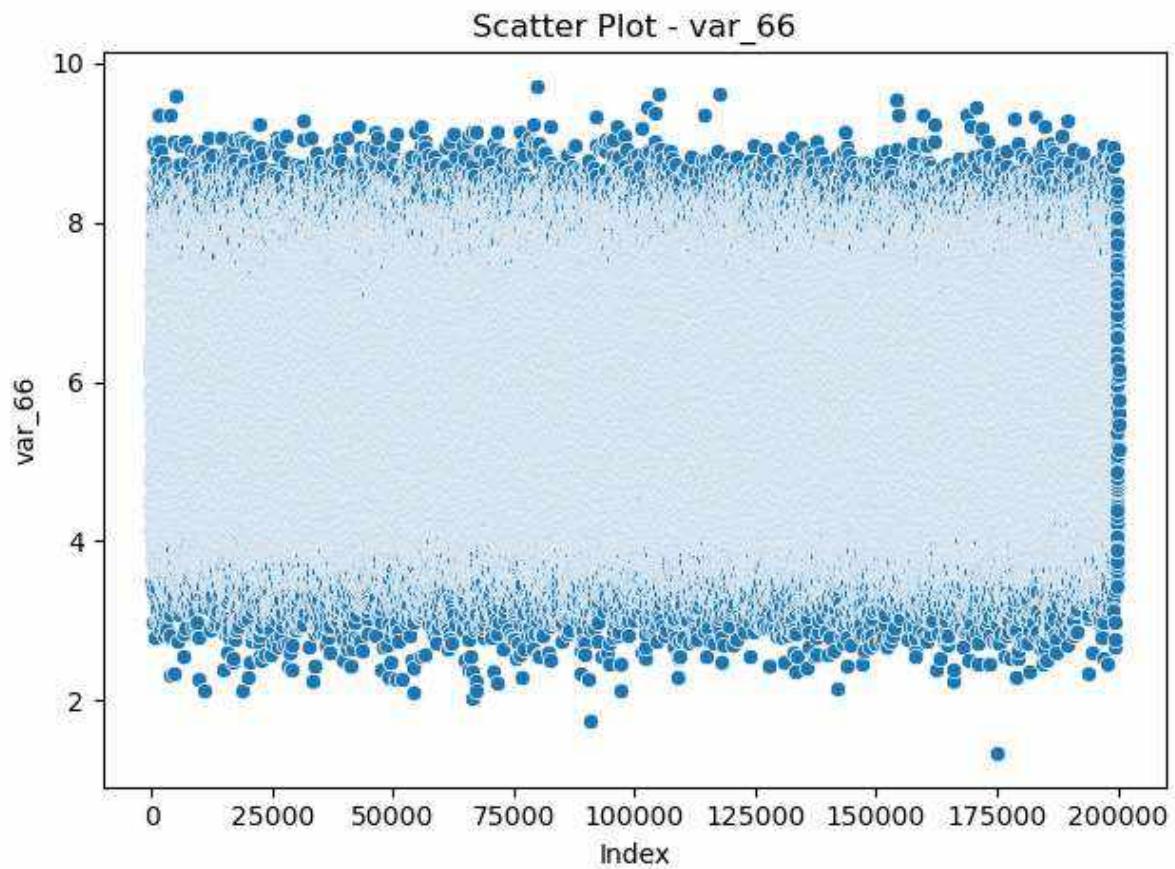
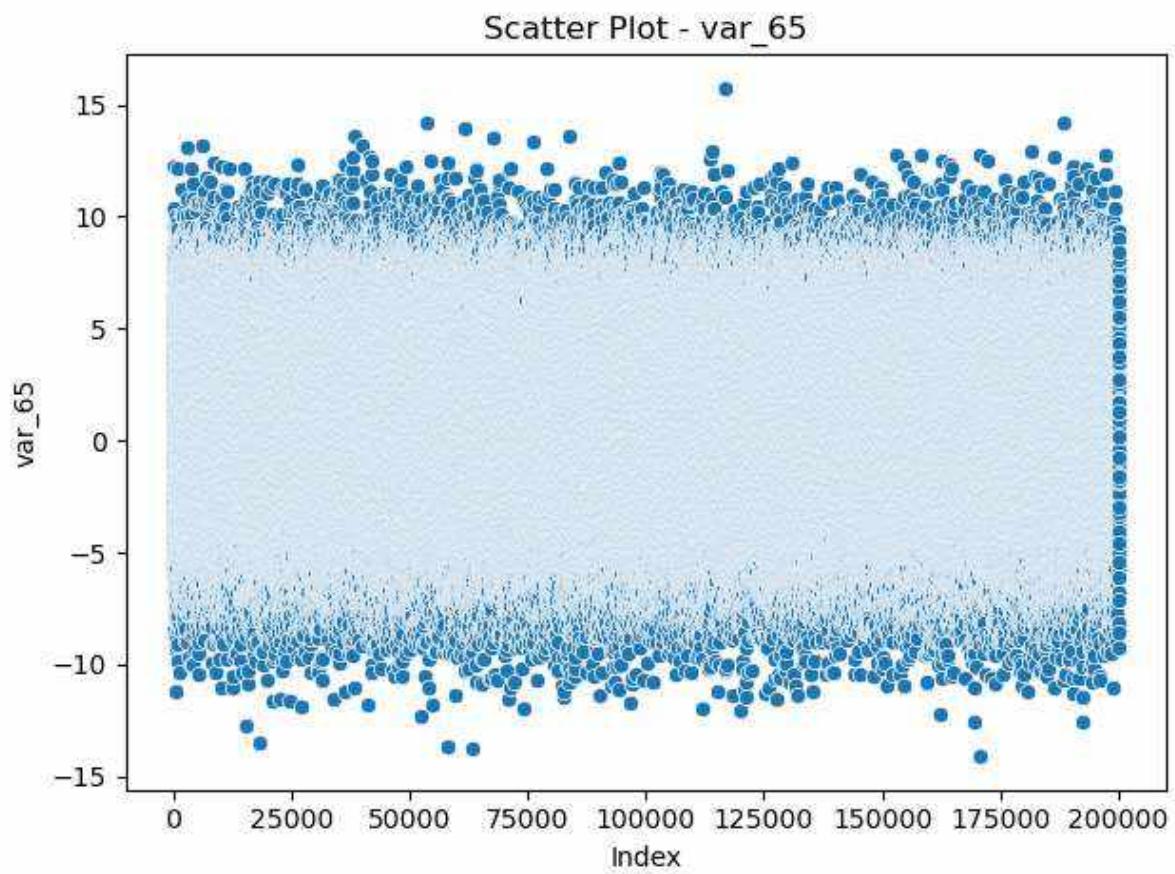


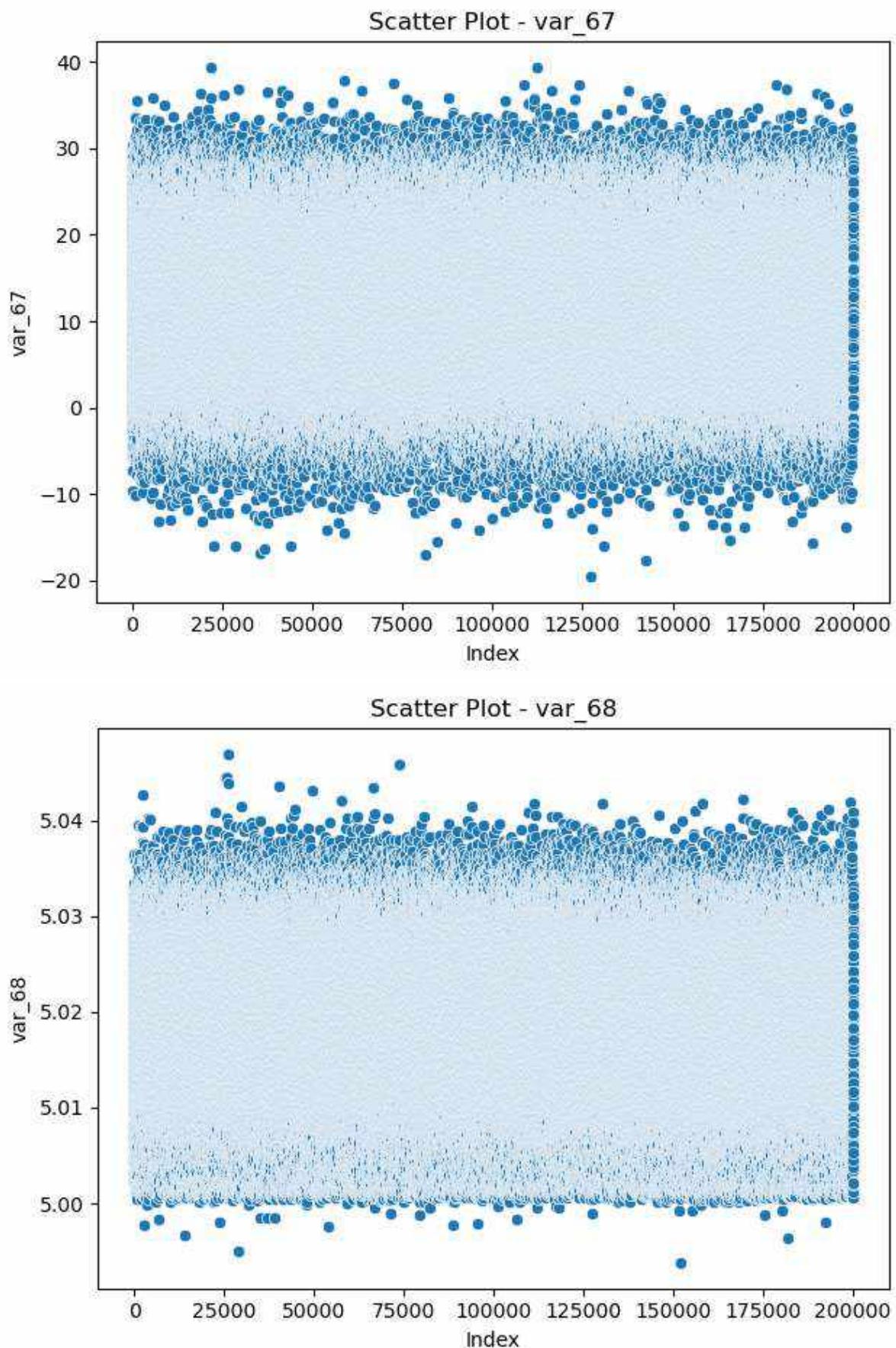
Scatter Plot - var\_63



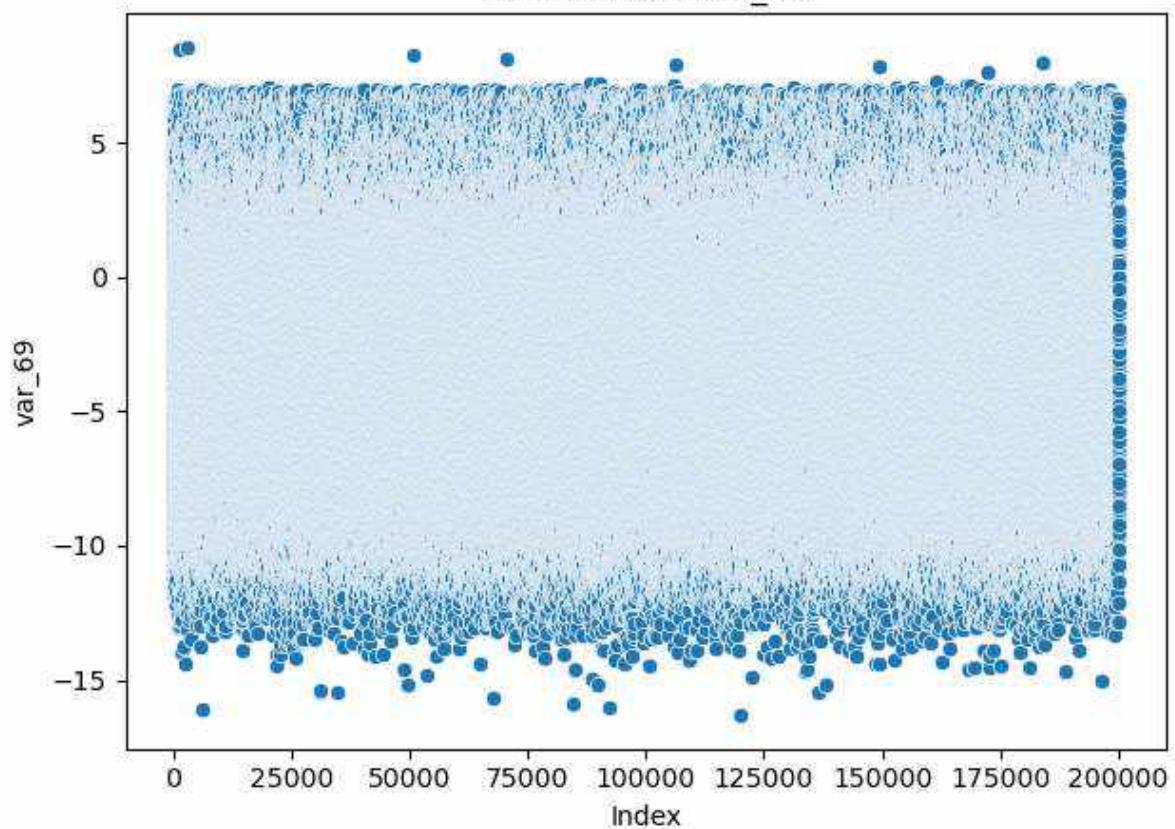
Scatter Plot - var\_64



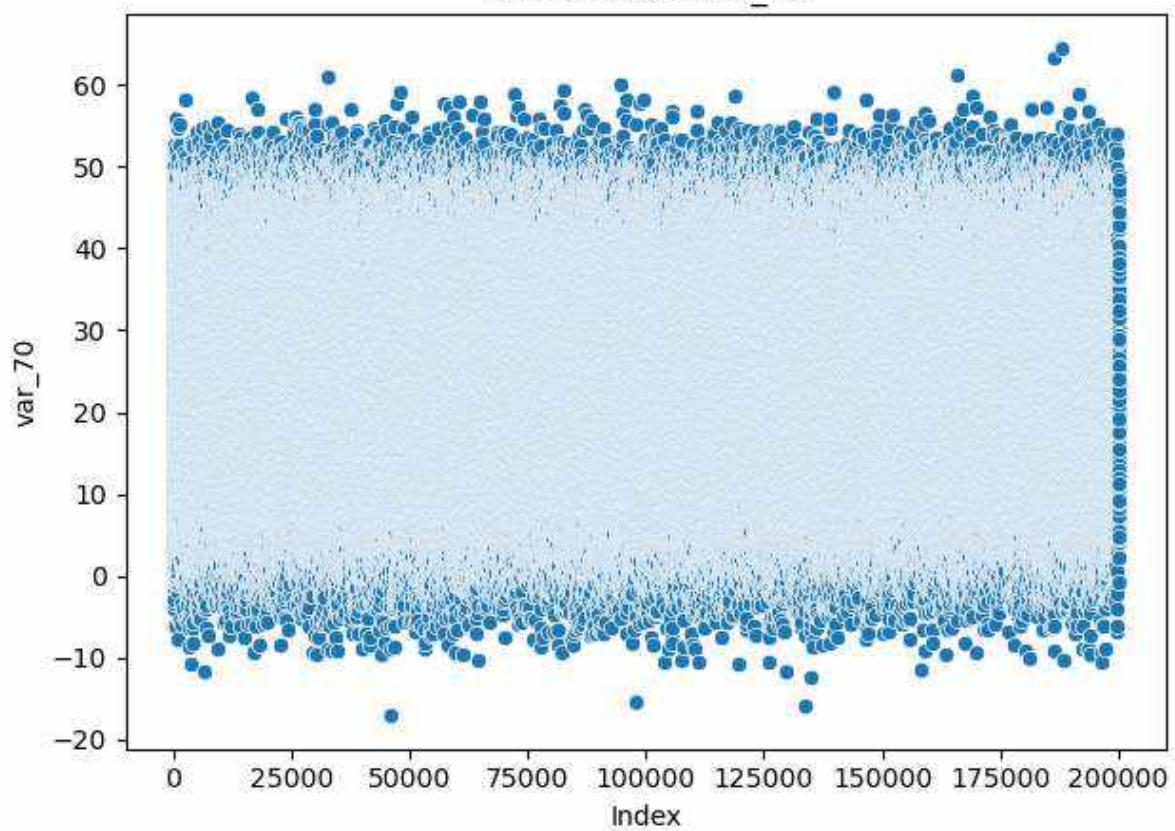




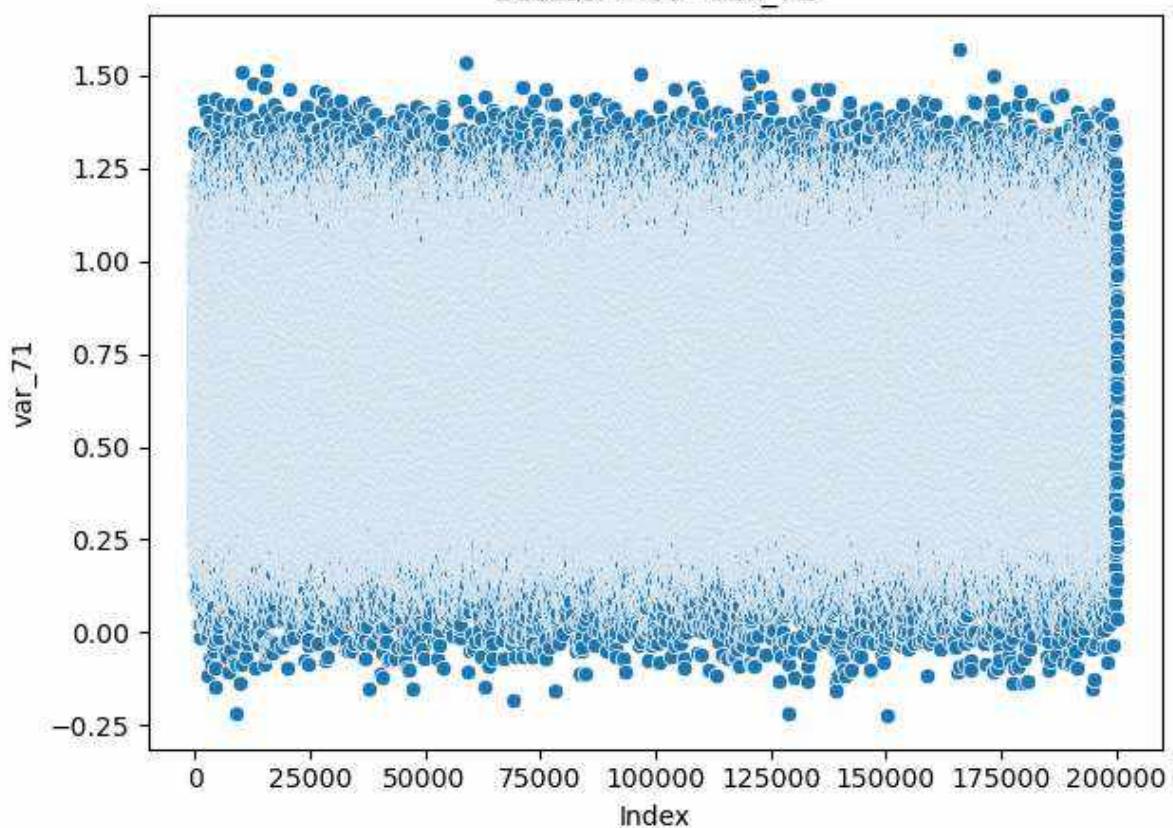
Scatter Plot - var\_69



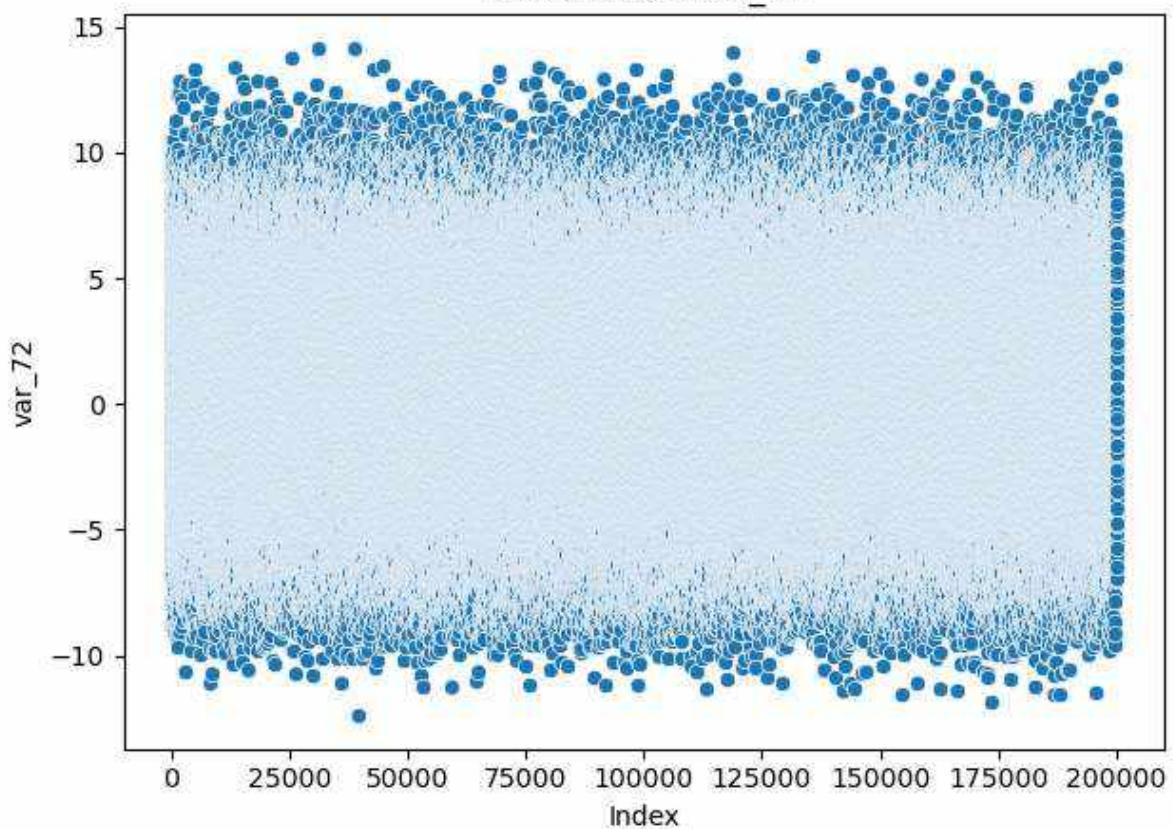
Scatter Plot - var\_70



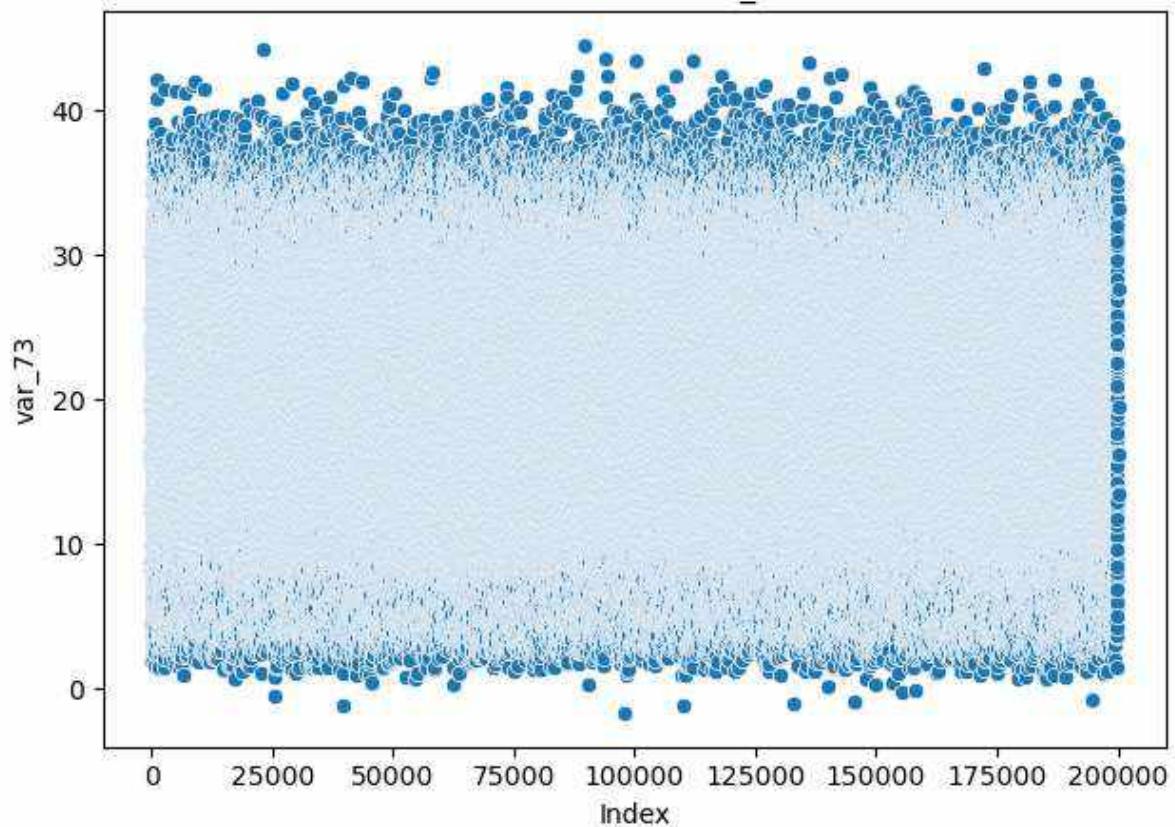
Scatter Plot - var\_71



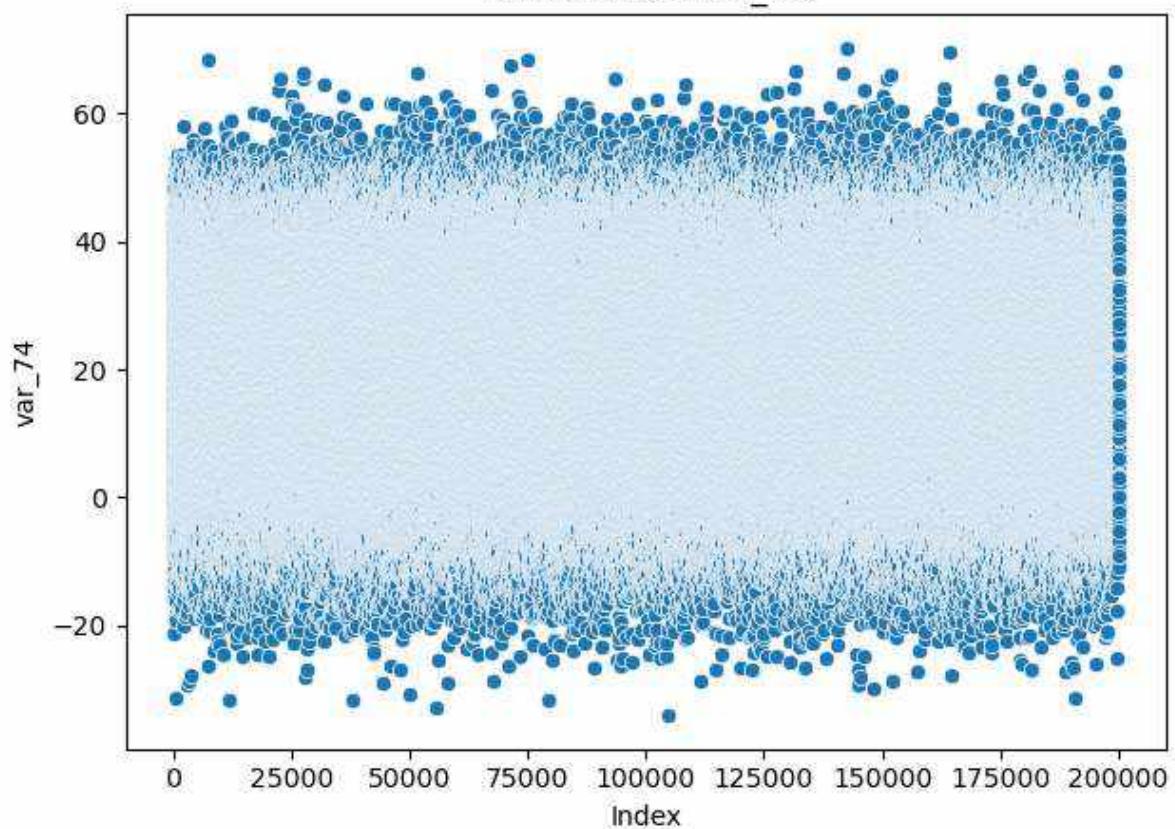
Scatter Plot - var\_72



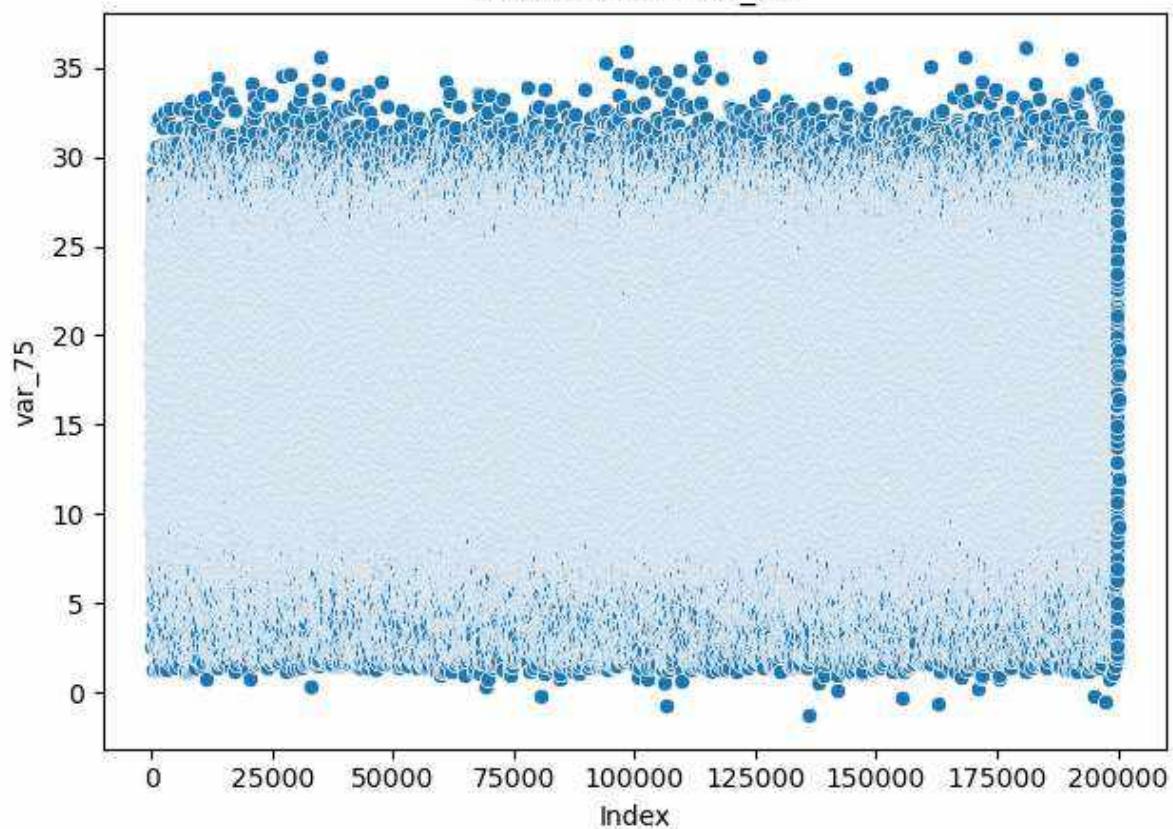
Scatter Plot - var\_73



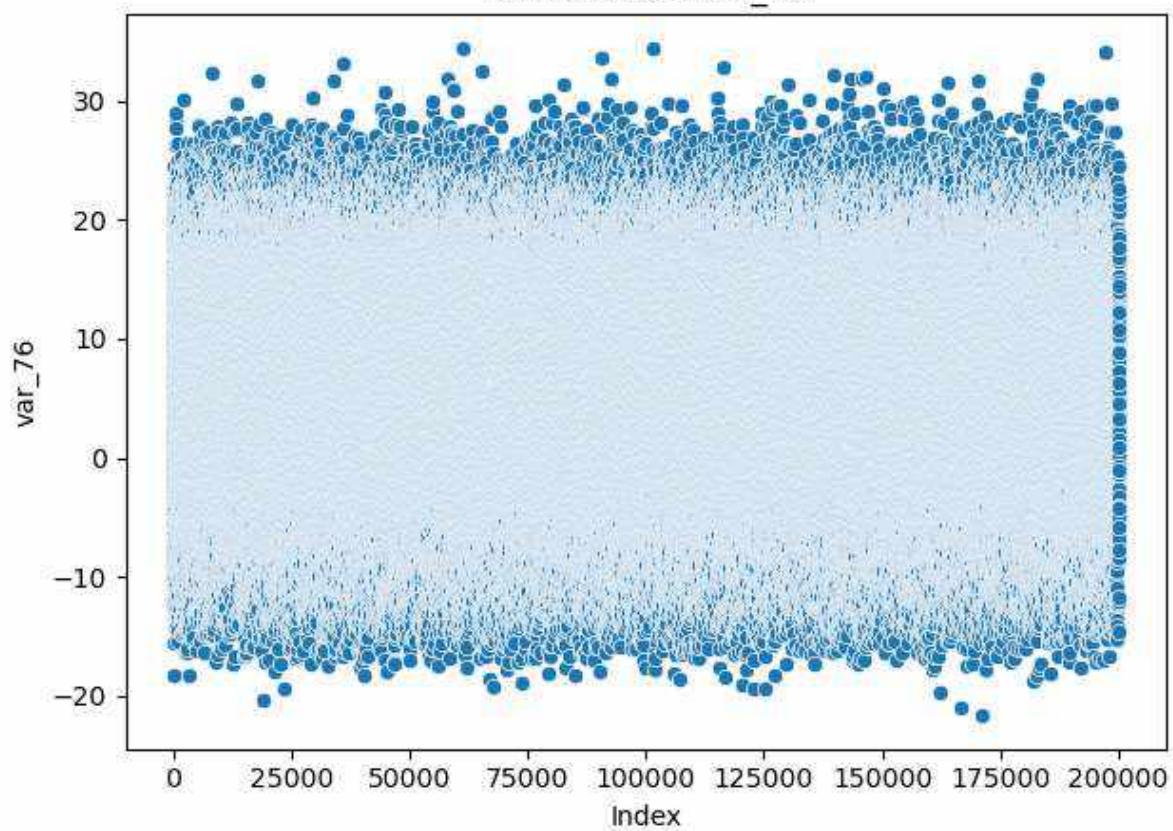
Scatter Plot - var\_74

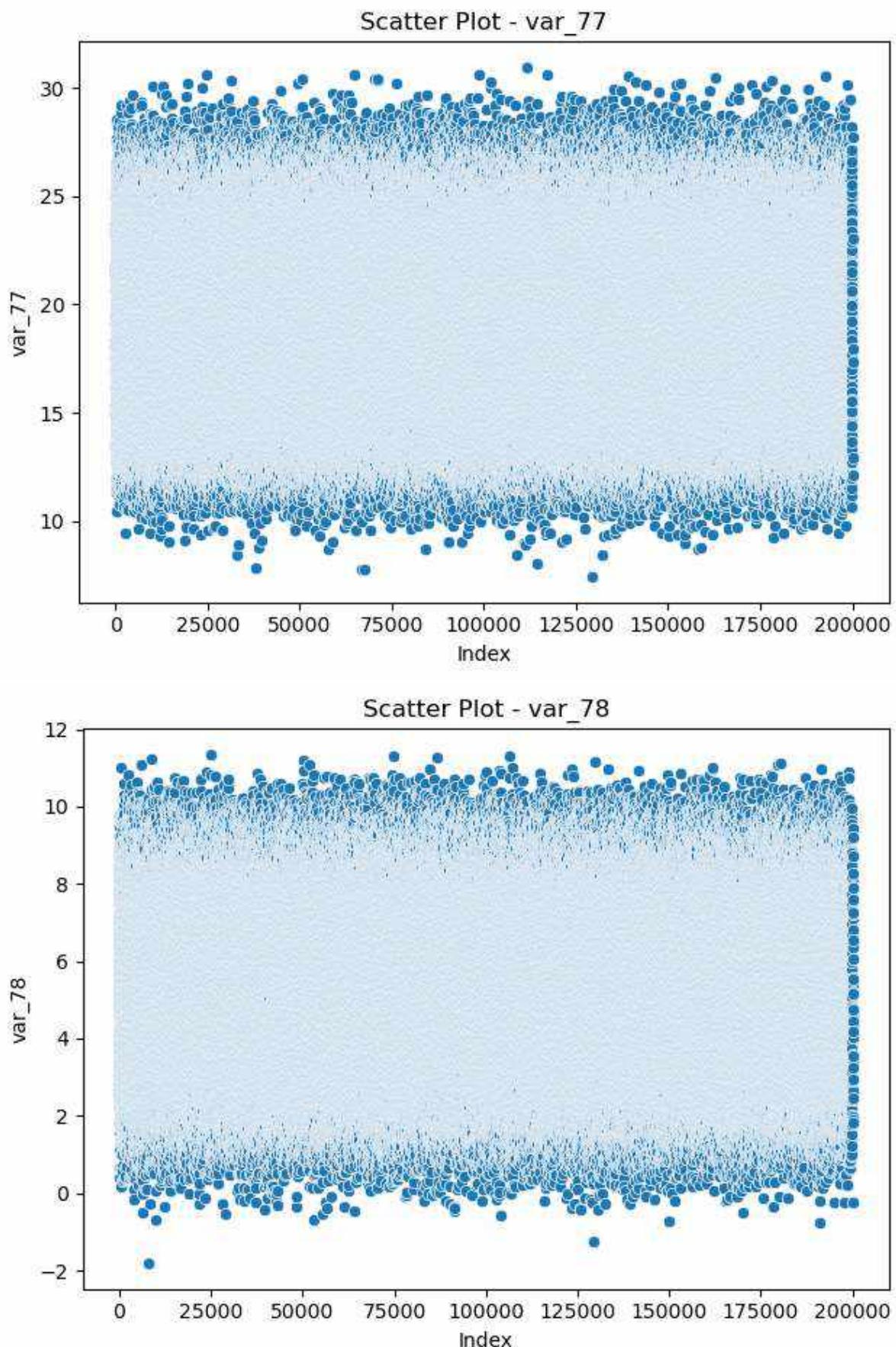


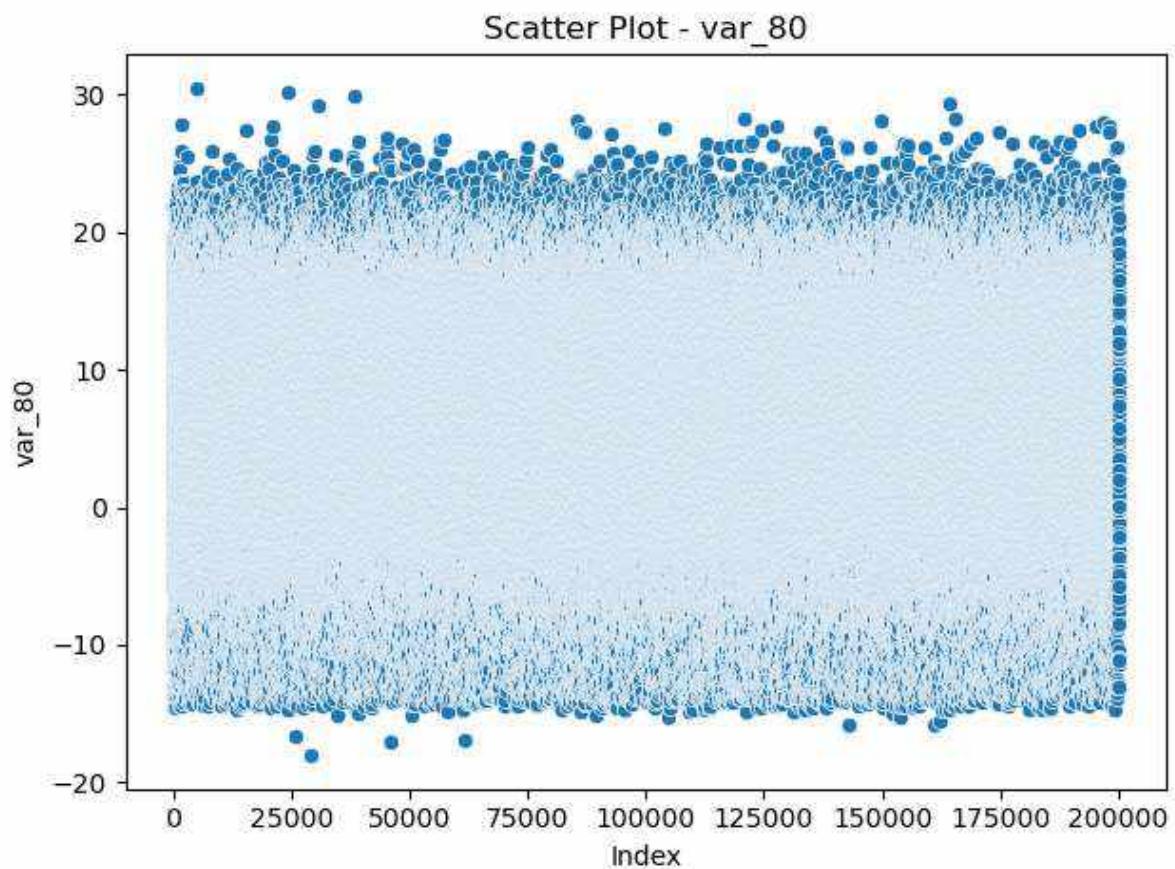
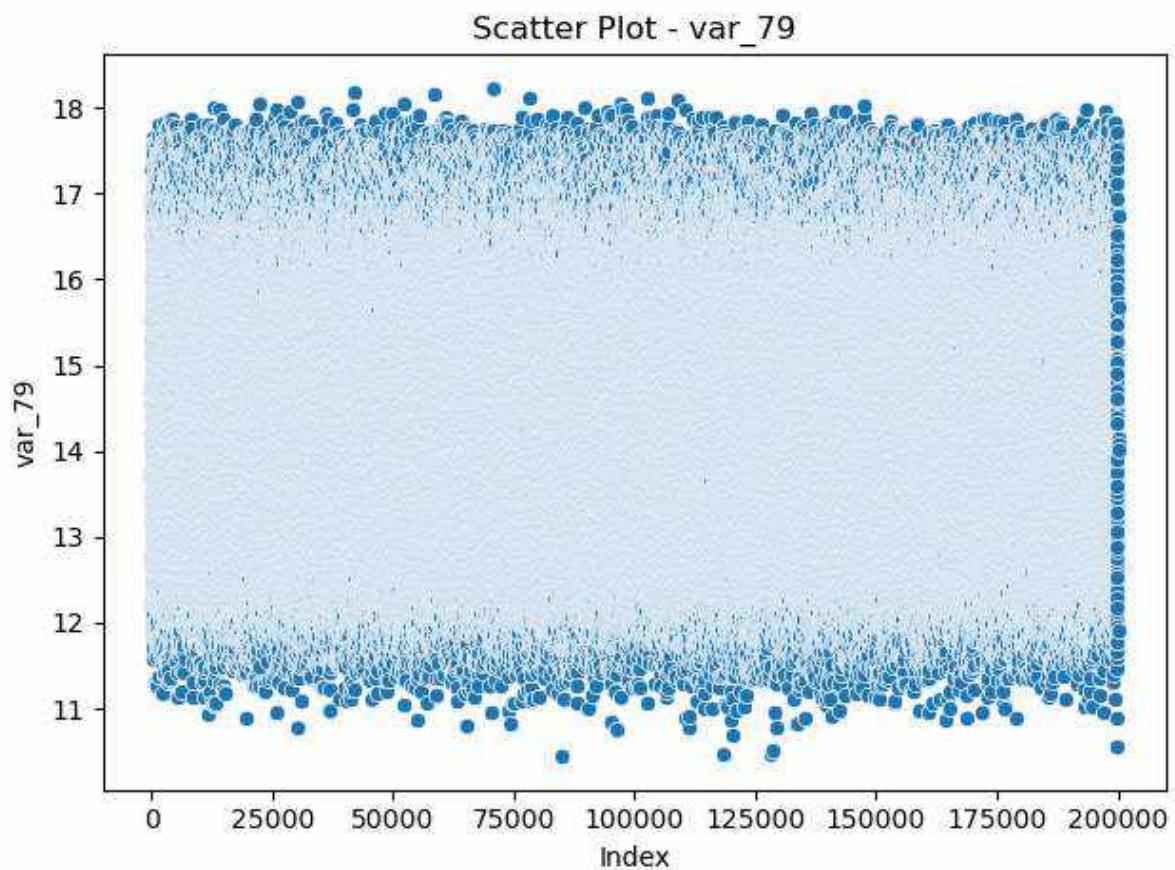
Scatter Plot - var\_75



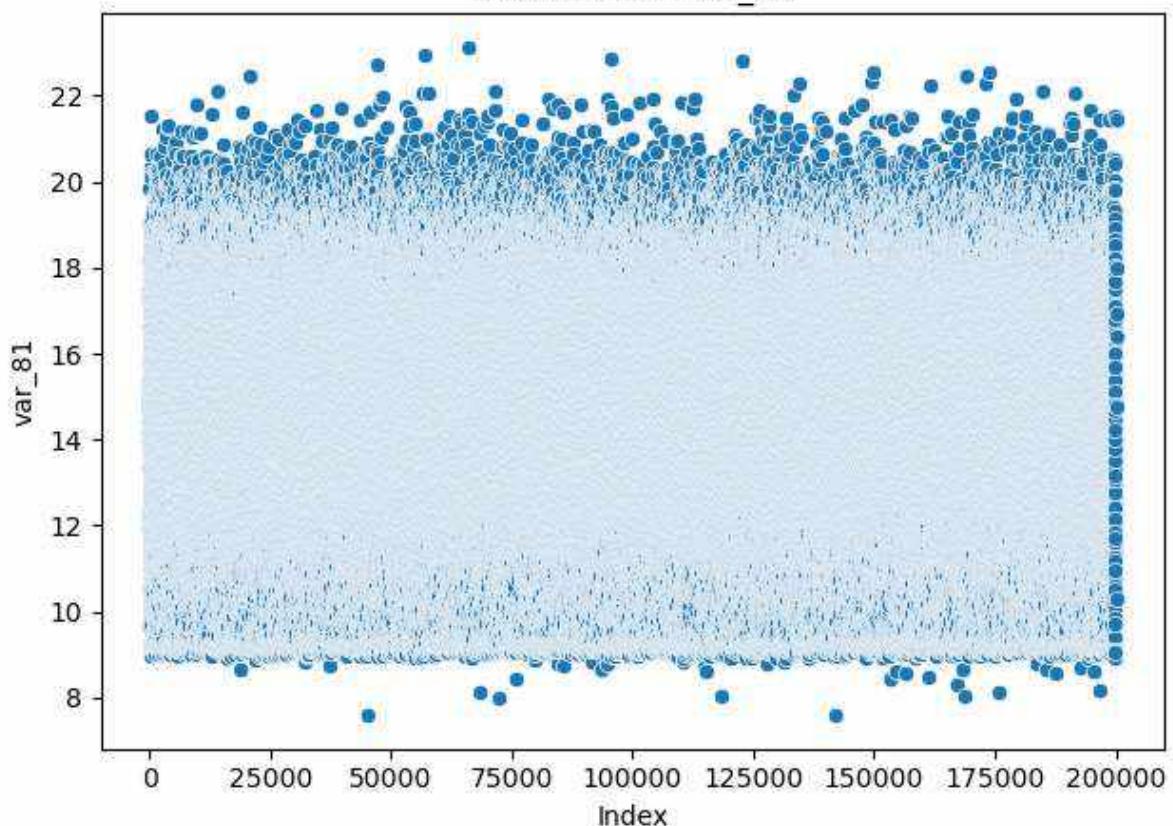
Scatter Plot - var\_76



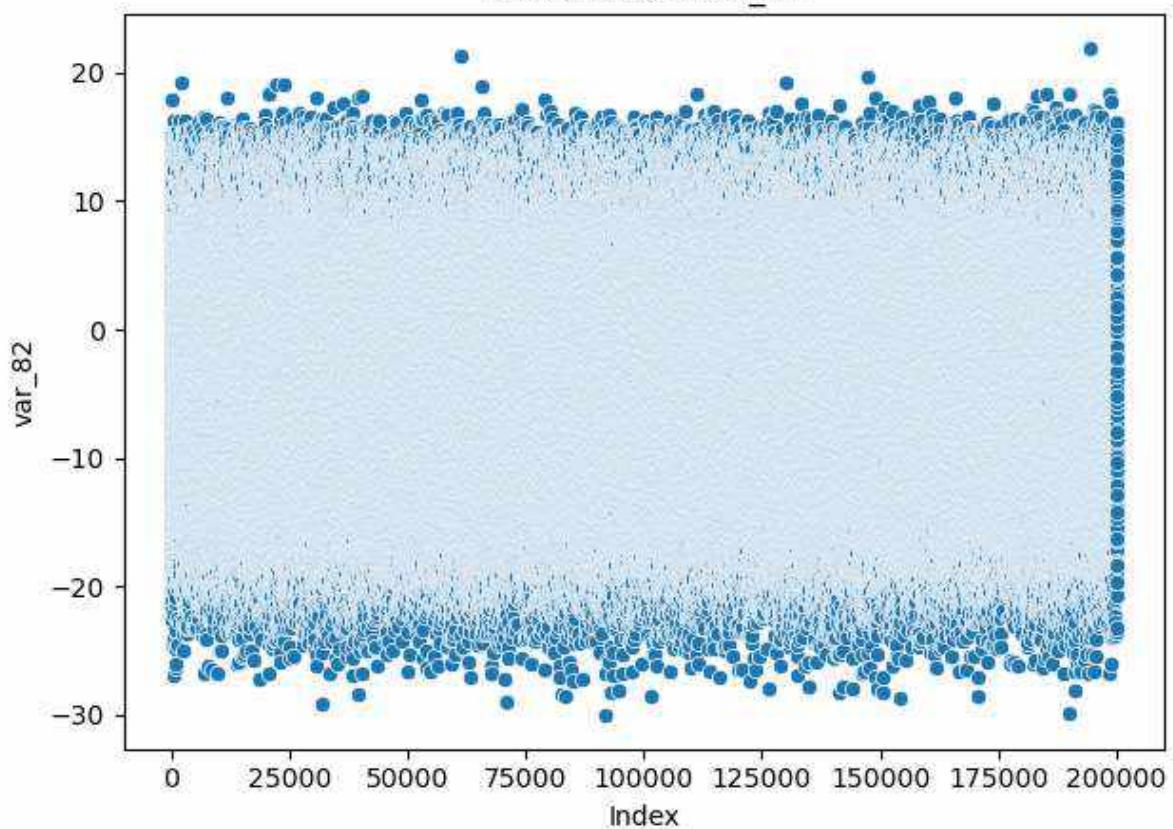


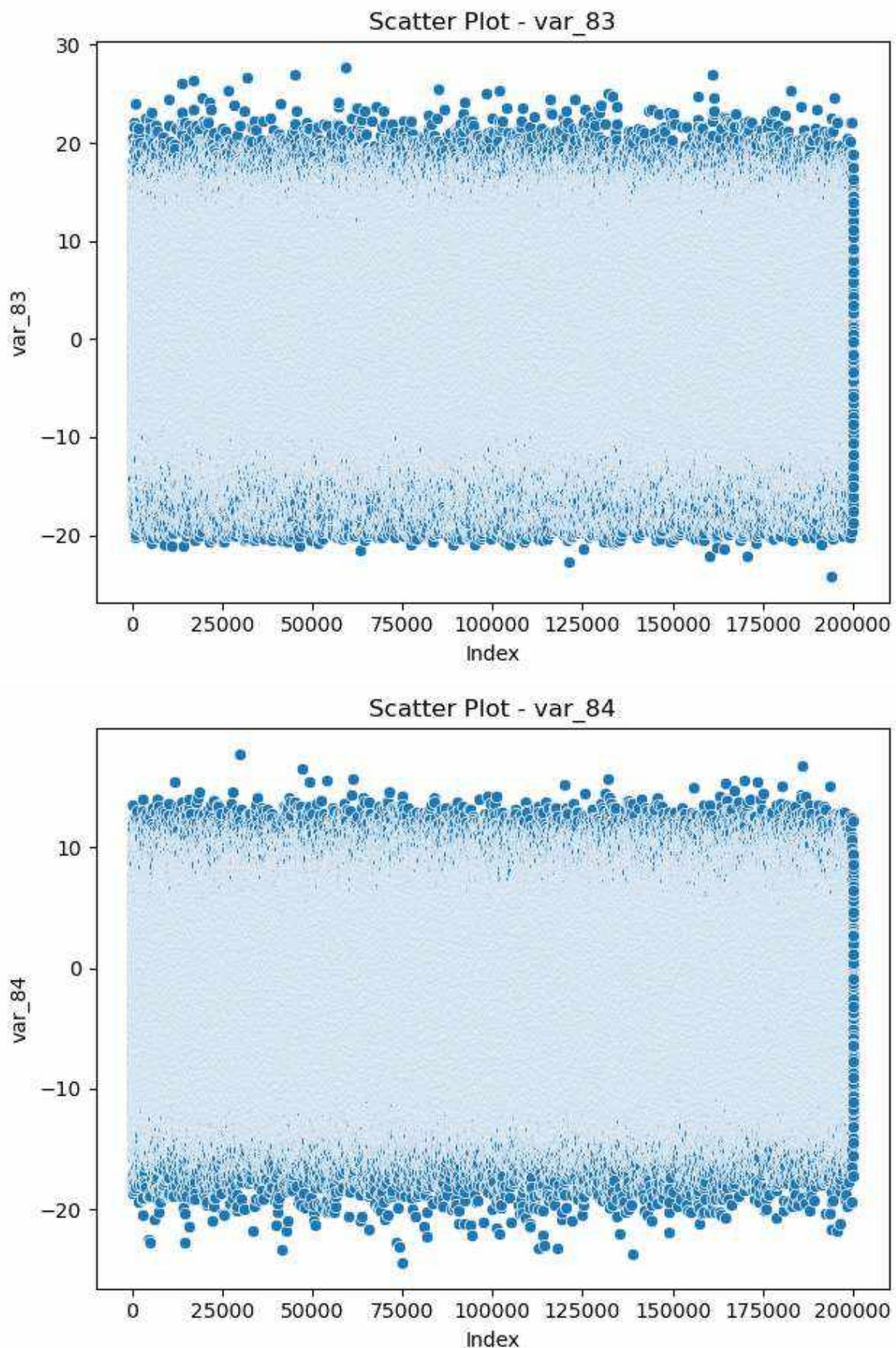


Scatter Plot - var\_81

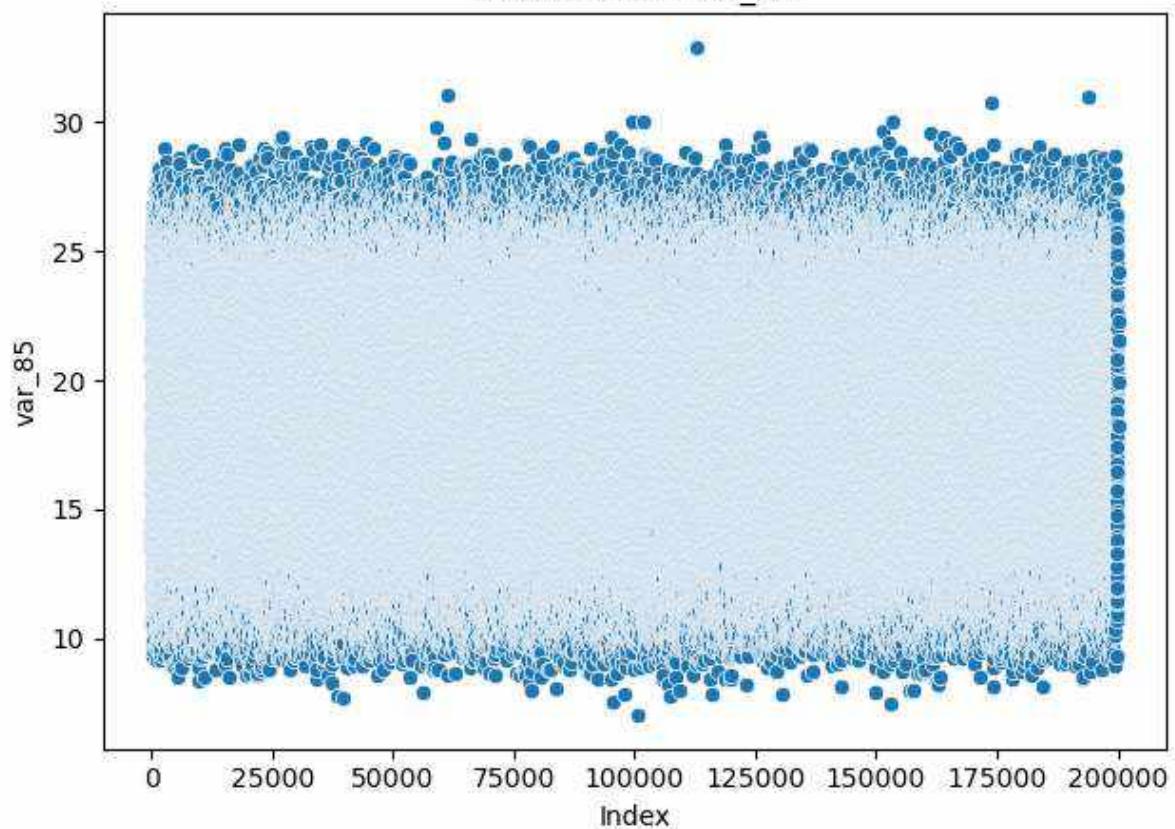


Scatter Plot - var\_82

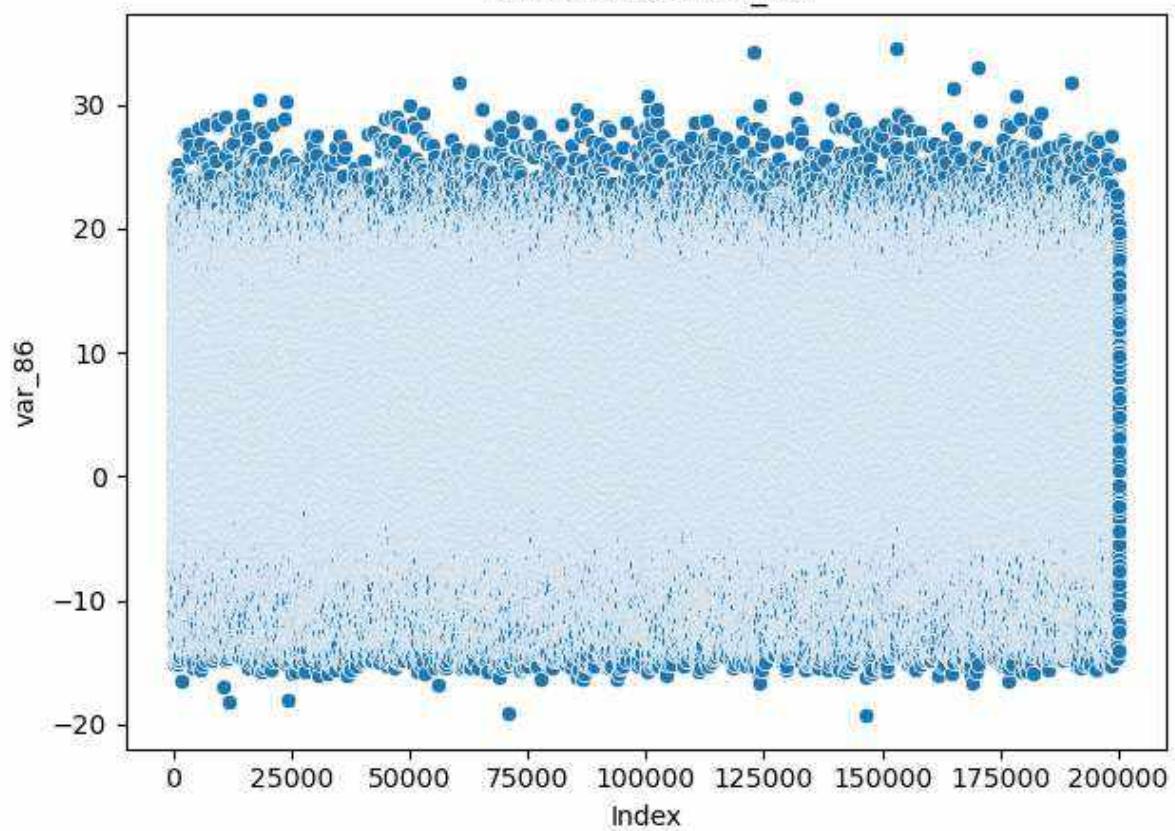




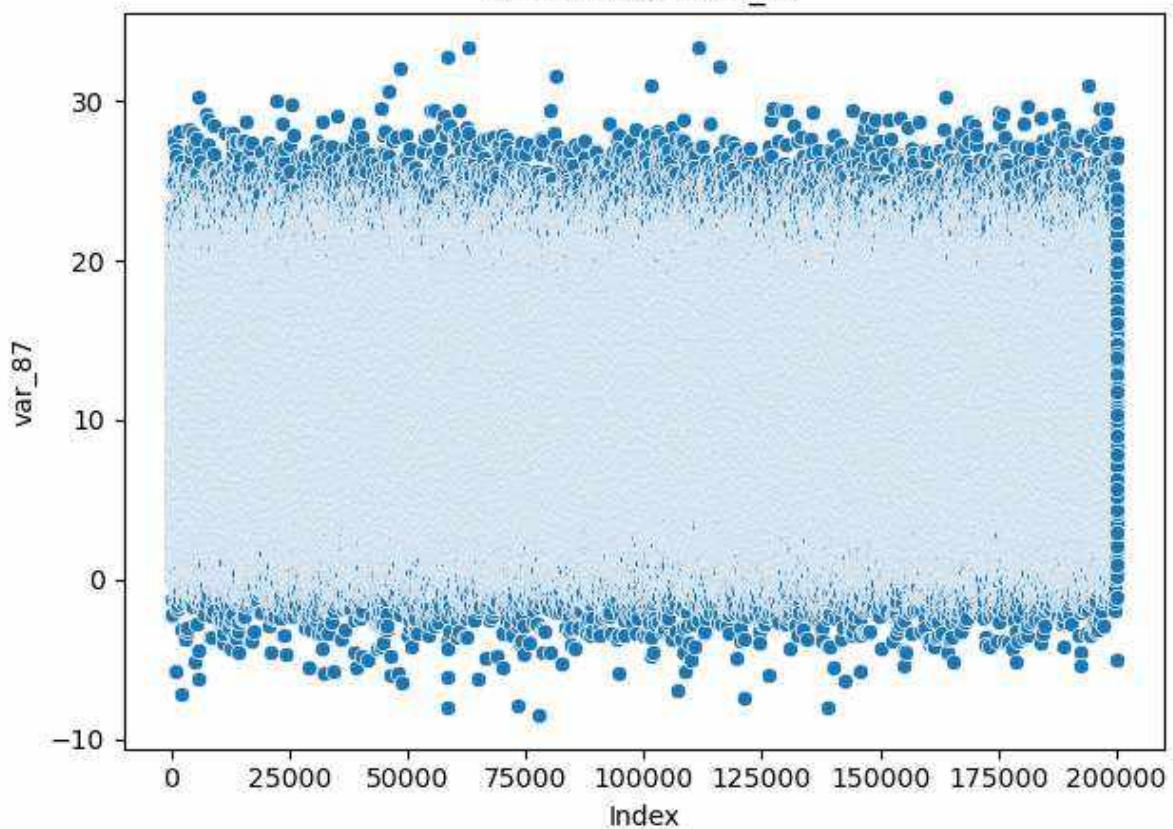
Scatter Plot - var\_85



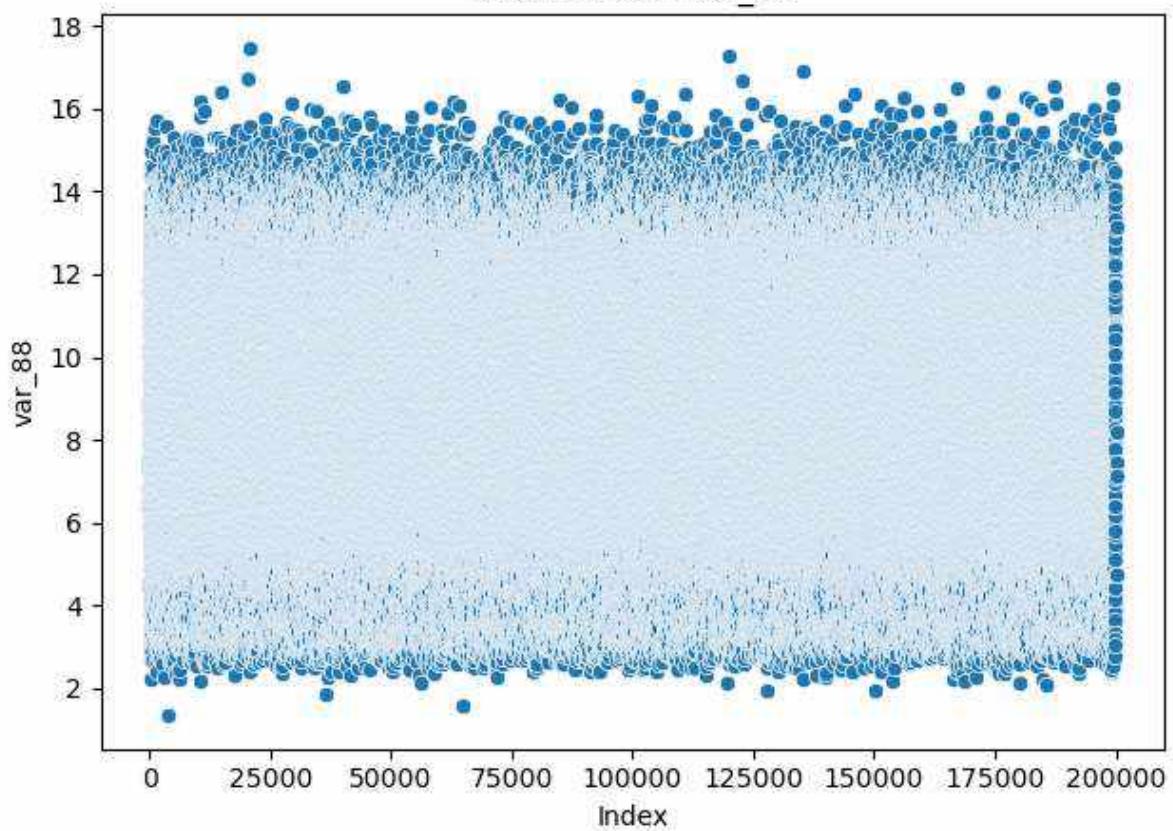
Scatter Plot - var\_86

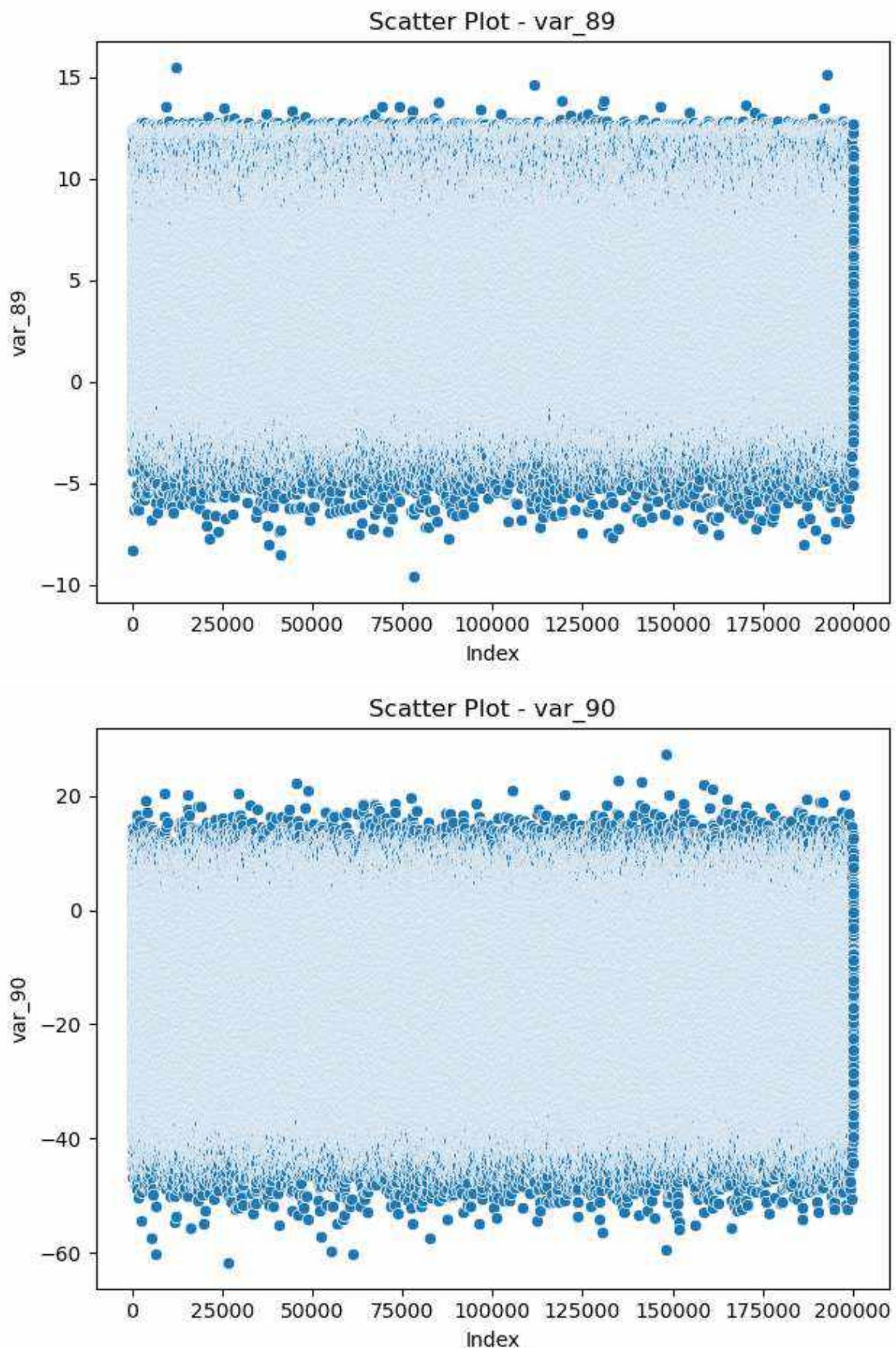


Scatter Plot - var\_87

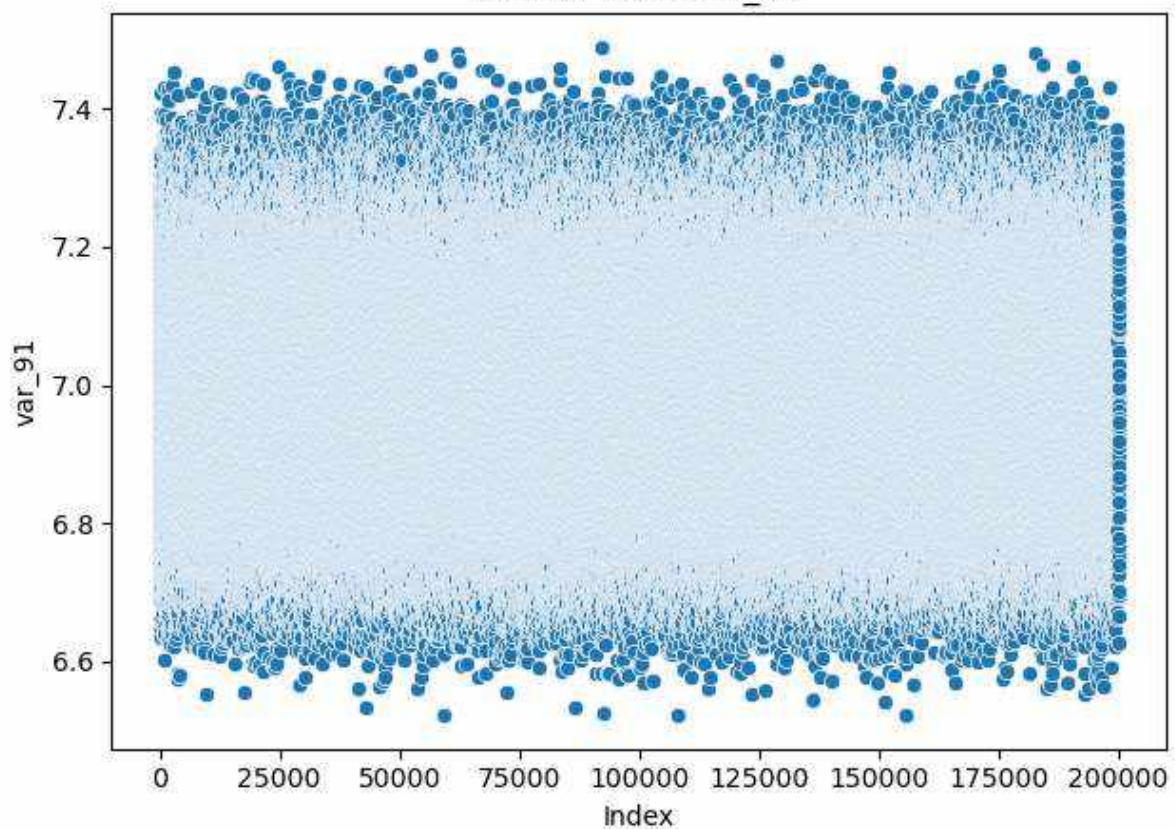


Scatter Plot - var\_88

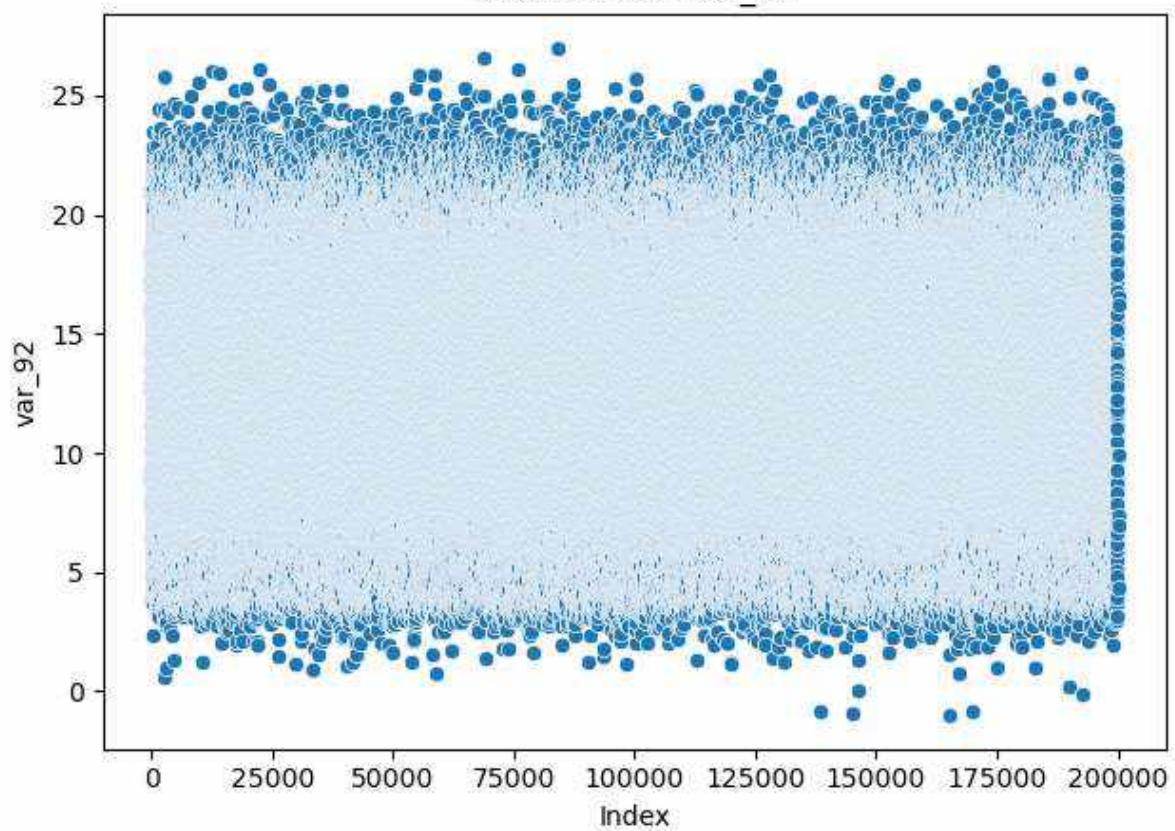




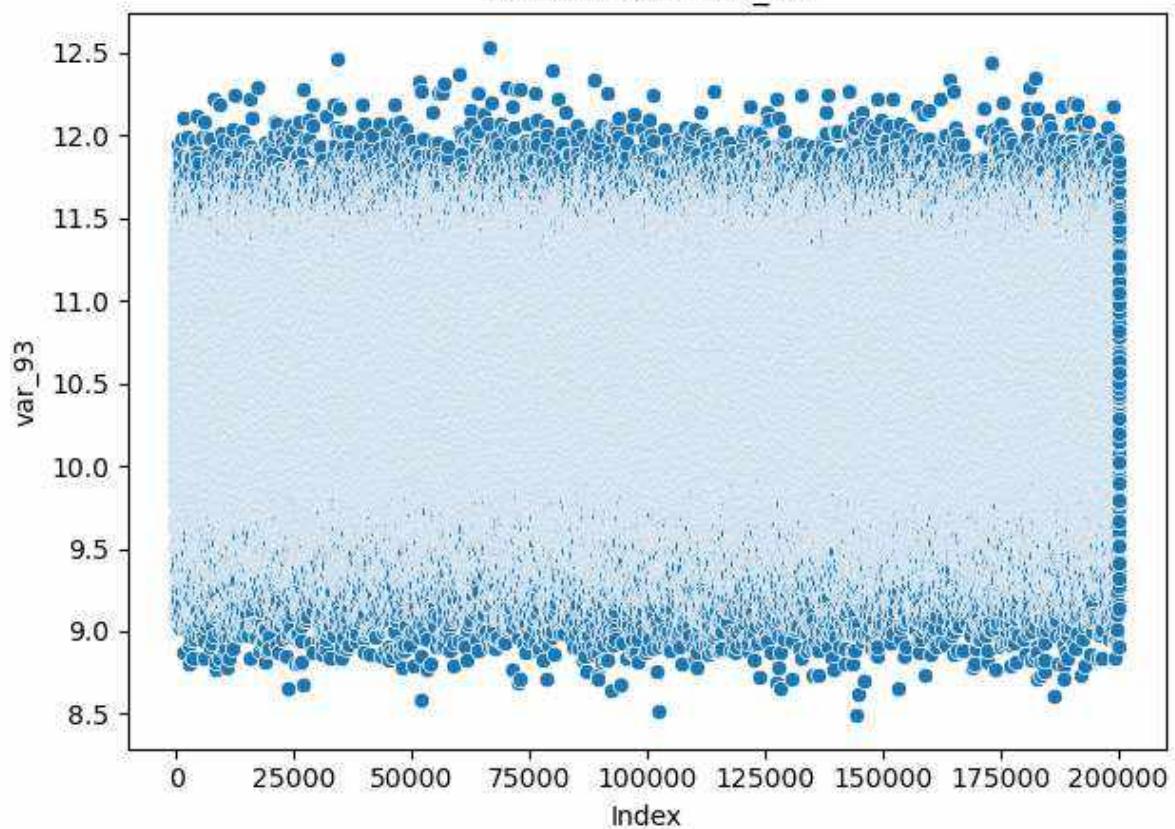
Scatter Plot - var\_91



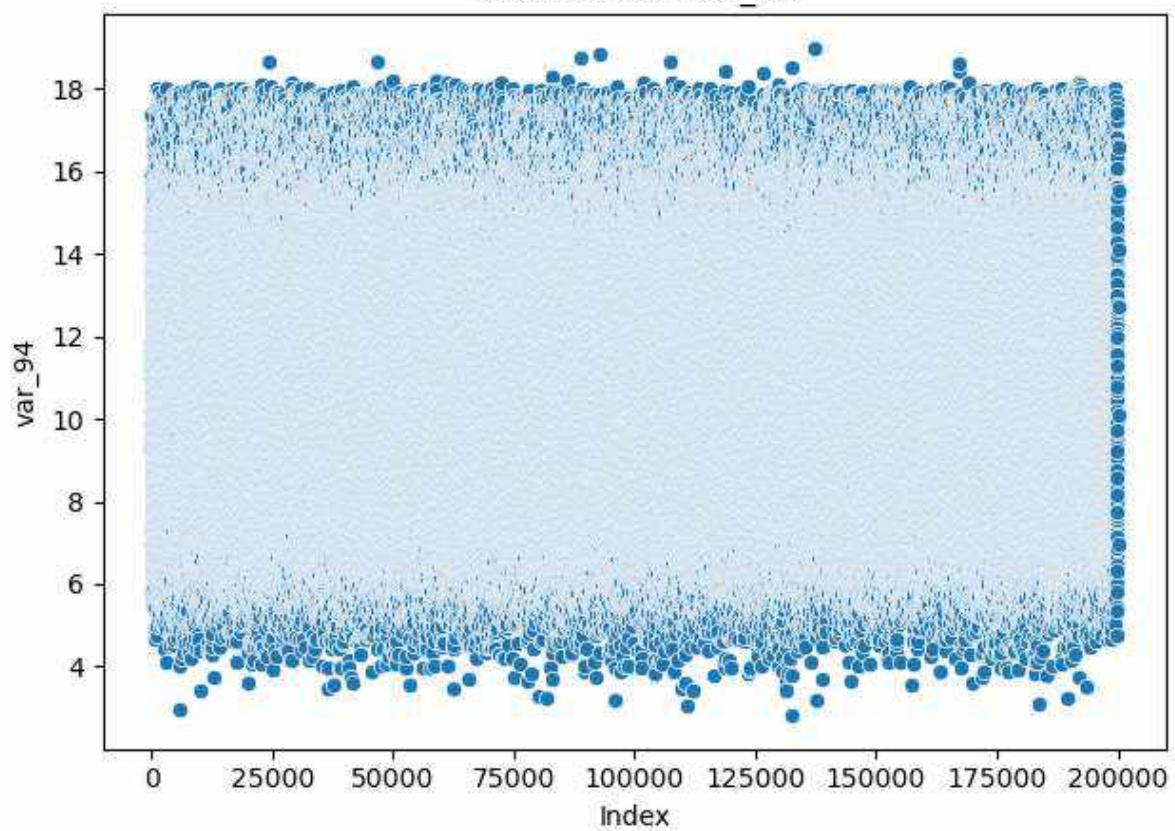
Scatter Plot - var\_92

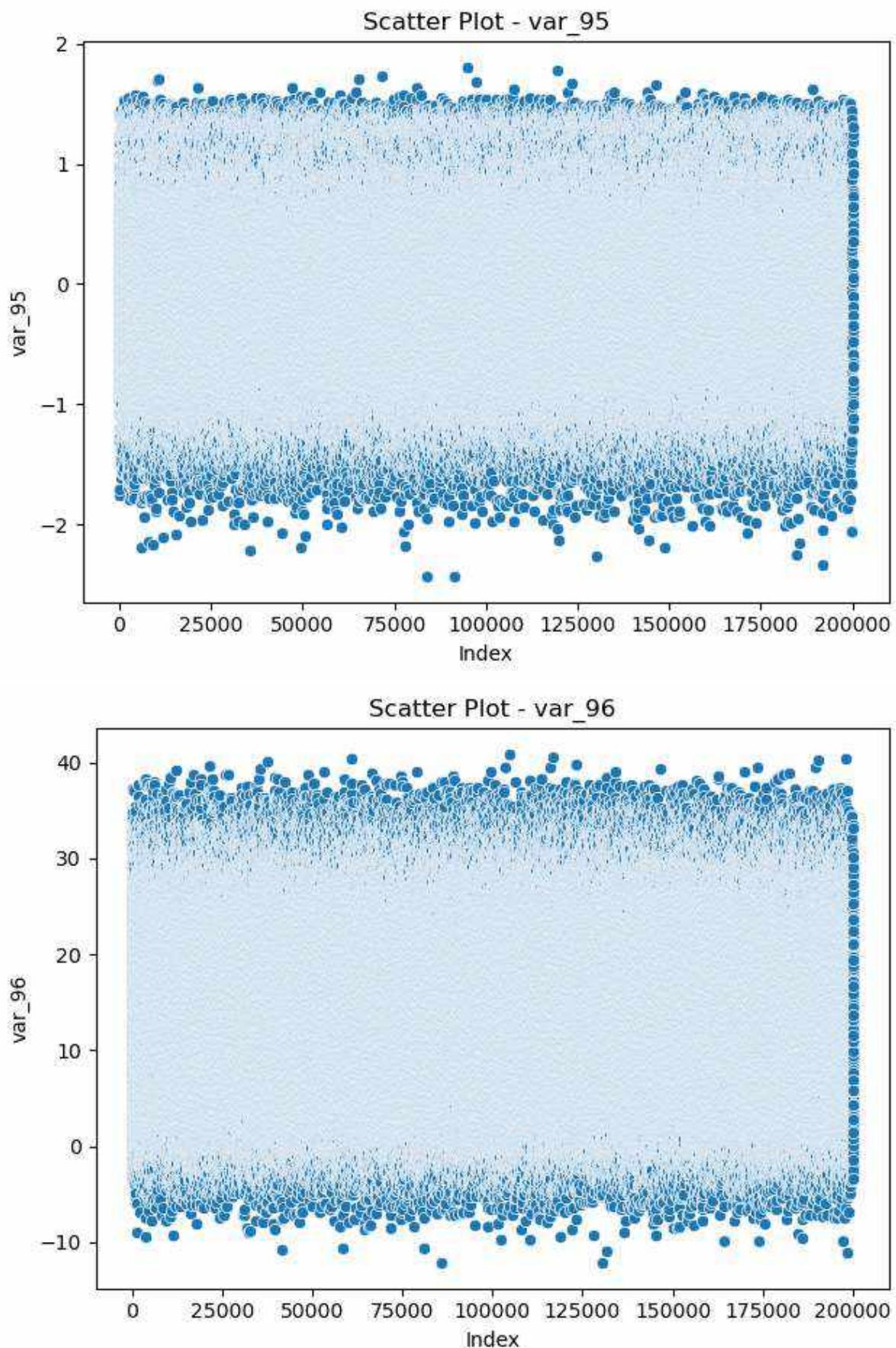


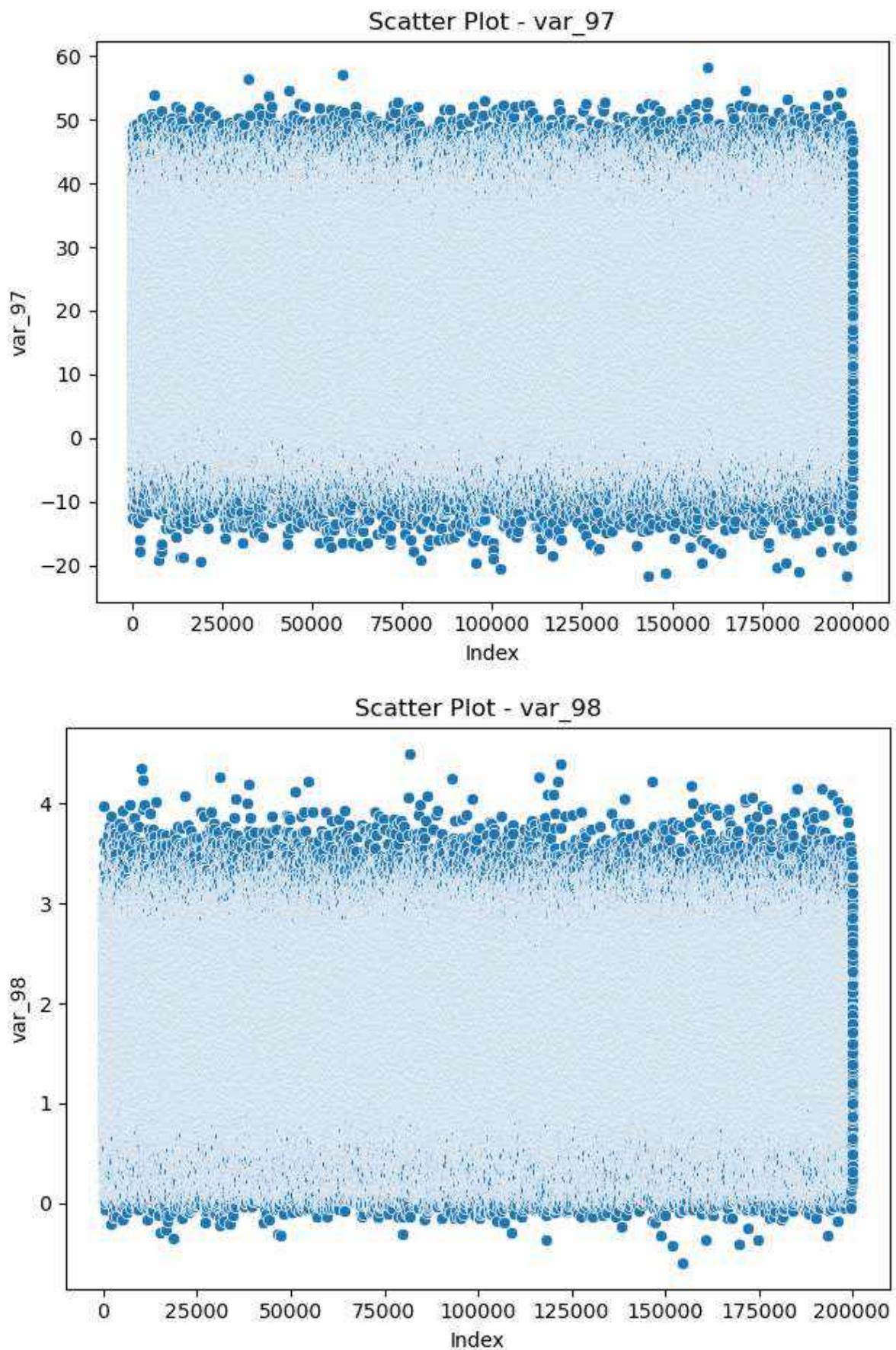
Scatter Plot - var\_93



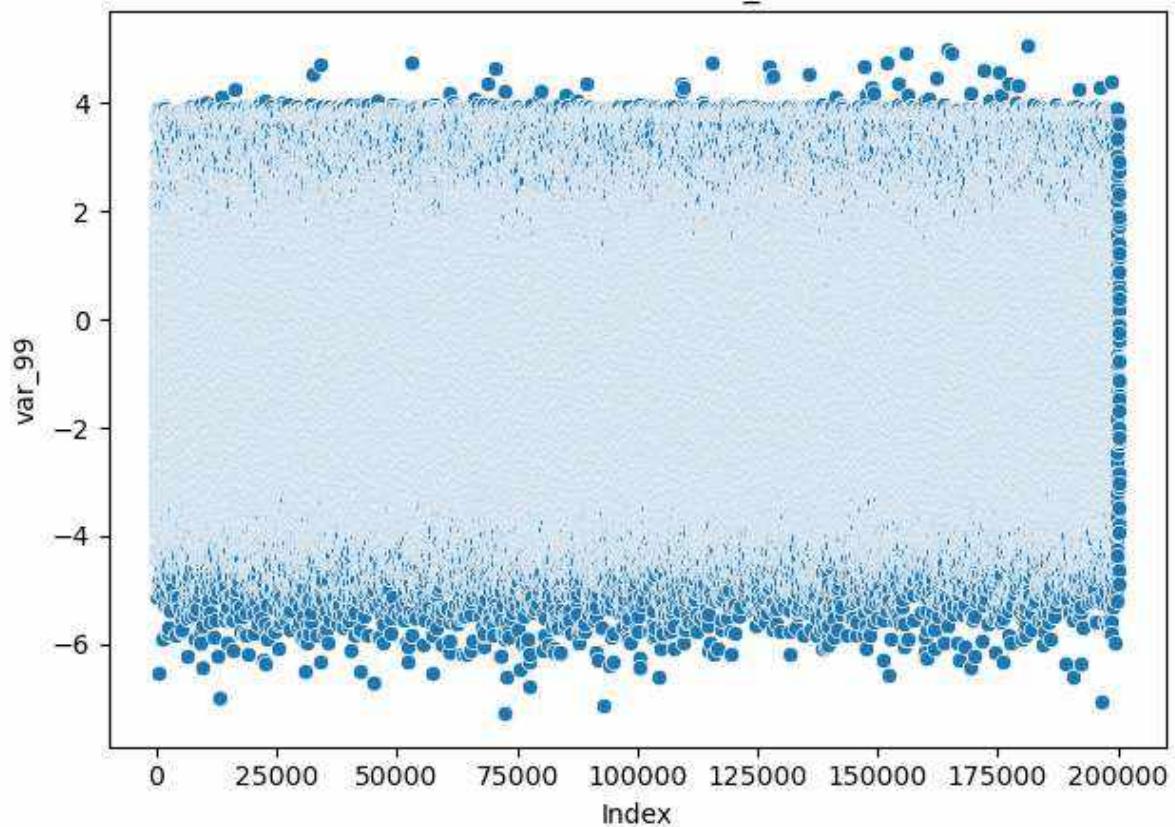
Scatter Plot - var\_94



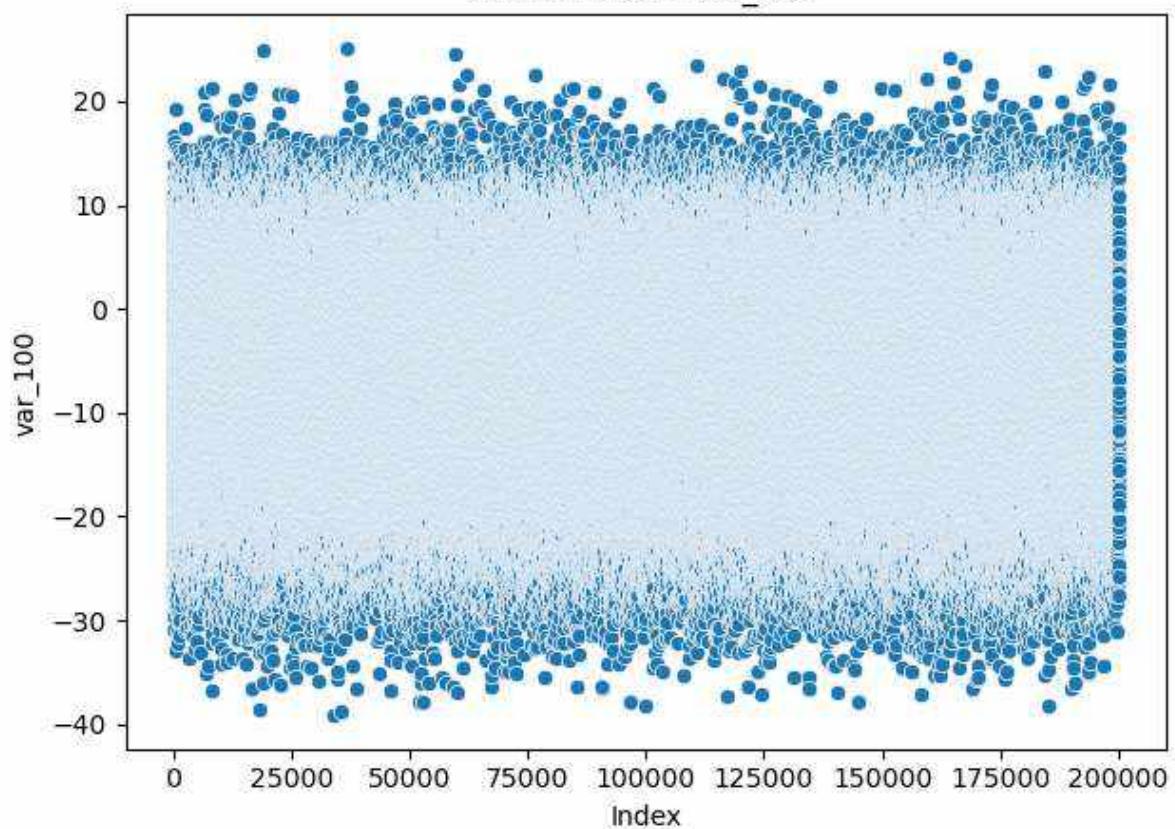




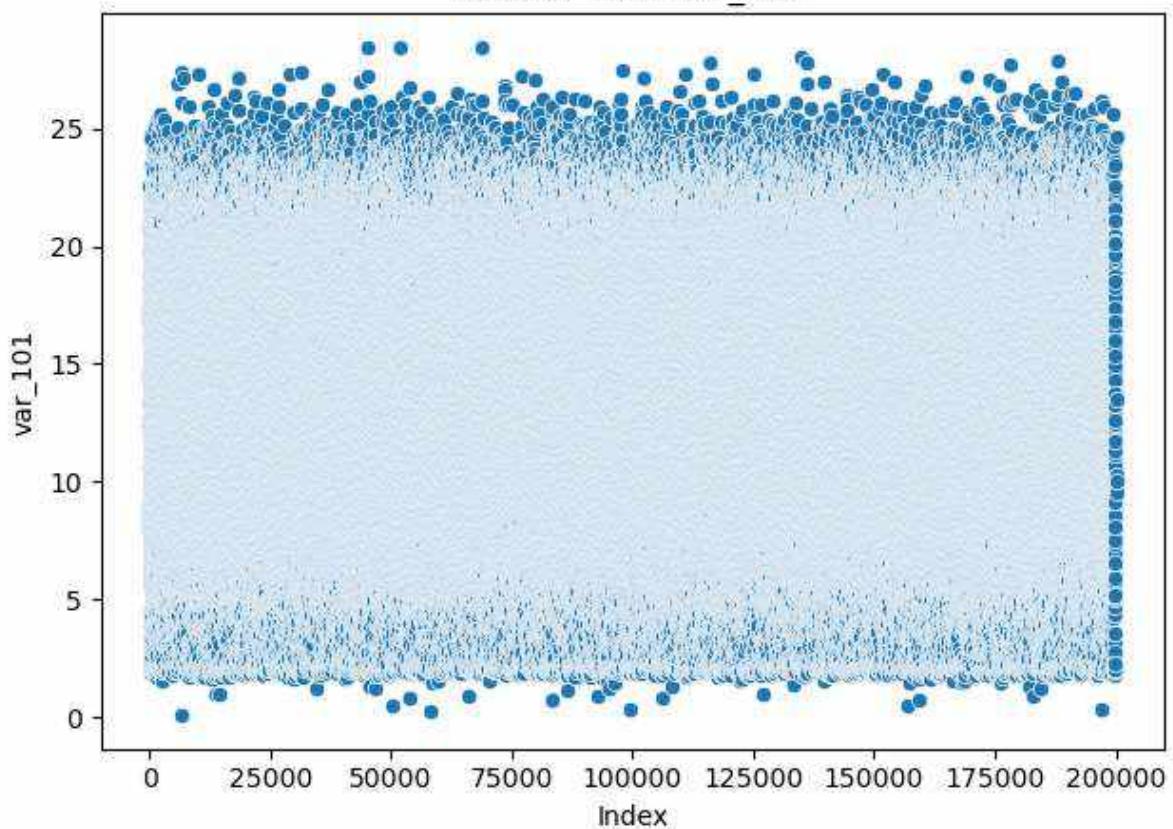
Scatter Plot - var\_99



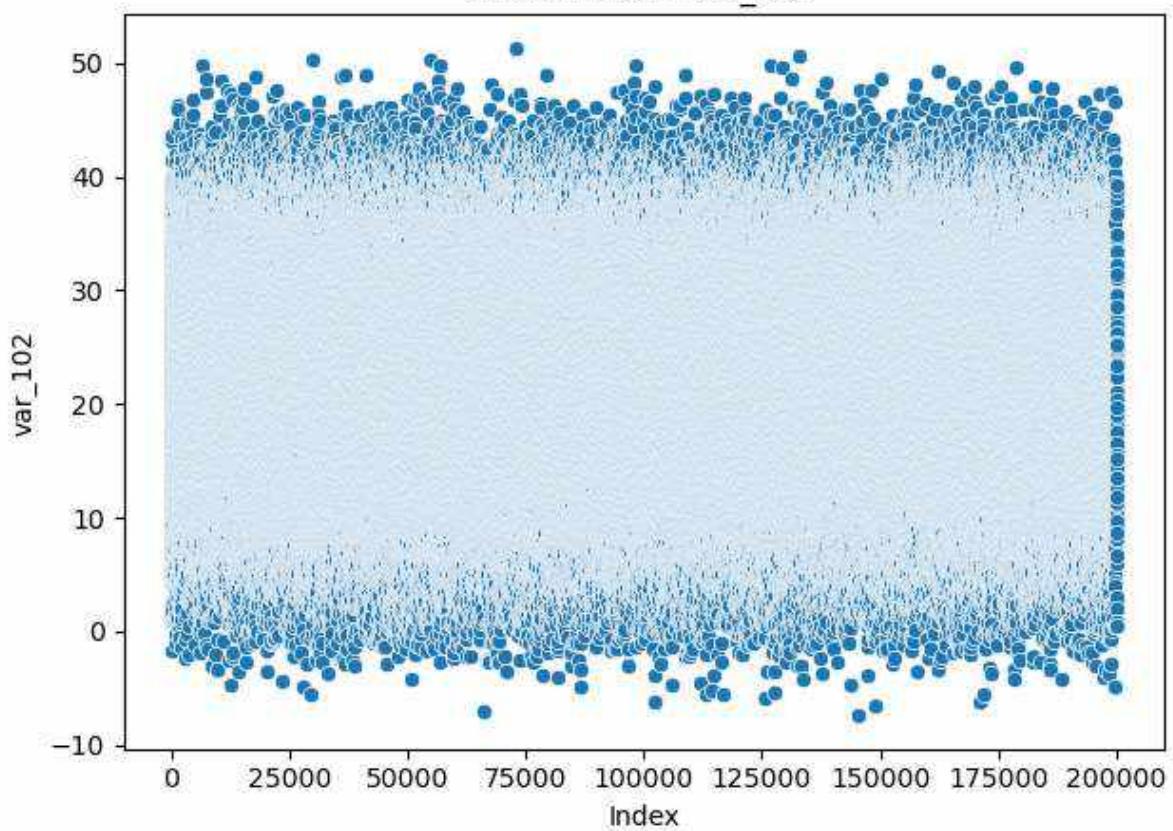
Scatter Plot - var\_100



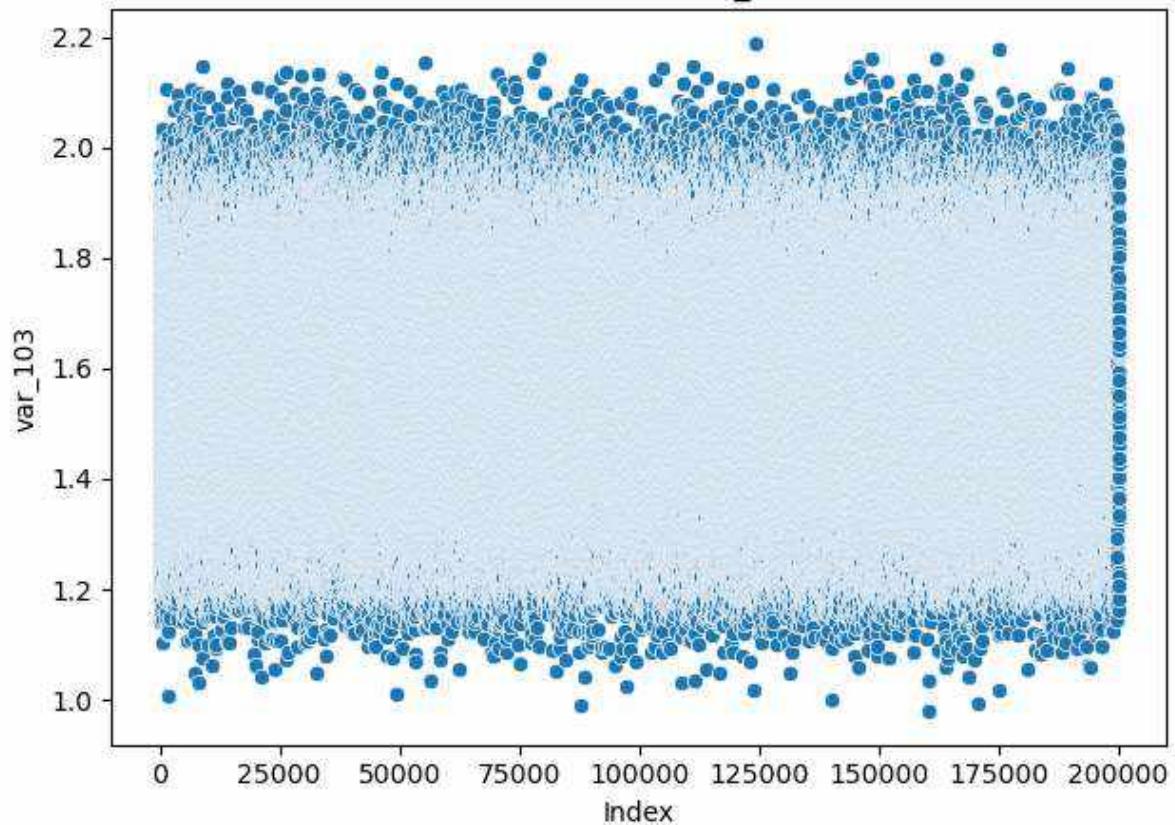
Scatter Plot - var\_101



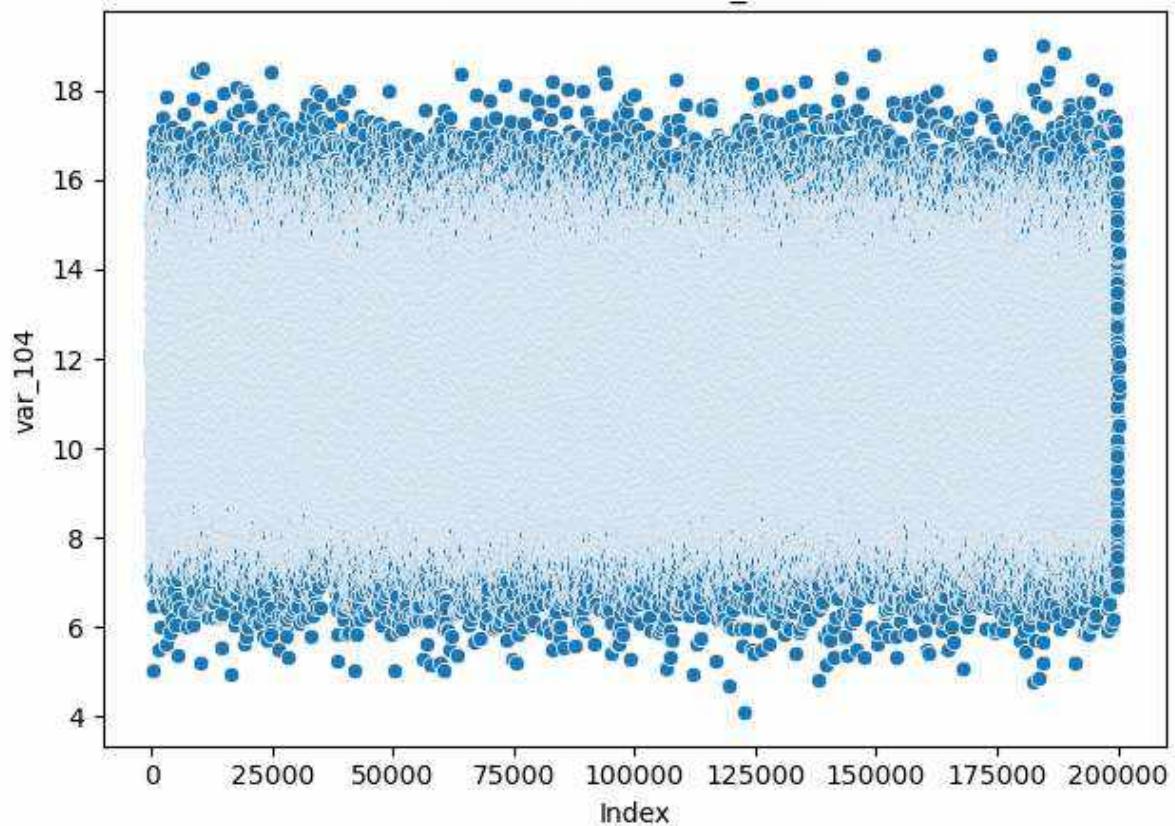
Scatter Plot - var\_102



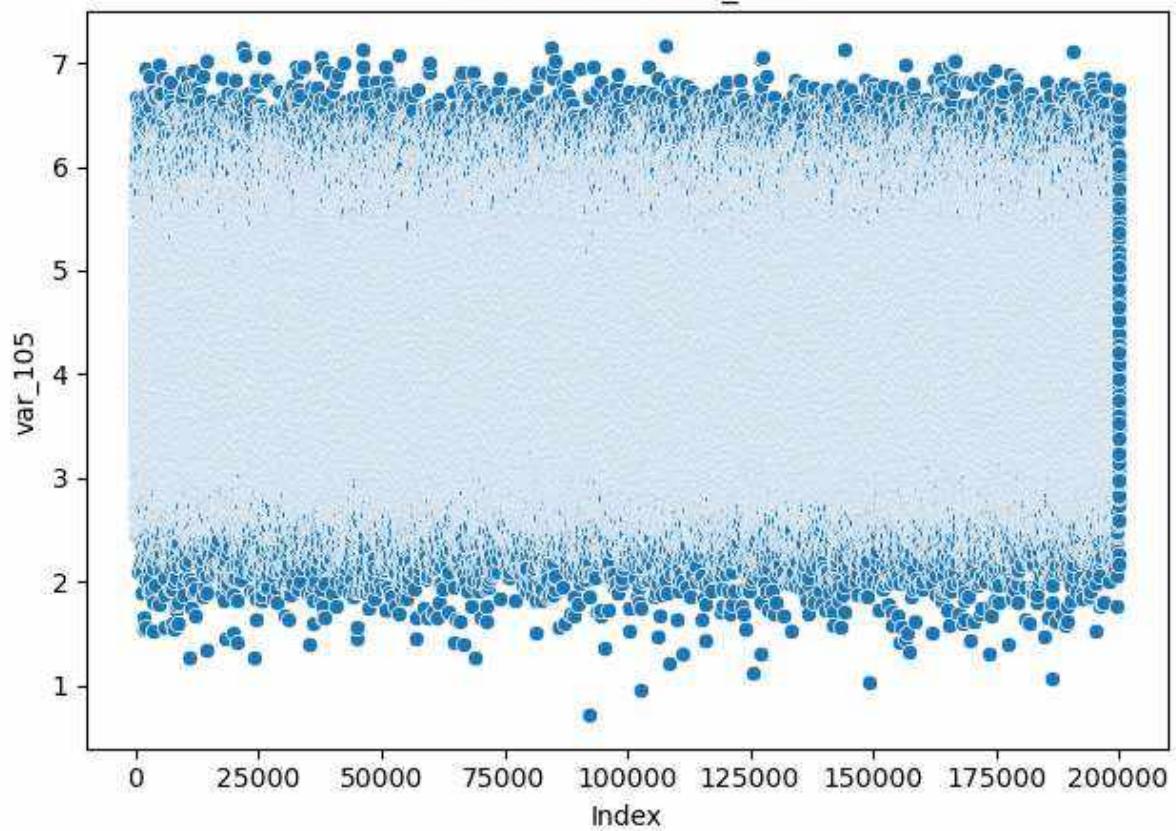
Scatter Plot - var\_103



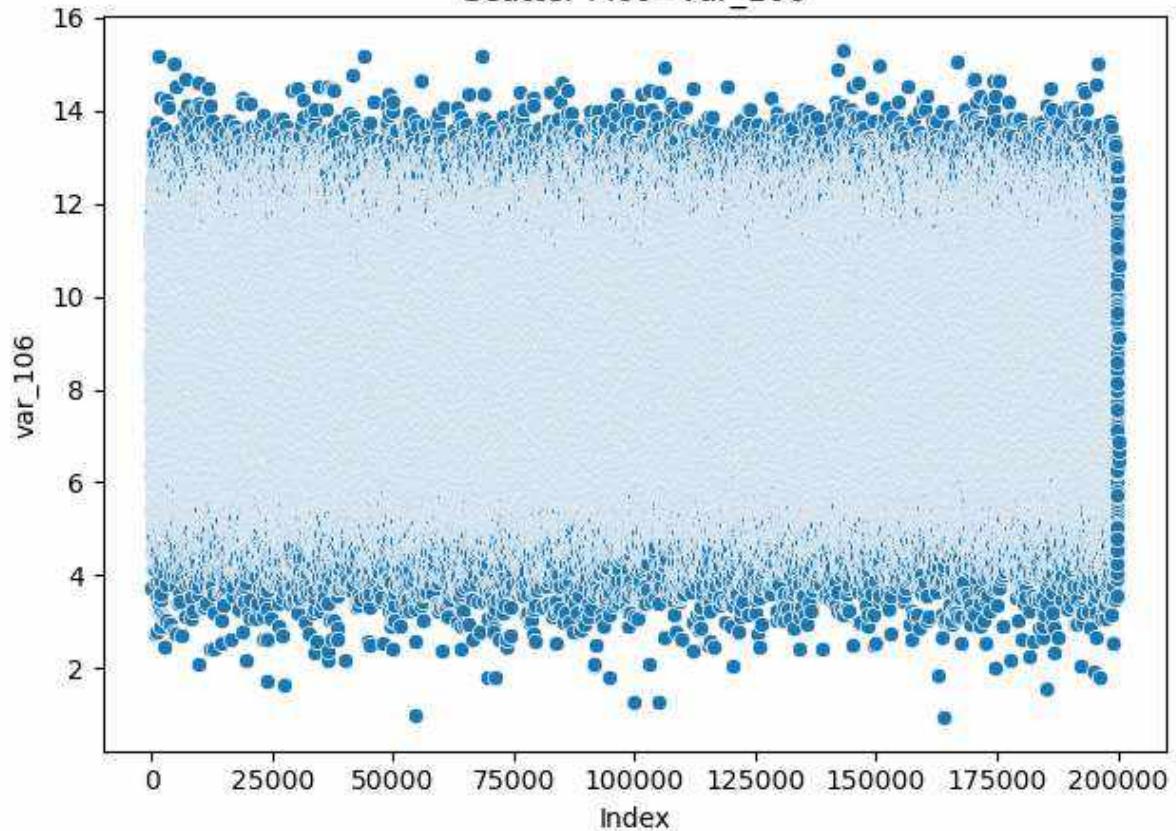
Scatter Plot - var\_104



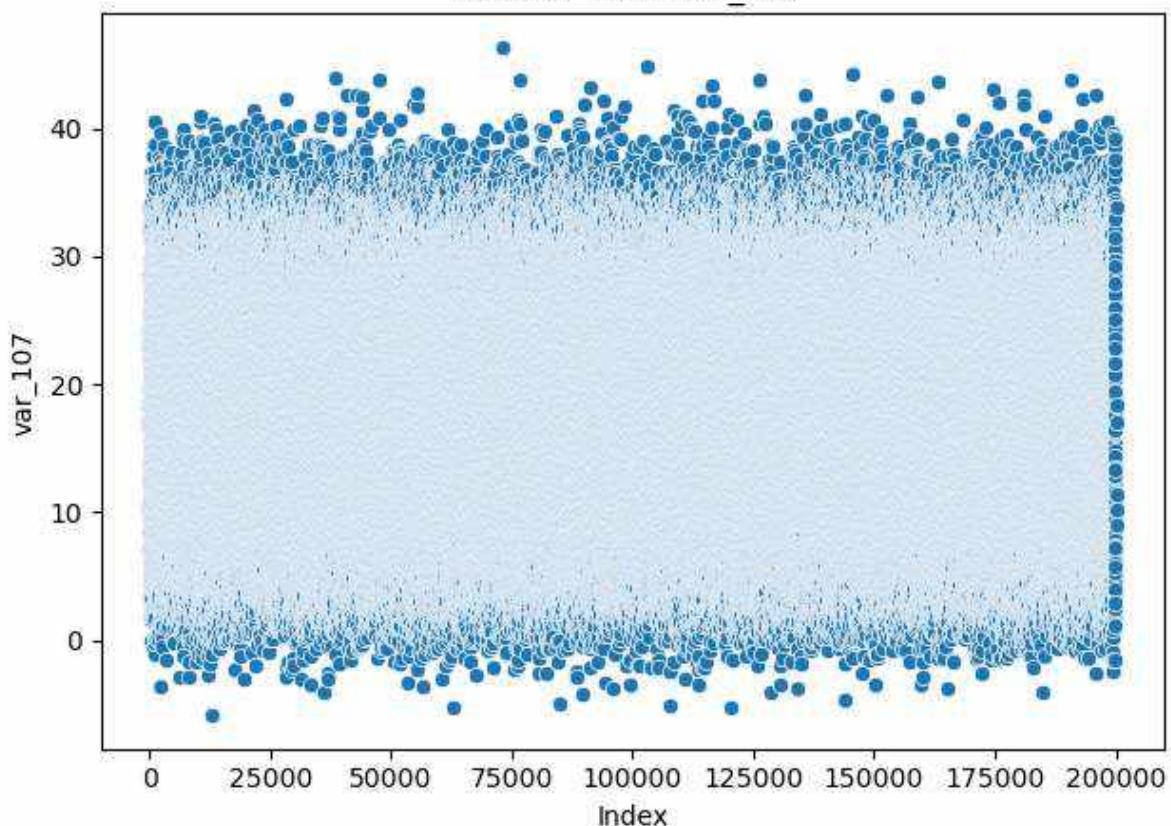
Scatter Plot - var\_105



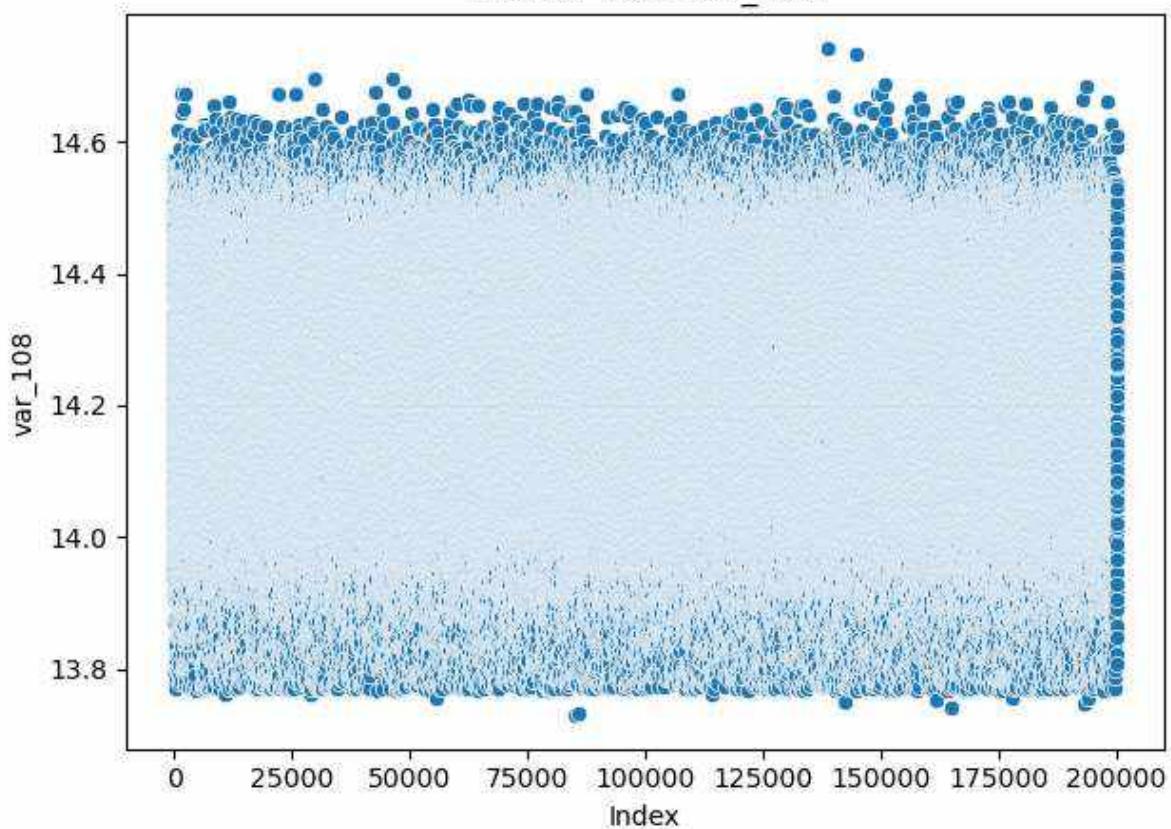
Scatter Plot - var\_106



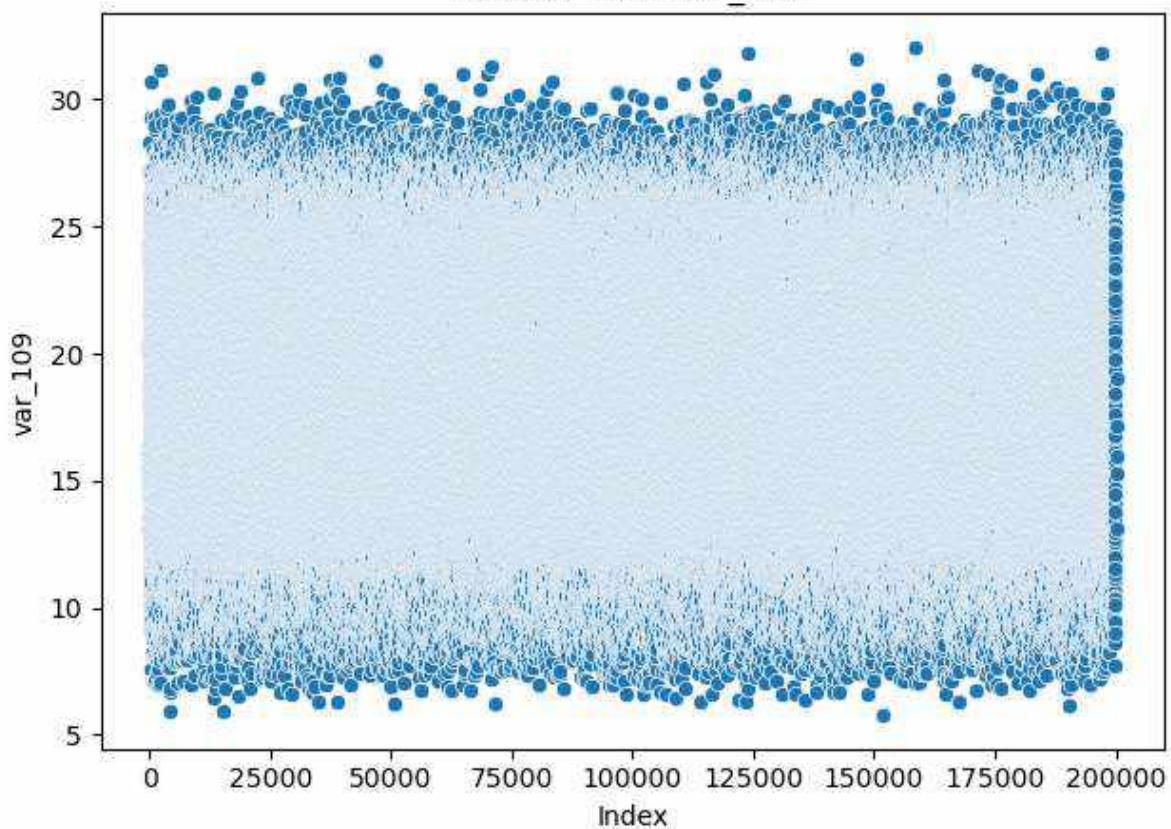
Scatter Plot - var\_107



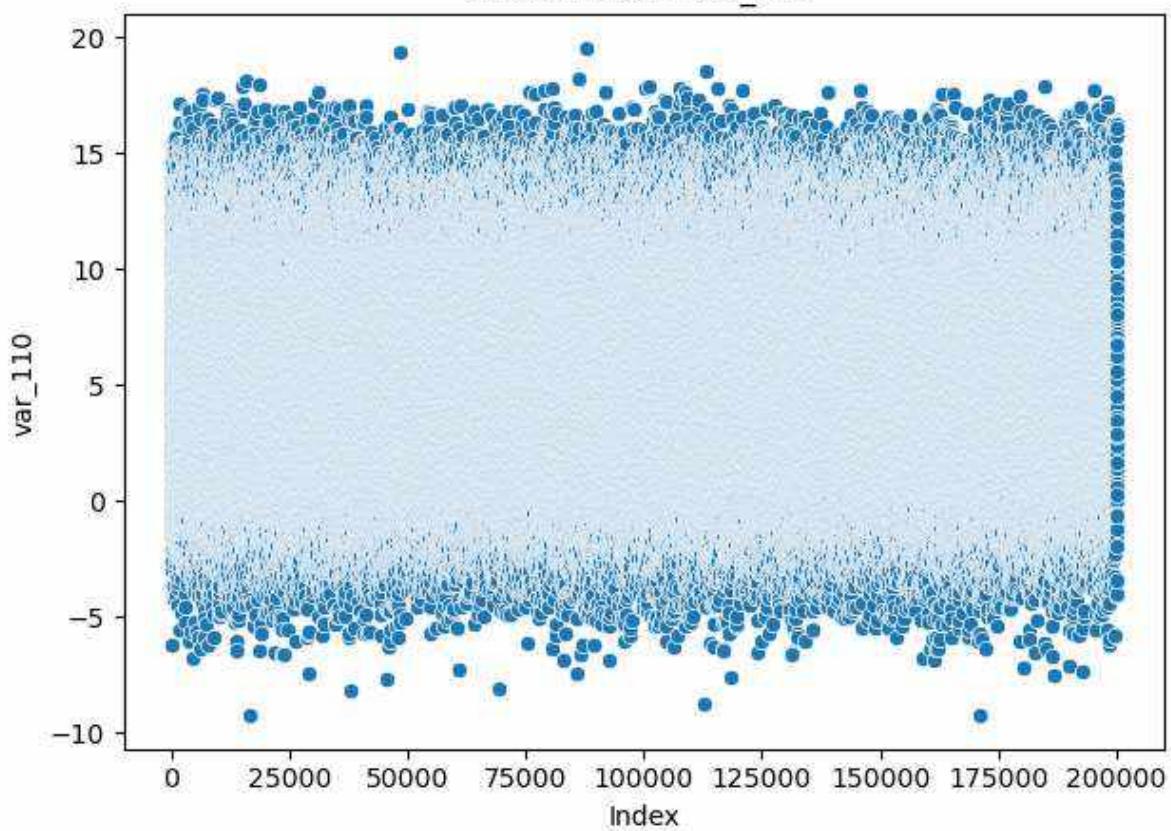
Scatter Plot - var\_108

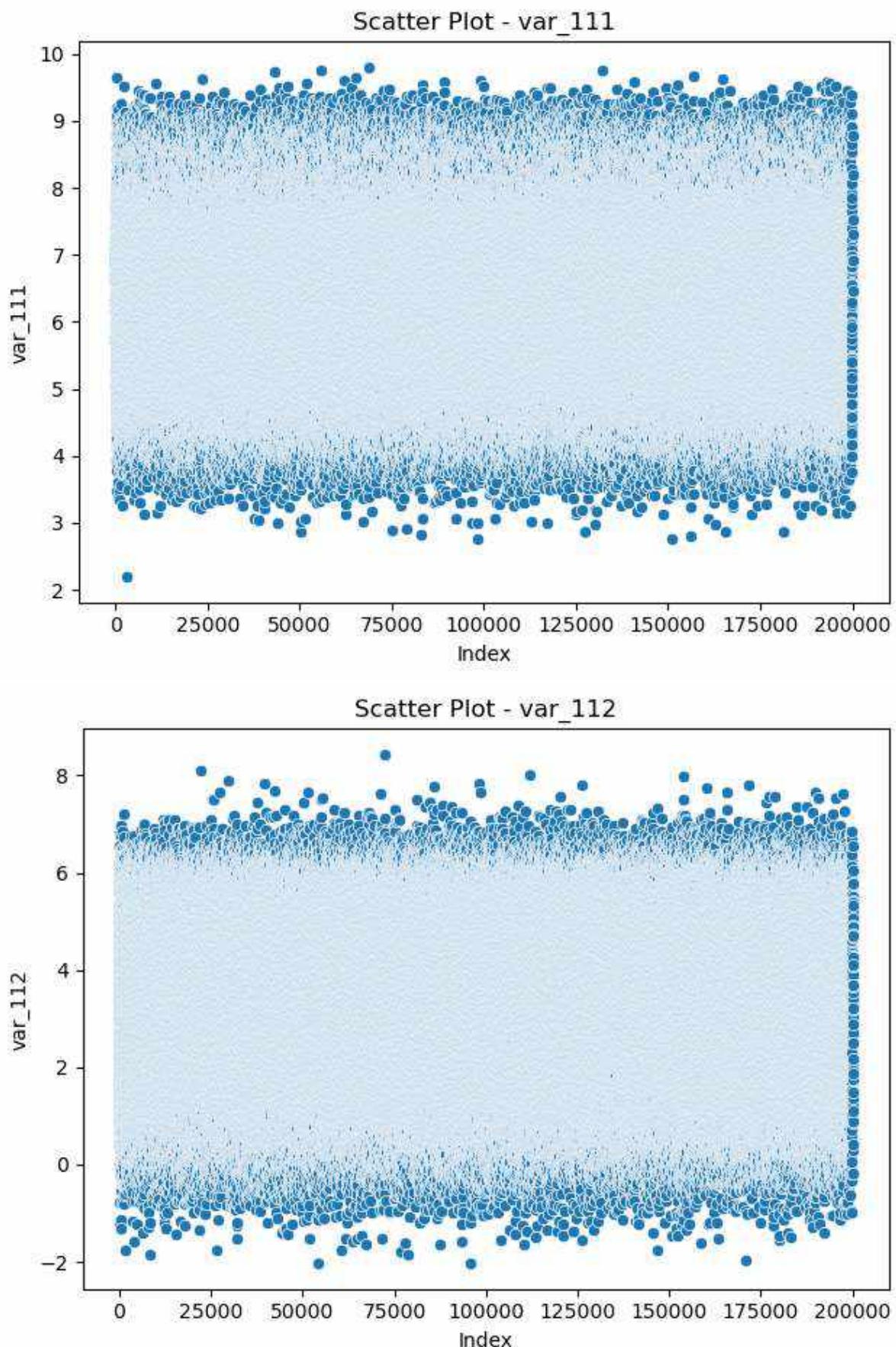


Scatter Plot - var\_109

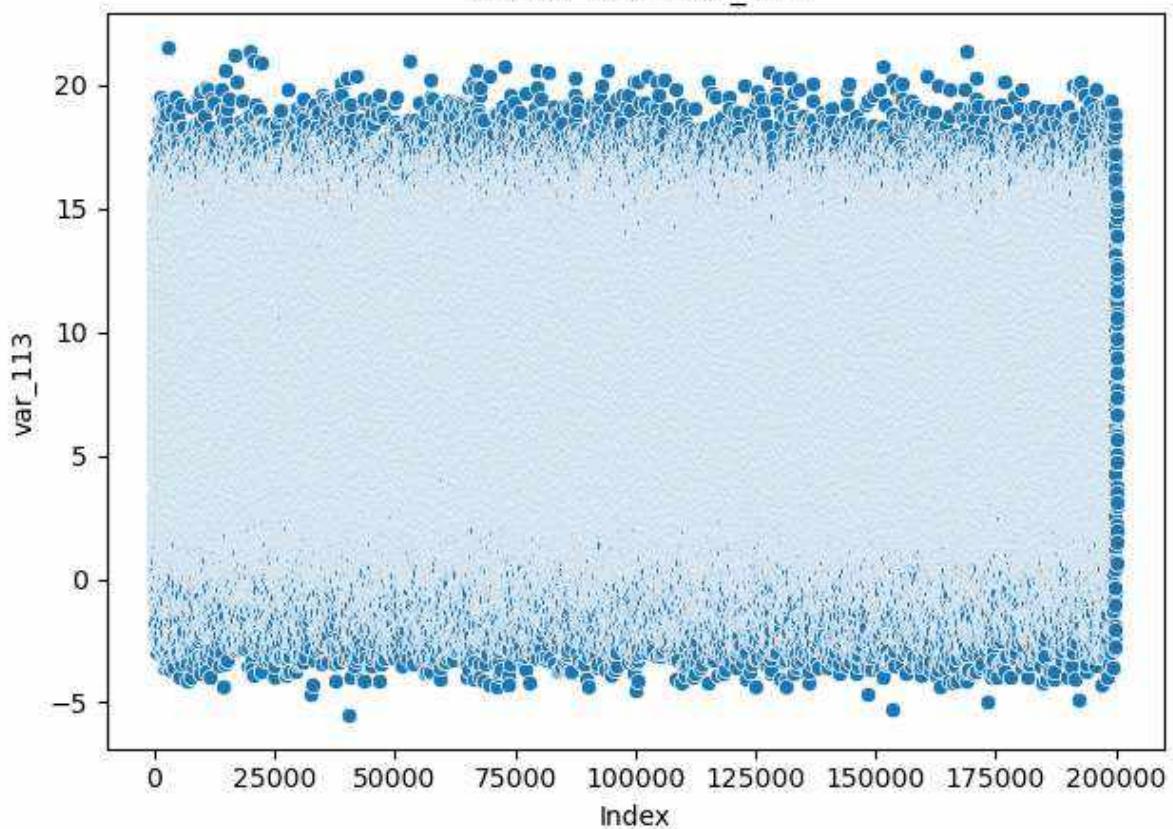


Scatter Plot - var\_110

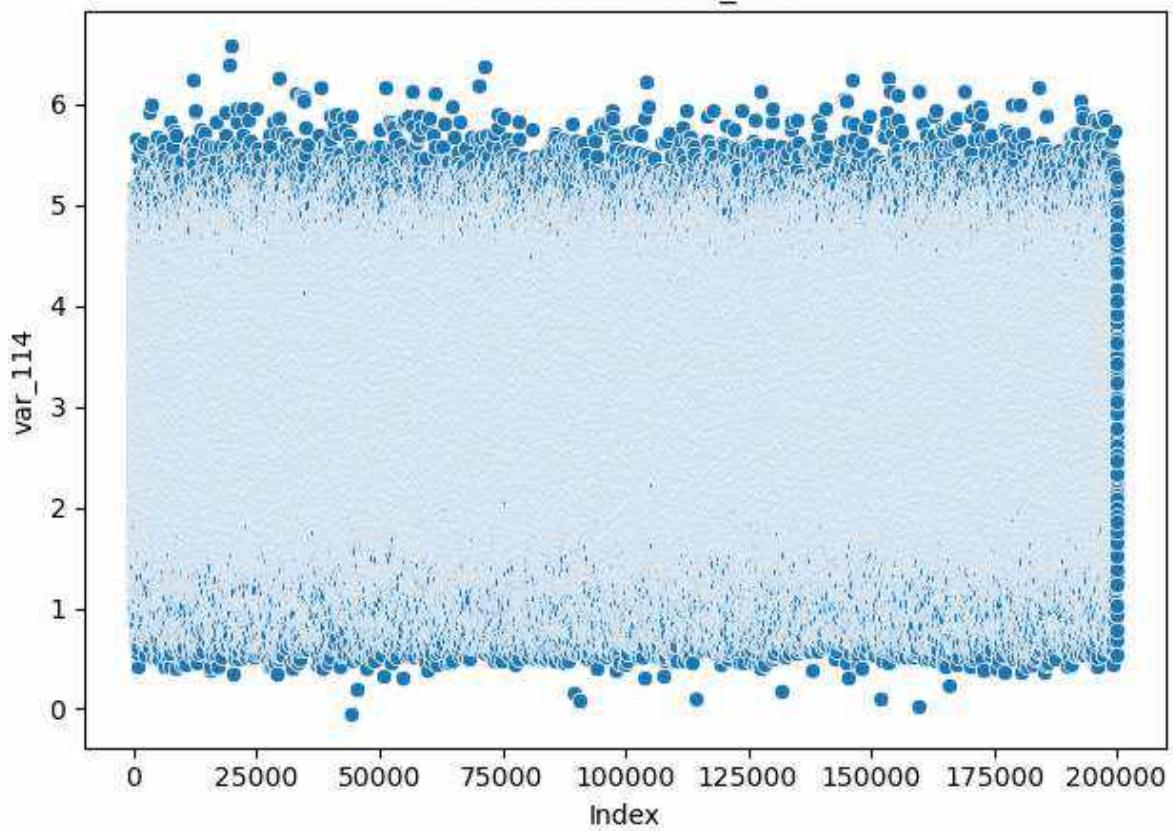


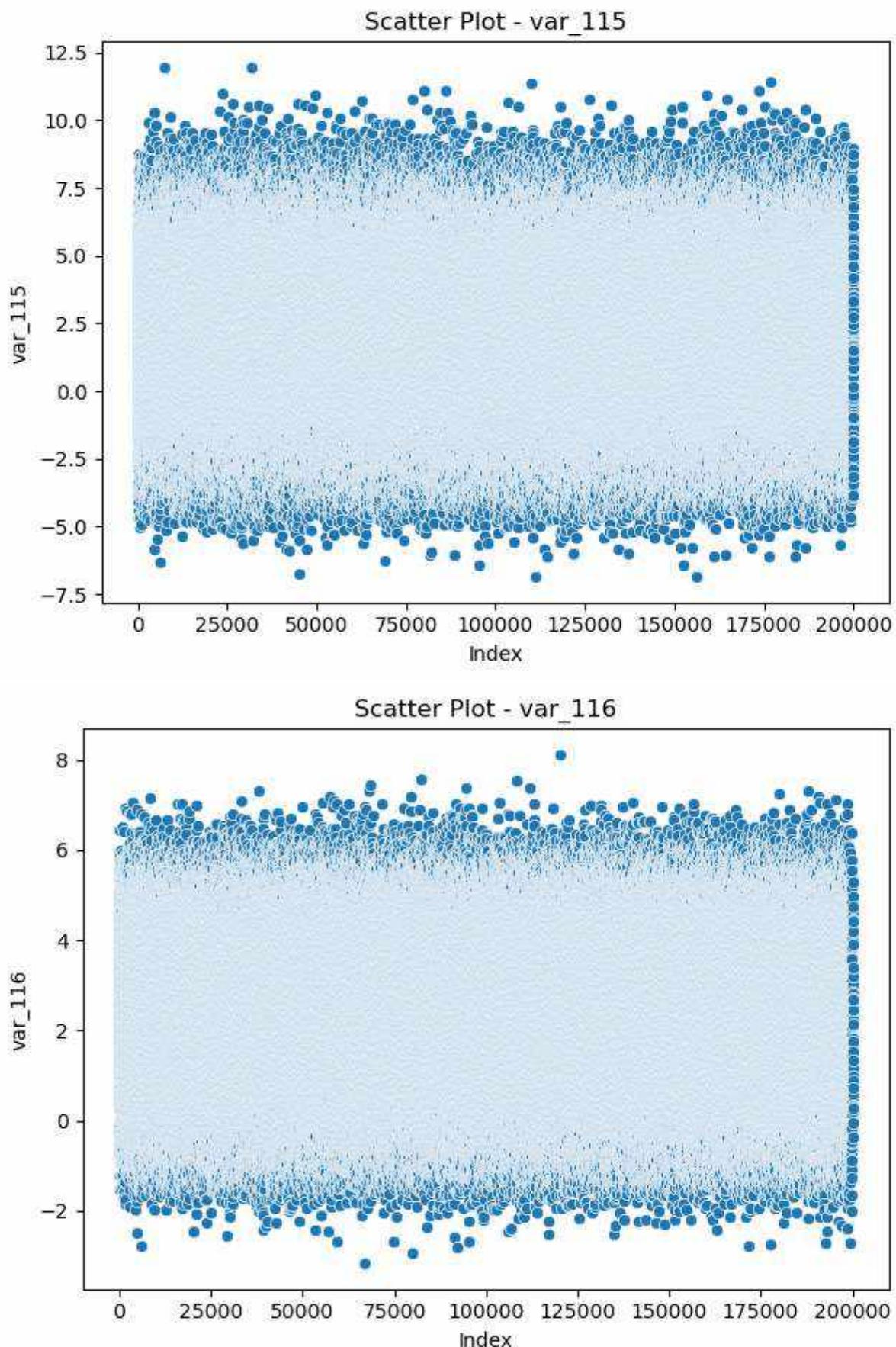


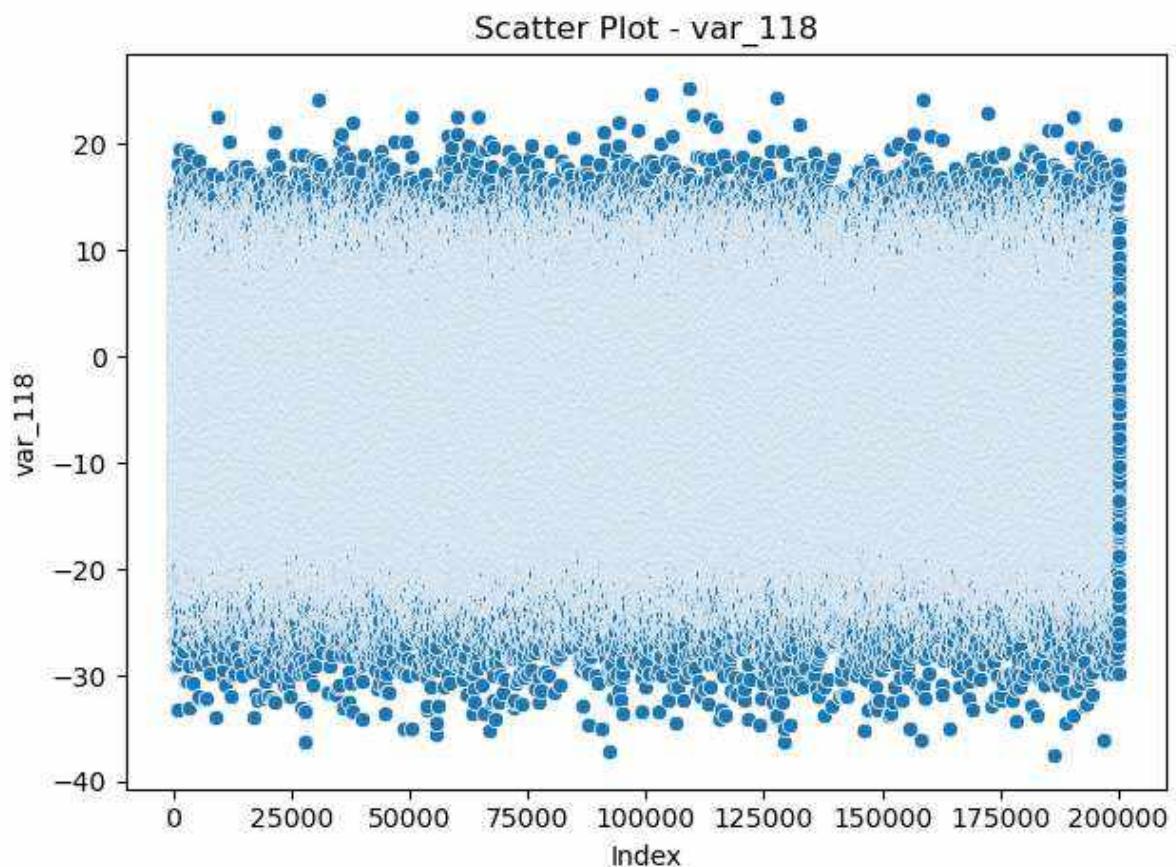
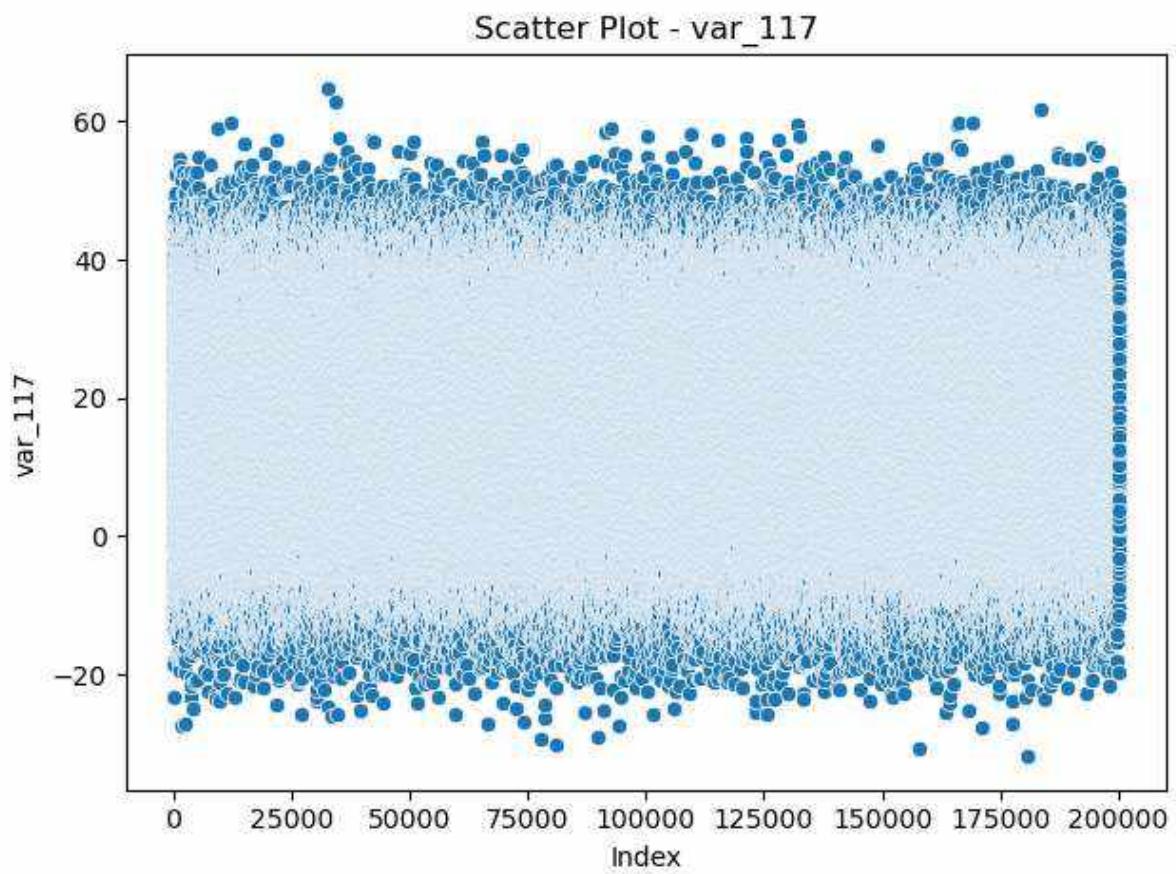
Scatter Plot - var\_113



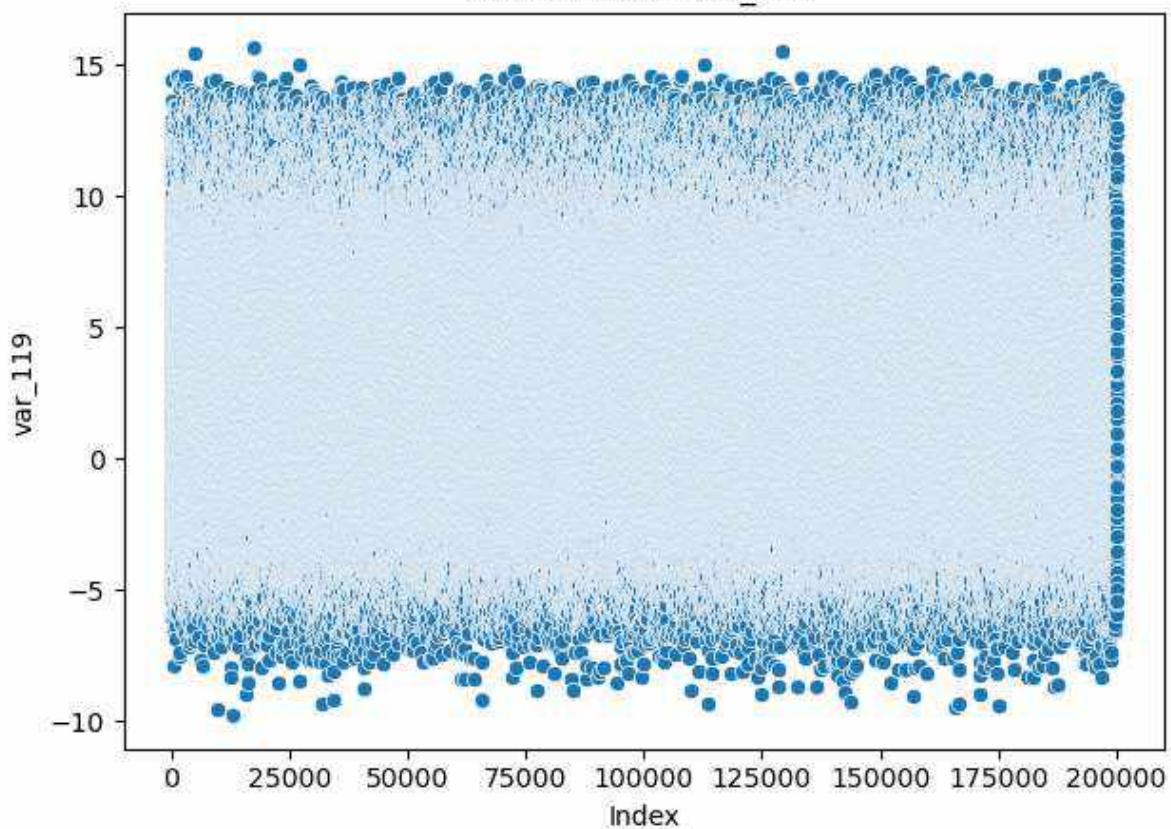
Scatter Plot - var\_114



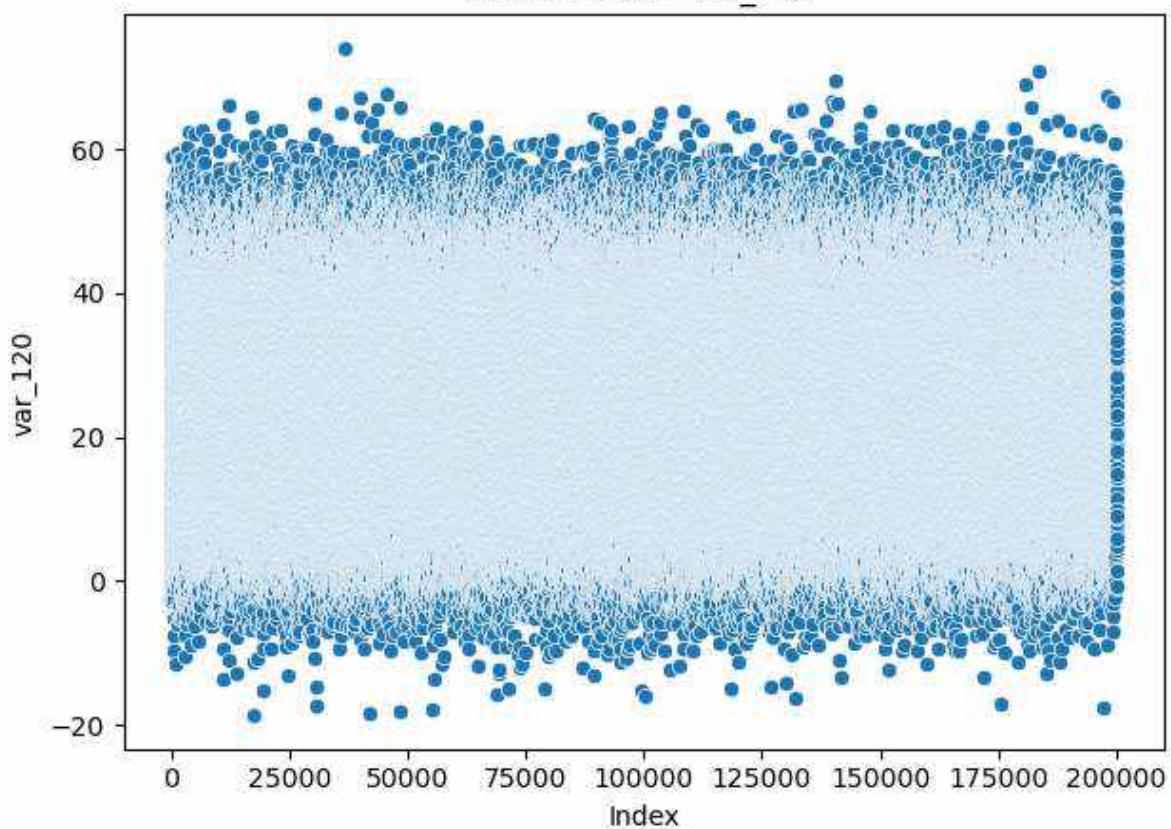




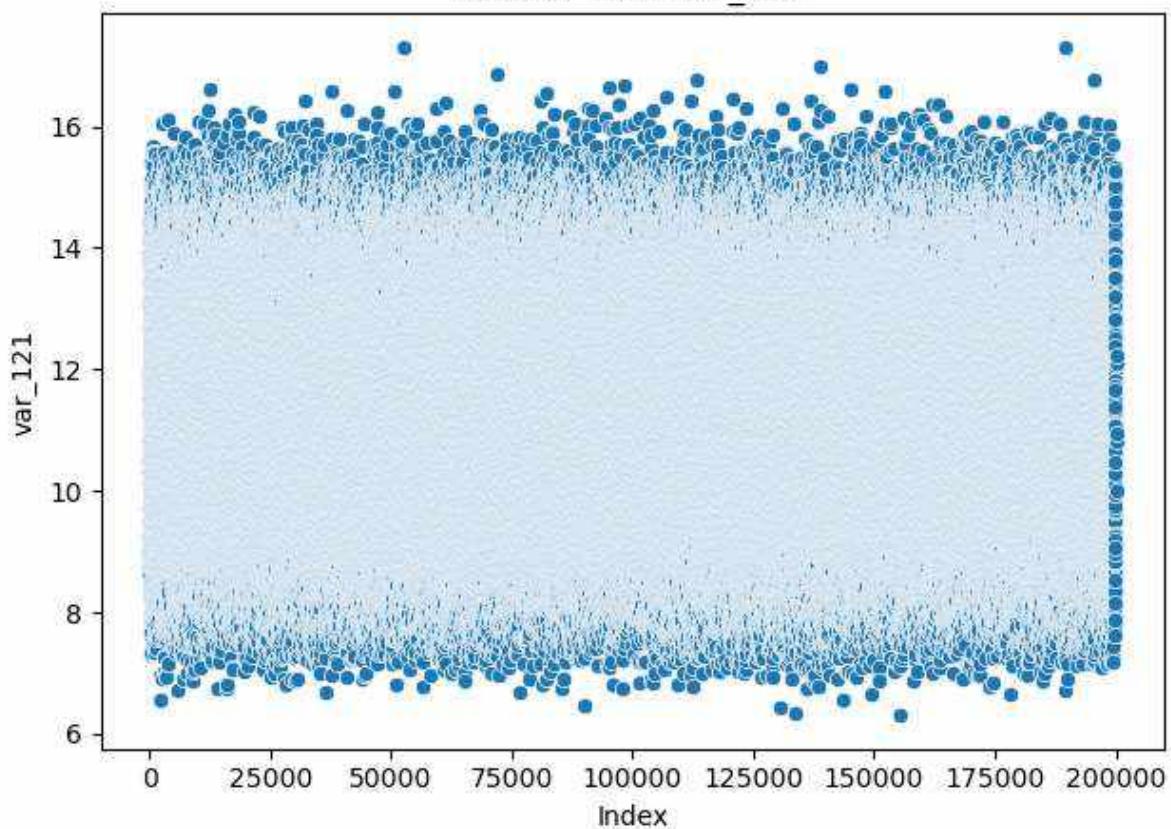
Scatter Plot - var\_119



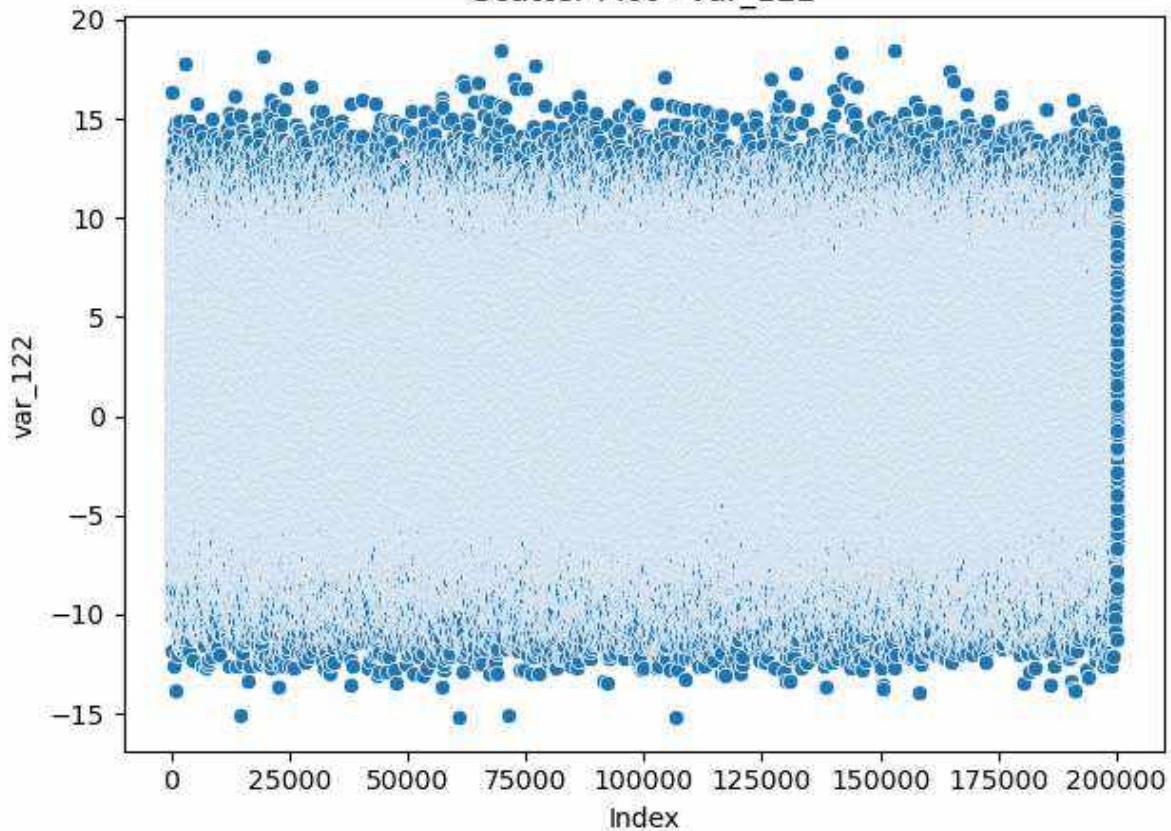
Scatter Plot - var\_120



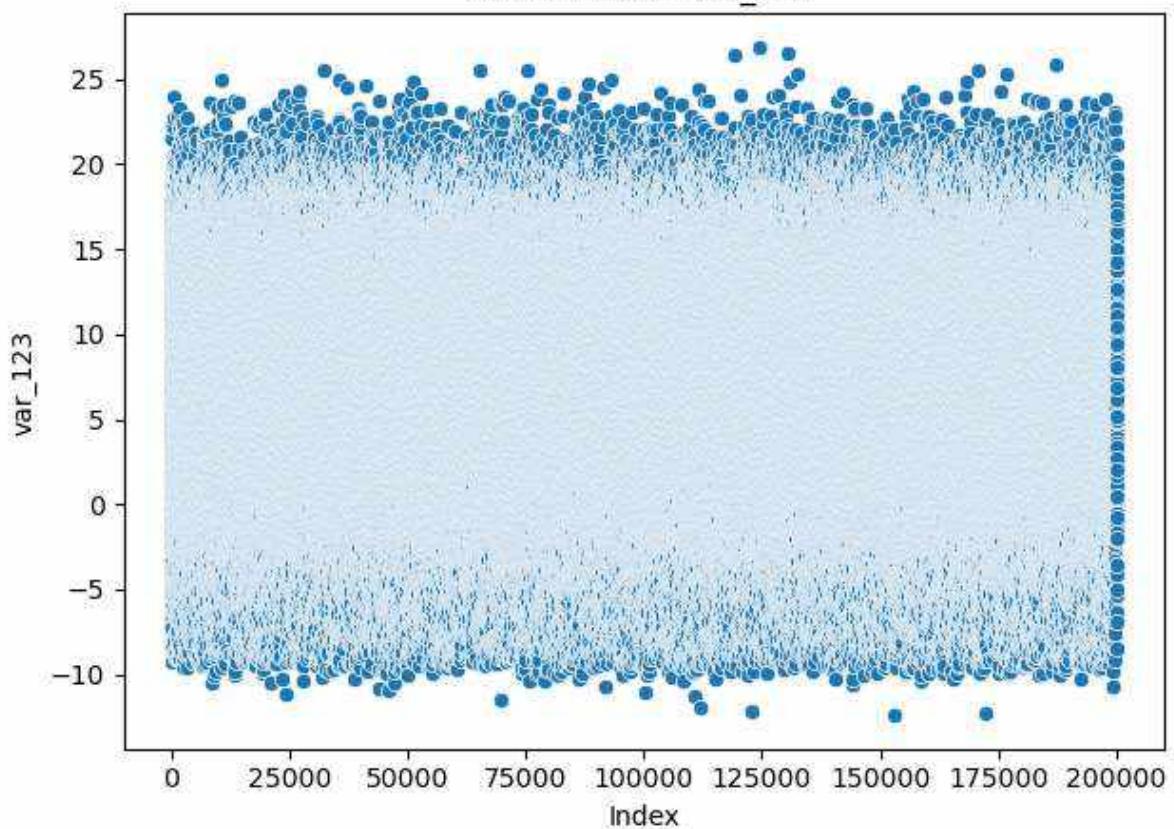
Scatter Plot - var\_121



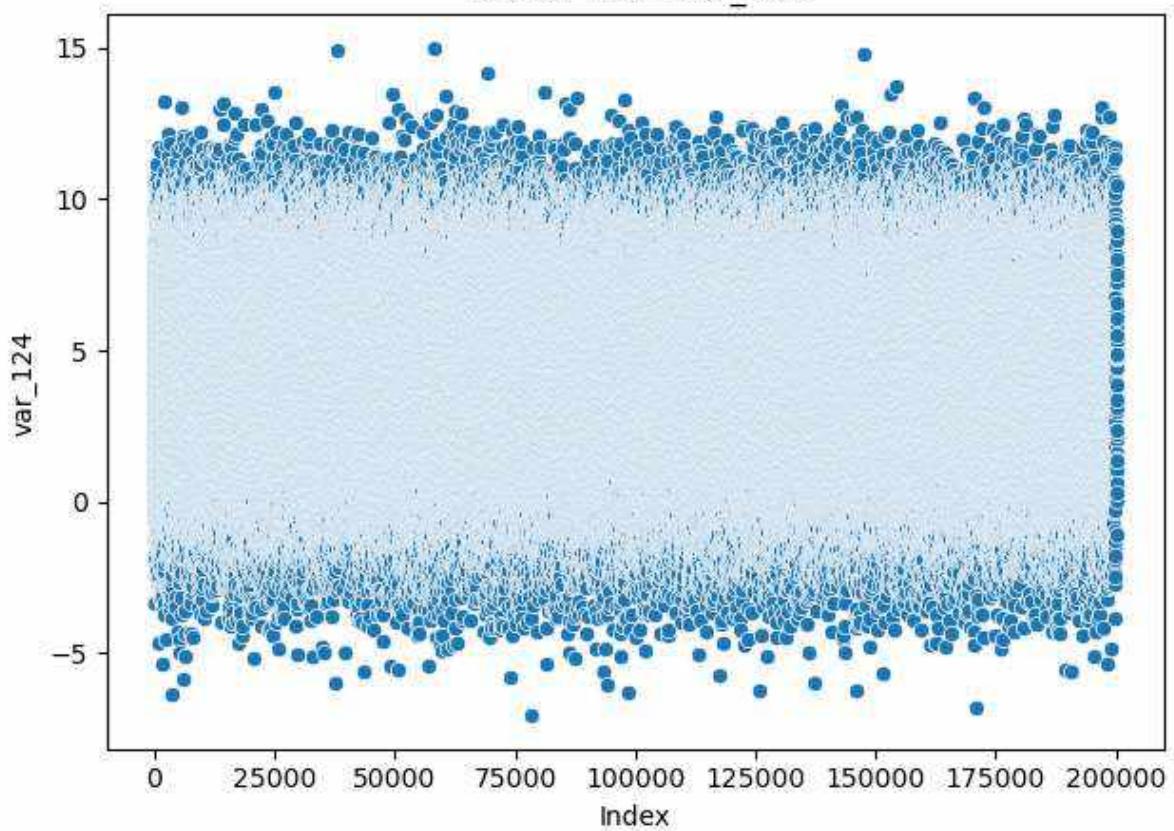
Scatter Plot - var\_122



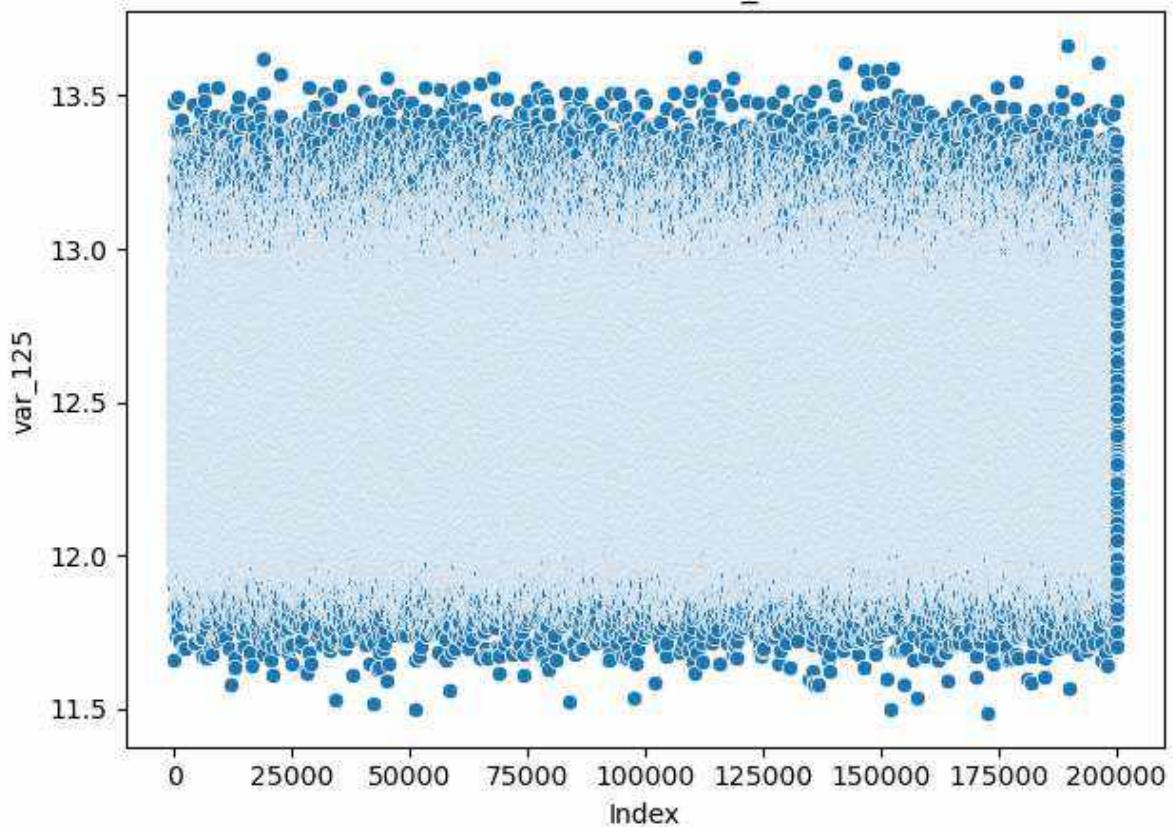
Scatter Plot - var\_123



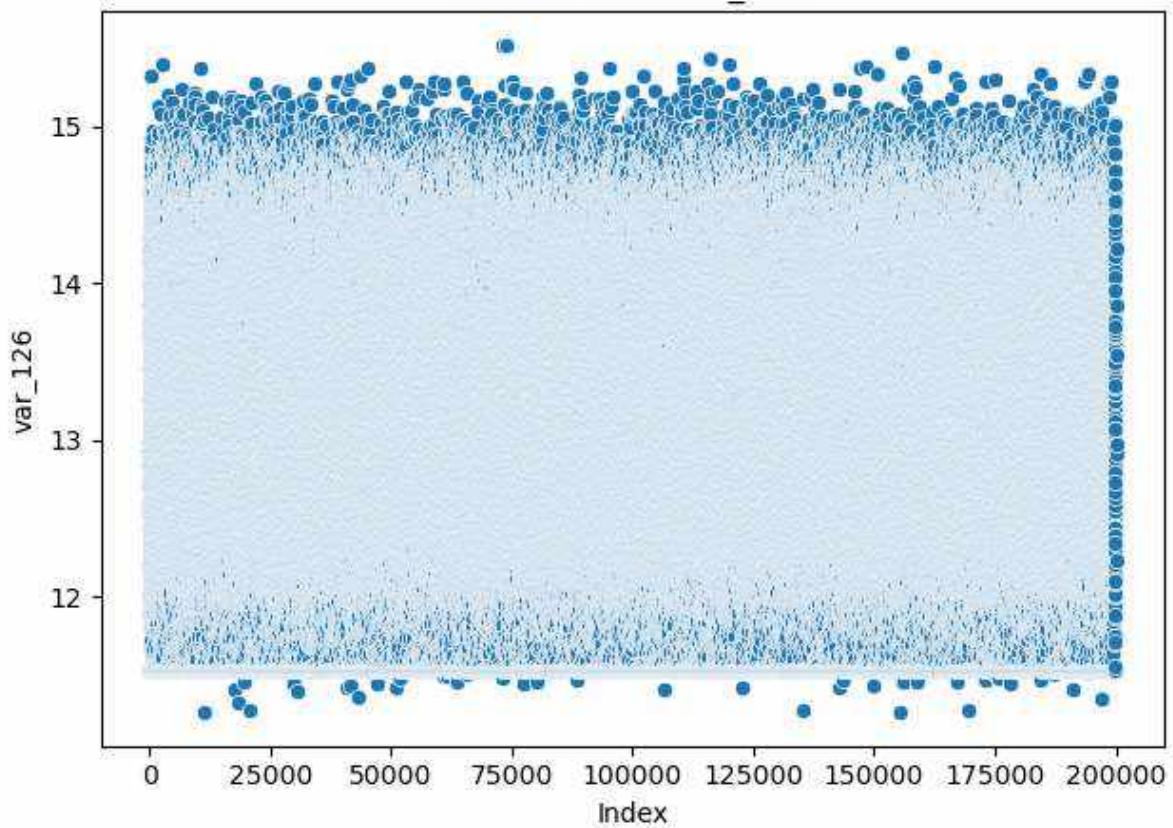
Scatter Plot - var\_124

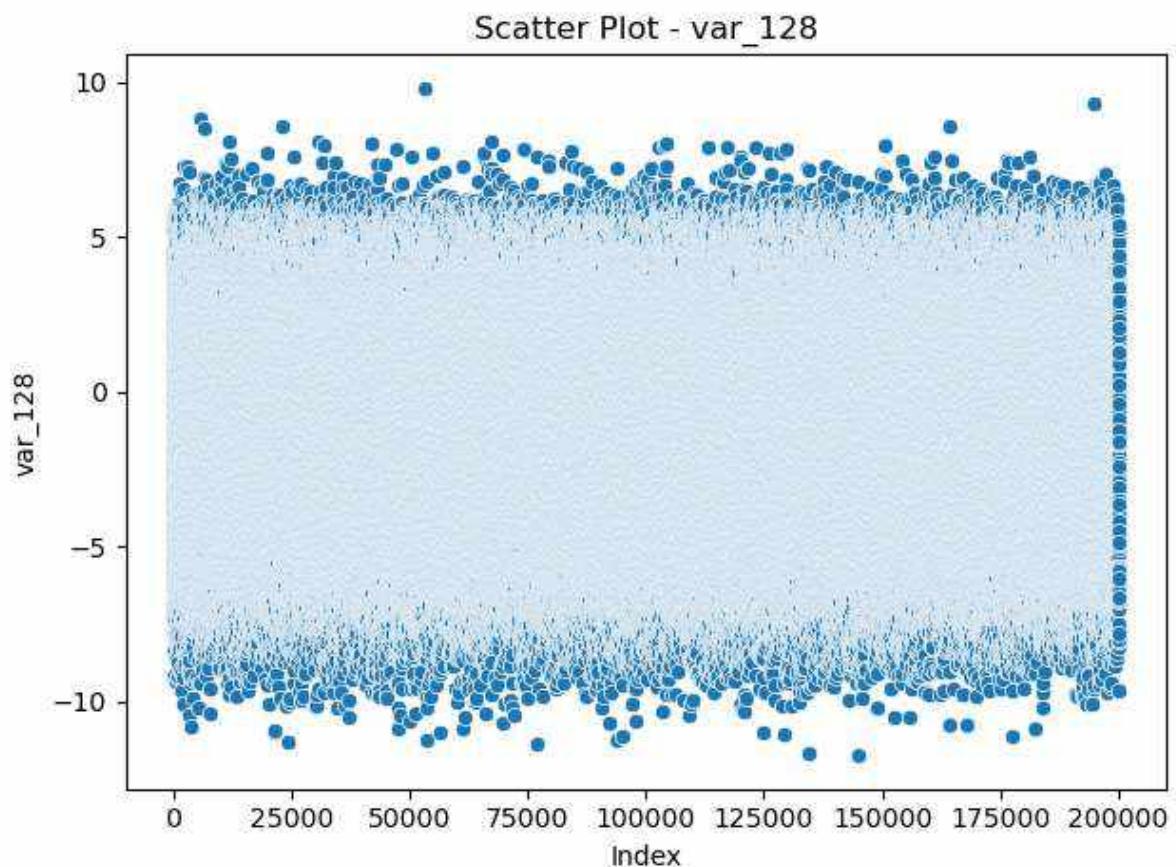
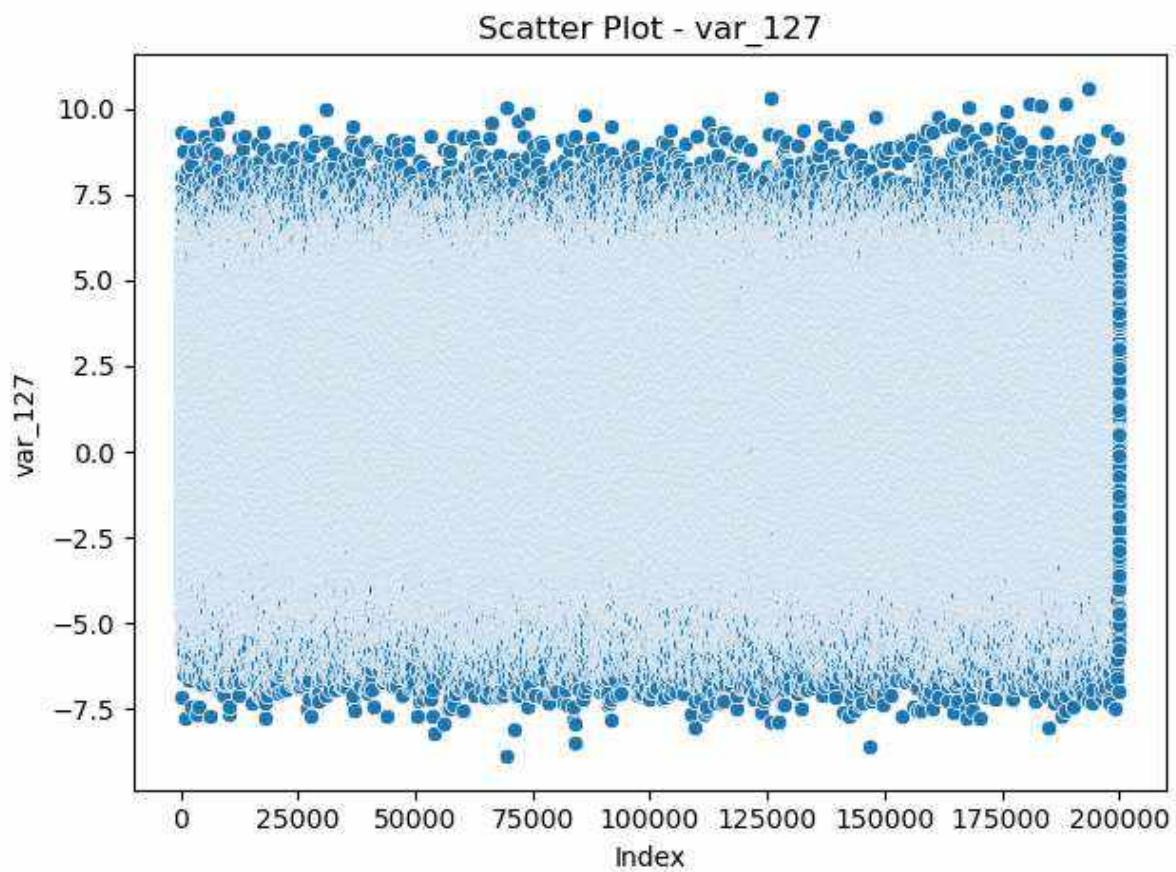


Scatter Plot - var\_125

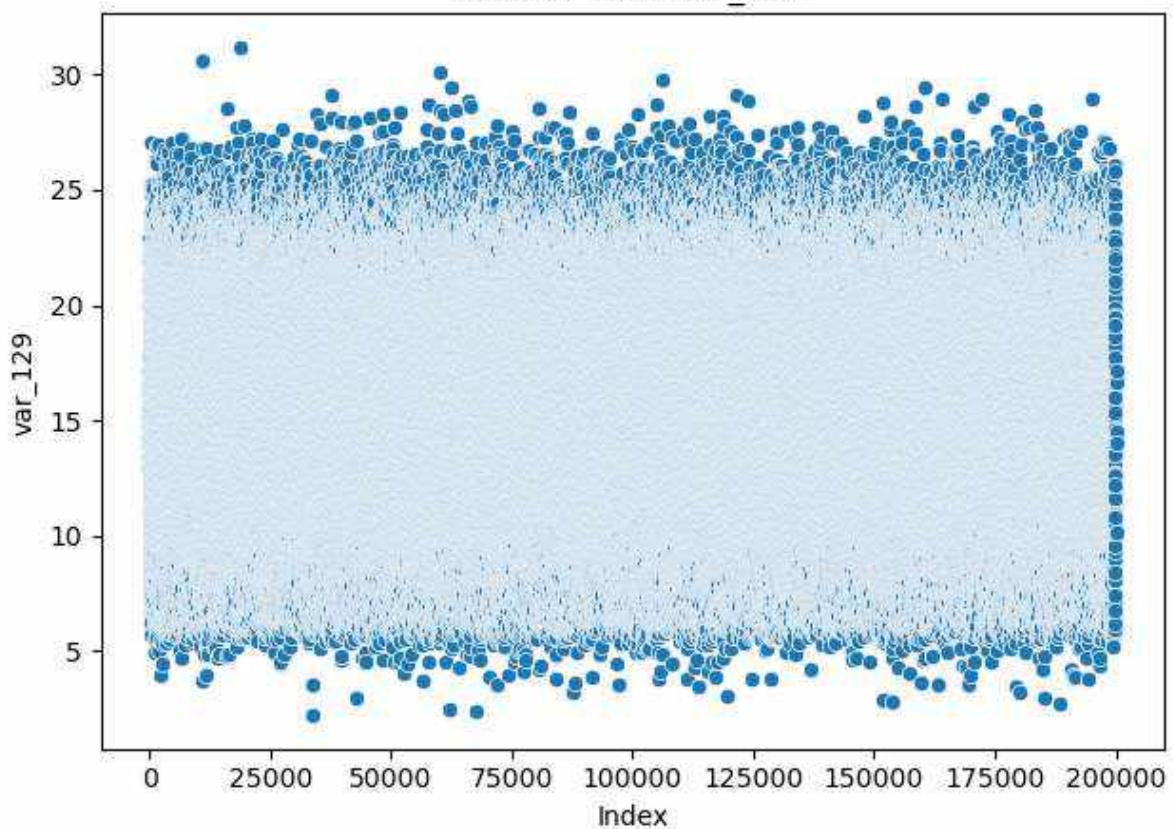


Scatter Plot - var\_126

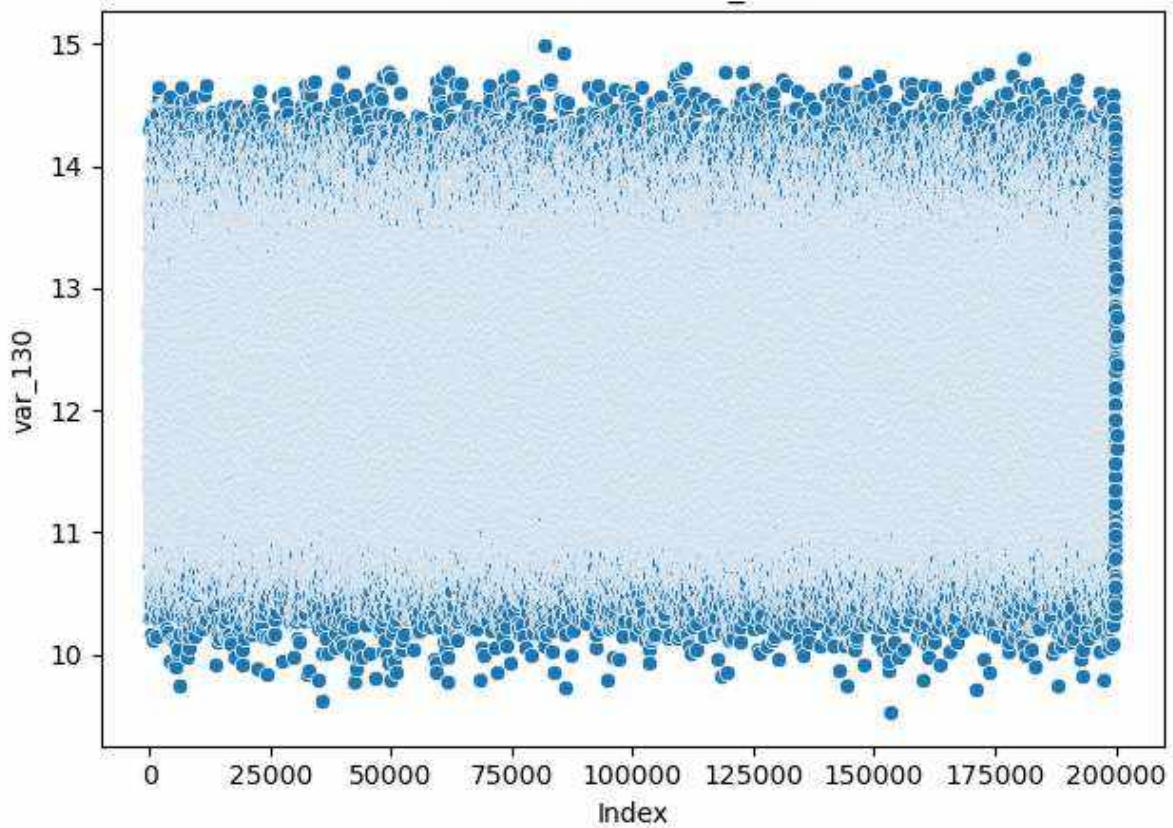




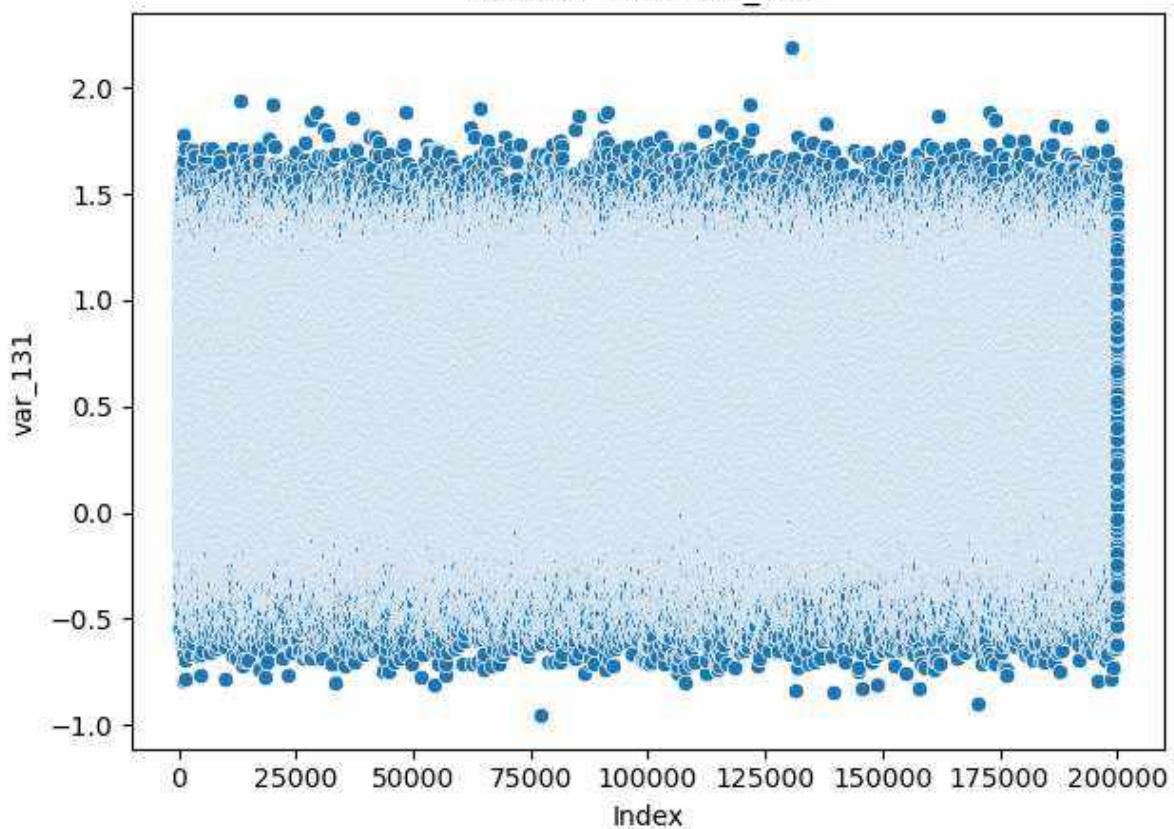
Scatter Plot - var\_129



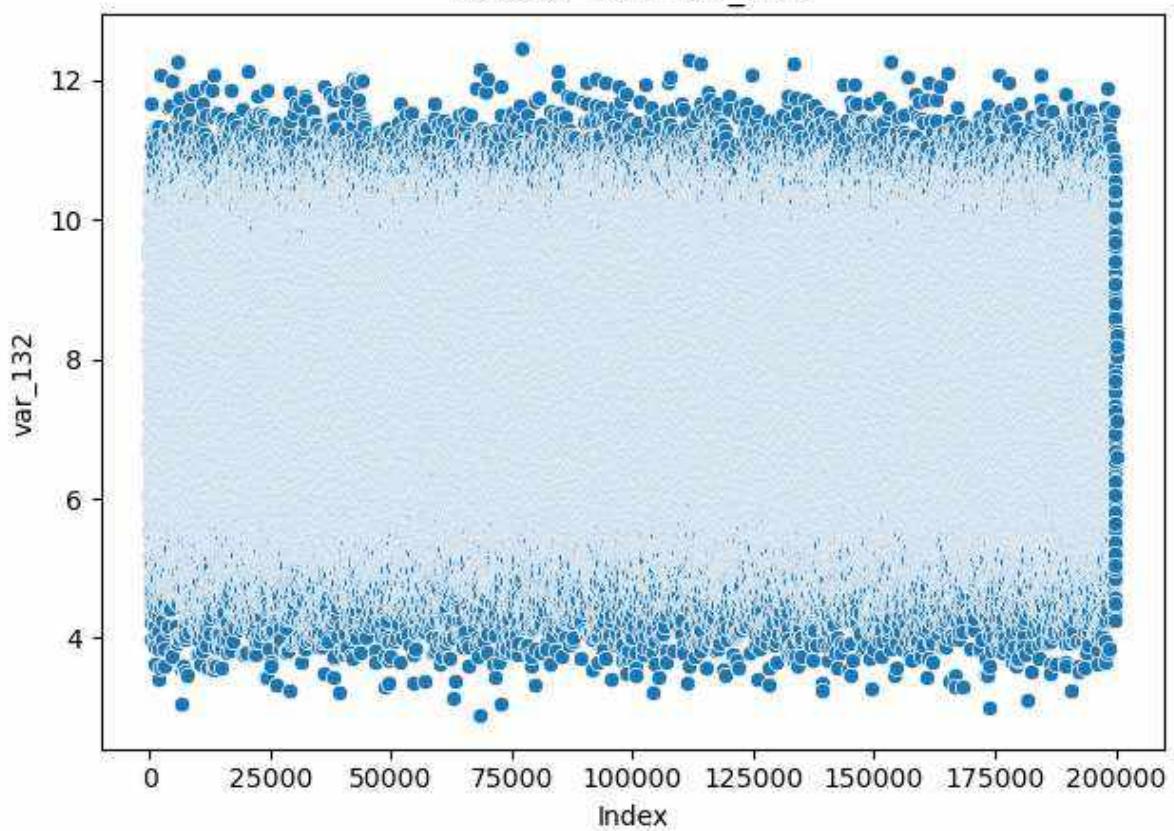
Scatter Plot - var\_130



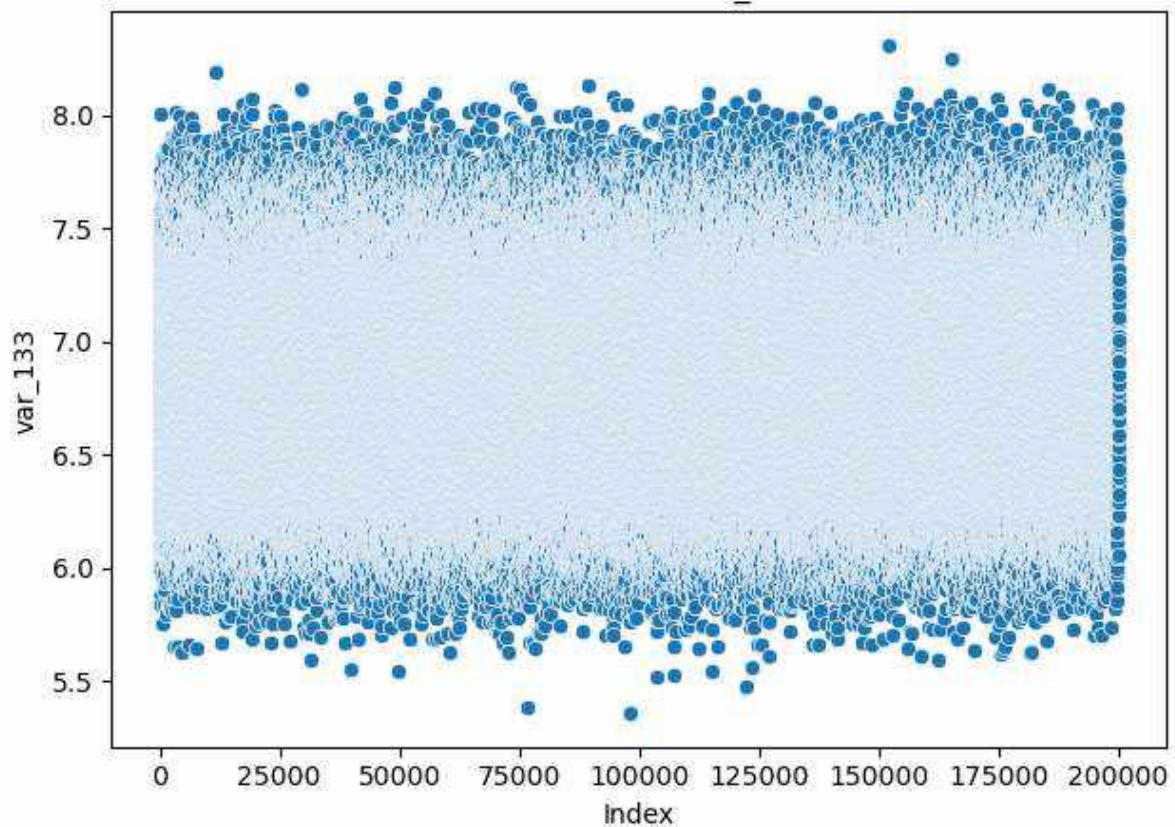
Scatter Plot - var\_131



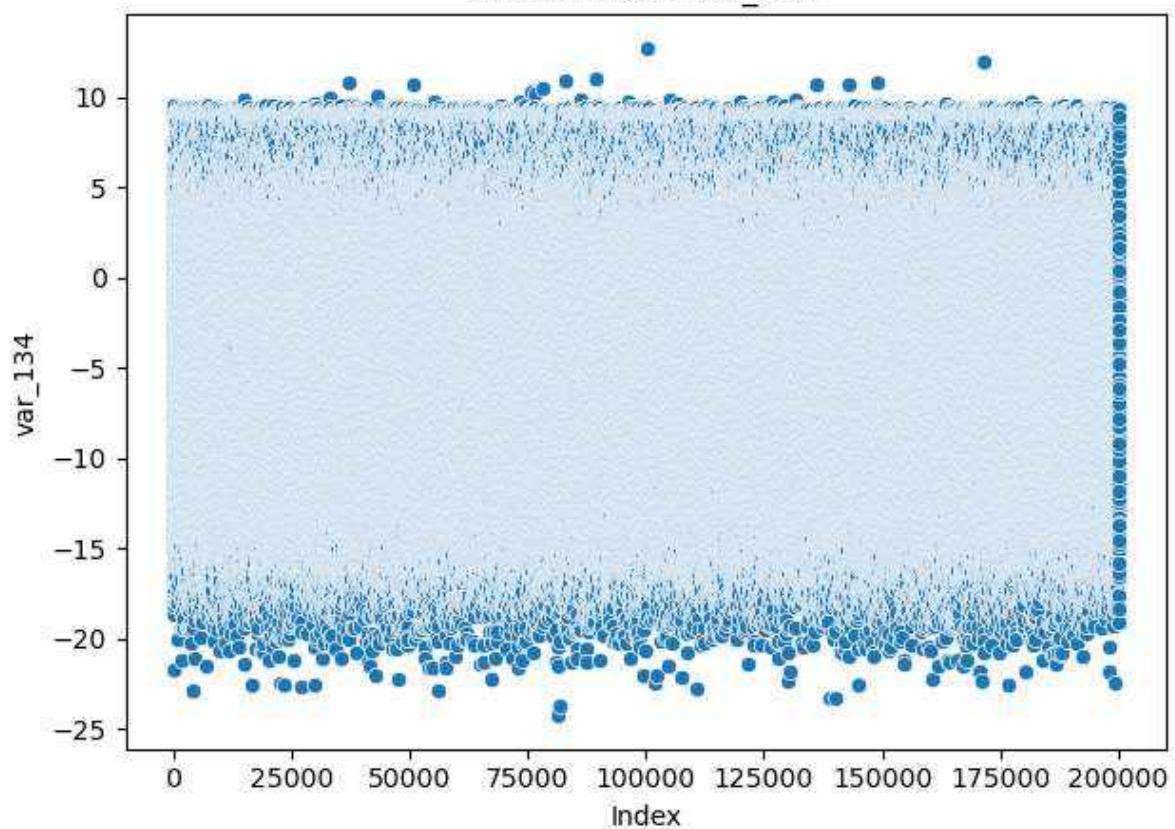
Scatter Plot - var\_132



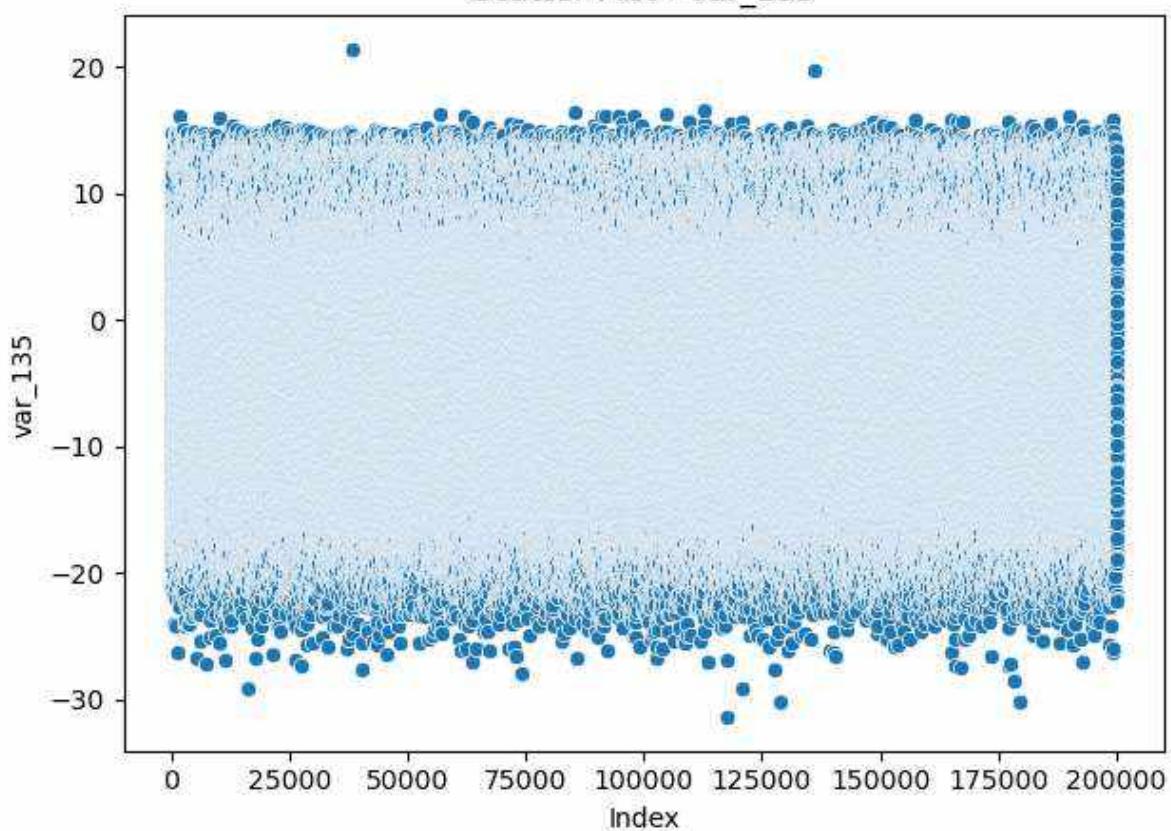
Scatter Plot - var\_133



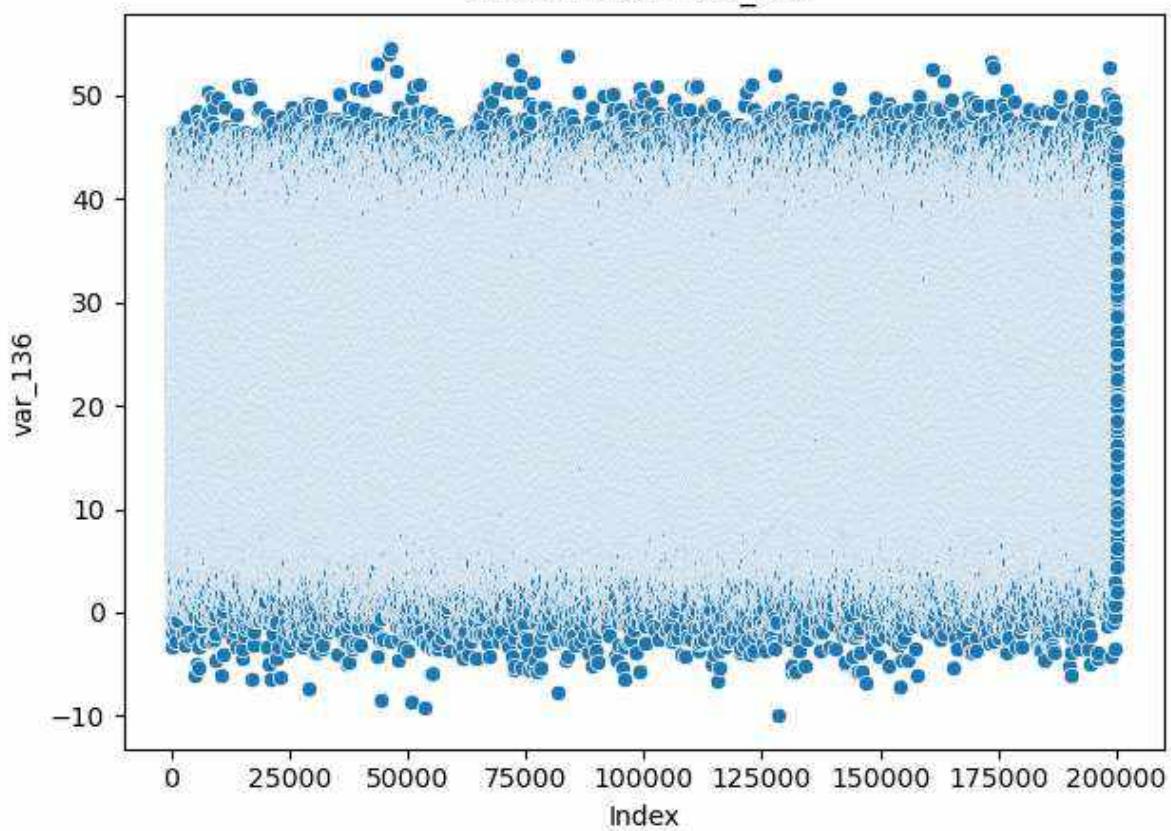
Scatter Plot - var\_134

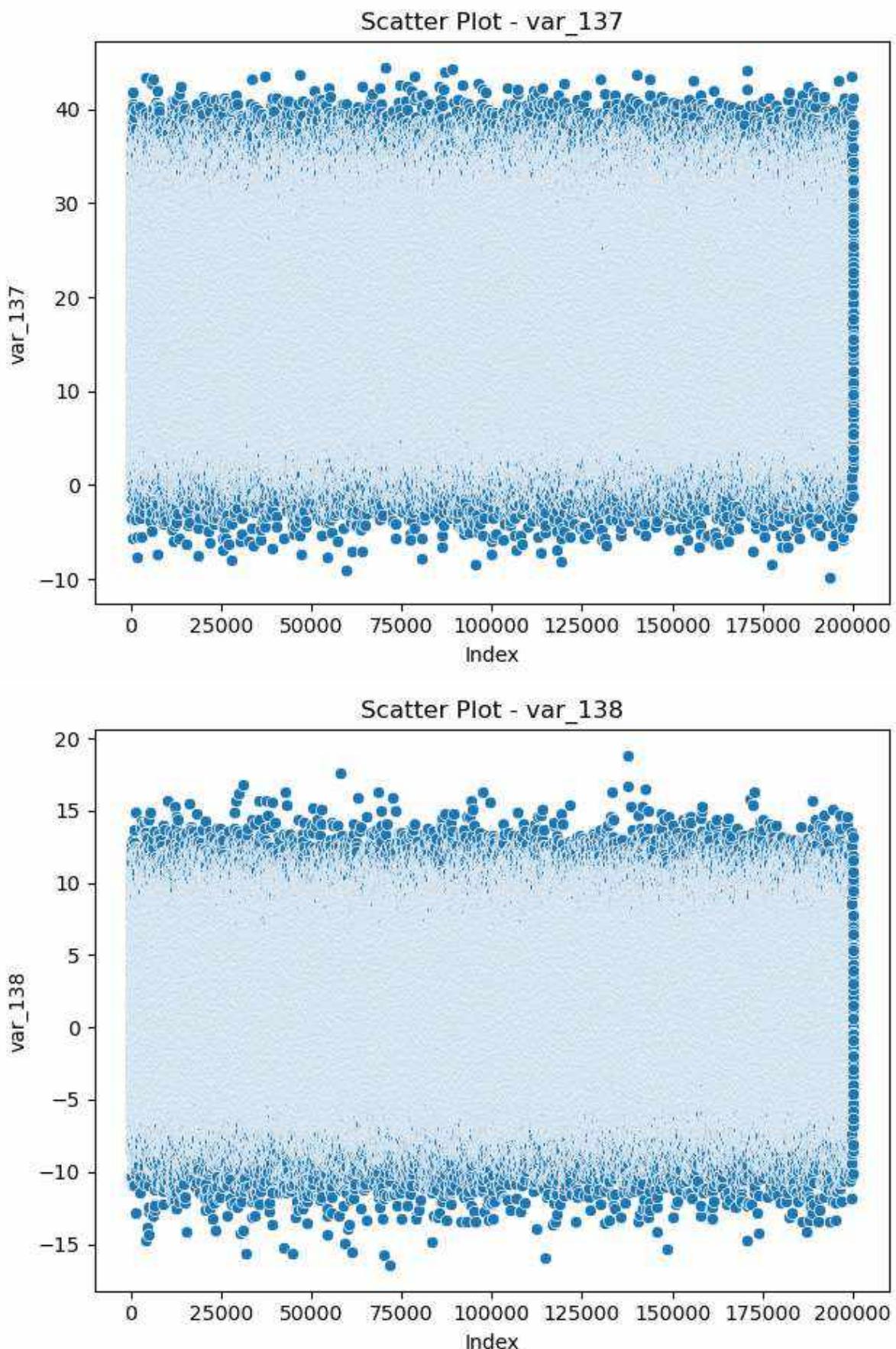


Scatter Plot - var\_135

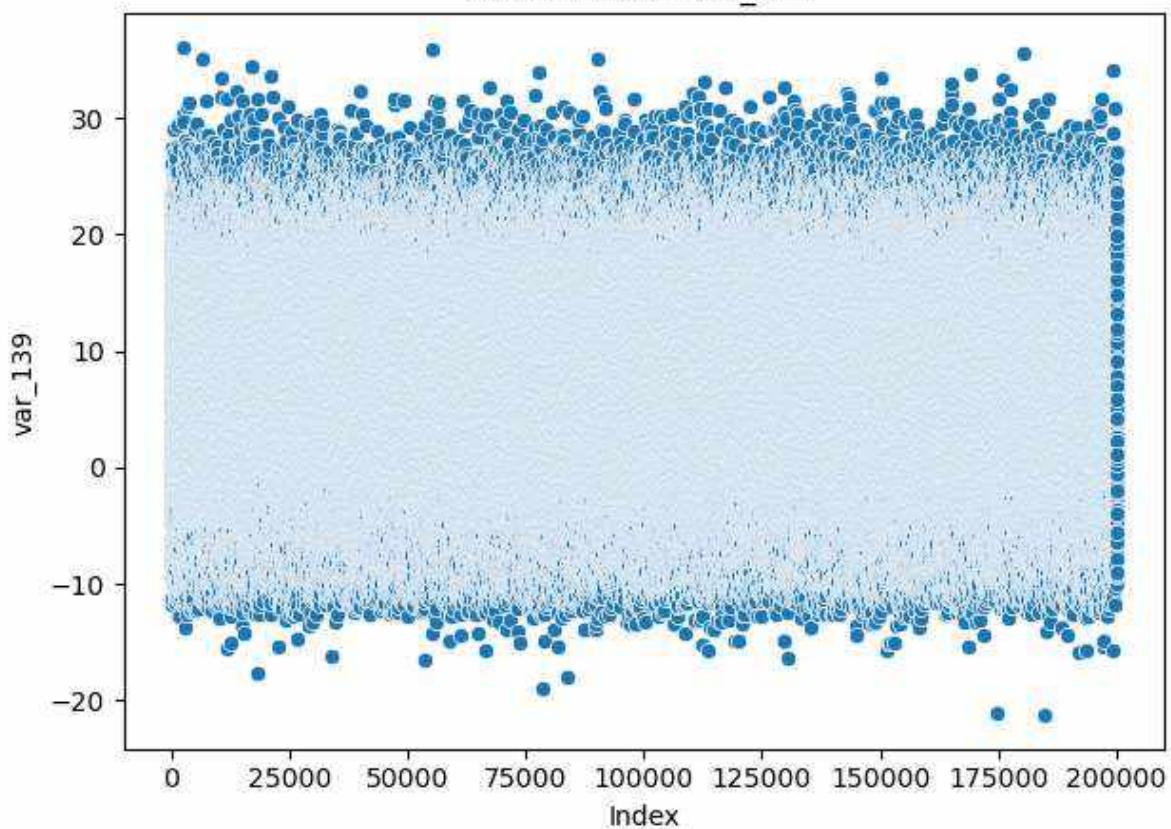


Scatter Plot - var\_136

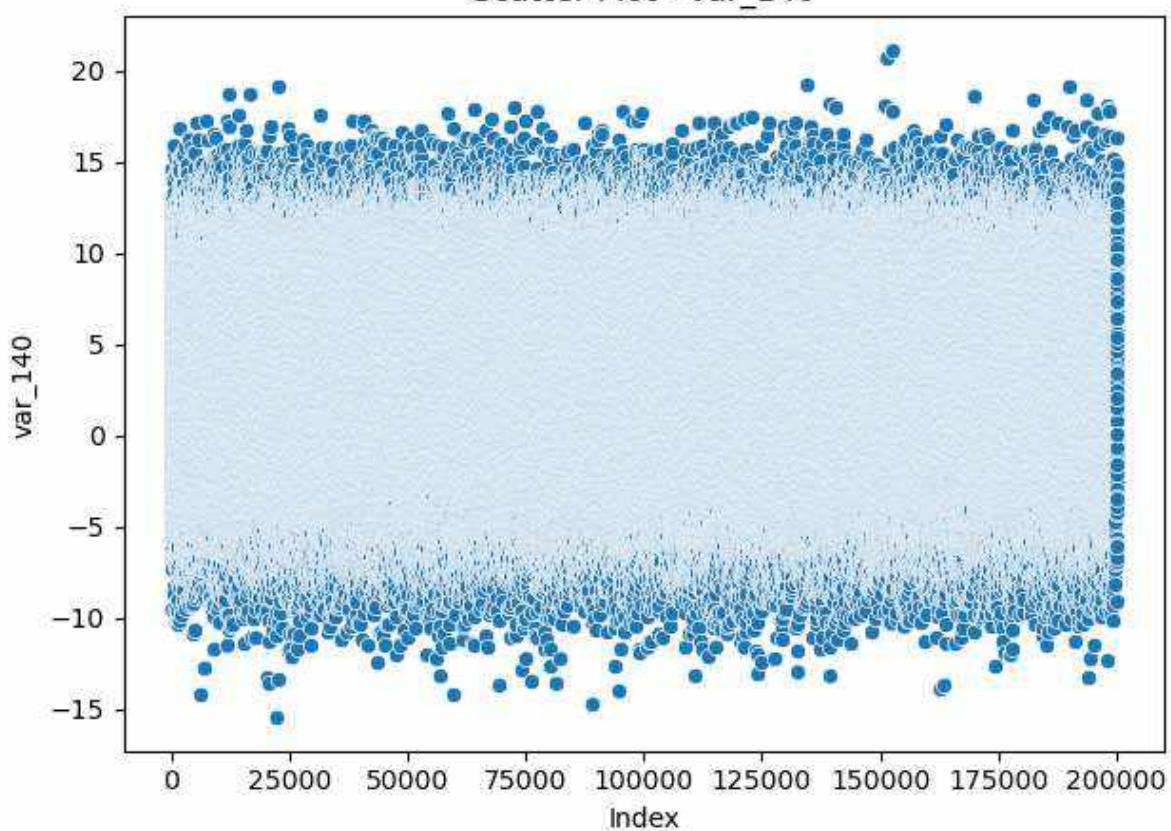


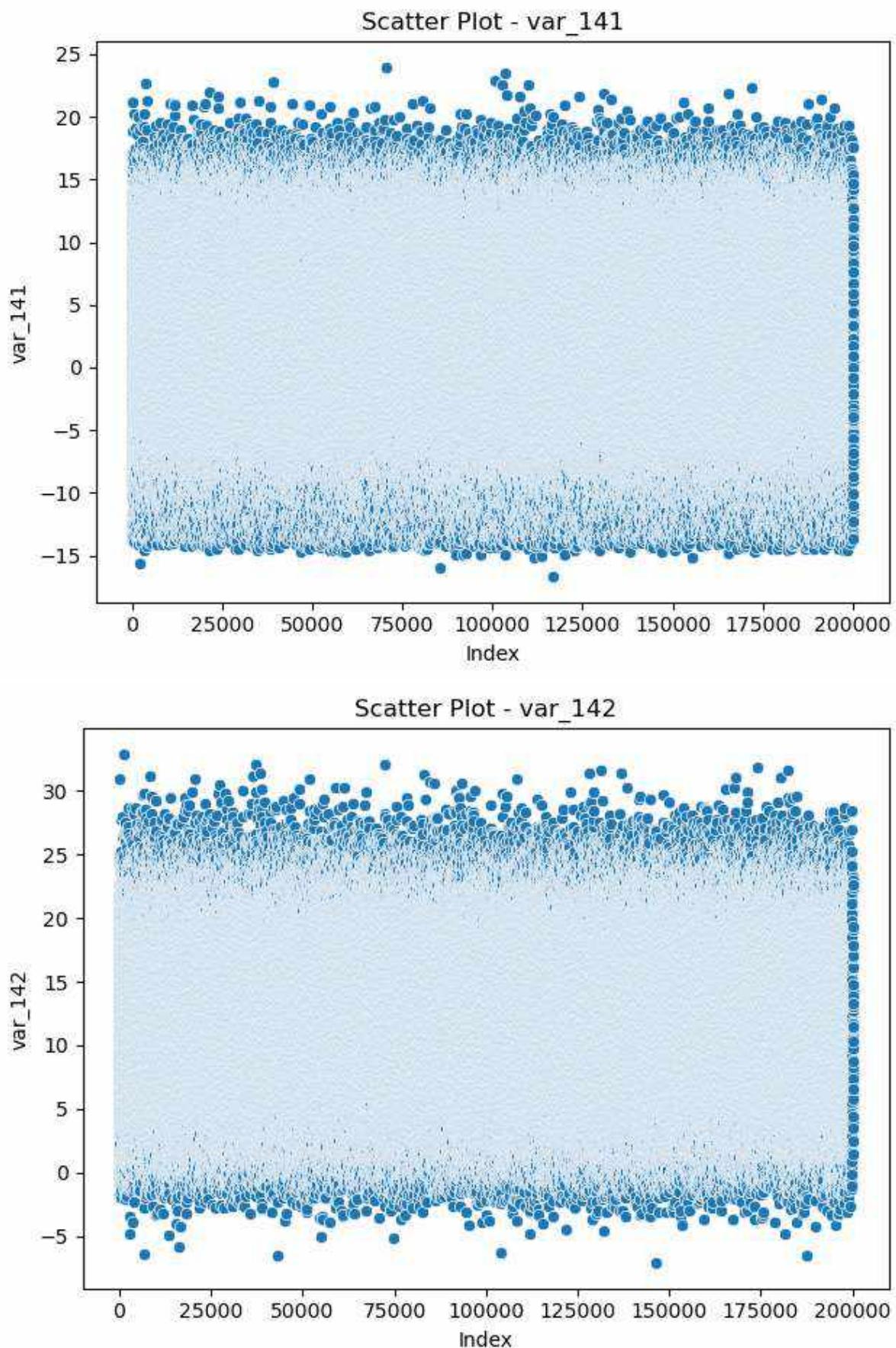


Scatter Plot - var\_139

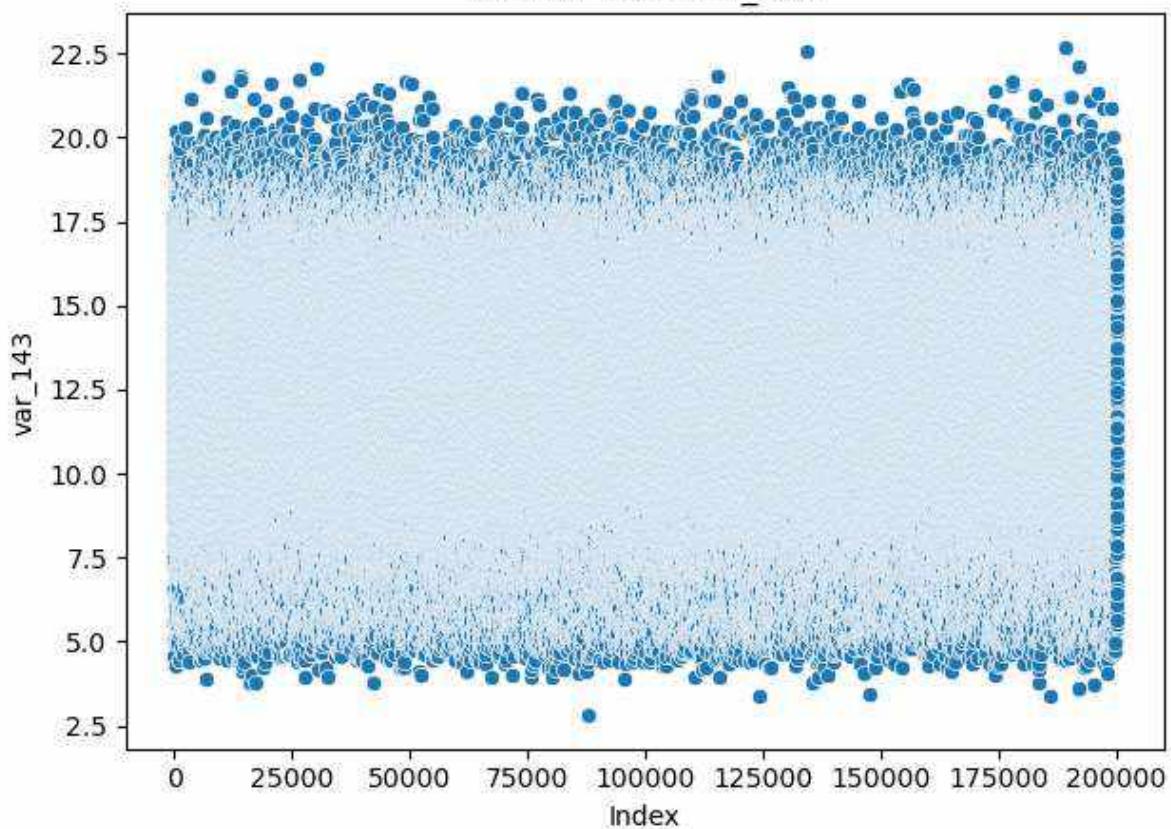


Scatter Plot - var\_140

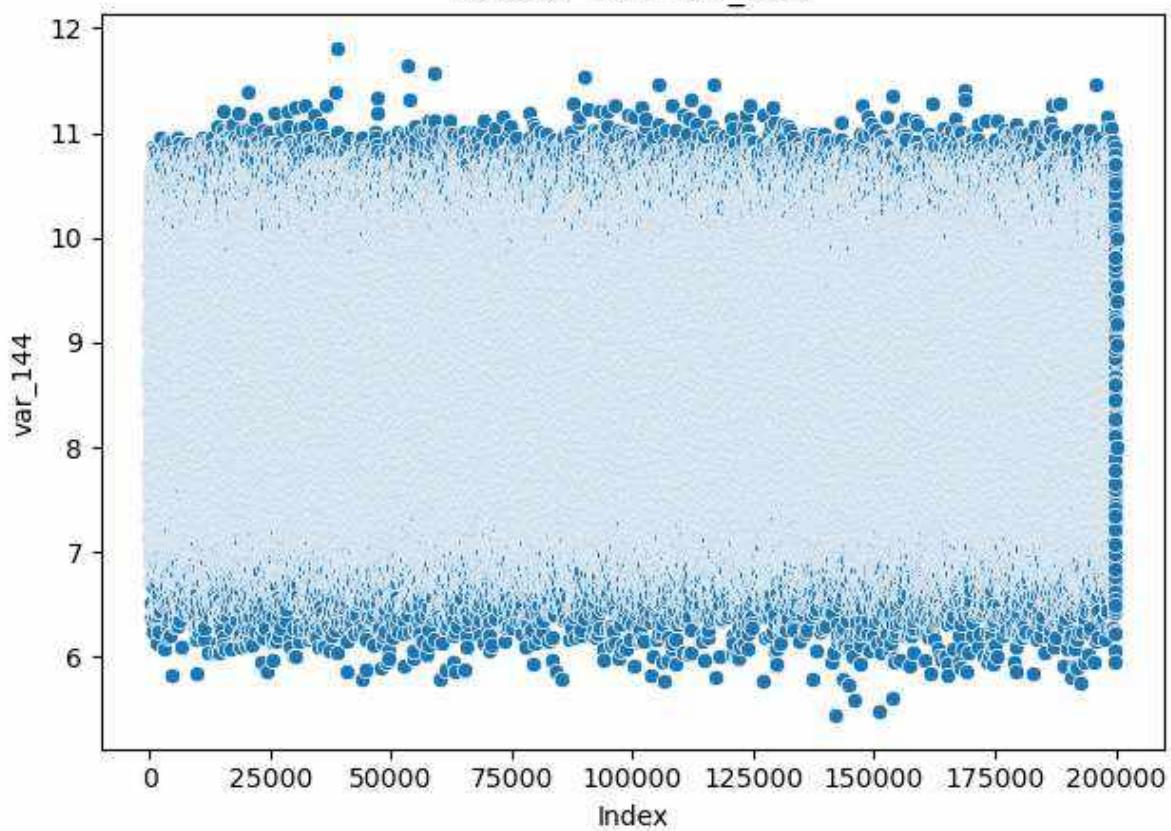




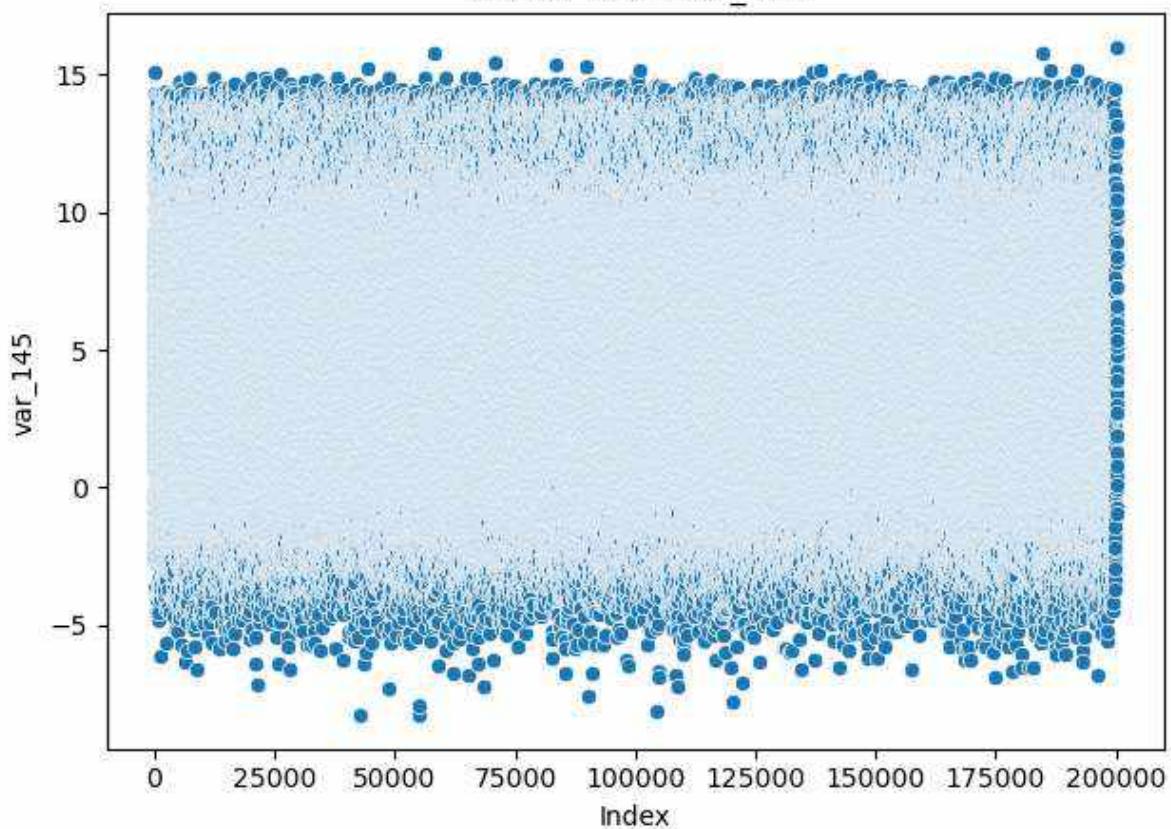
Scatter Plot - var\_143



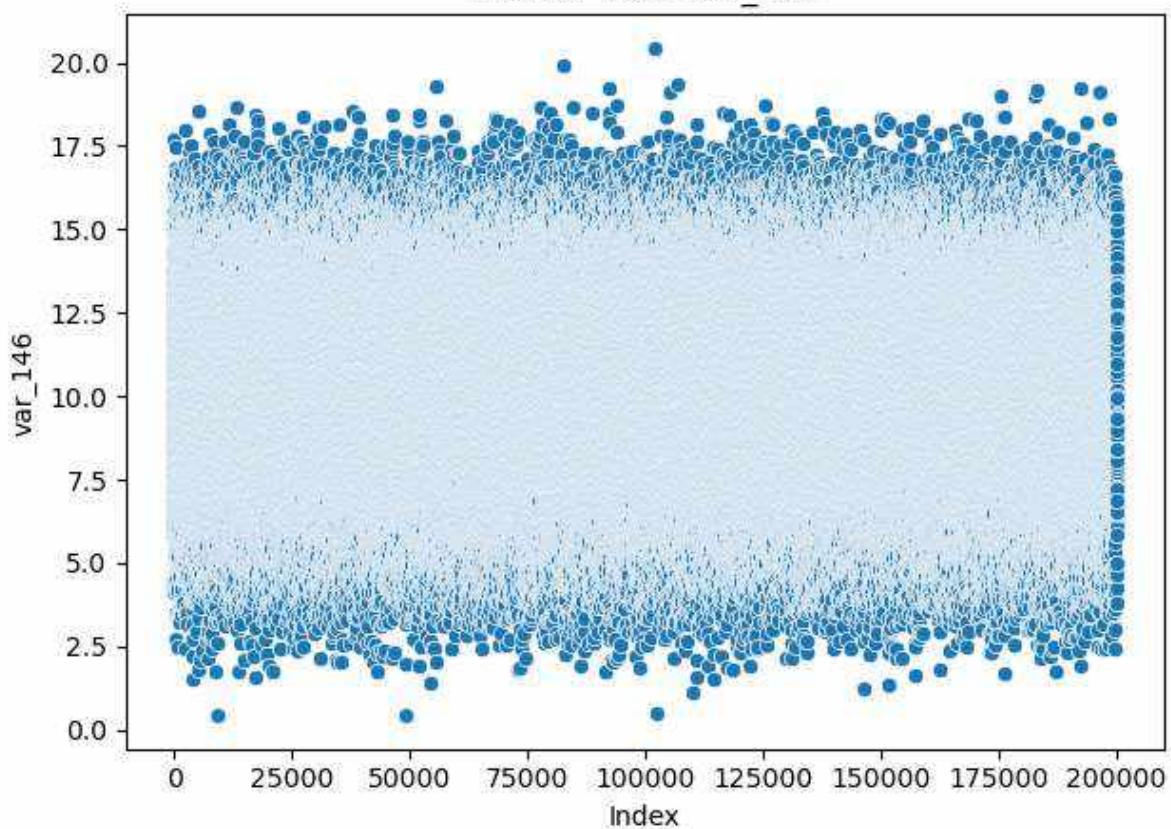
Scatter Plot - var\_144



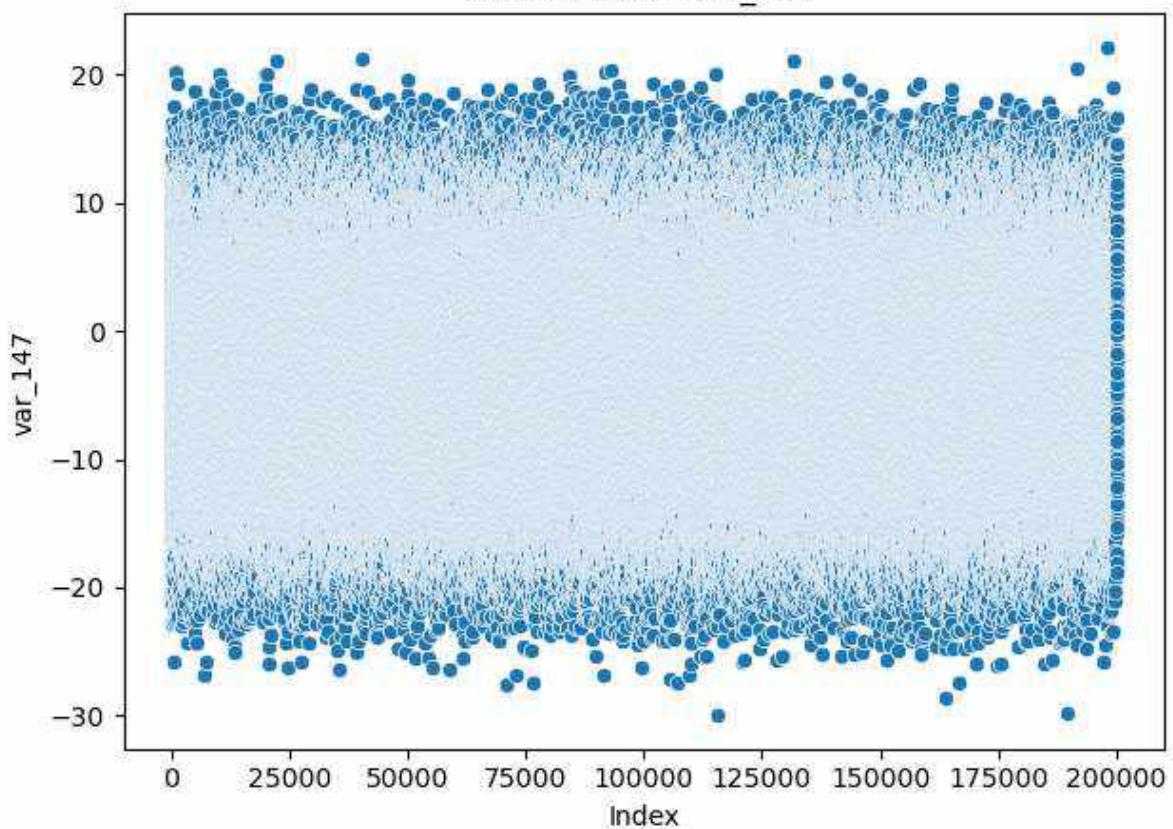
Scatter Plot - var\_145



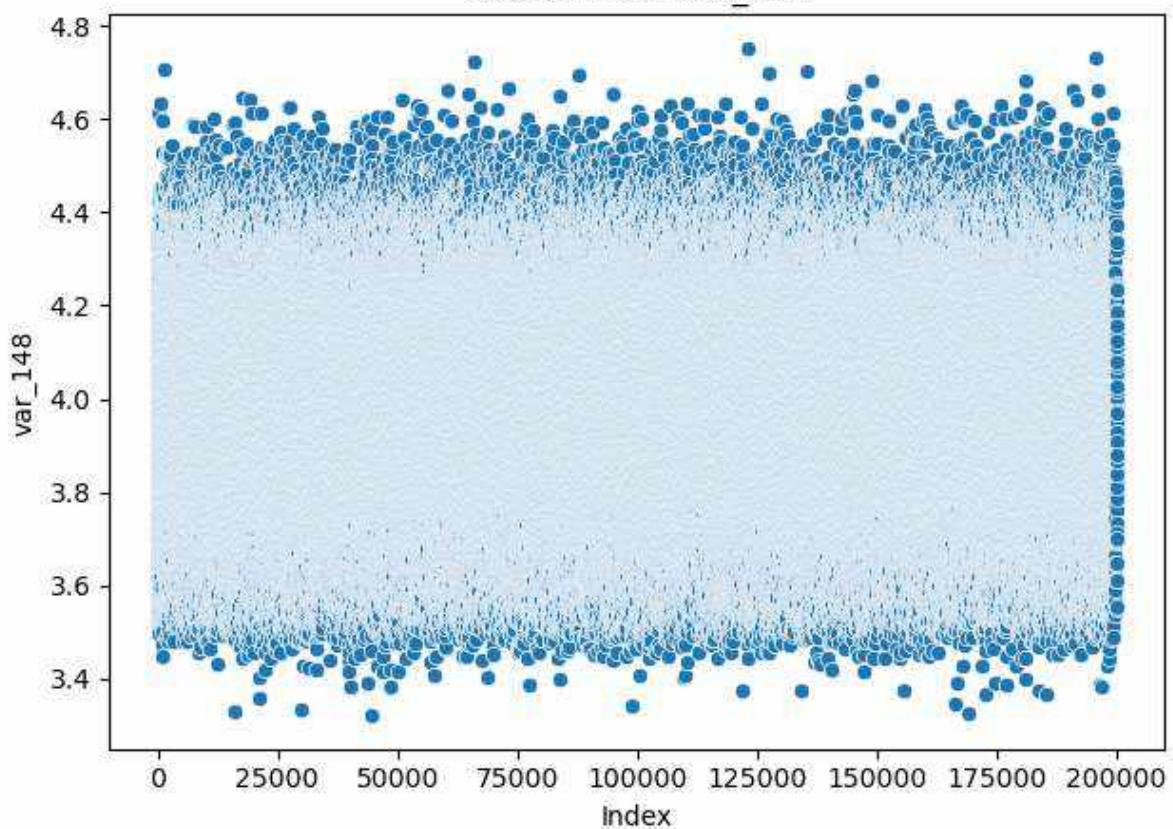
Scatter Plot - var\_146



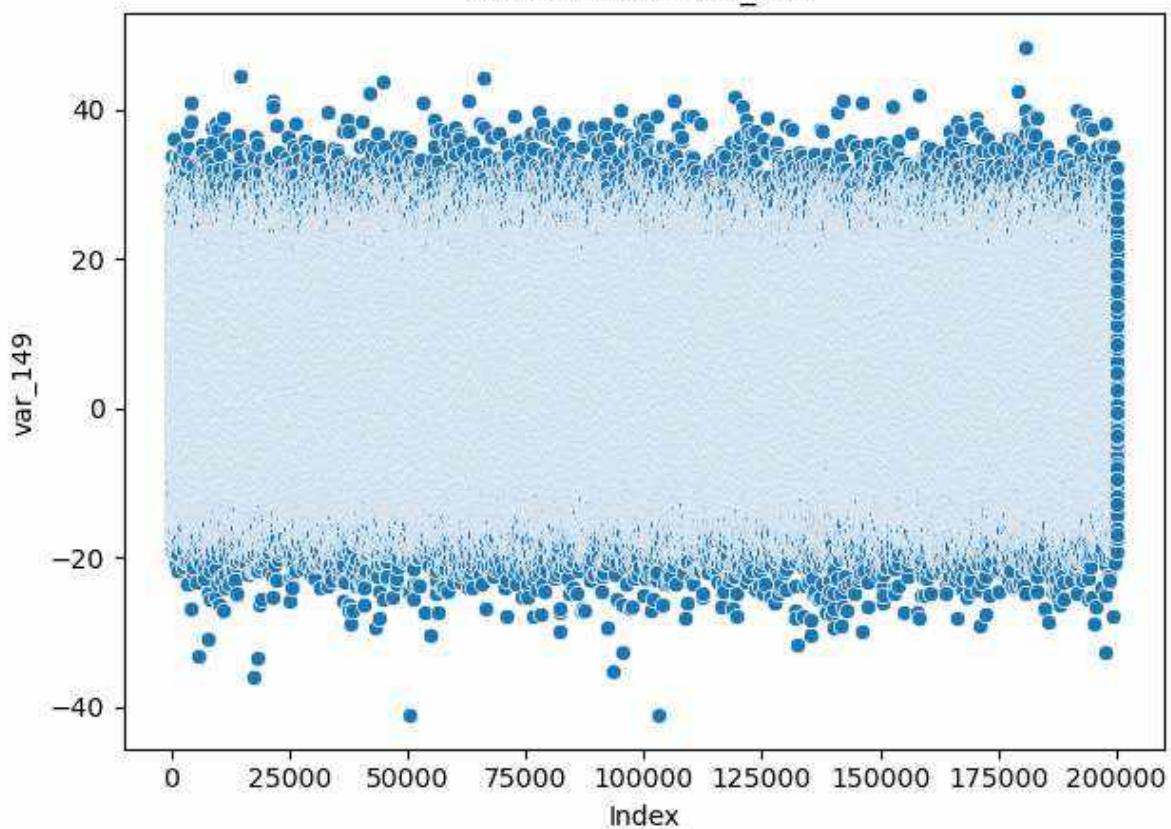
Scatter Plot - var\_147



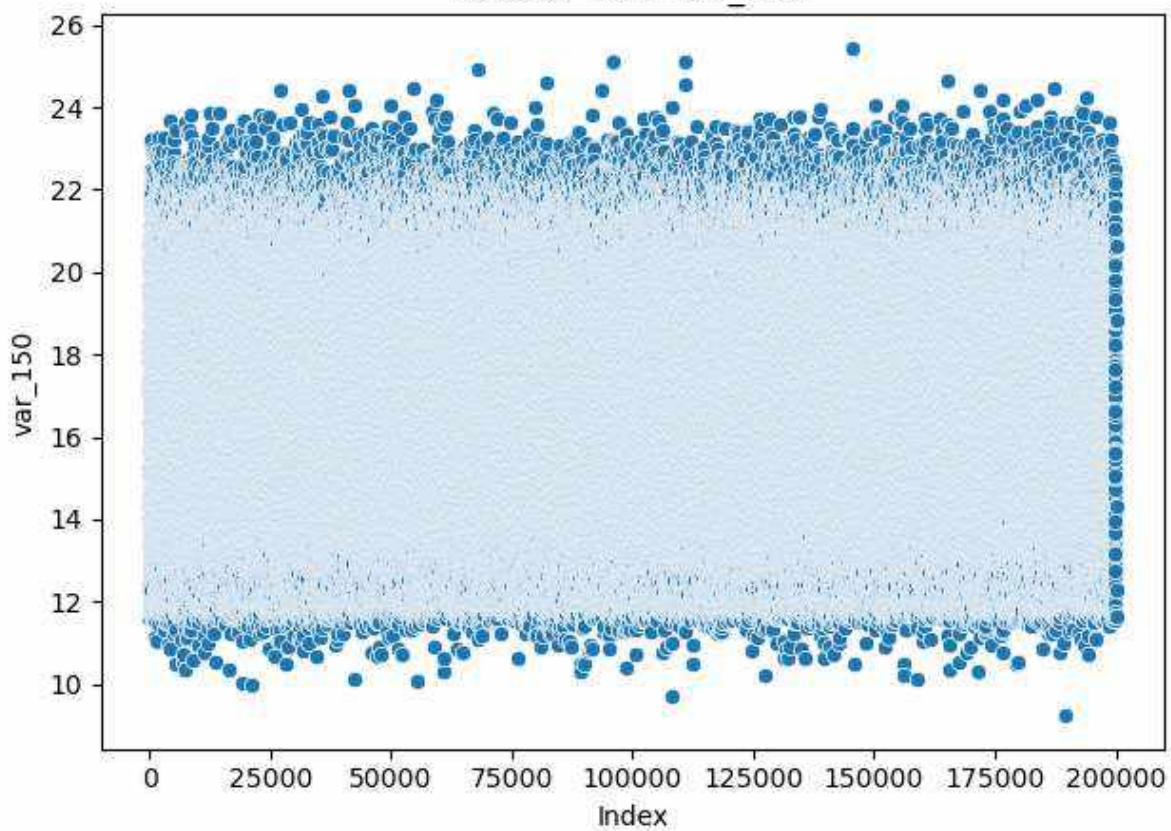
Scatter Plot - var\_148



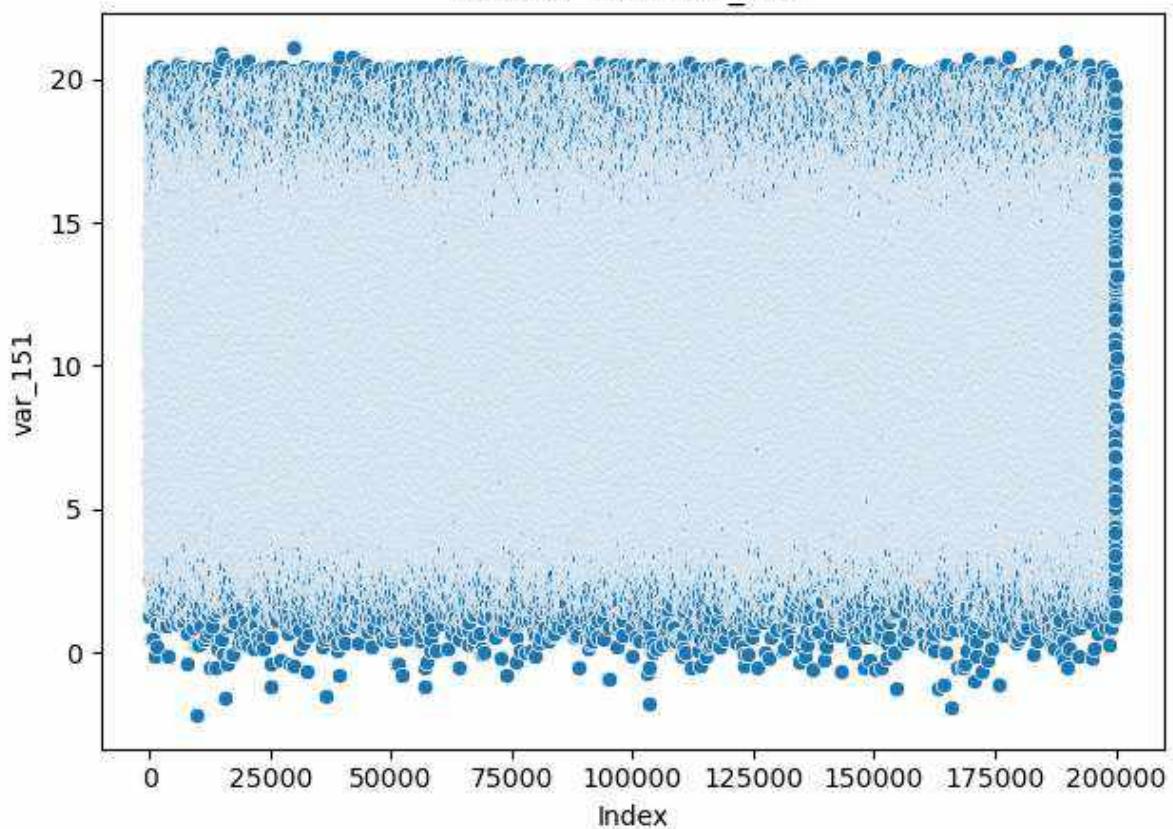
Scatter Plot - var\_149



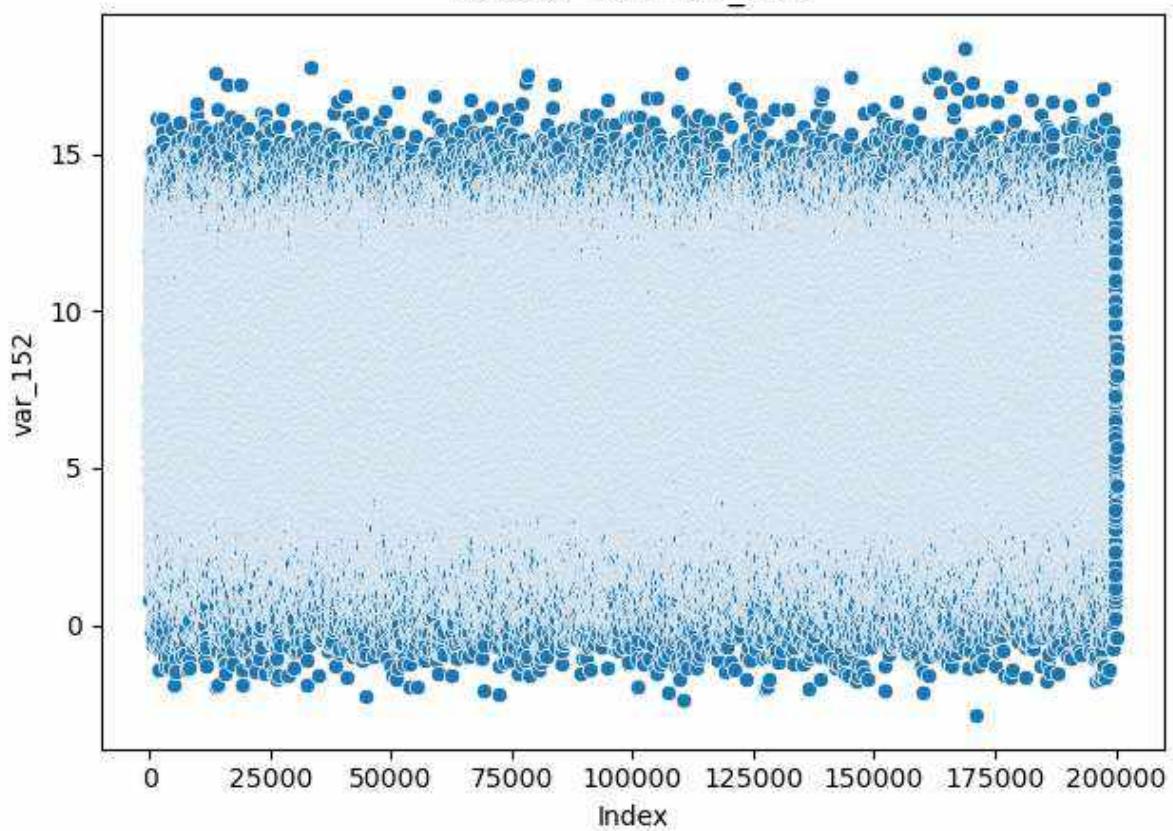
Scatter Plot - var\_150



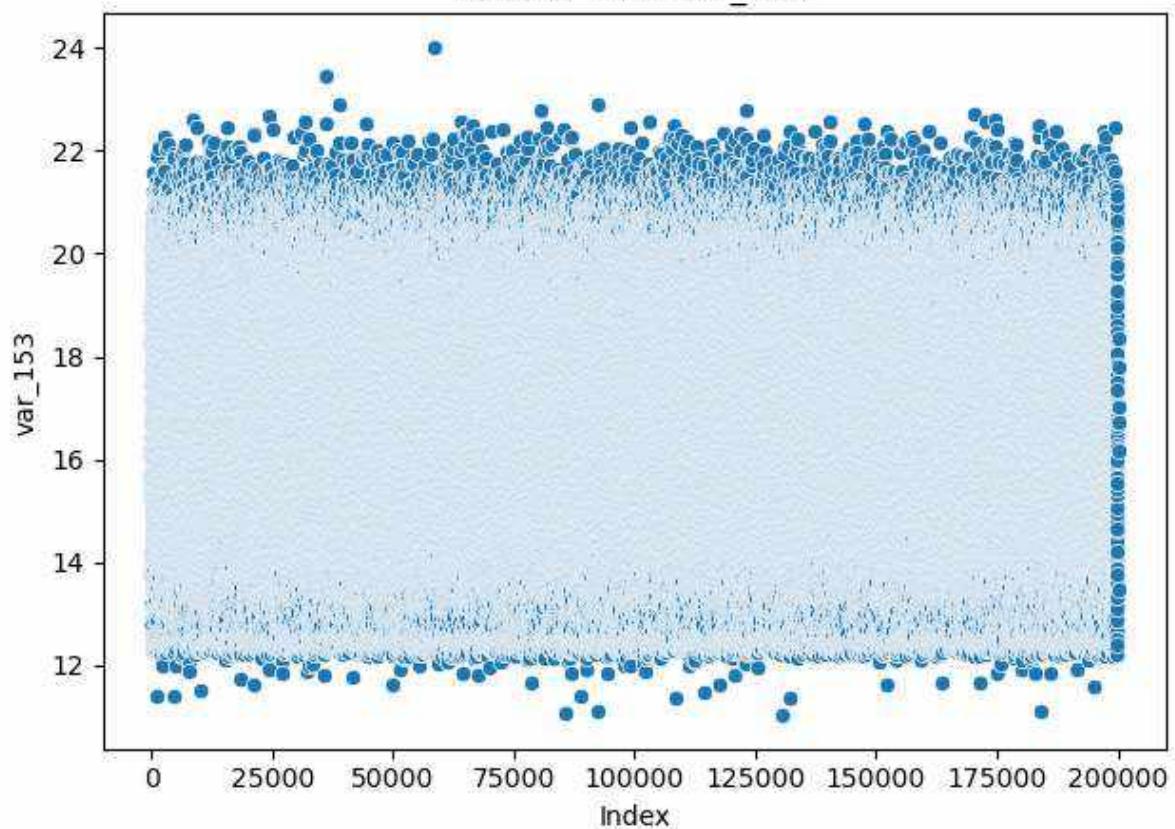
Scatter Plot - var\_151



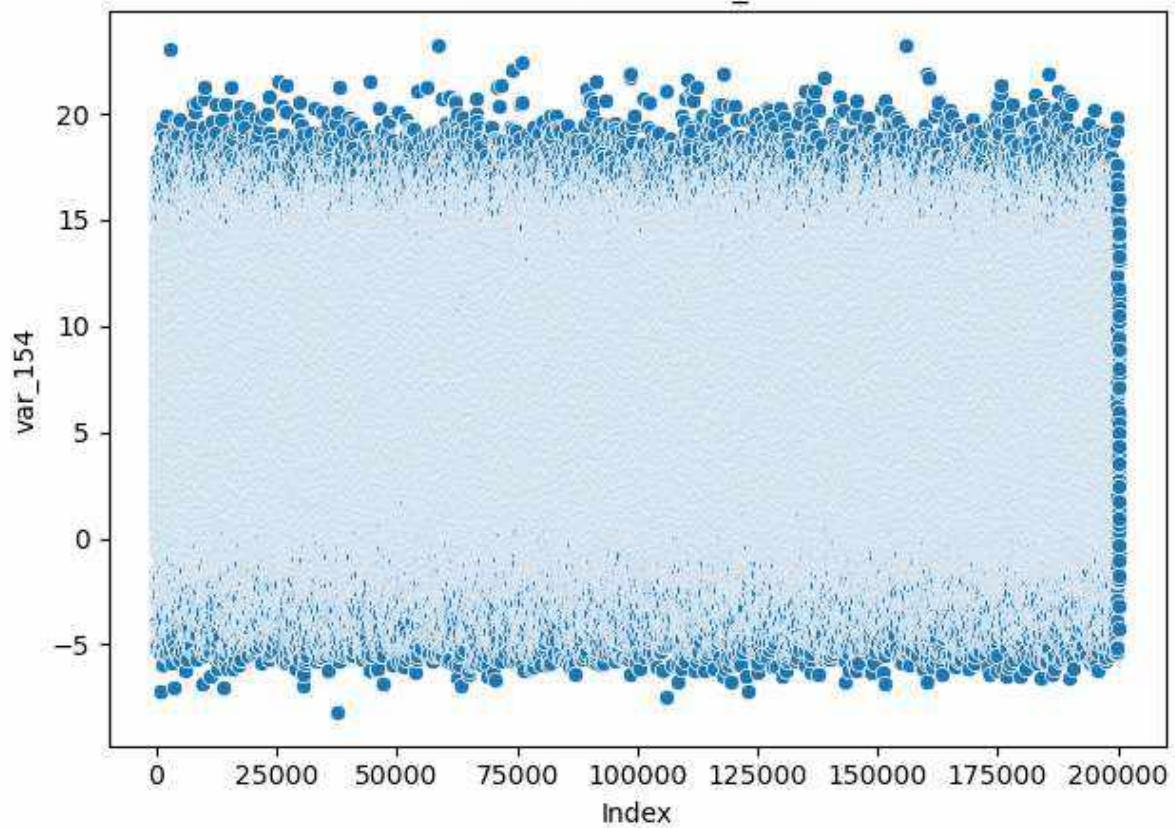
Scatter Plot - var\_152



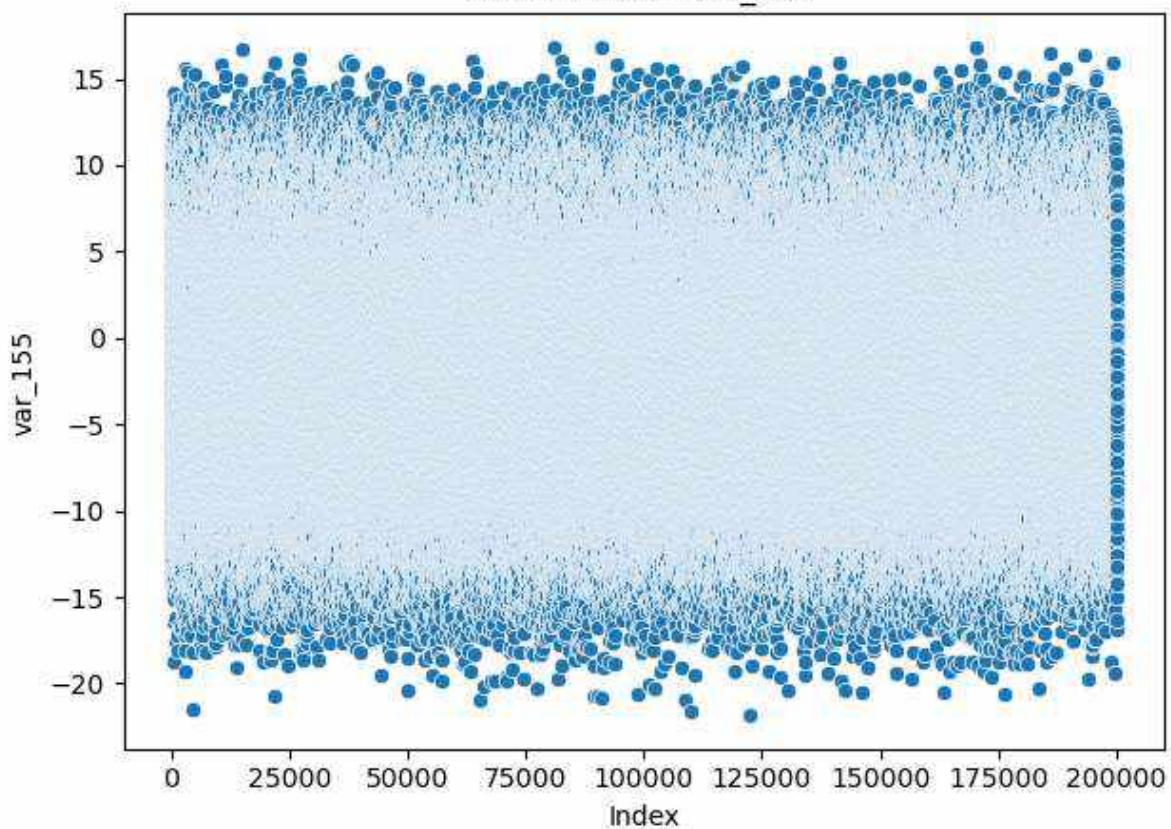
Scatter Plot - var\_153



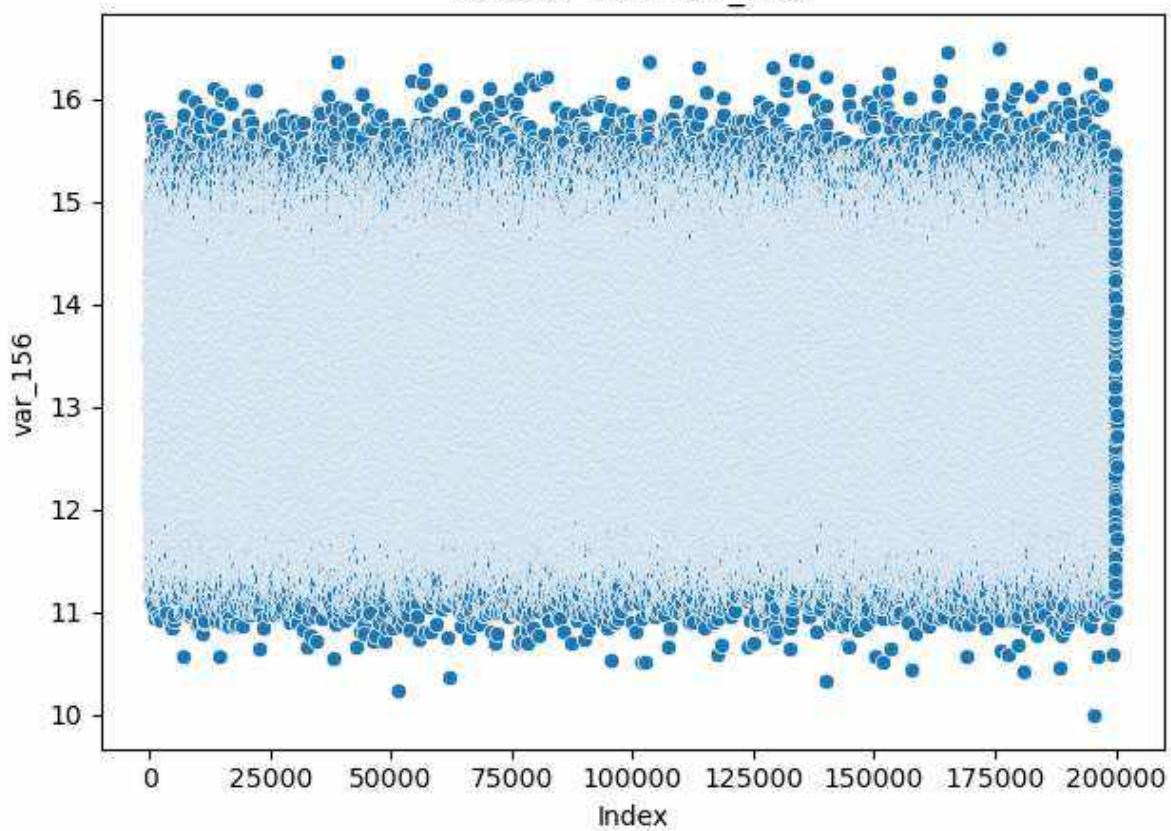
Scatter Plot - var\_154



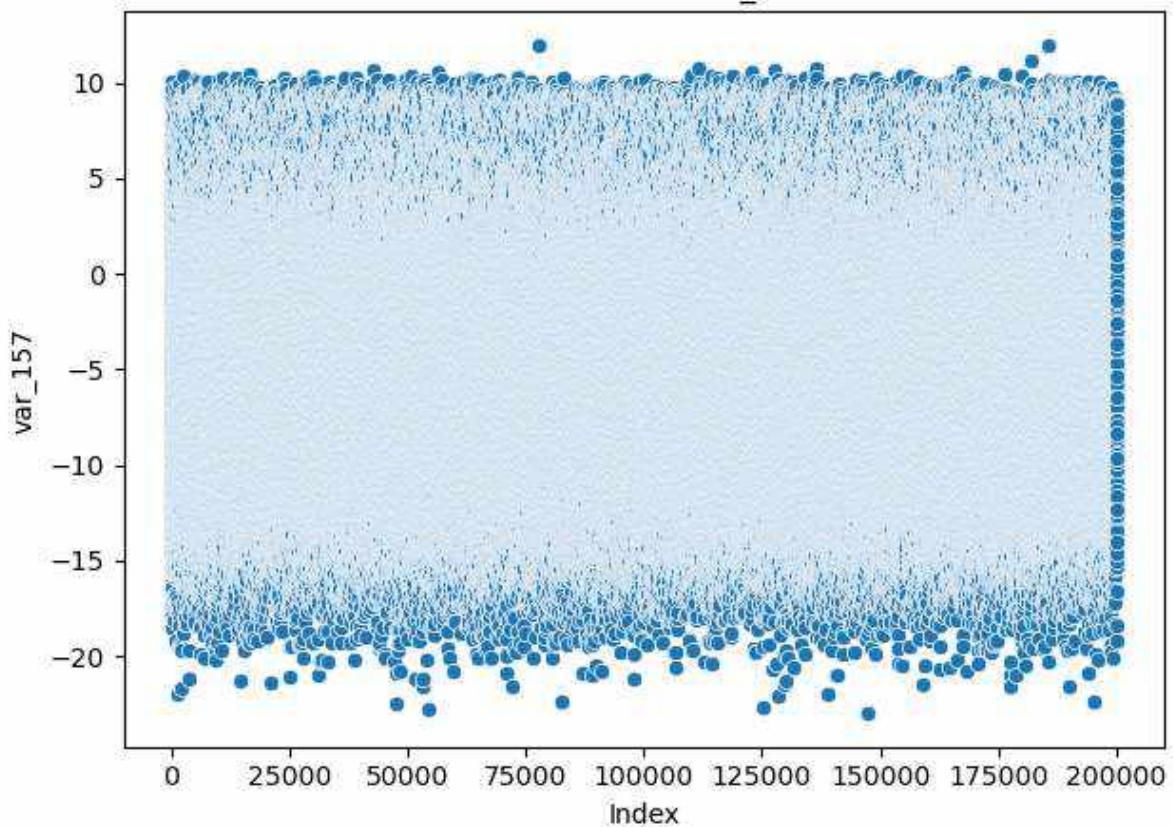
Scatter Plot - var\_155



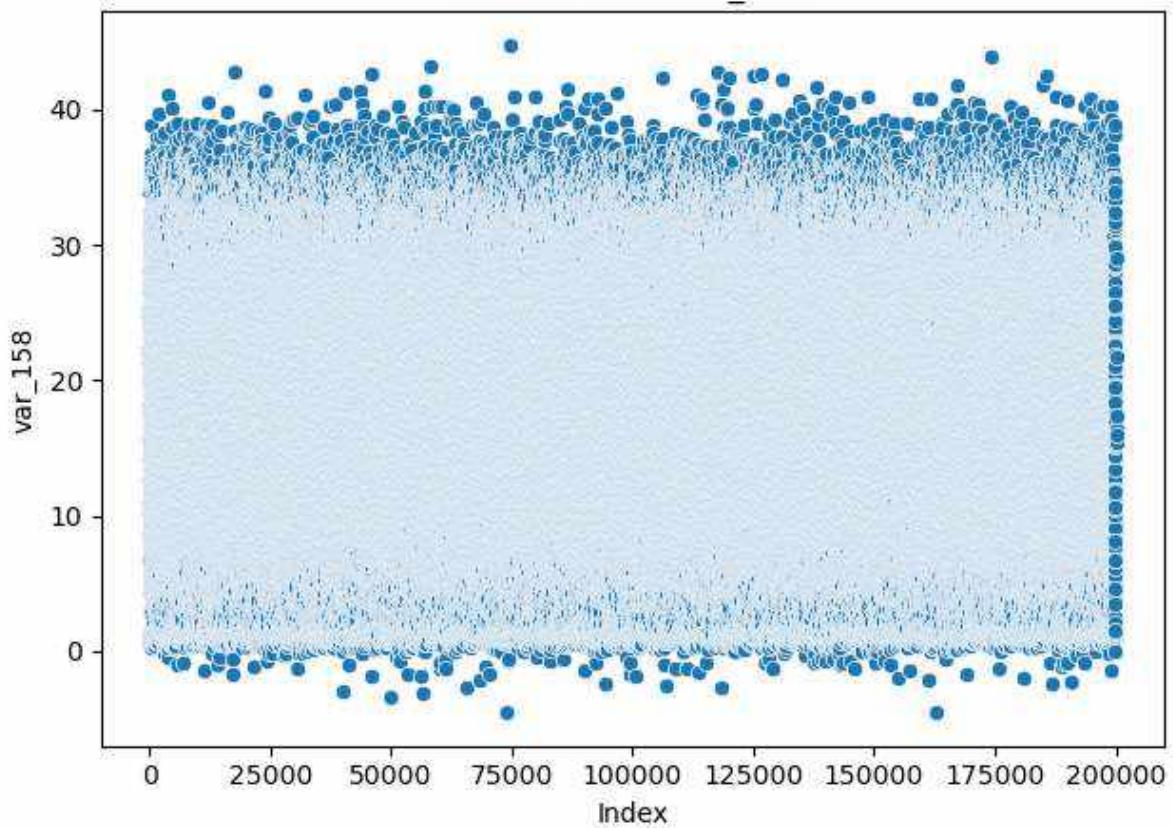
Scatter Plot - var\_156



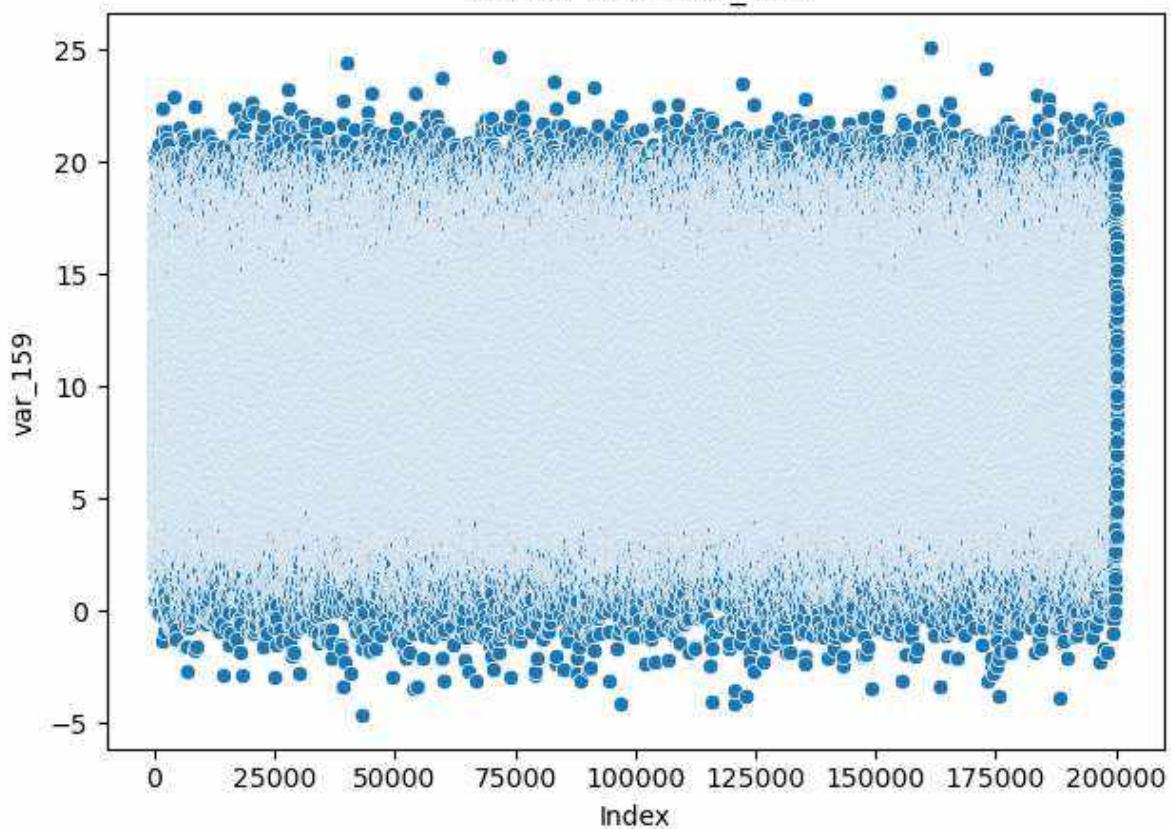
Scatter Plot - var\_157



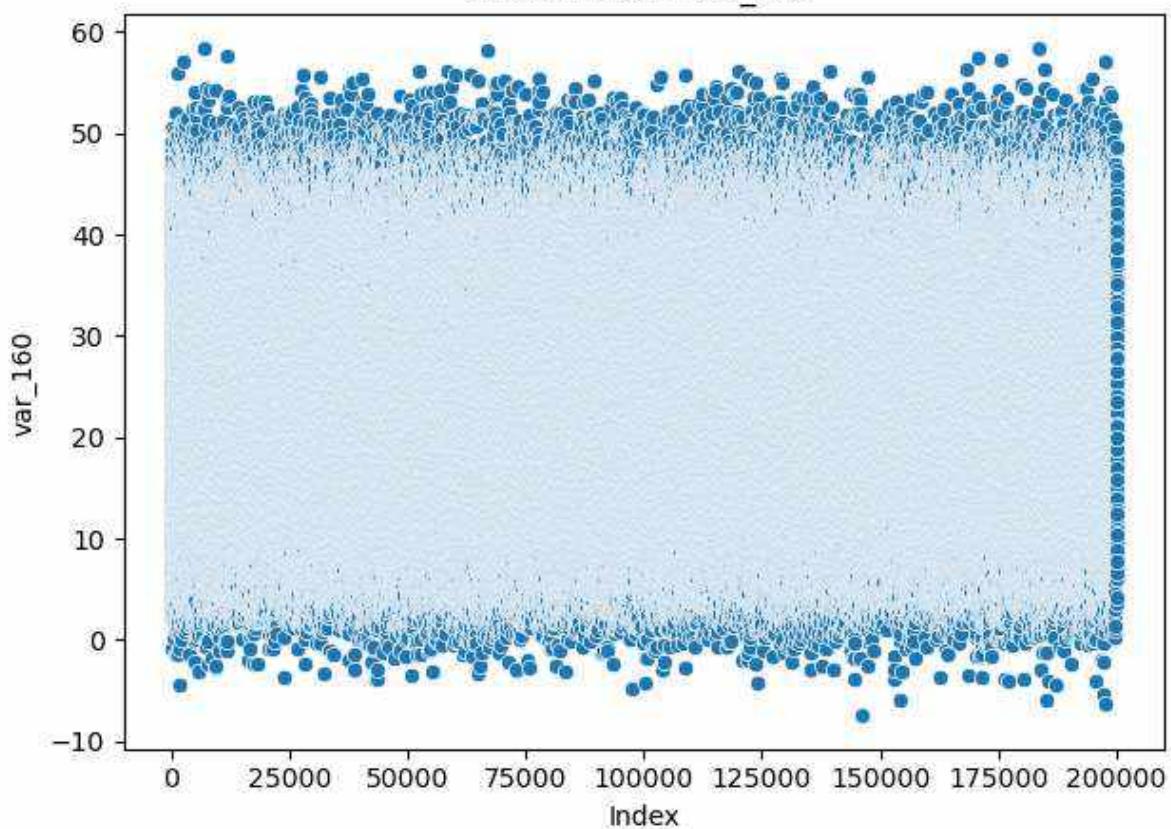
Scatter Plot - var\_158



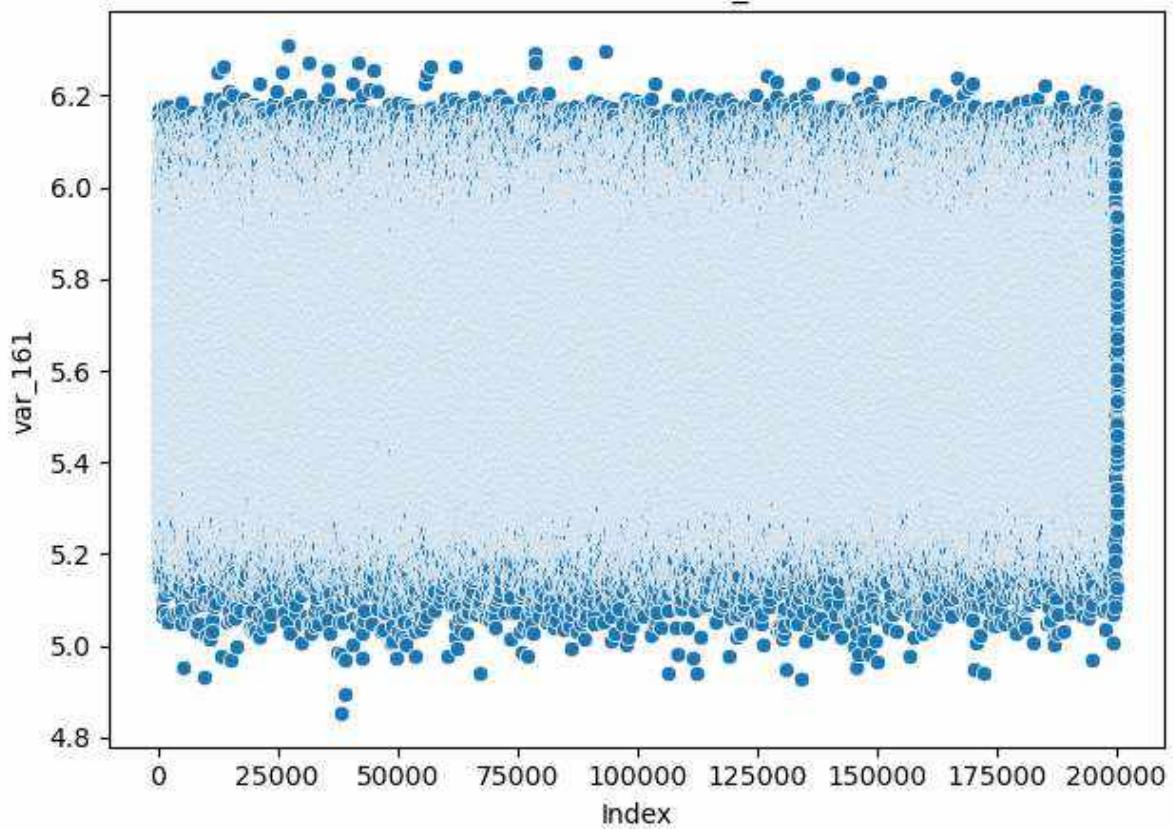
Scatter Plot - var\_159



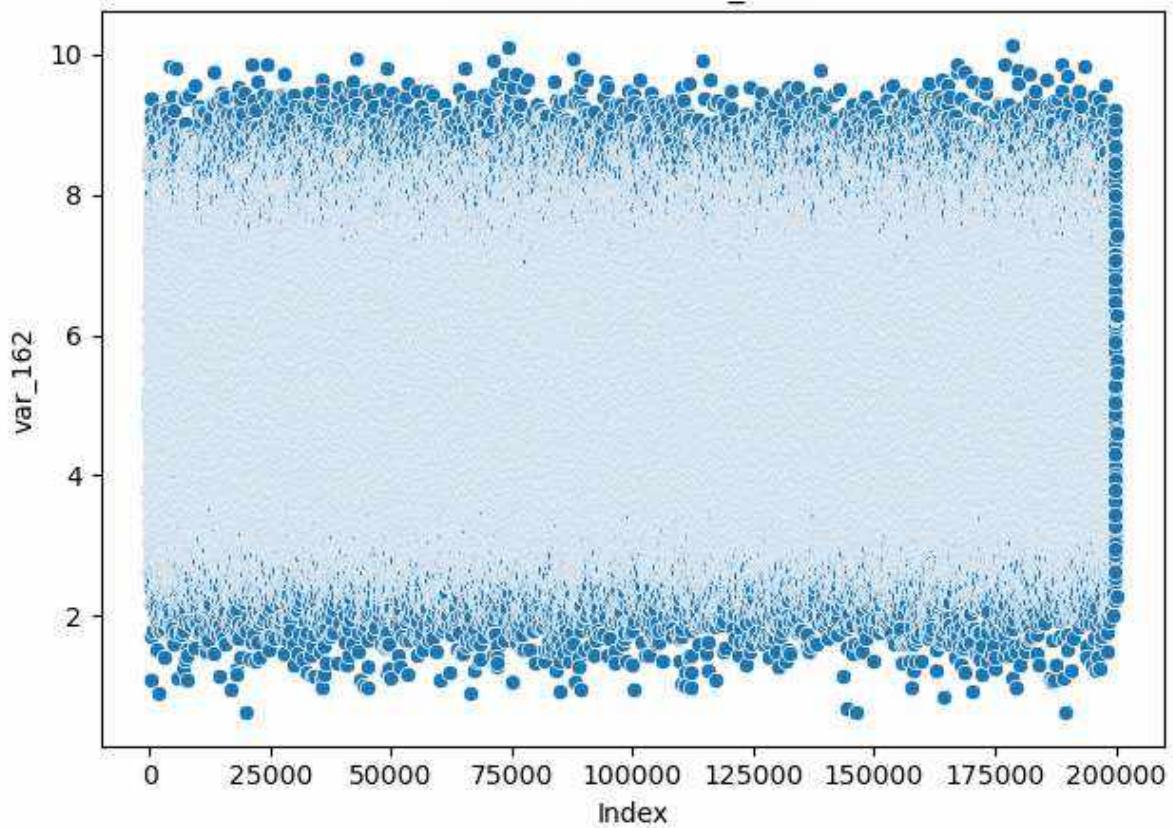
Scatter Plot - var\_160



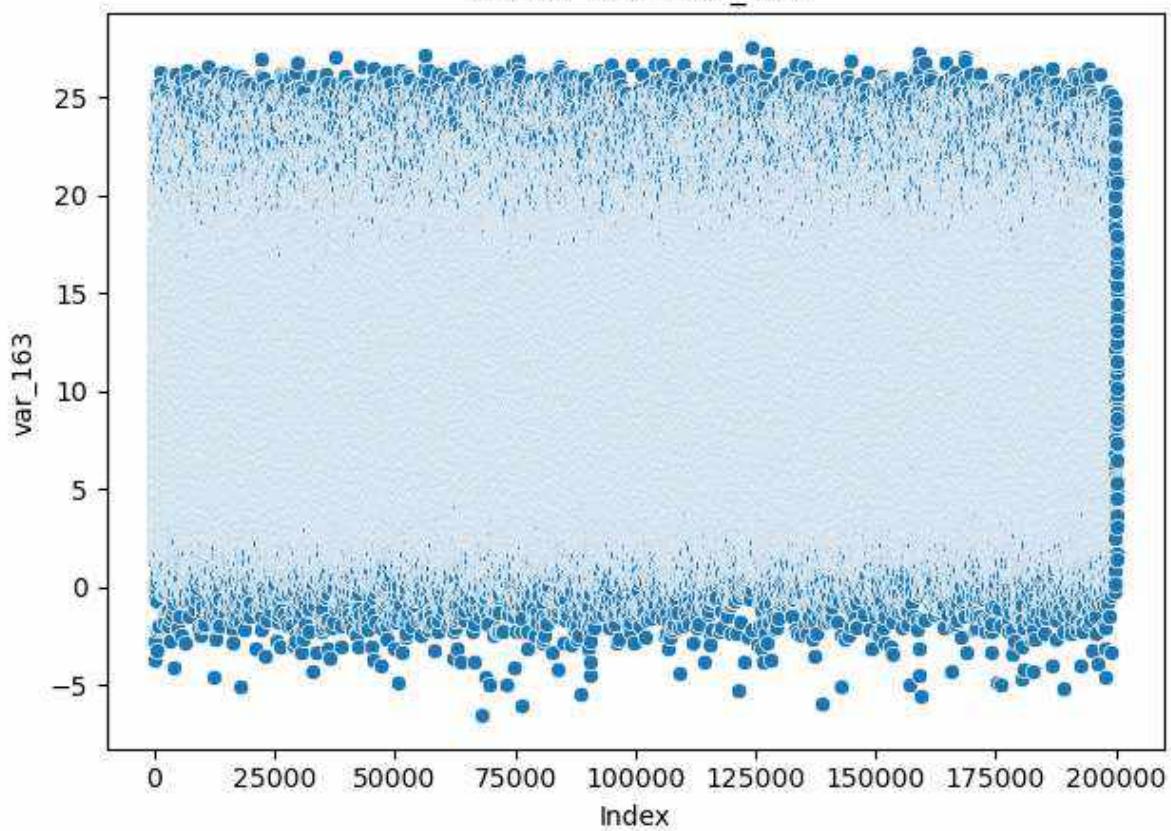
Scatter Plot - var\_161



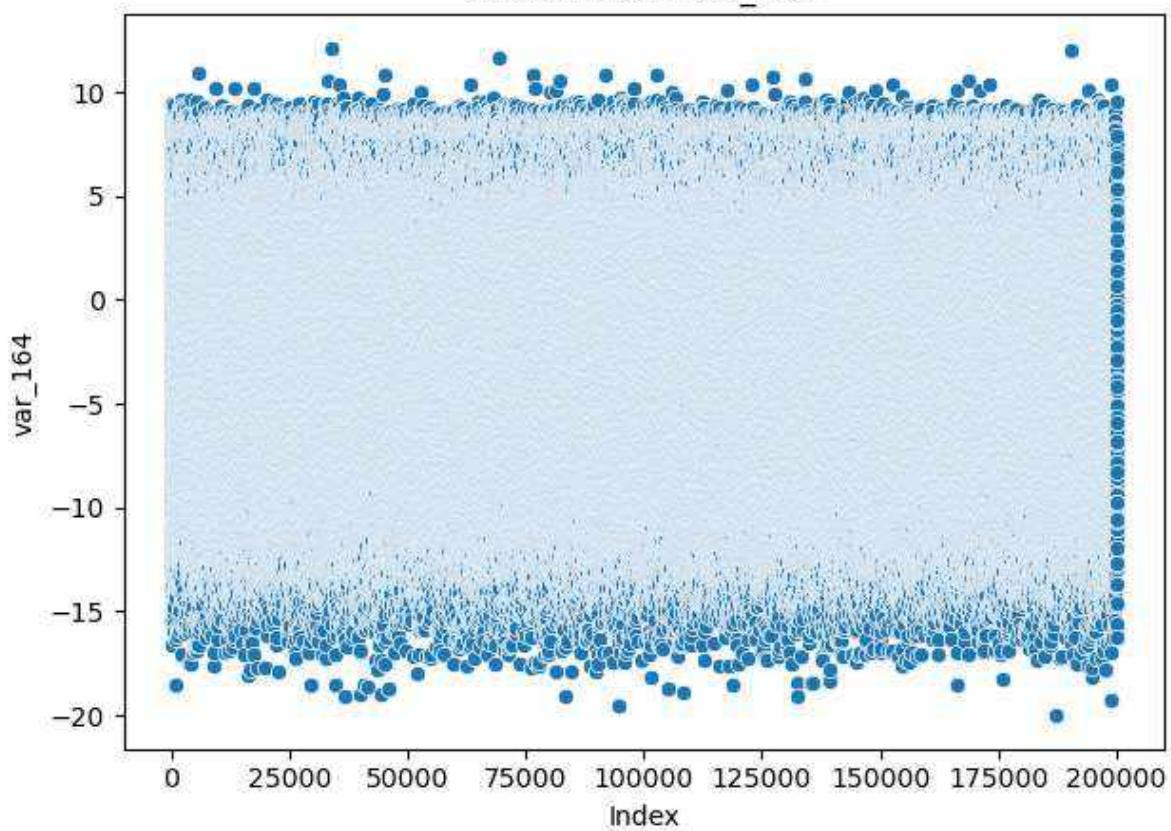
Scatter Plot - var\_162

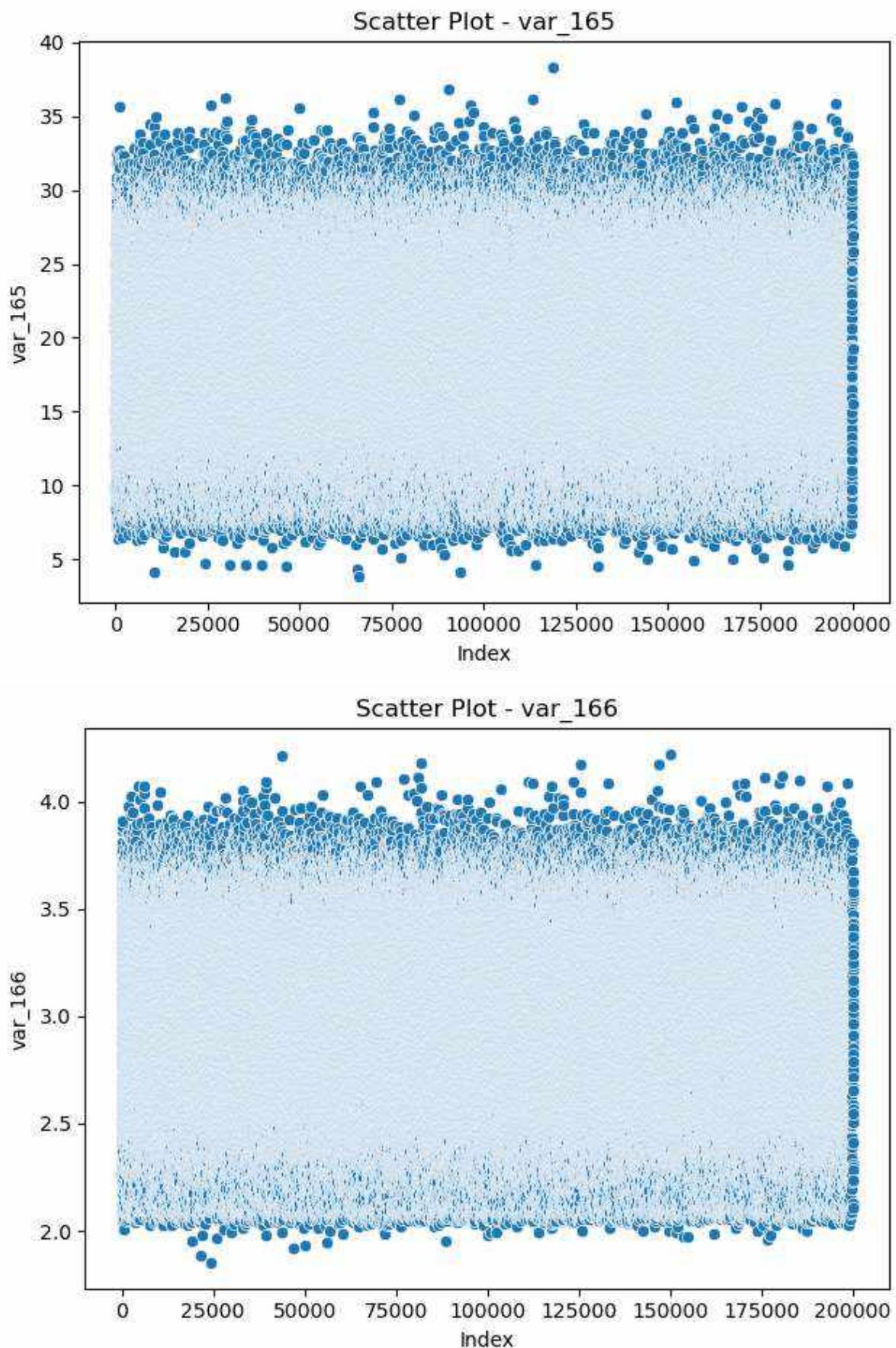


Scatter Plot - var\_163

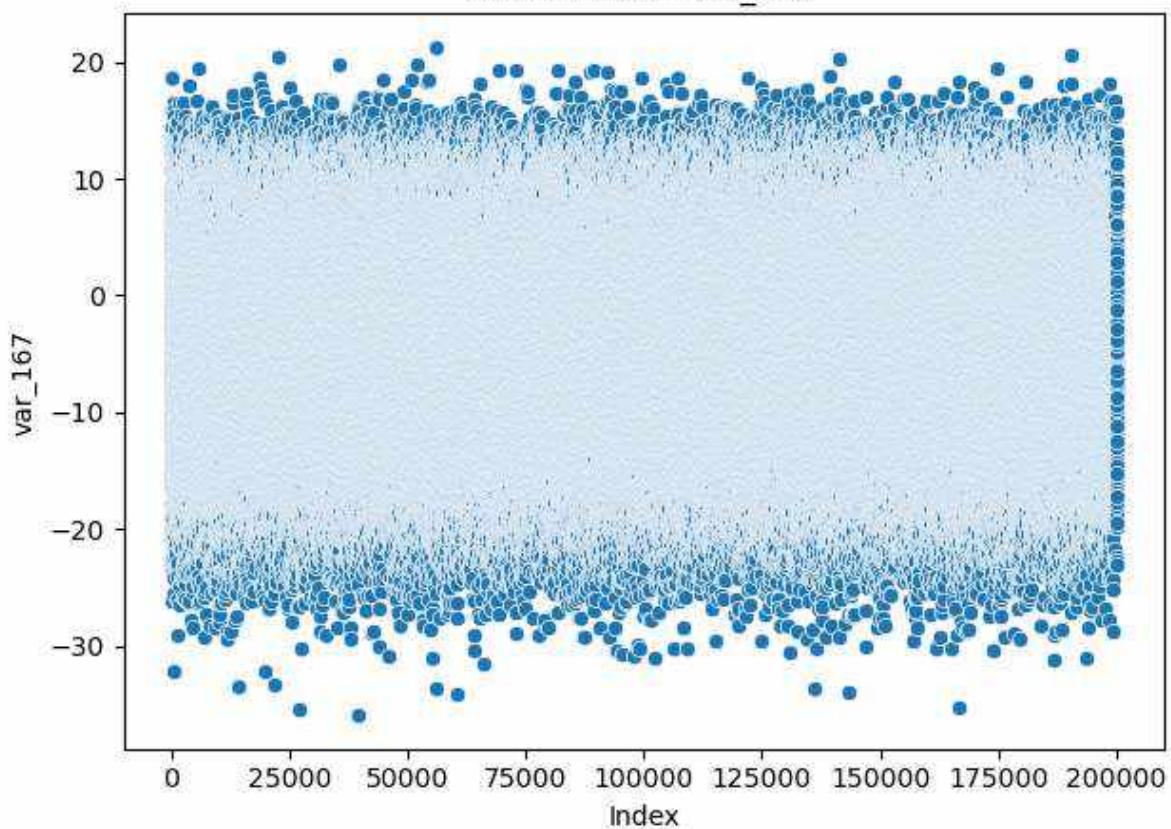


Scatter Plot - var\_164

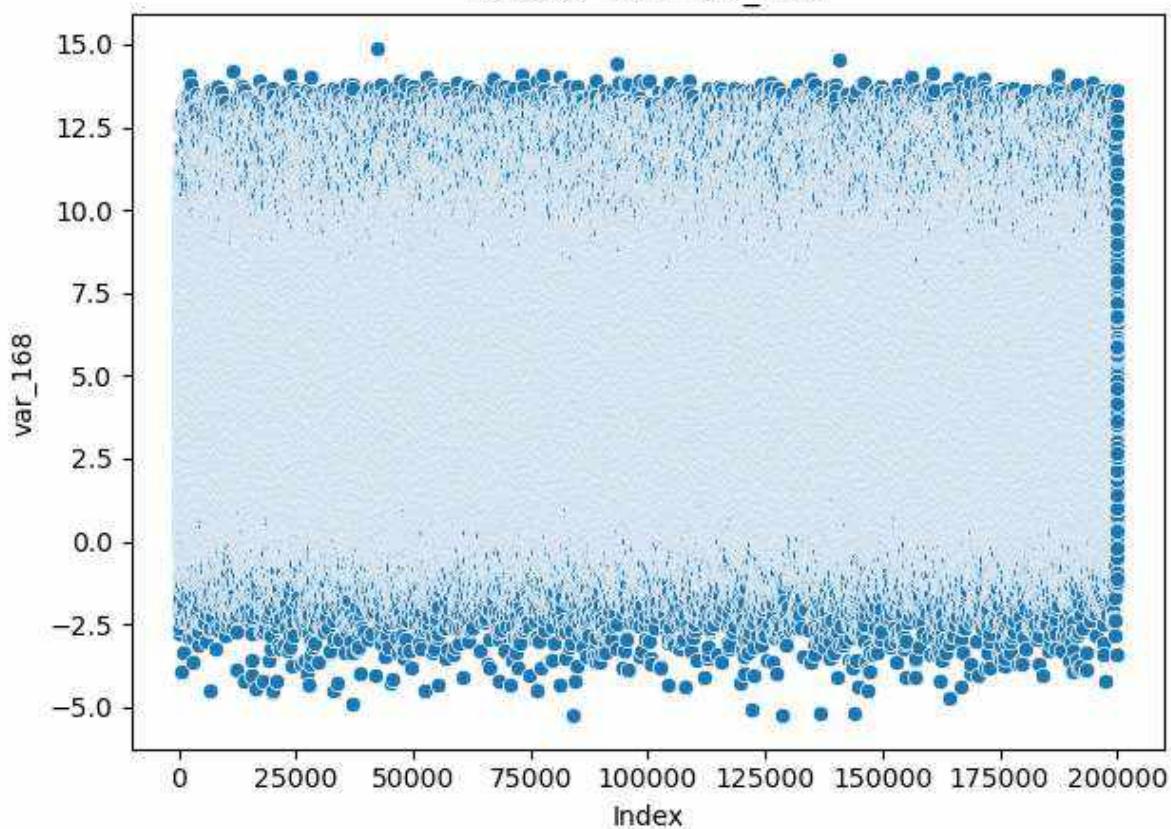




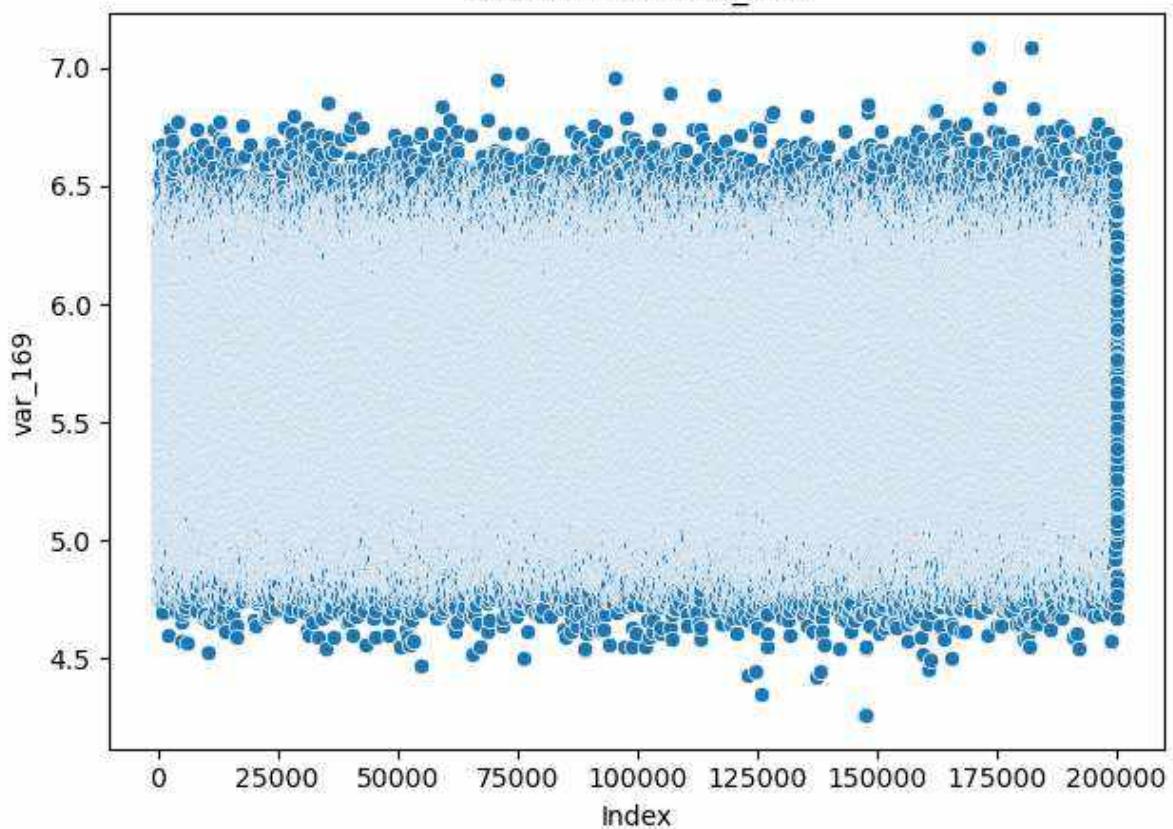
Scatter Plot - var\_167



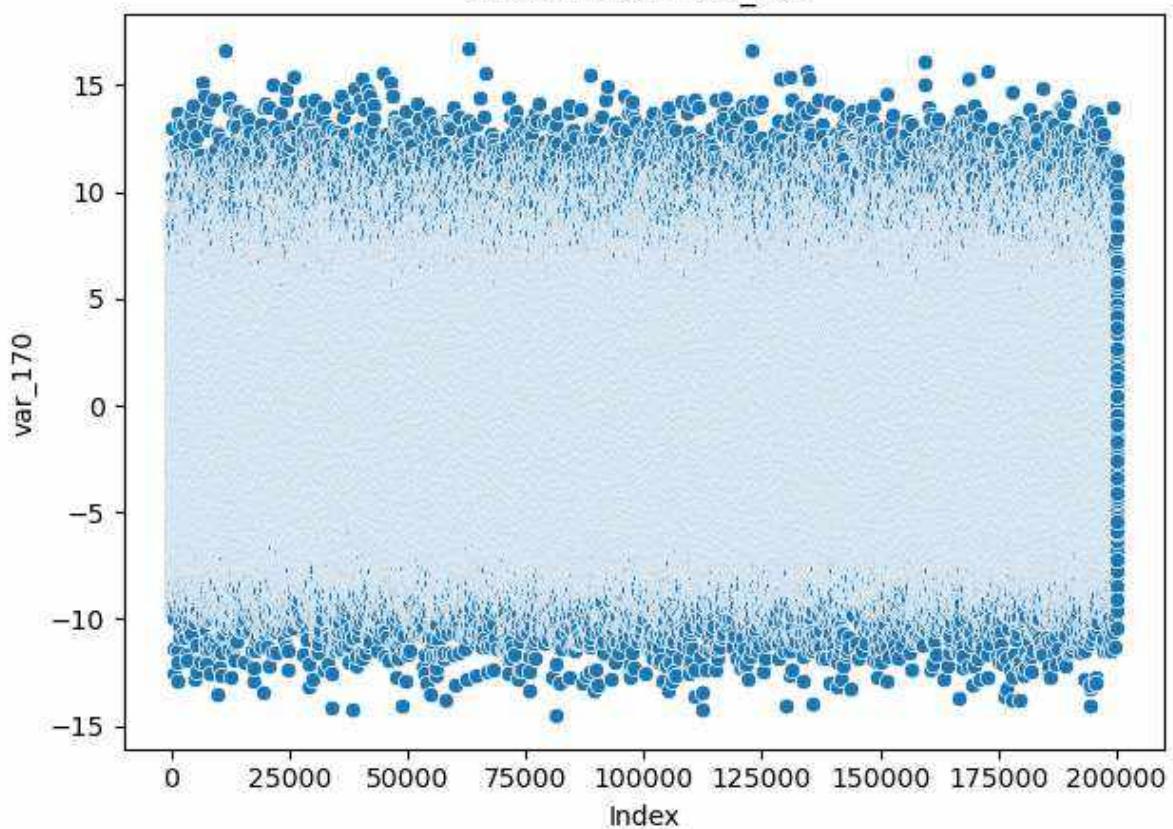
Scatter Plot - var\_168



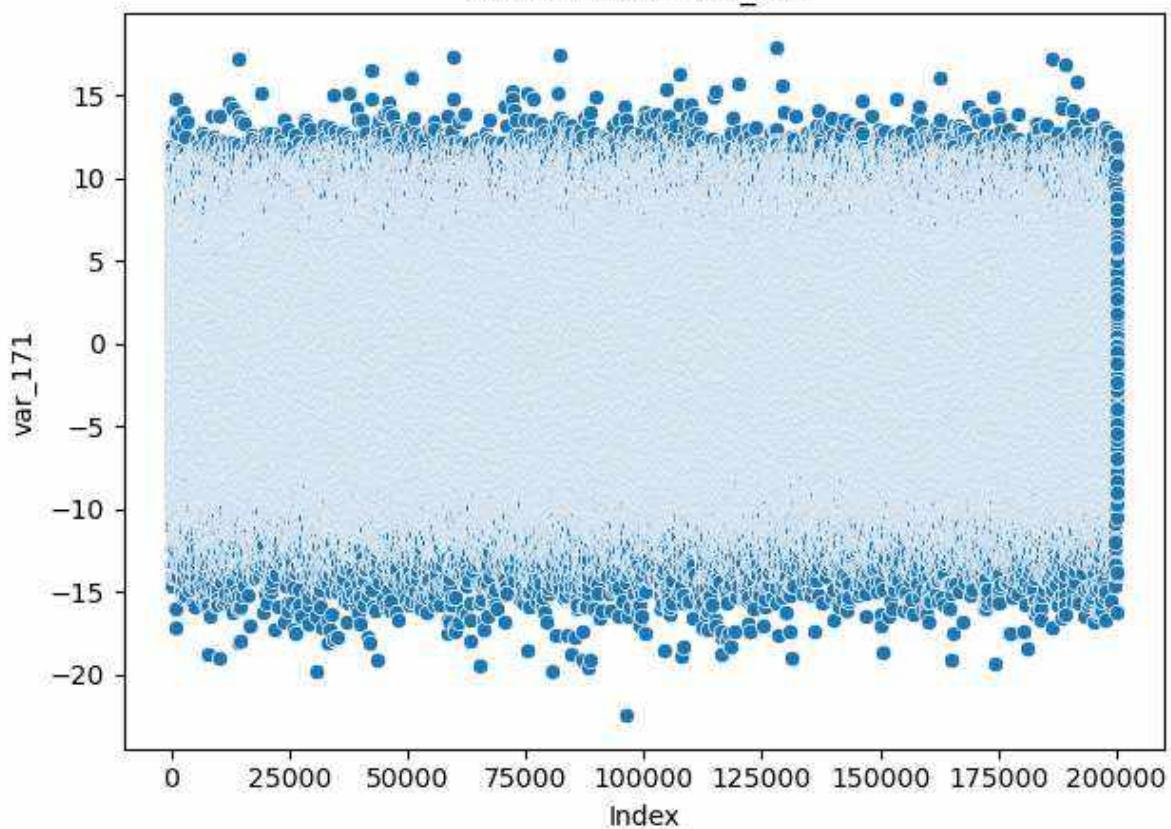
Scatter Plot - var\_169



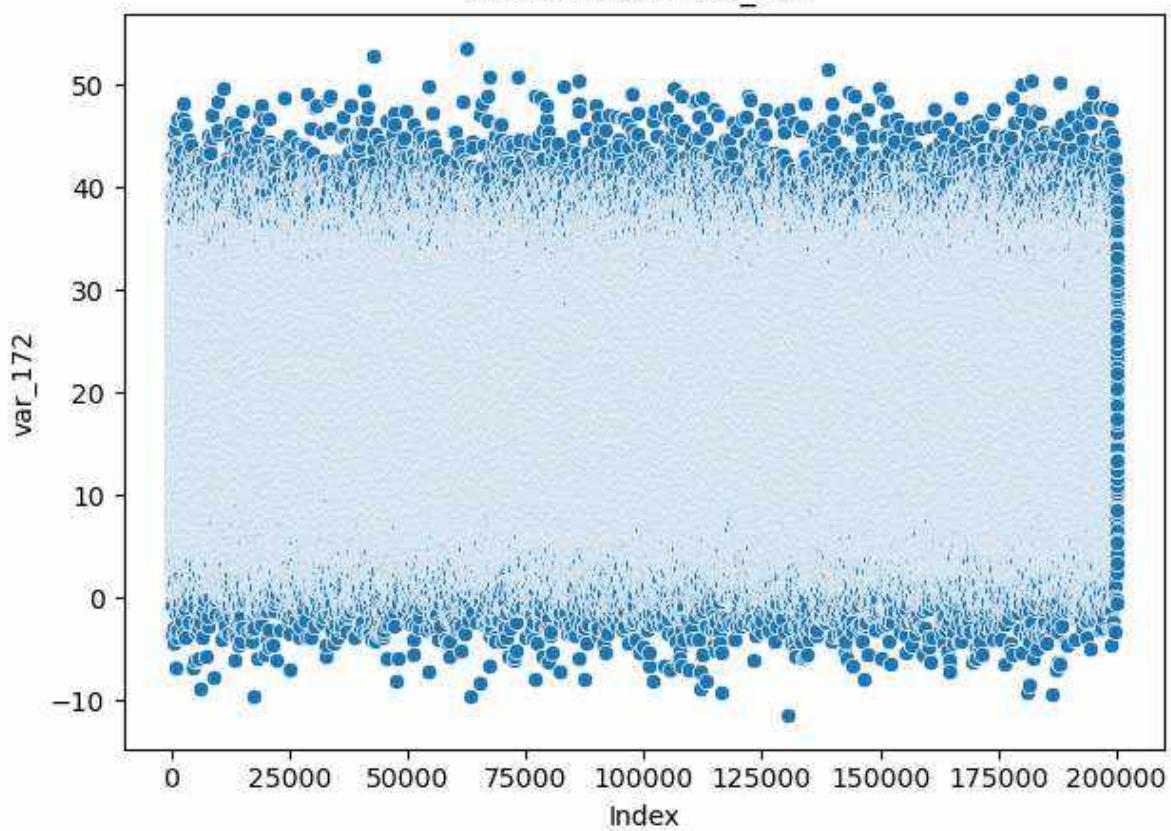
Scatter Plot - var\_170

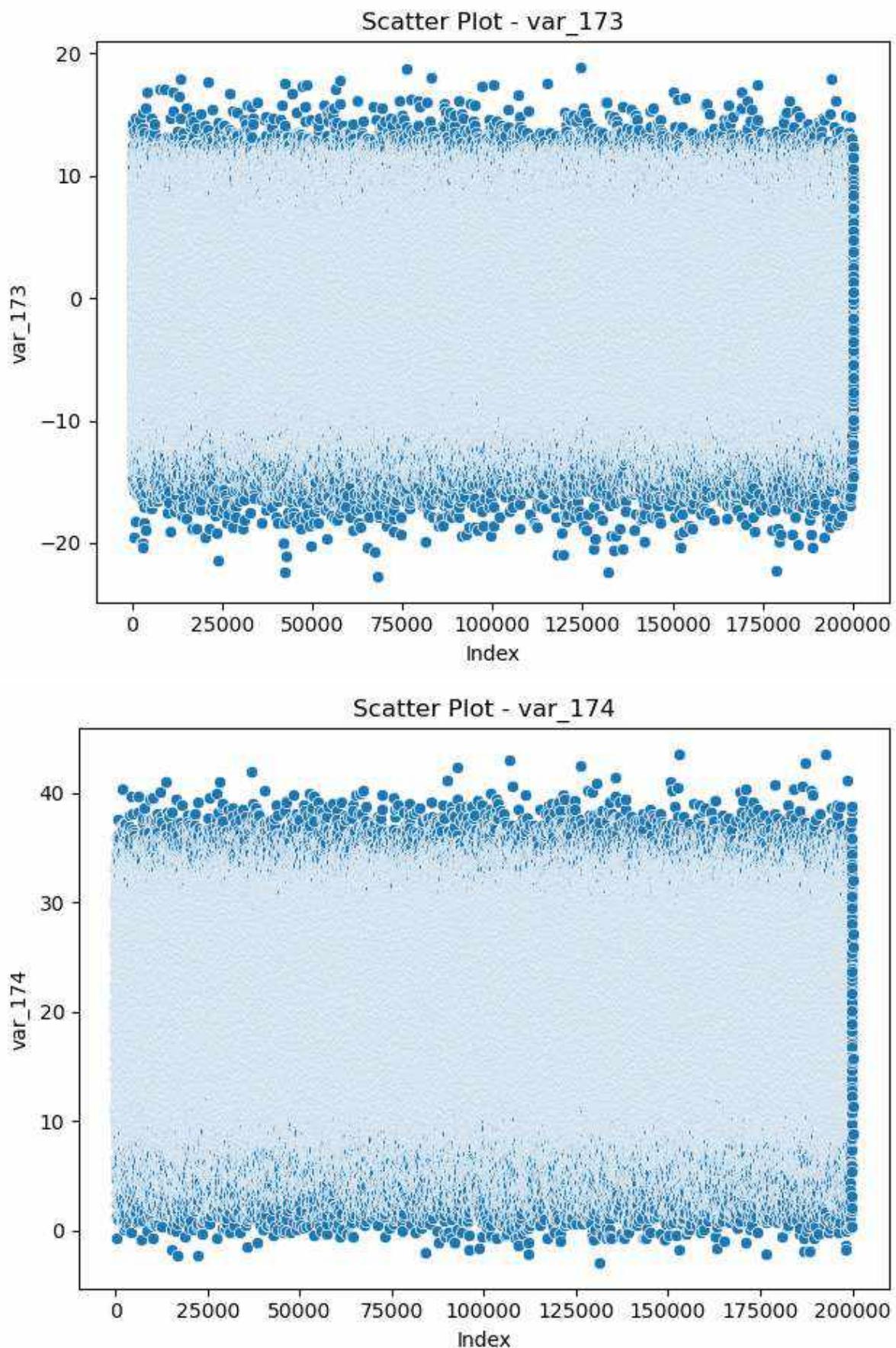


Scatter Plot - var\_171

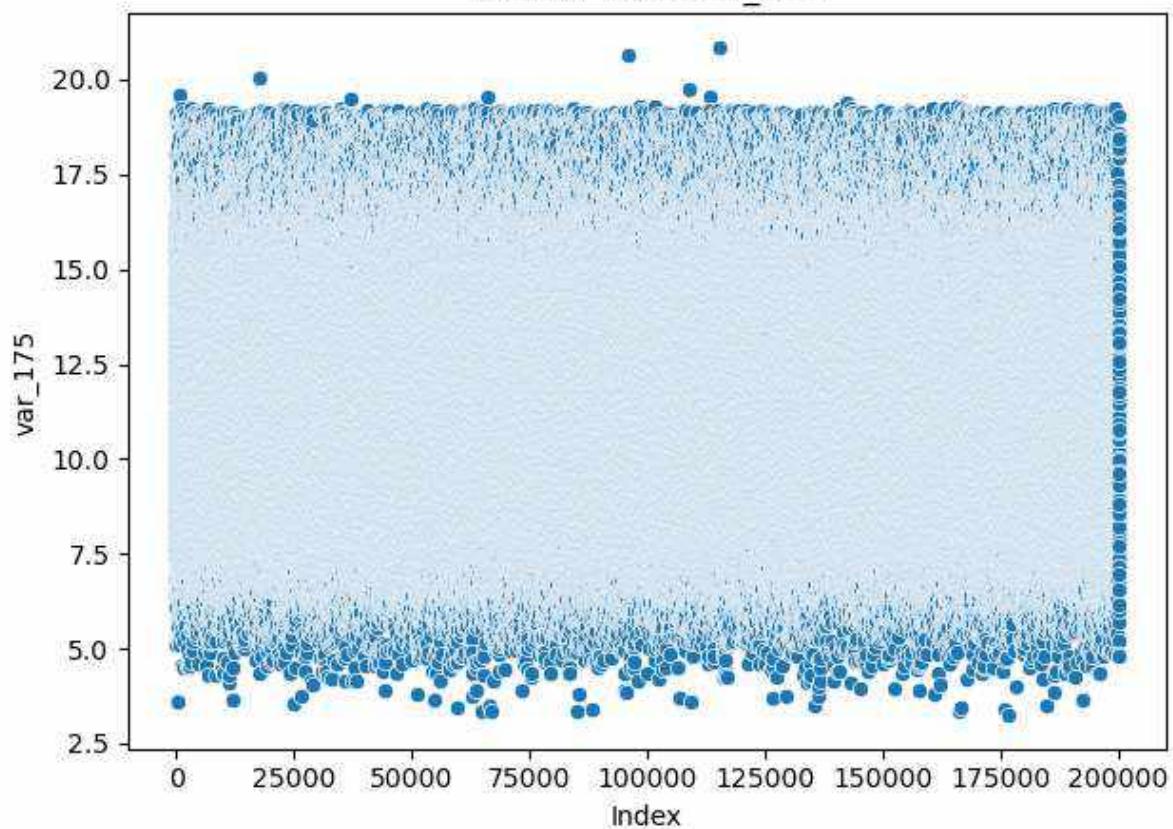


Scatter Plot - var\_172

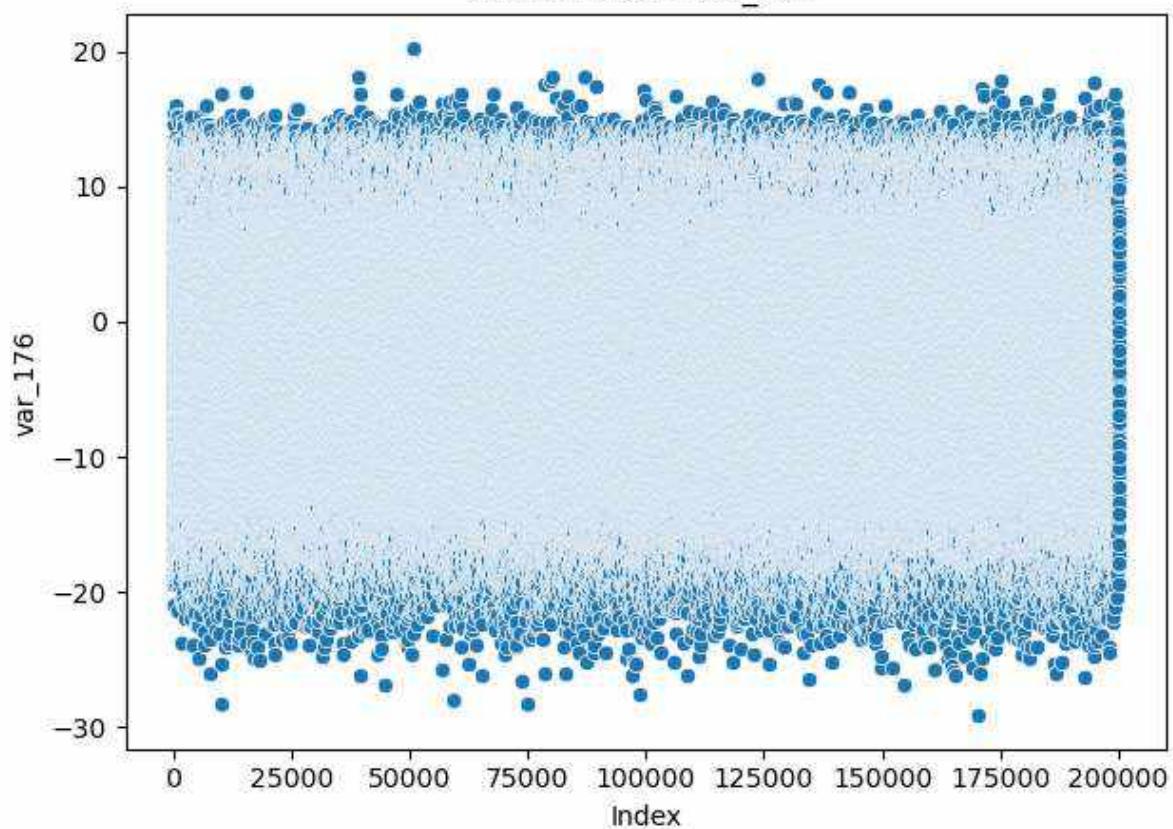




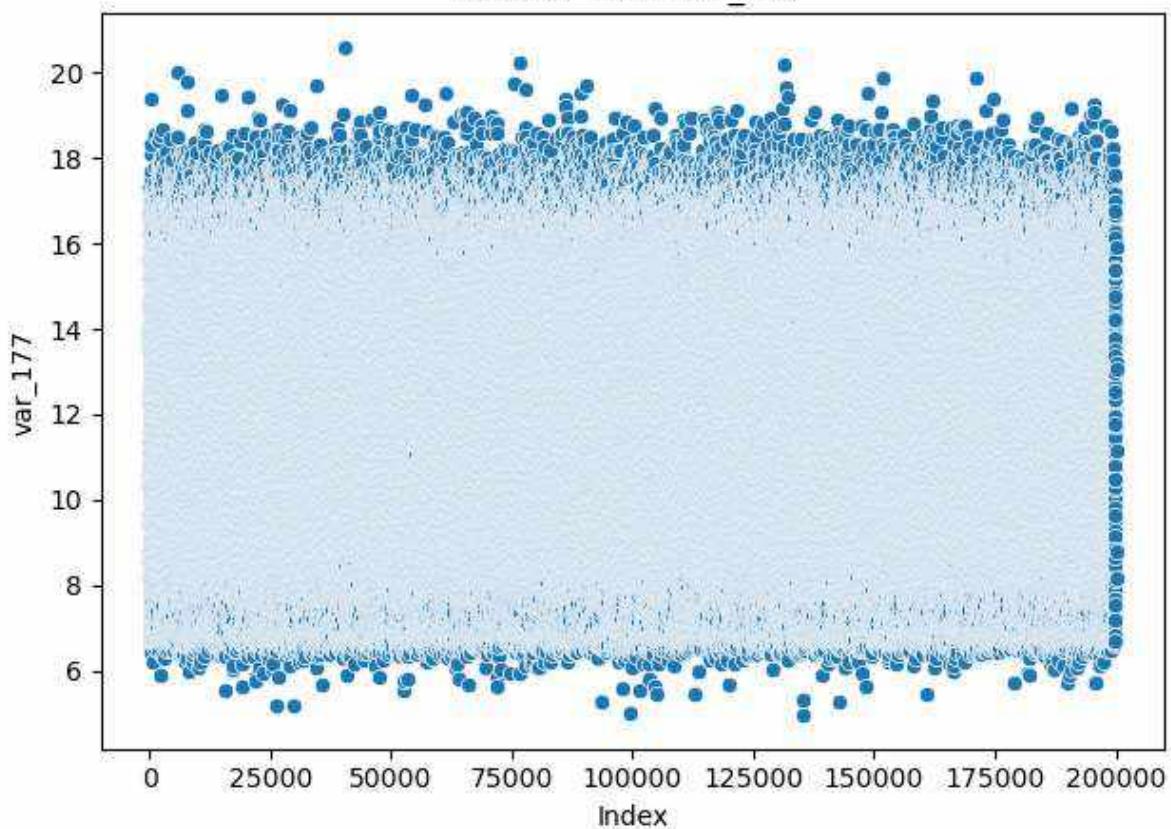
Scatter Plot - var\_175



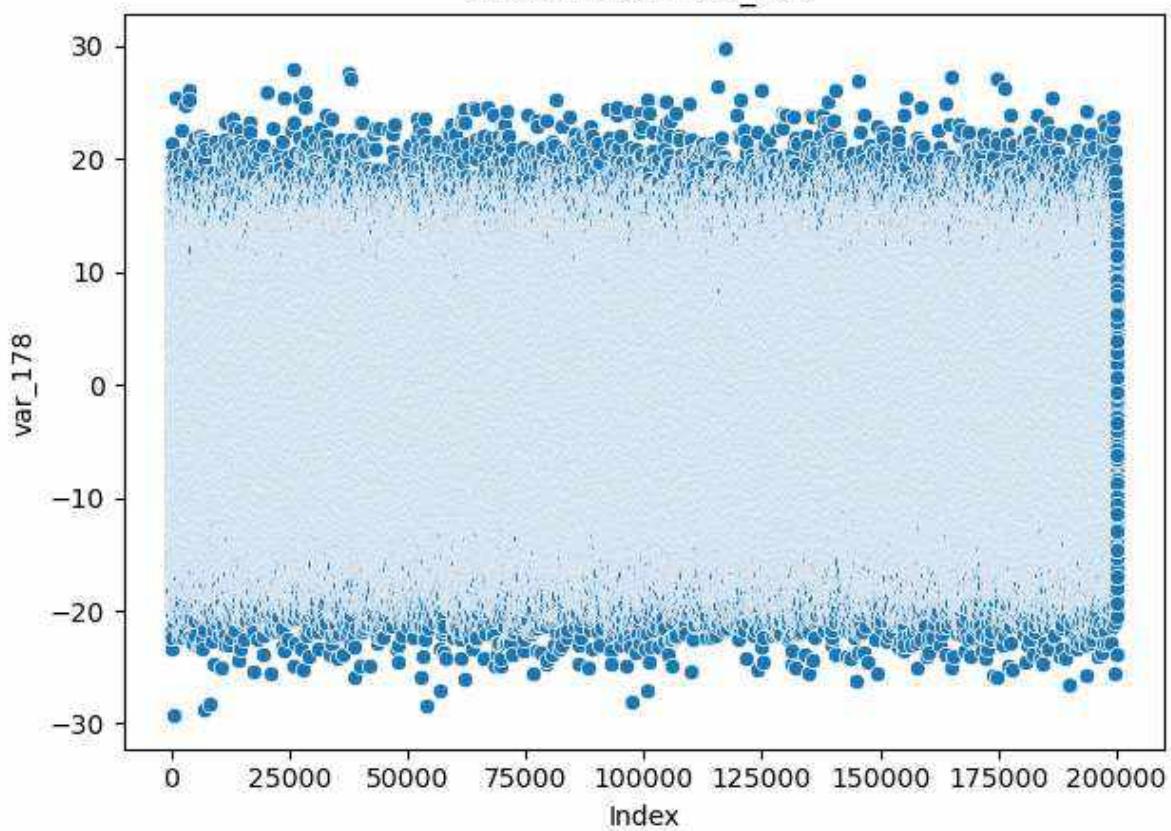
Scatter Plot - var\_176



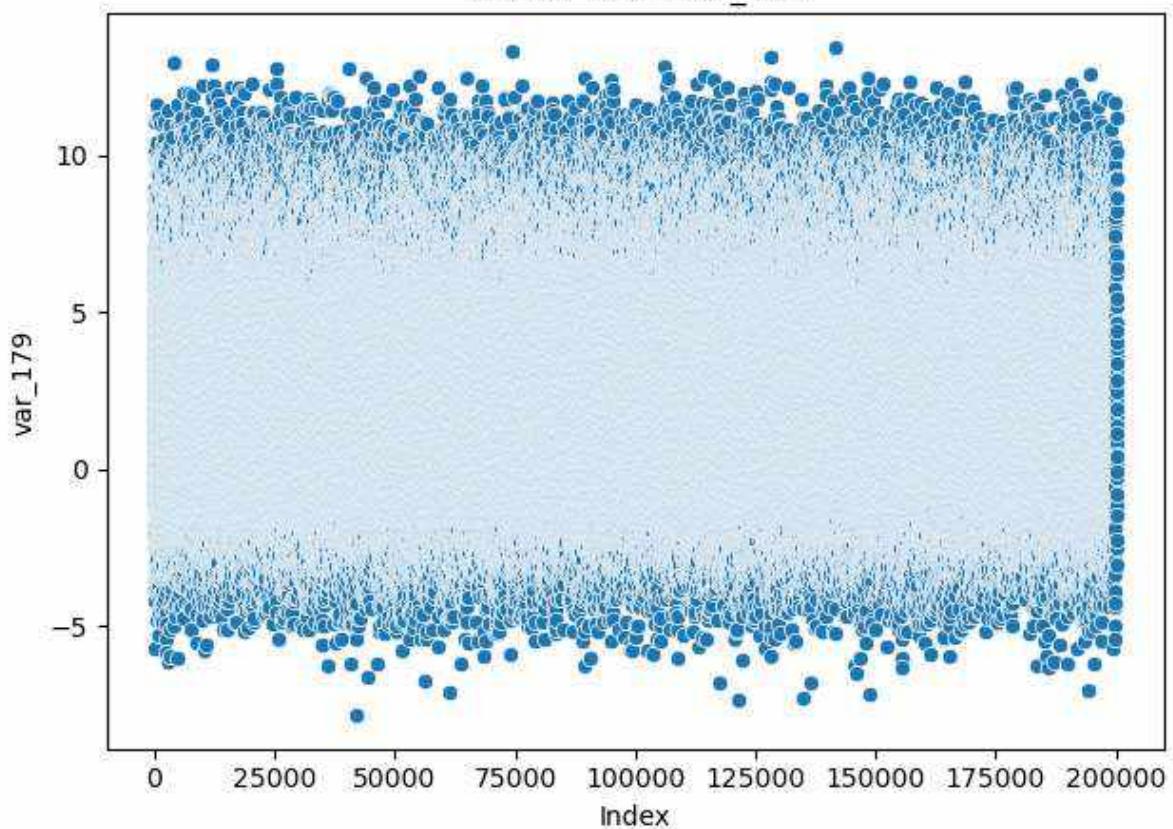
Scatter Plot - var\_177



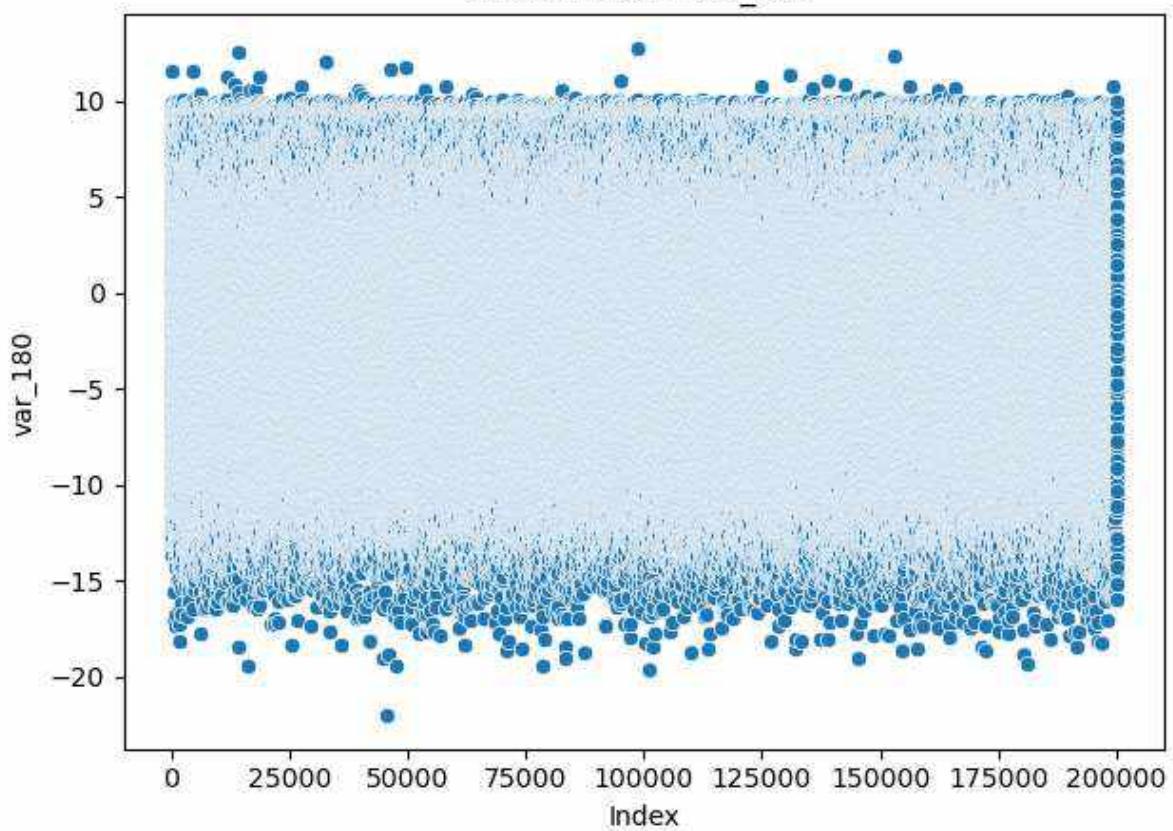
Scatter Plot - var\_178



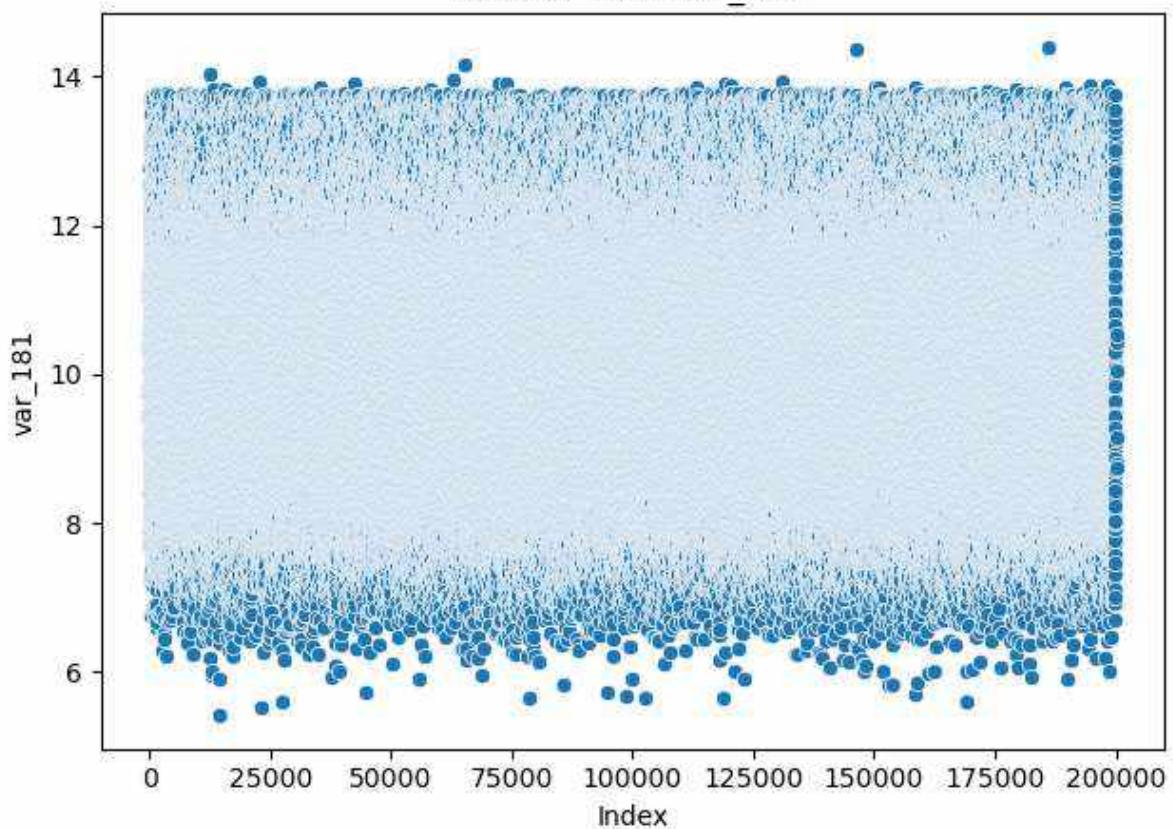
Scatter Plot - var\_179



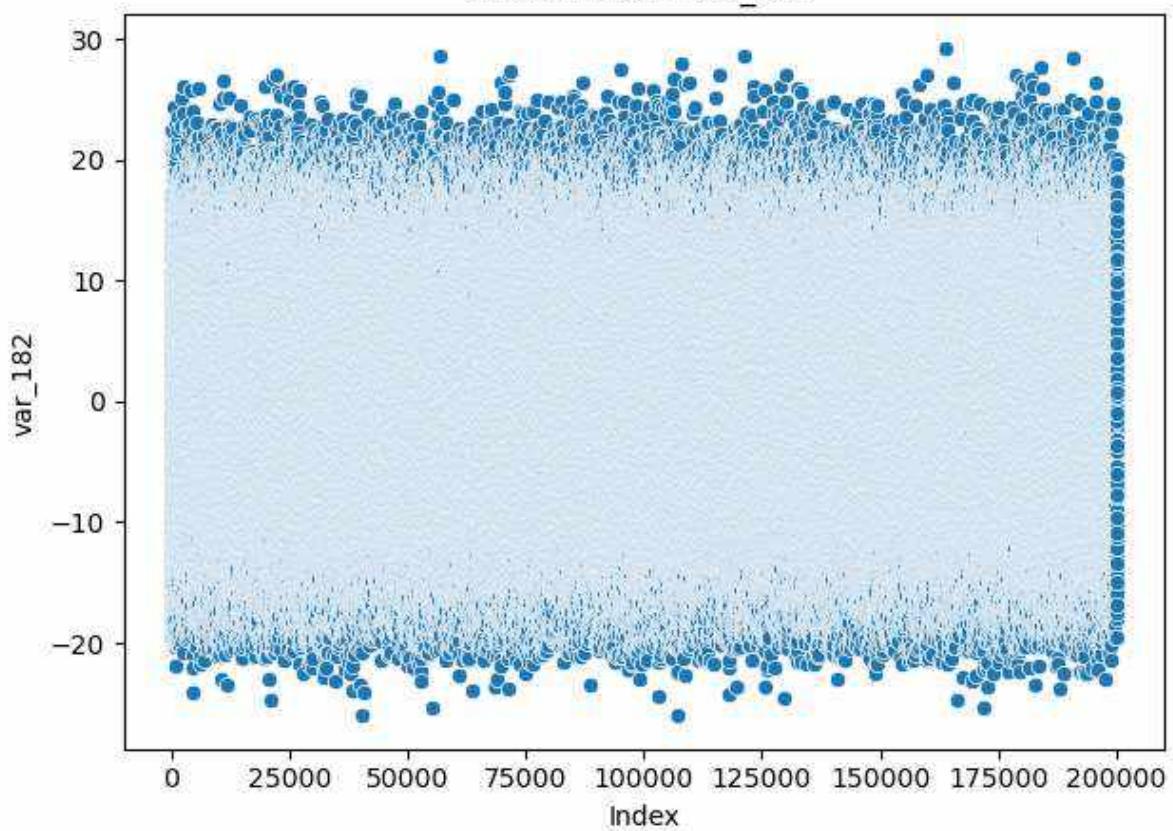
Scatter Plot - var\_180

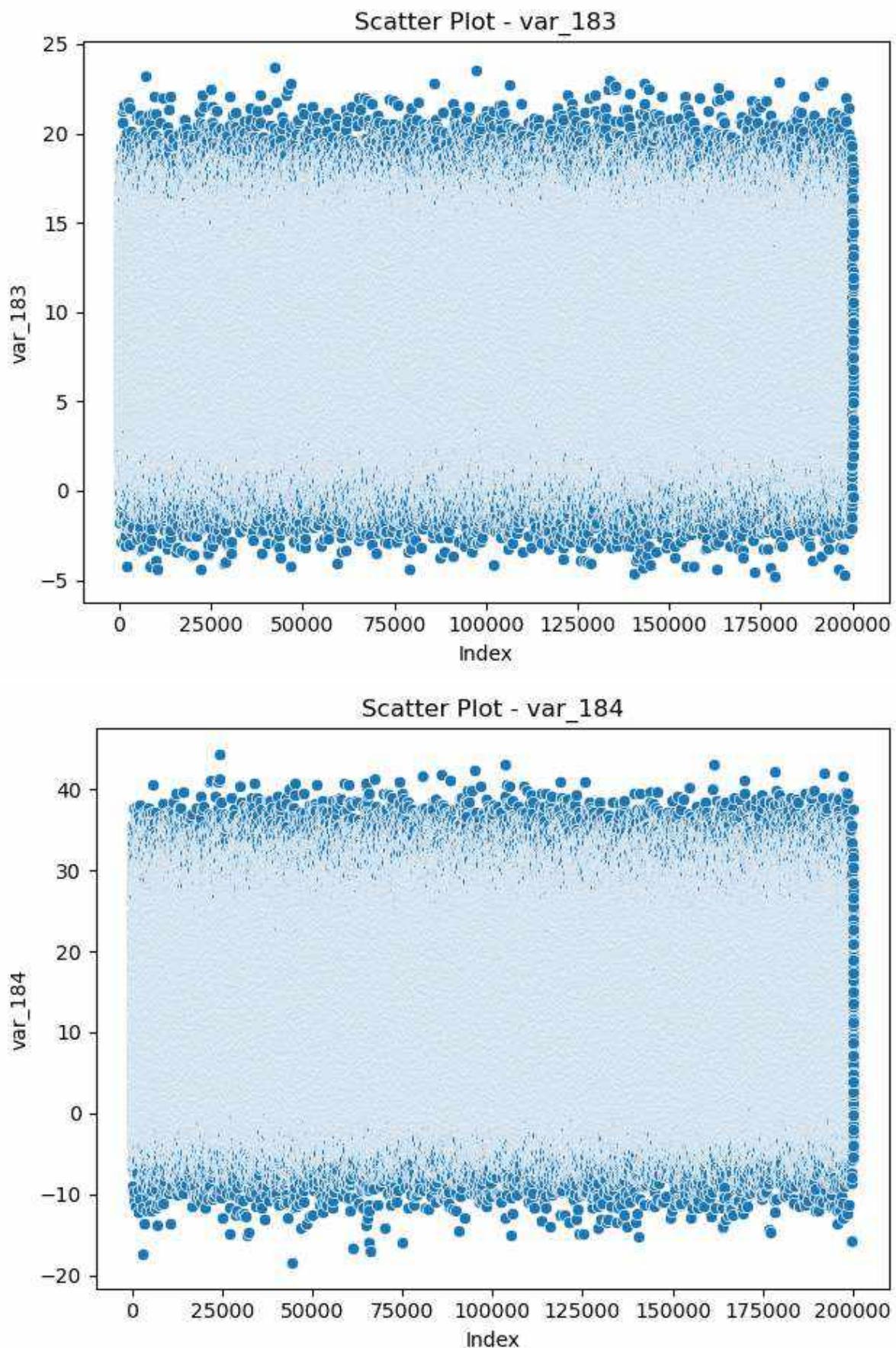


Scatter Plot - var\_181

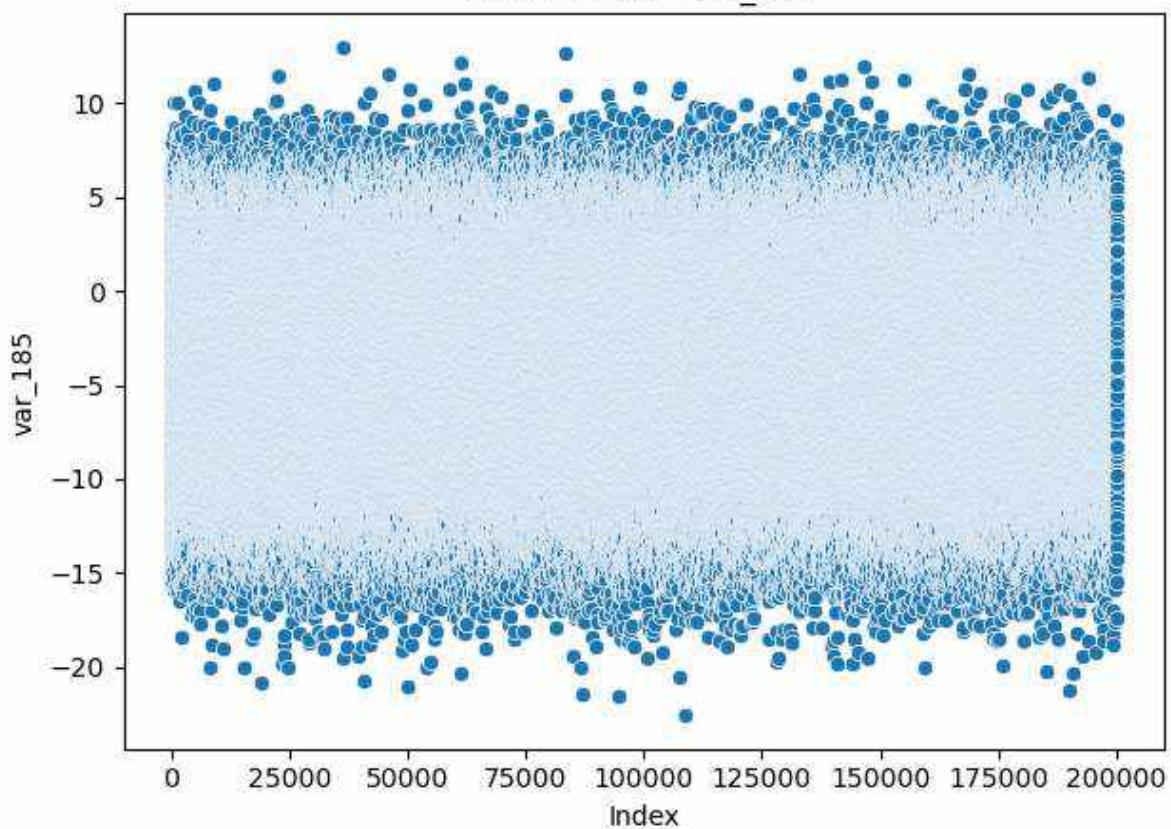


Scatter Plot - var\_182

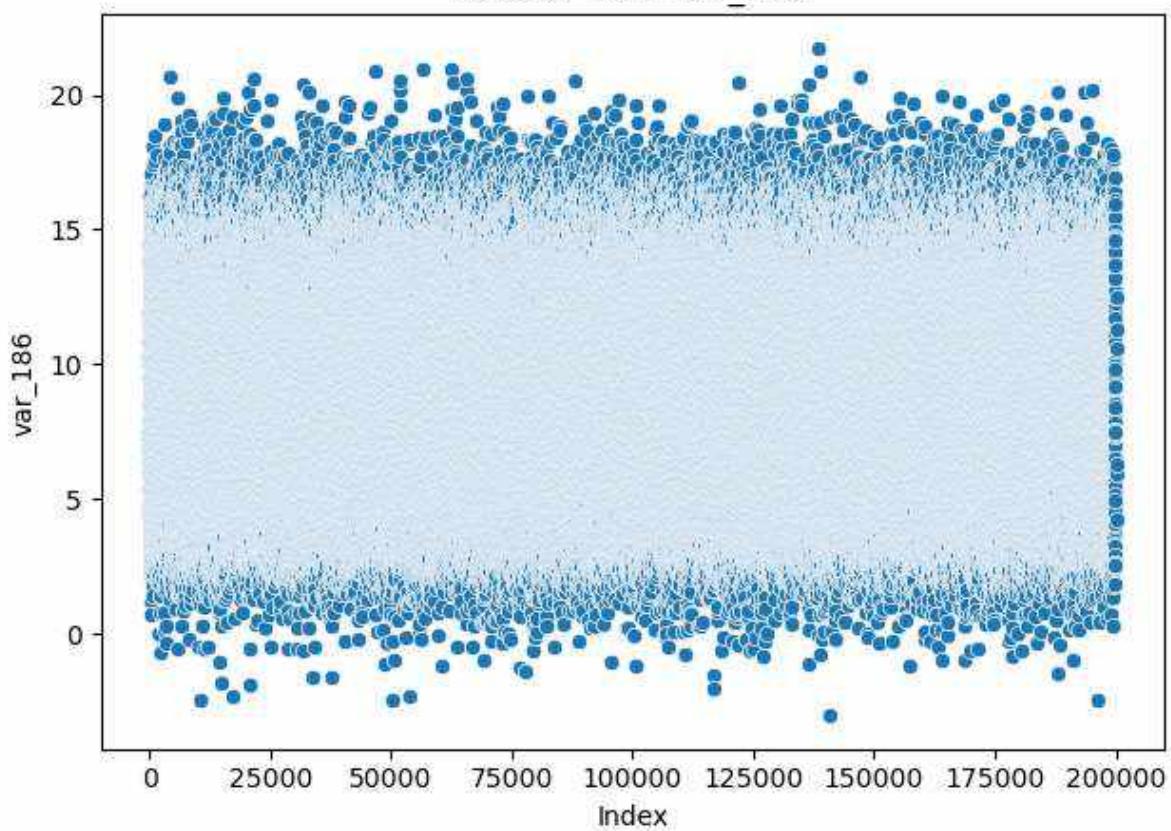




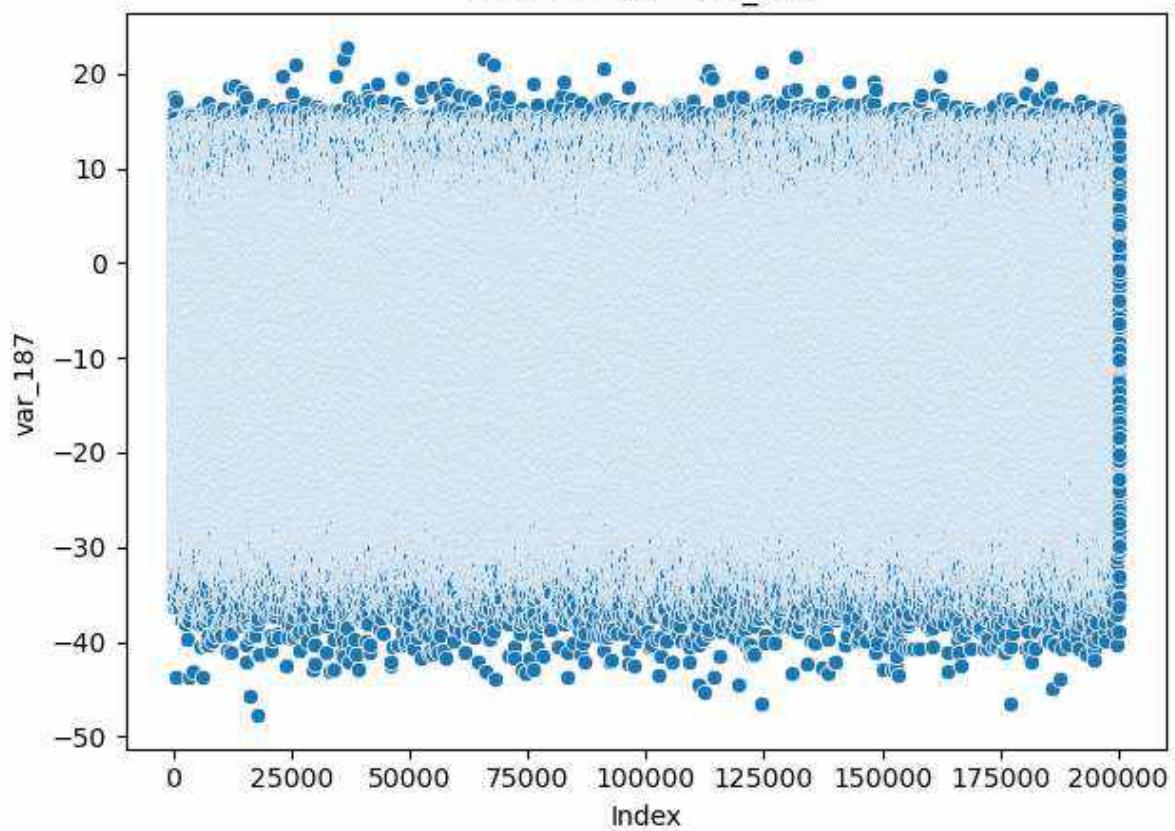
Scatter Plot - var\_185



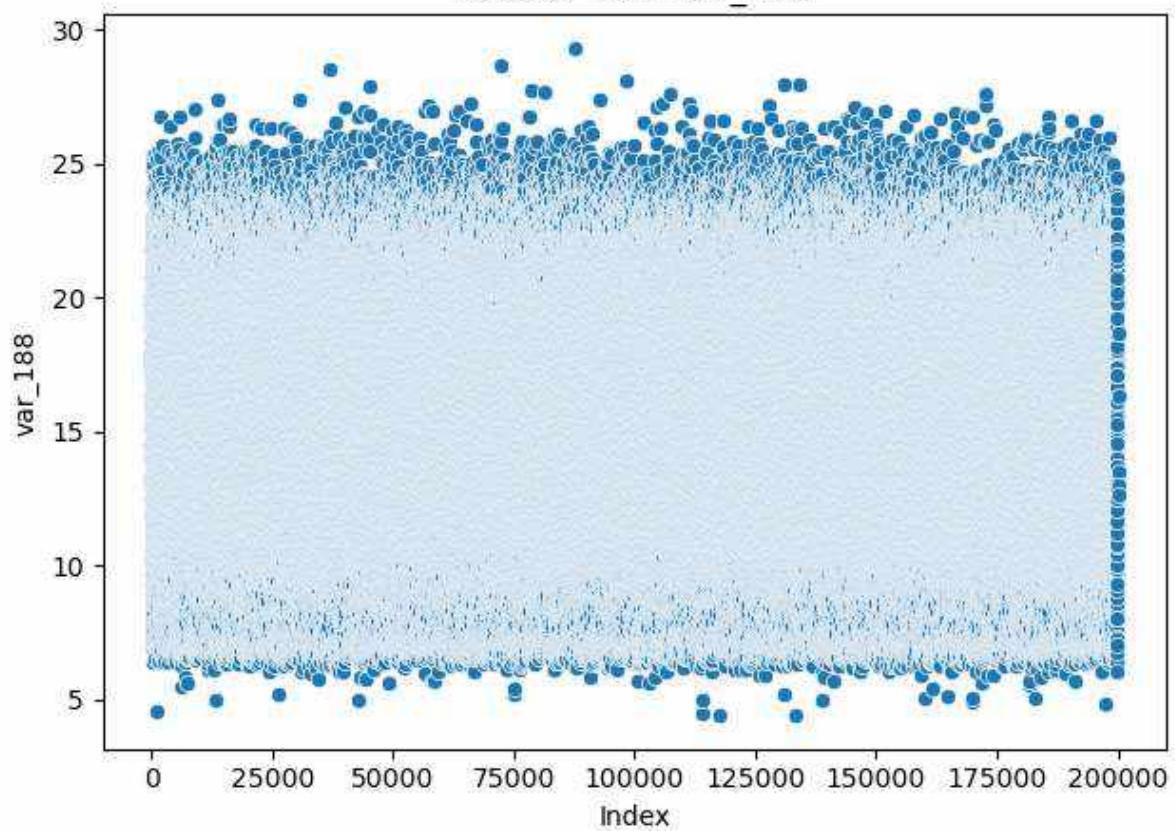
Scatter Plot - var\_186



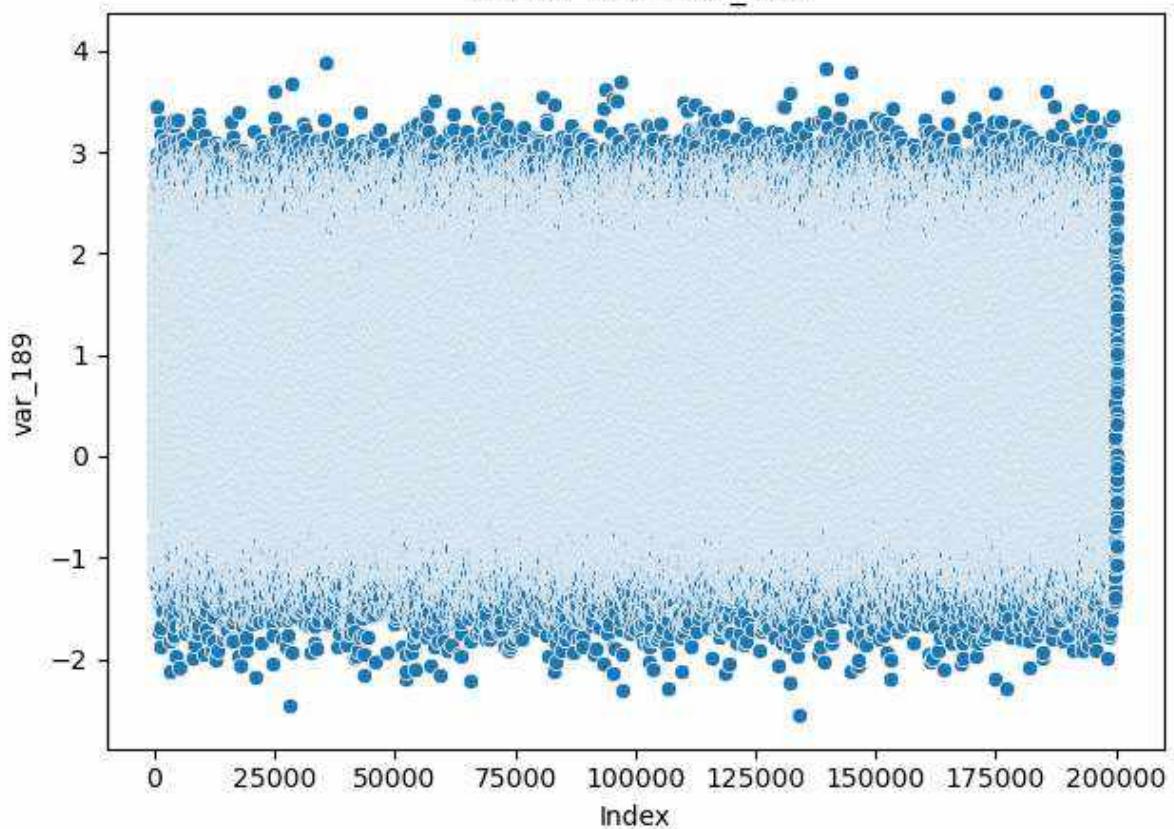
Scatter Plot - var\_187



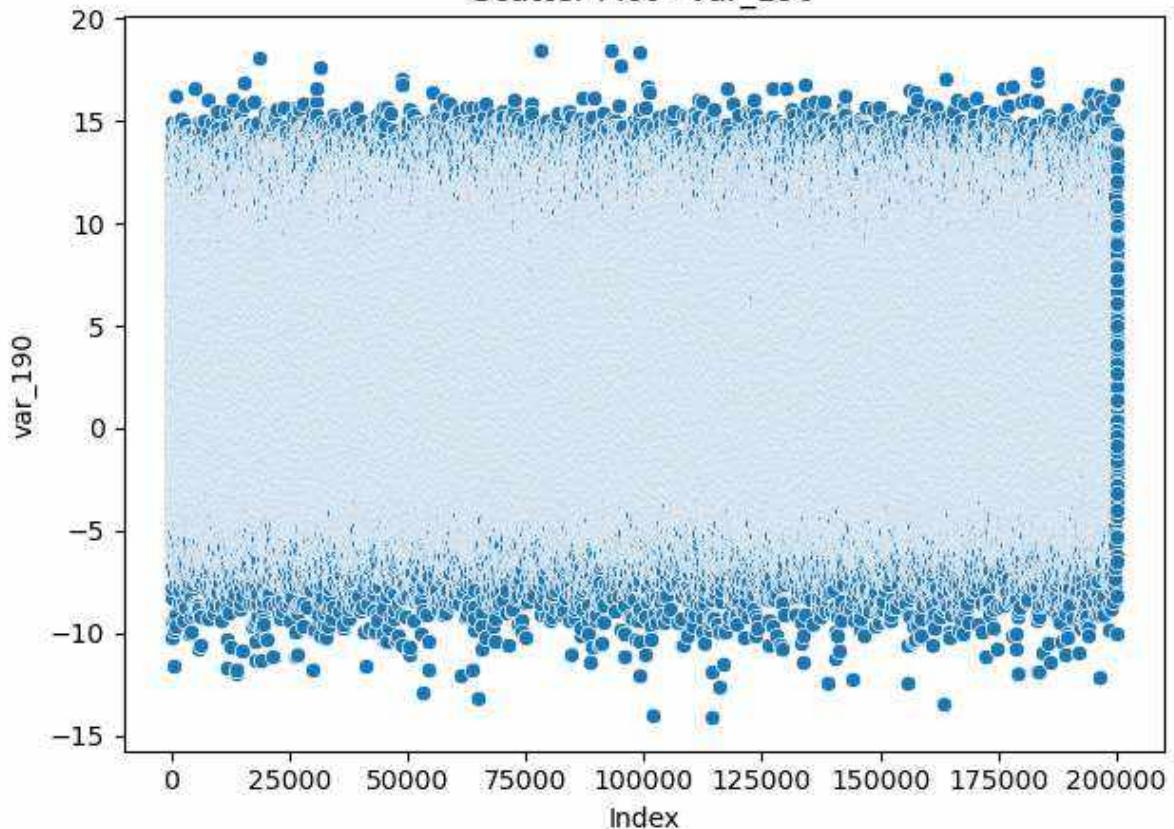
Scatter Plot - var\_188

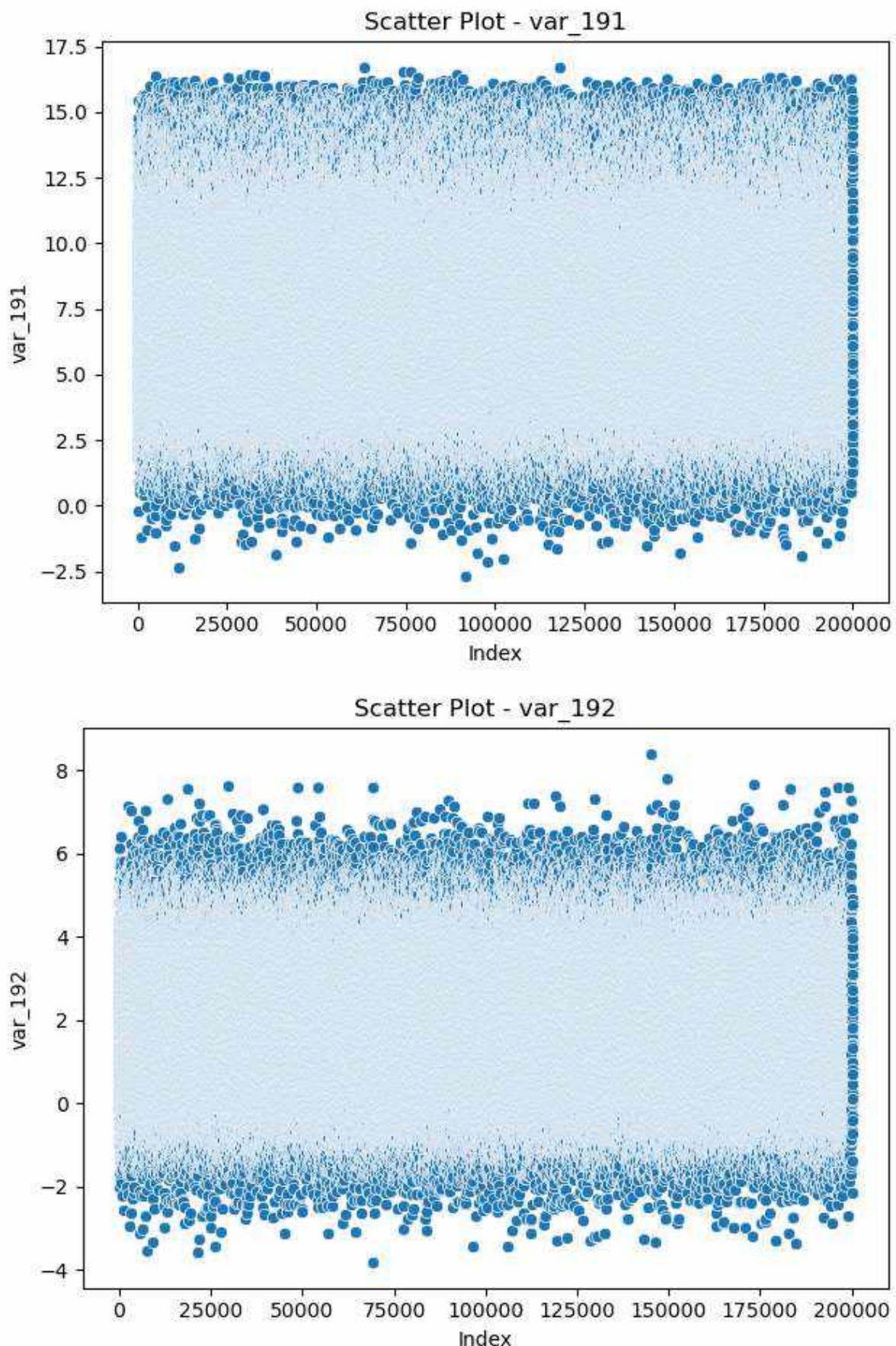


Scatter Plot - var\_189

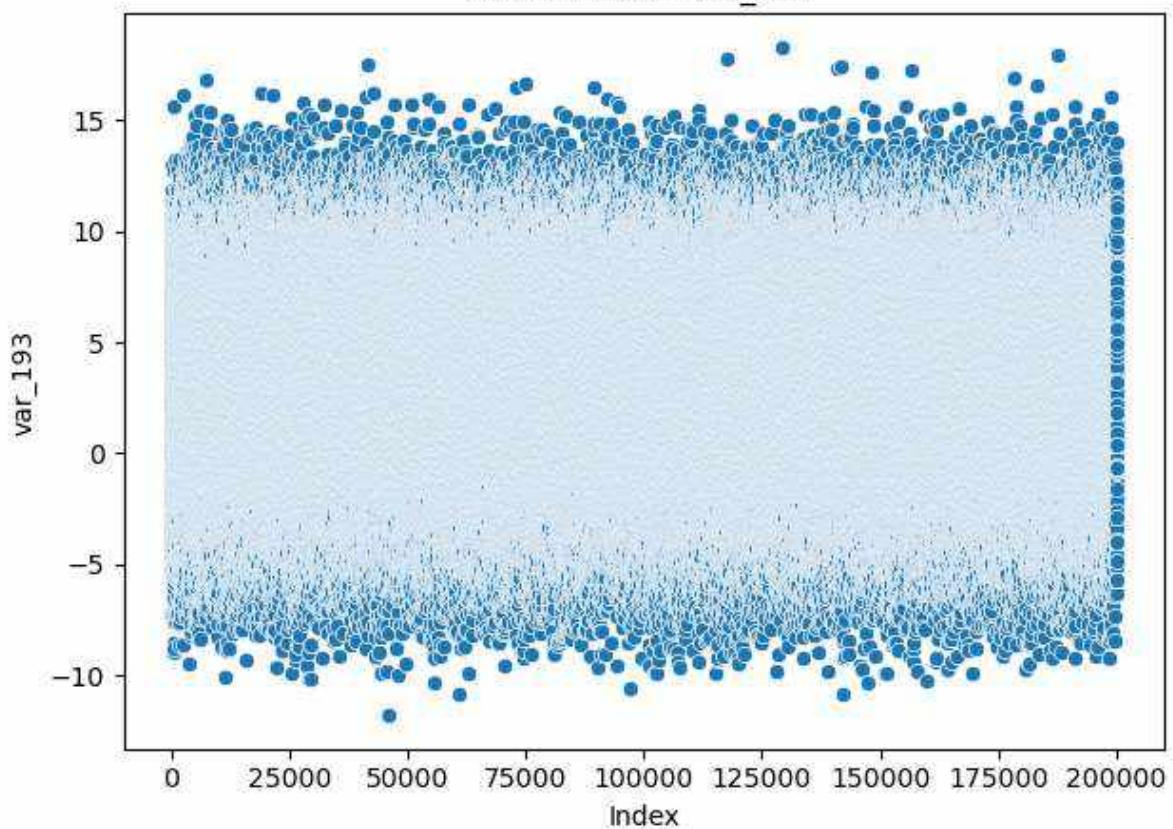


Scatter Plot - var\_190

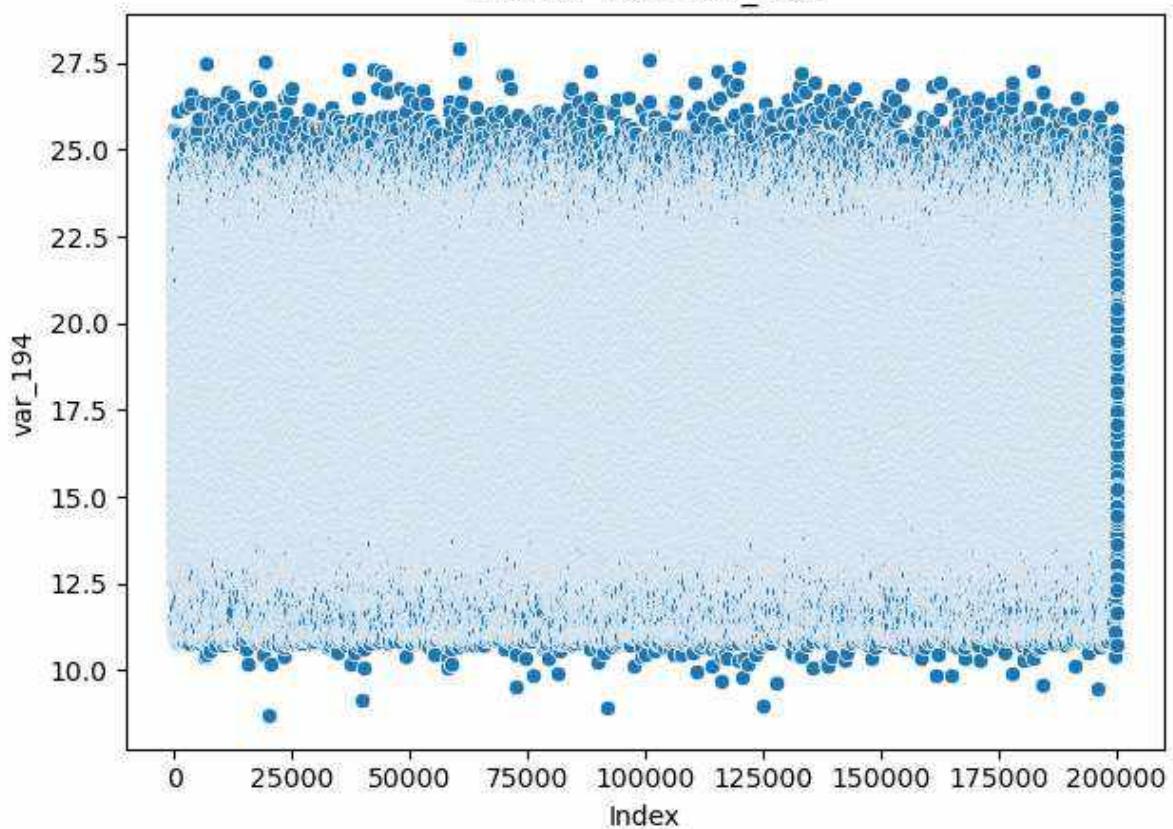




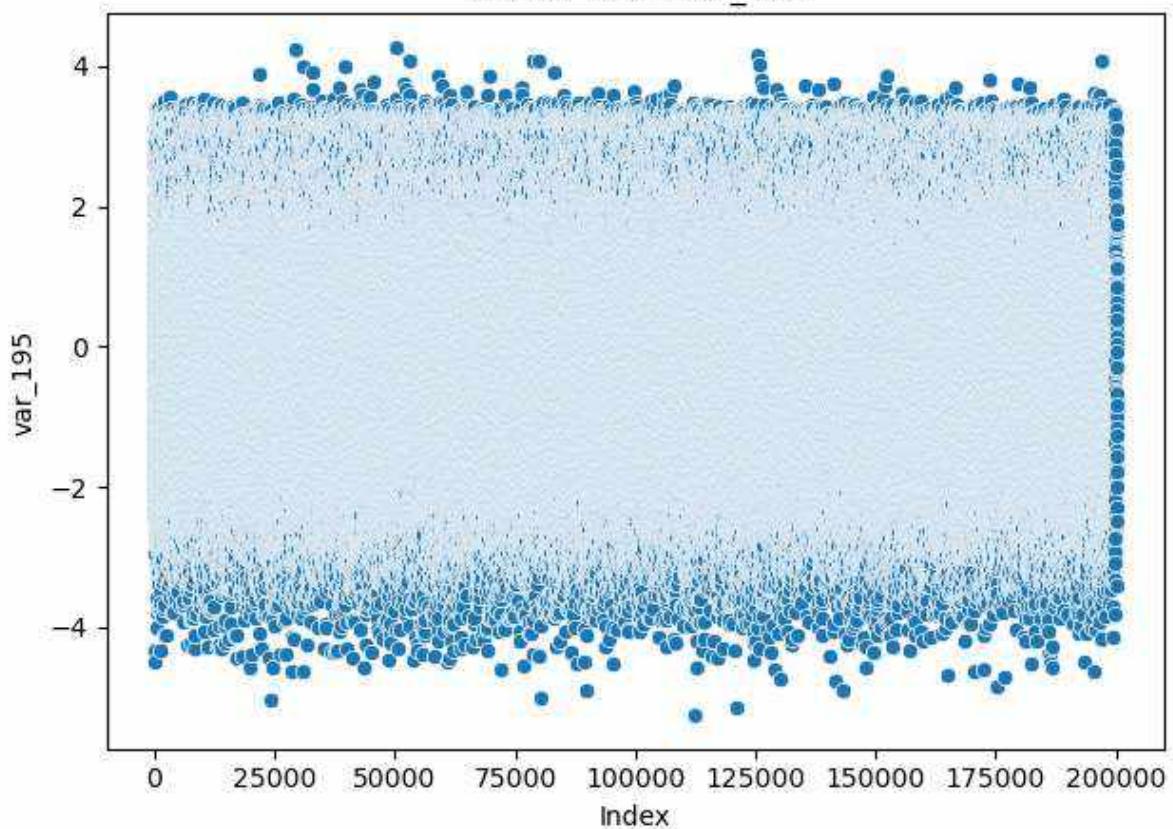
Scatter Plot - var\_193



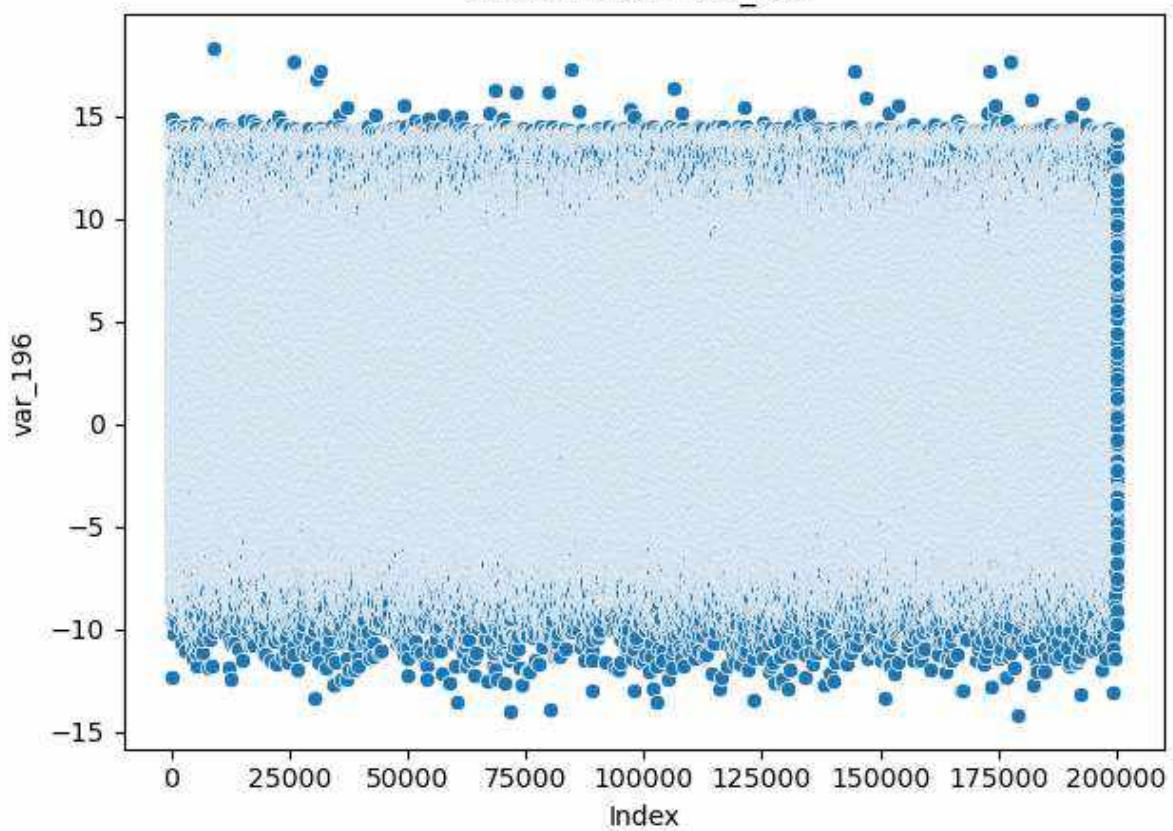
Scatter Plot - var\_194



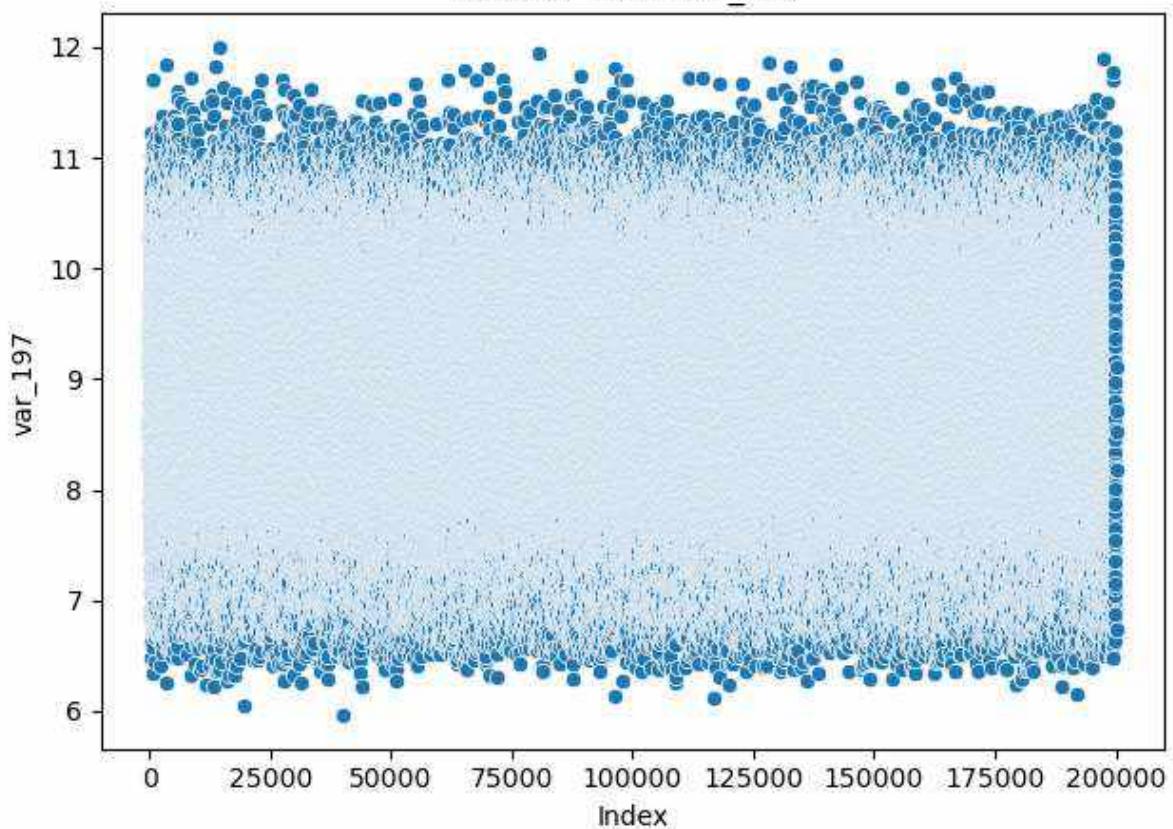
Scatter Plot - var\_195



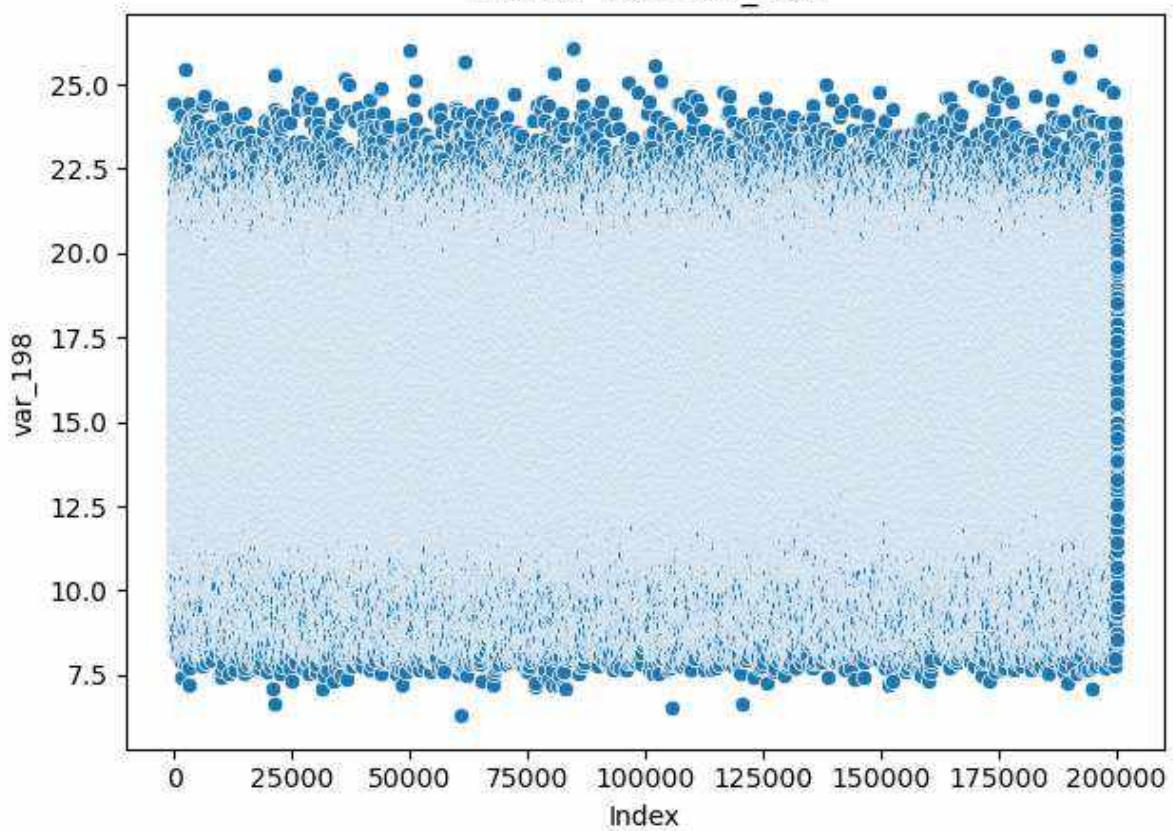
Scatter Plot - var\_196

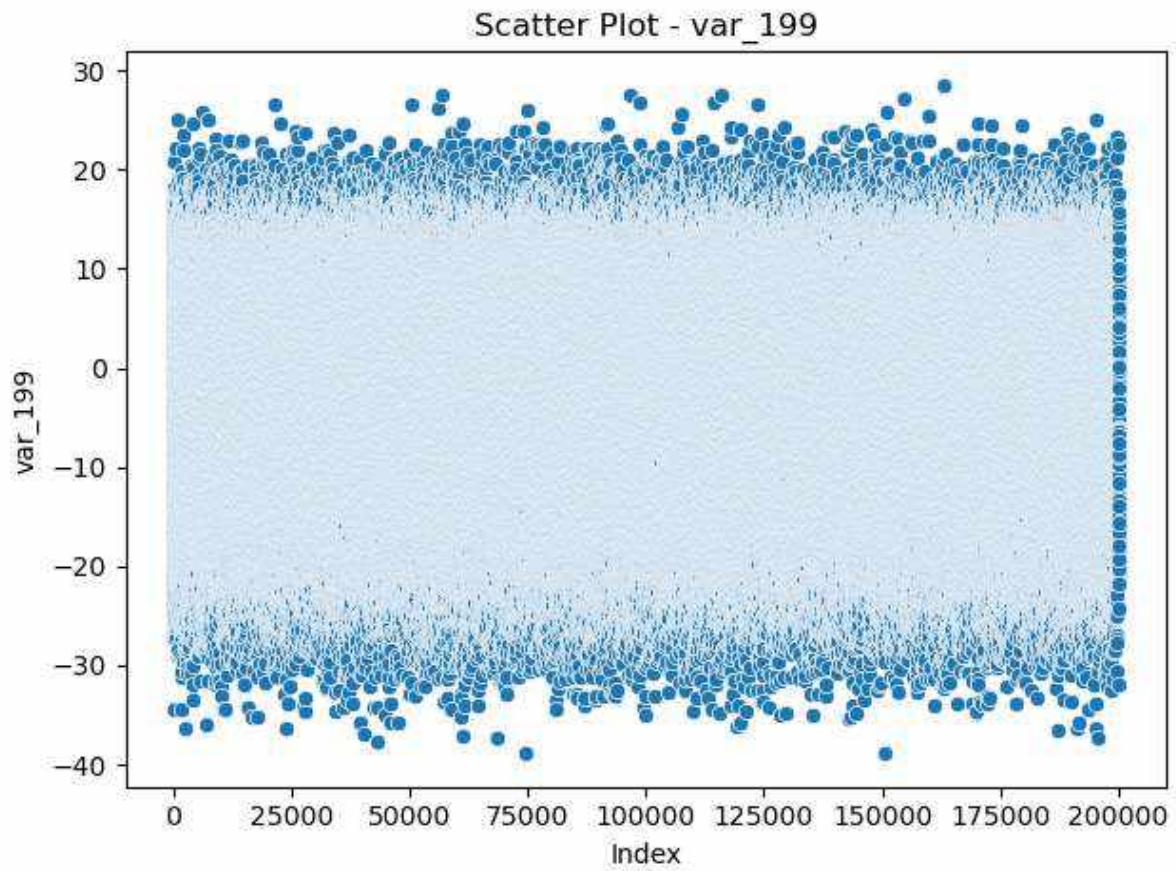


Scatter Plot - var\_197



Scatter Plot - var\_198

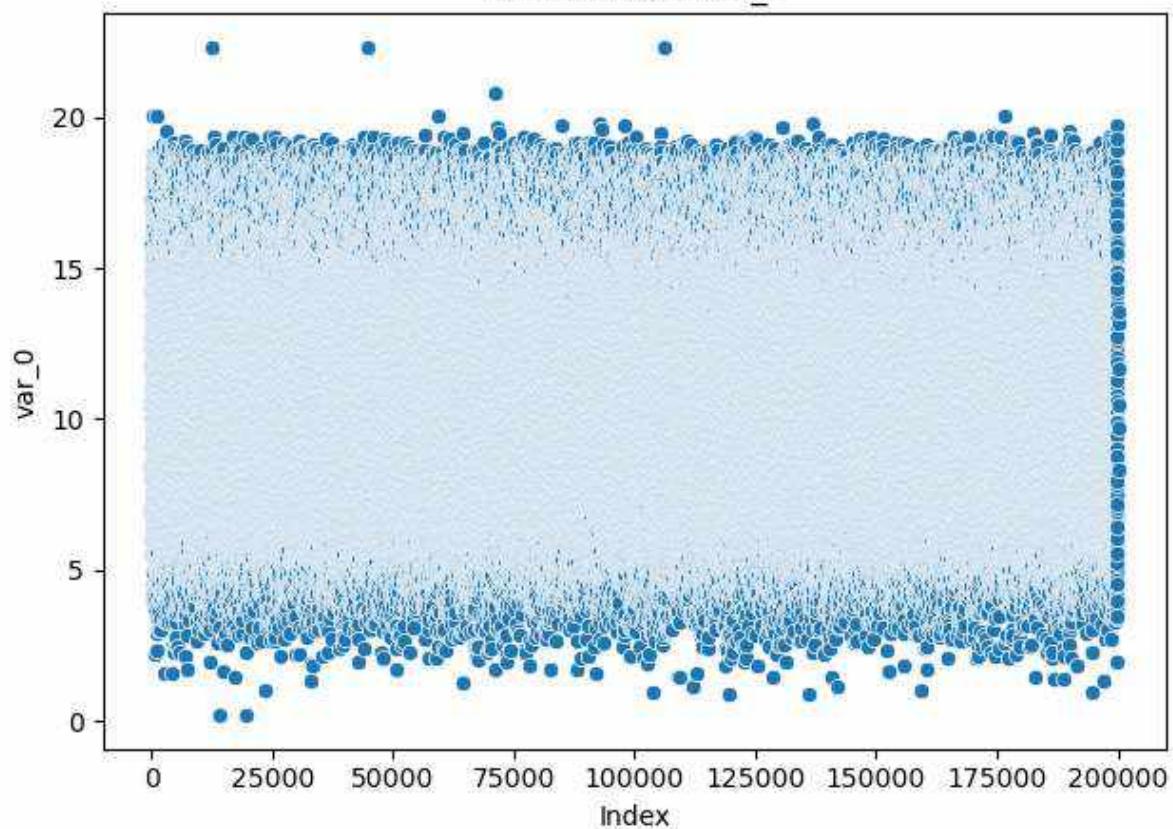




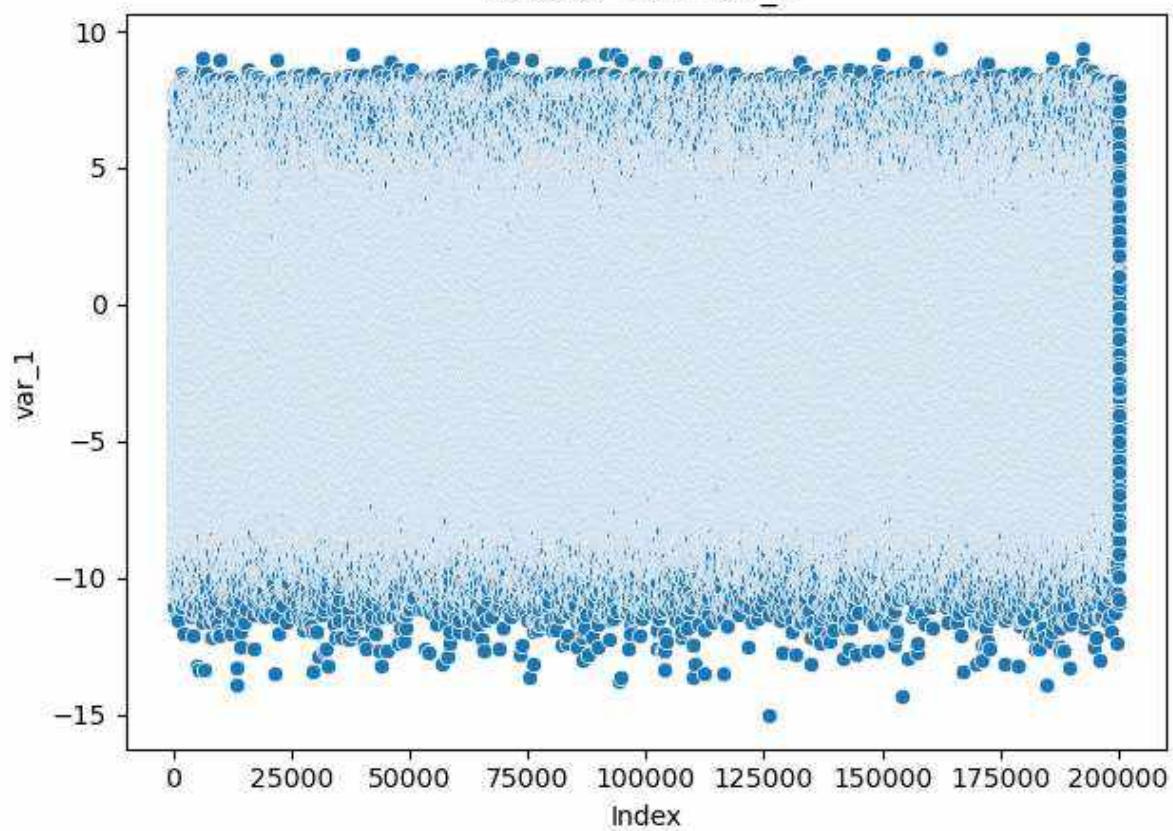
In [26]: # Scatter plot of test set

```
for column in data_test_set.columns:
    if column != 'ID_code':
        sns.scatterplot(x=range(len(data_test_set)), y=data_test_set[column])
        plt.title(f'Scatter Plot - {column}')
        plt.xlabel('Index')
        plt.ylabel(column)
        plt.tight_layout()
        plt.show()
```

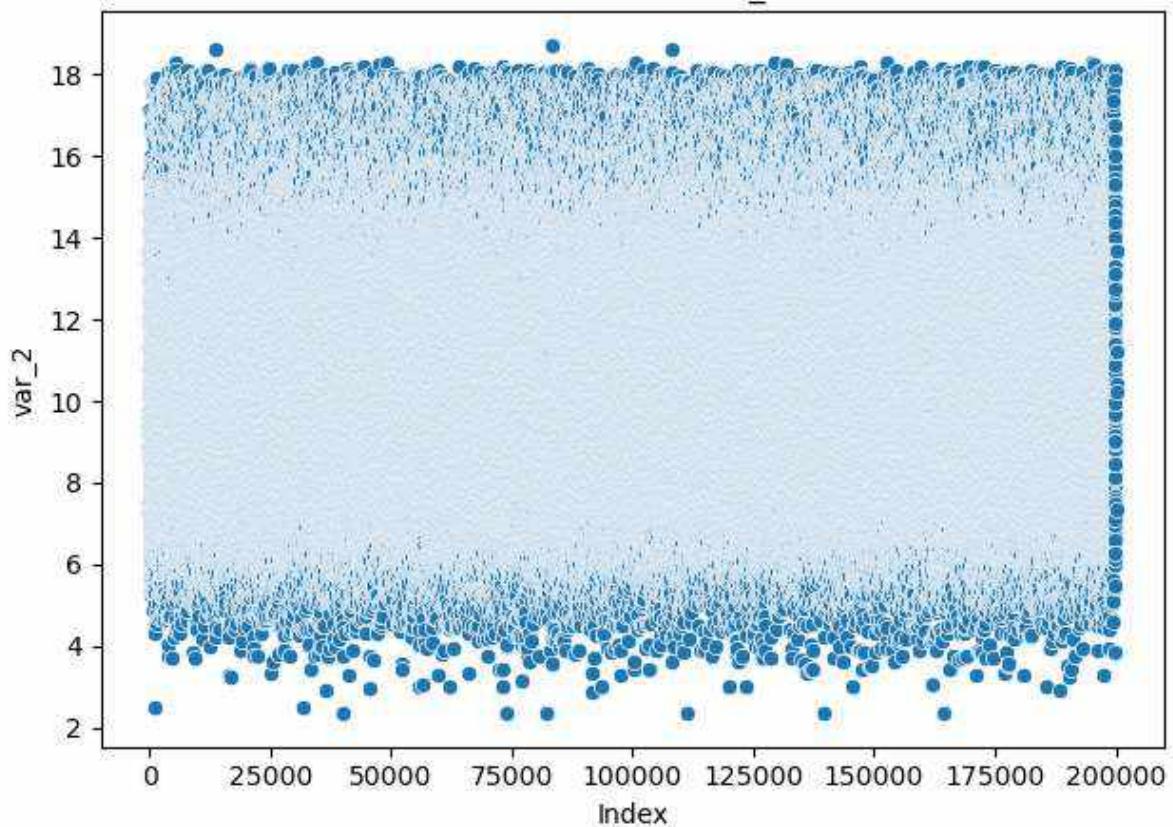
Scatter Plot - var\_0



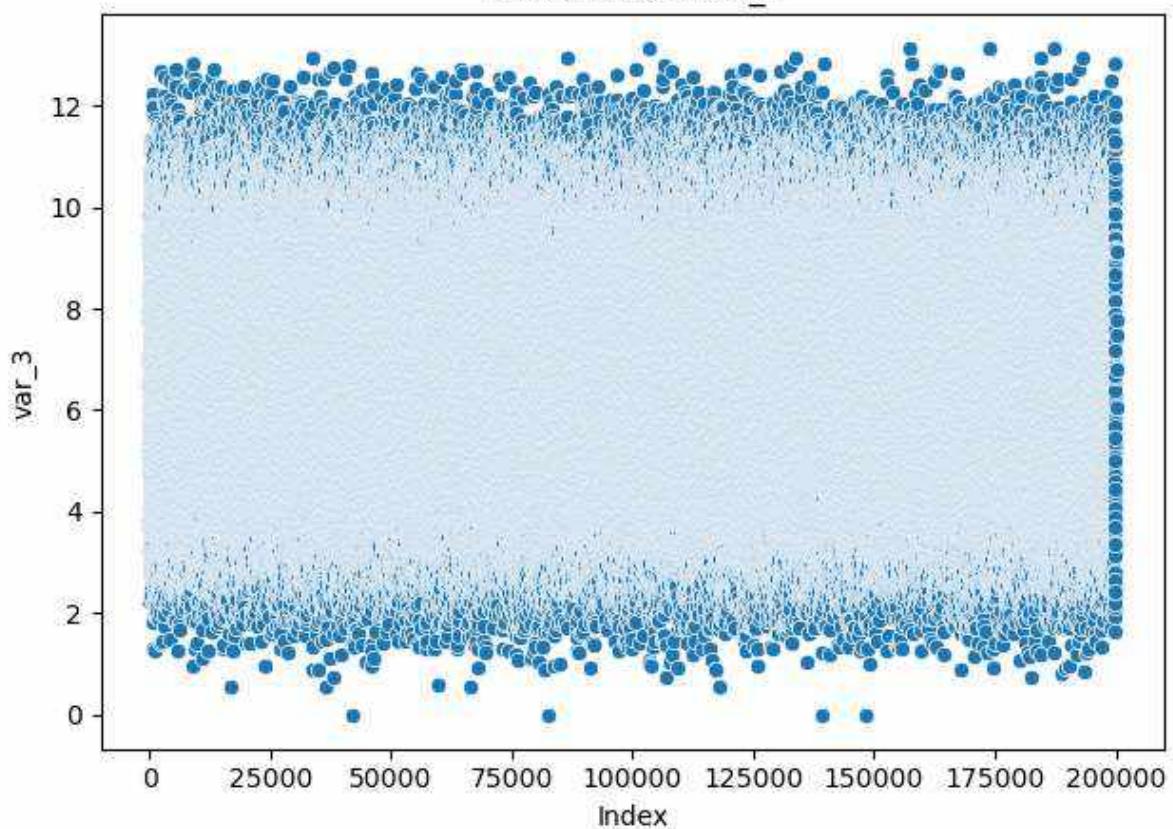
Scatter Plot - var\_1

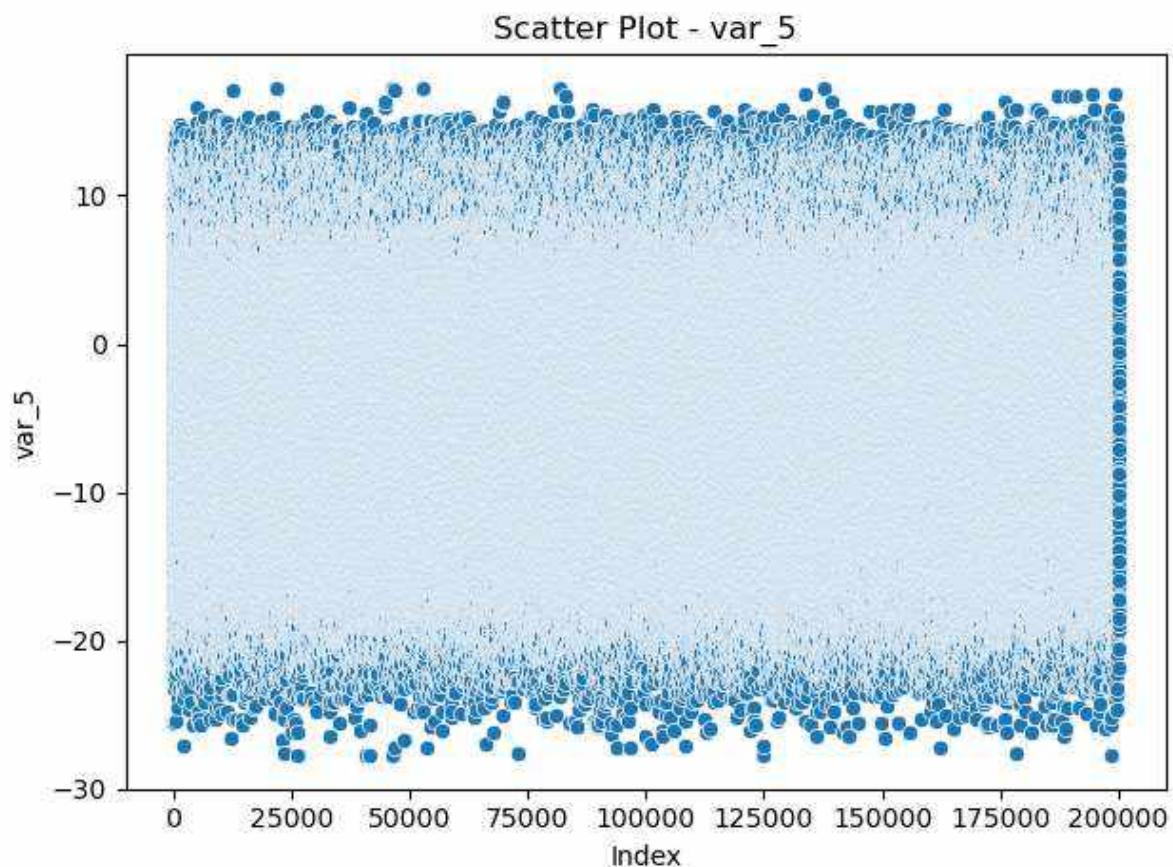
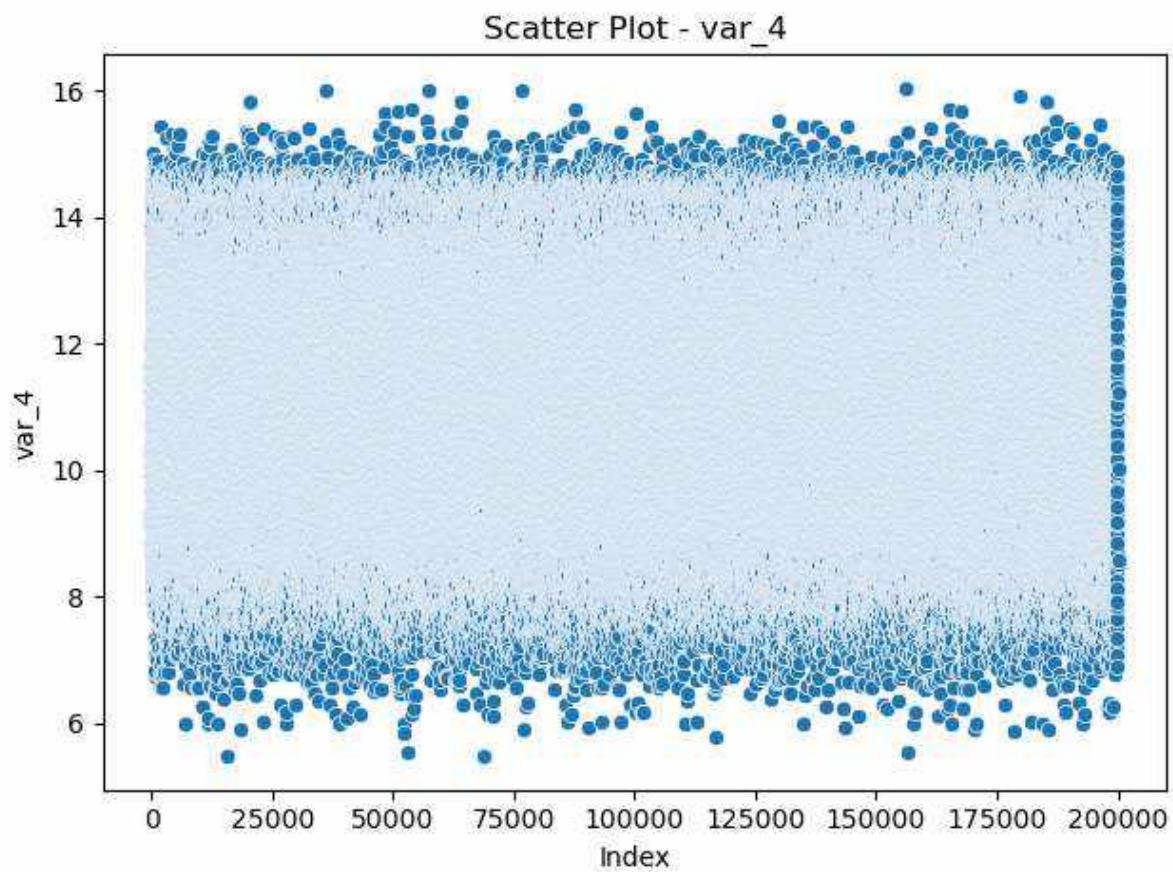


Scatter Plot - var\_2

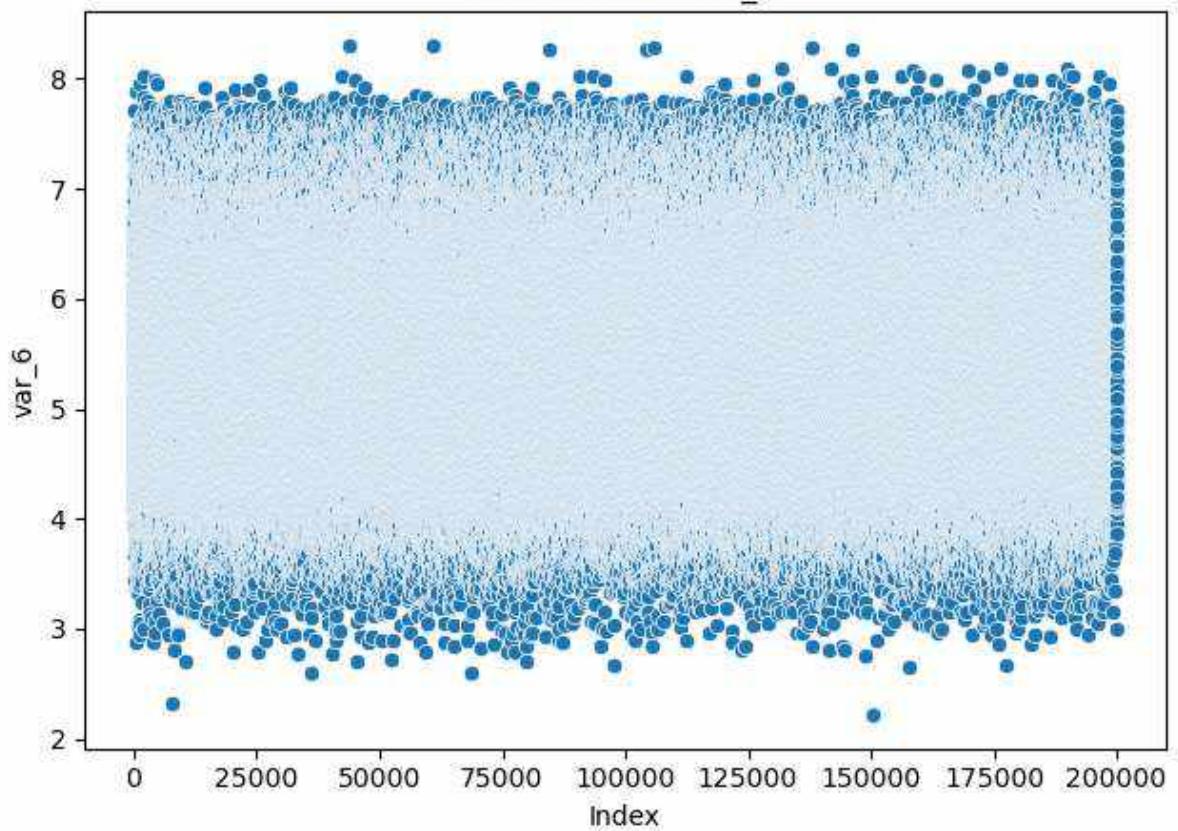


Scatter Plot - var\_3

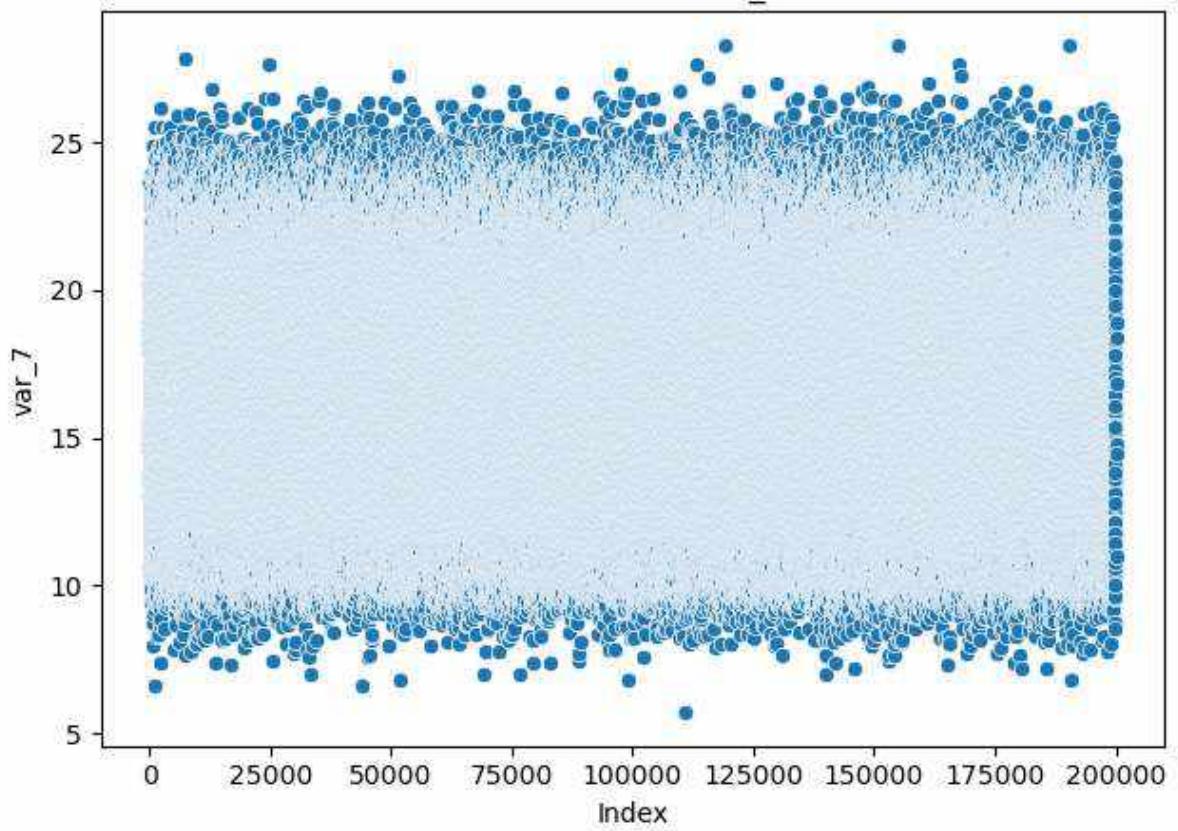


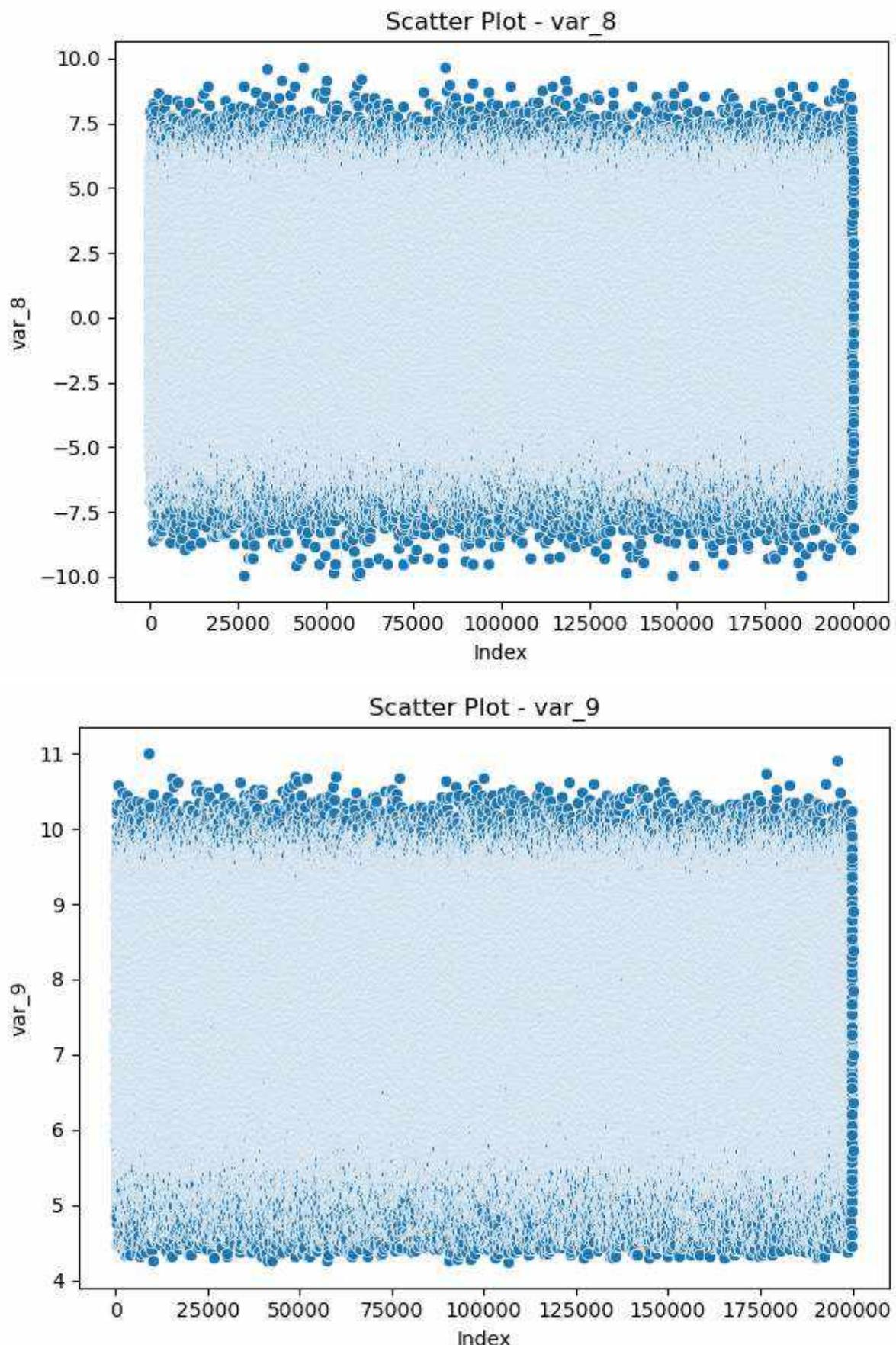


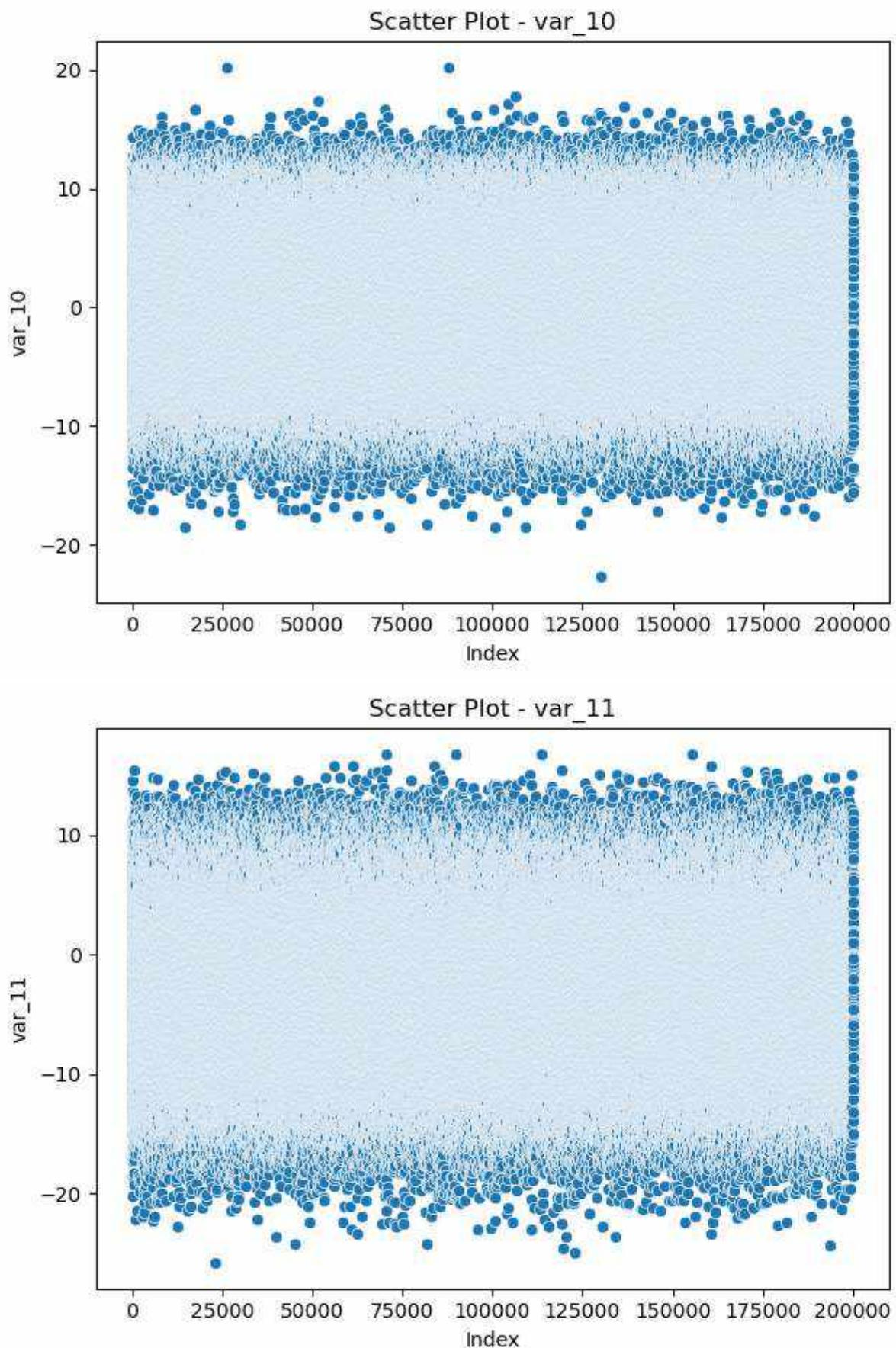
Scatter Plot - var\_6

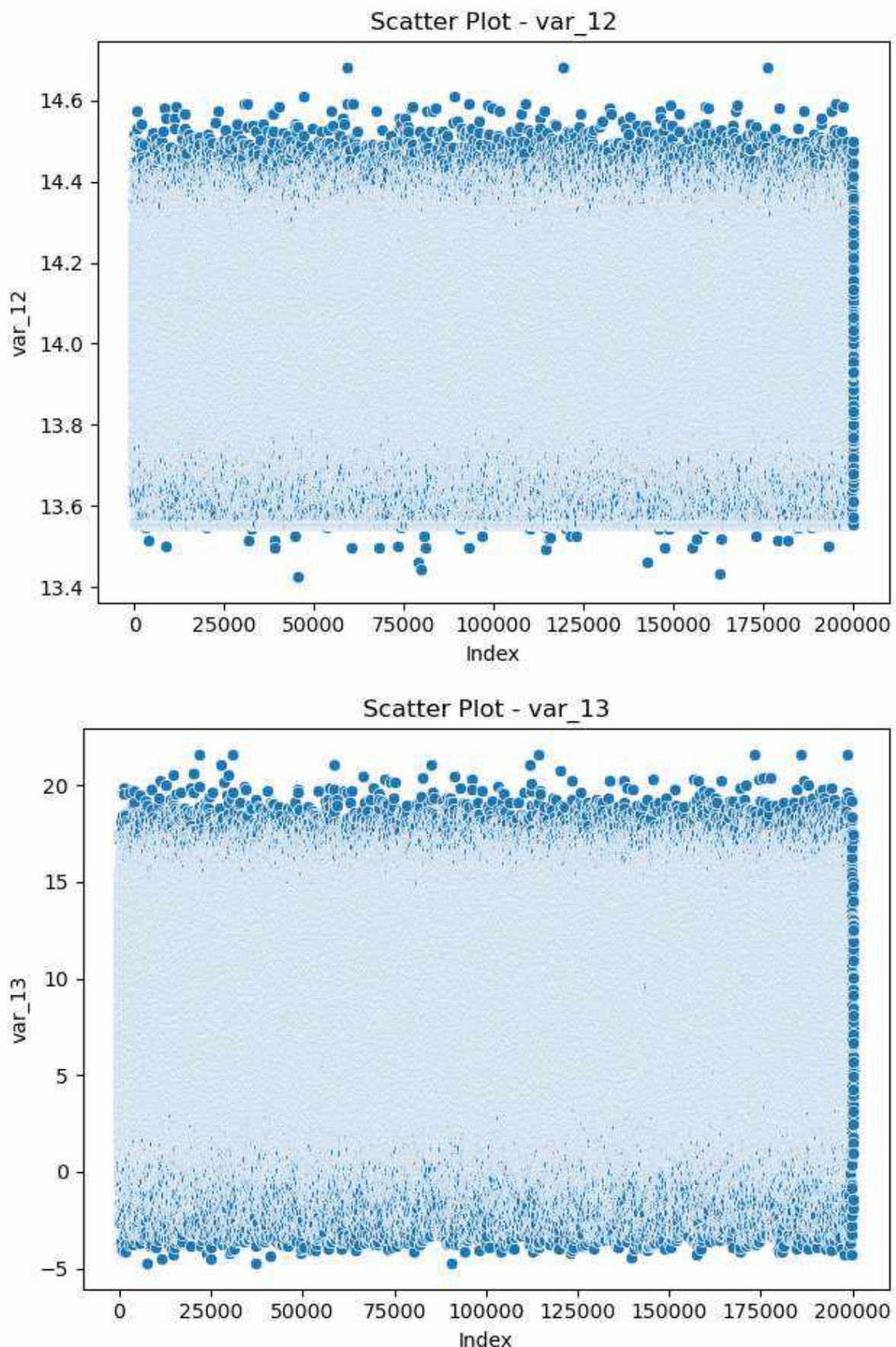


Scatter Plot - var\_7

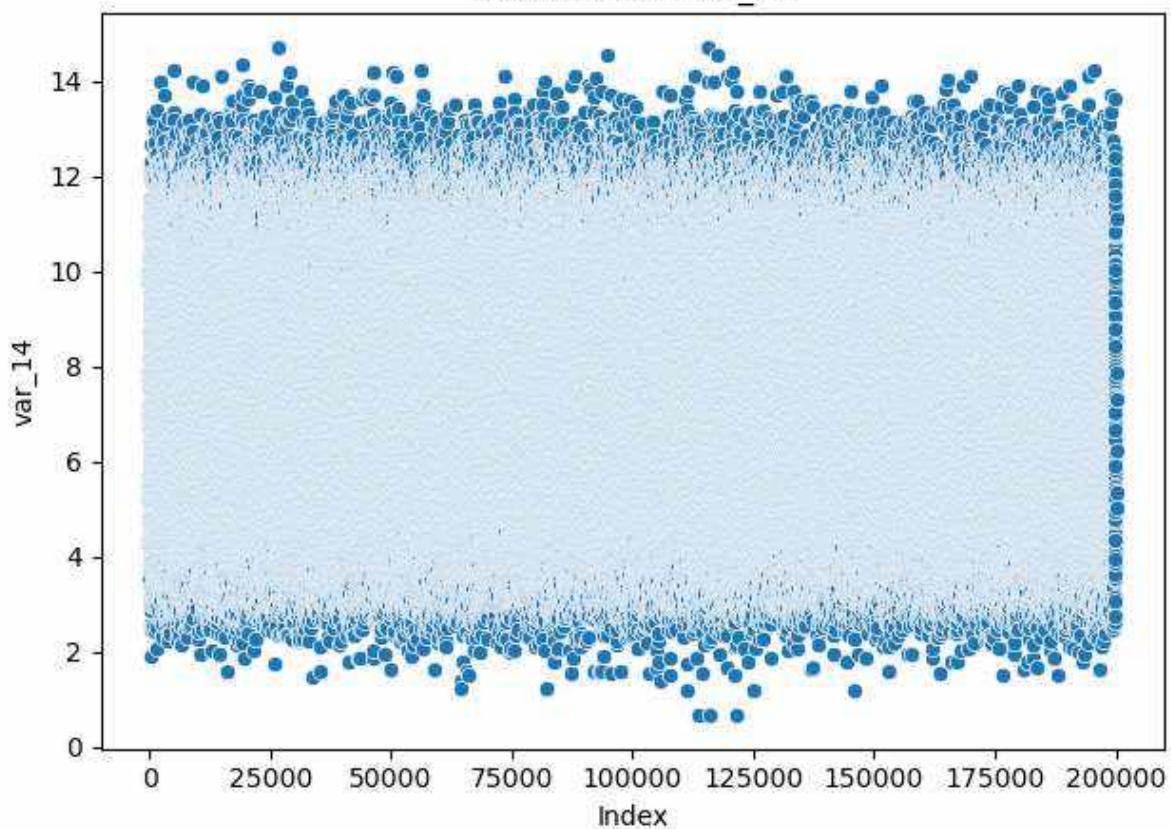




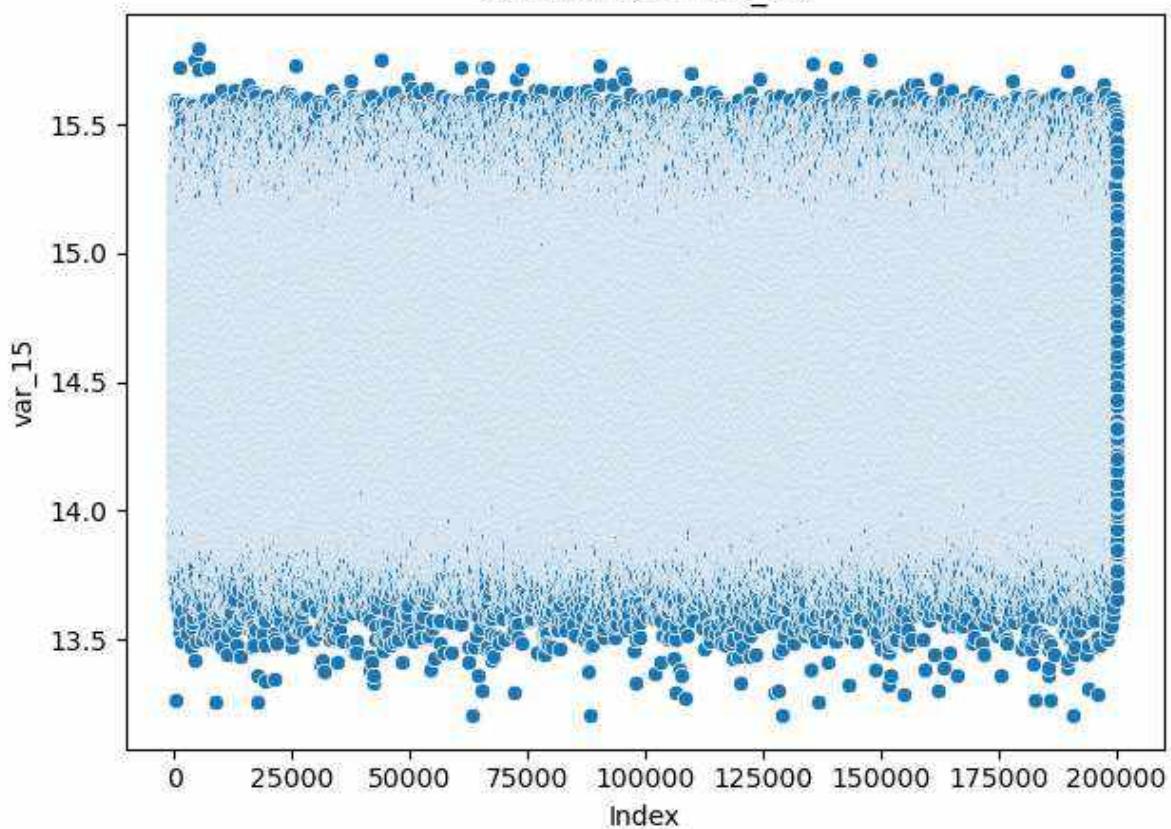


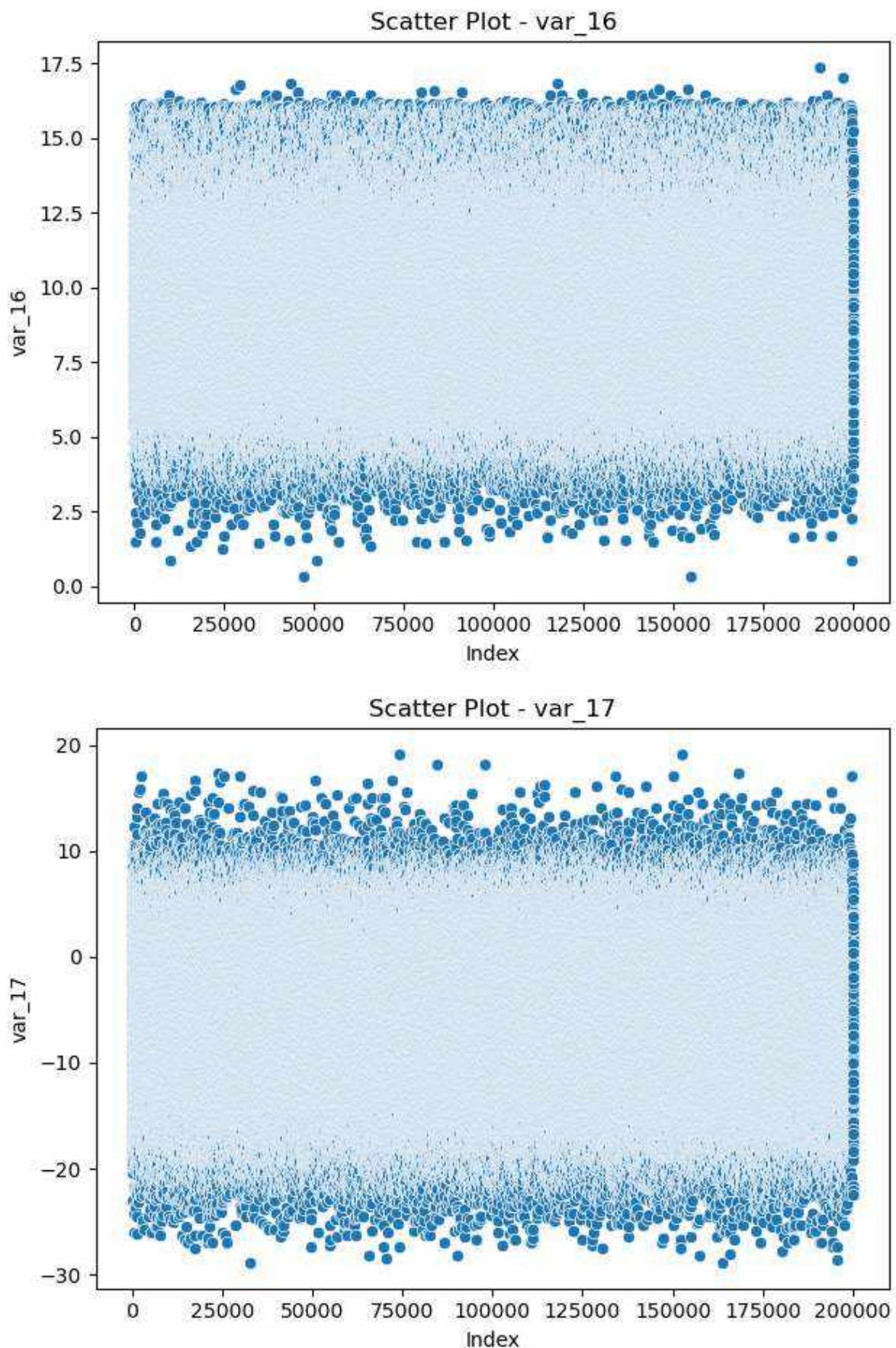


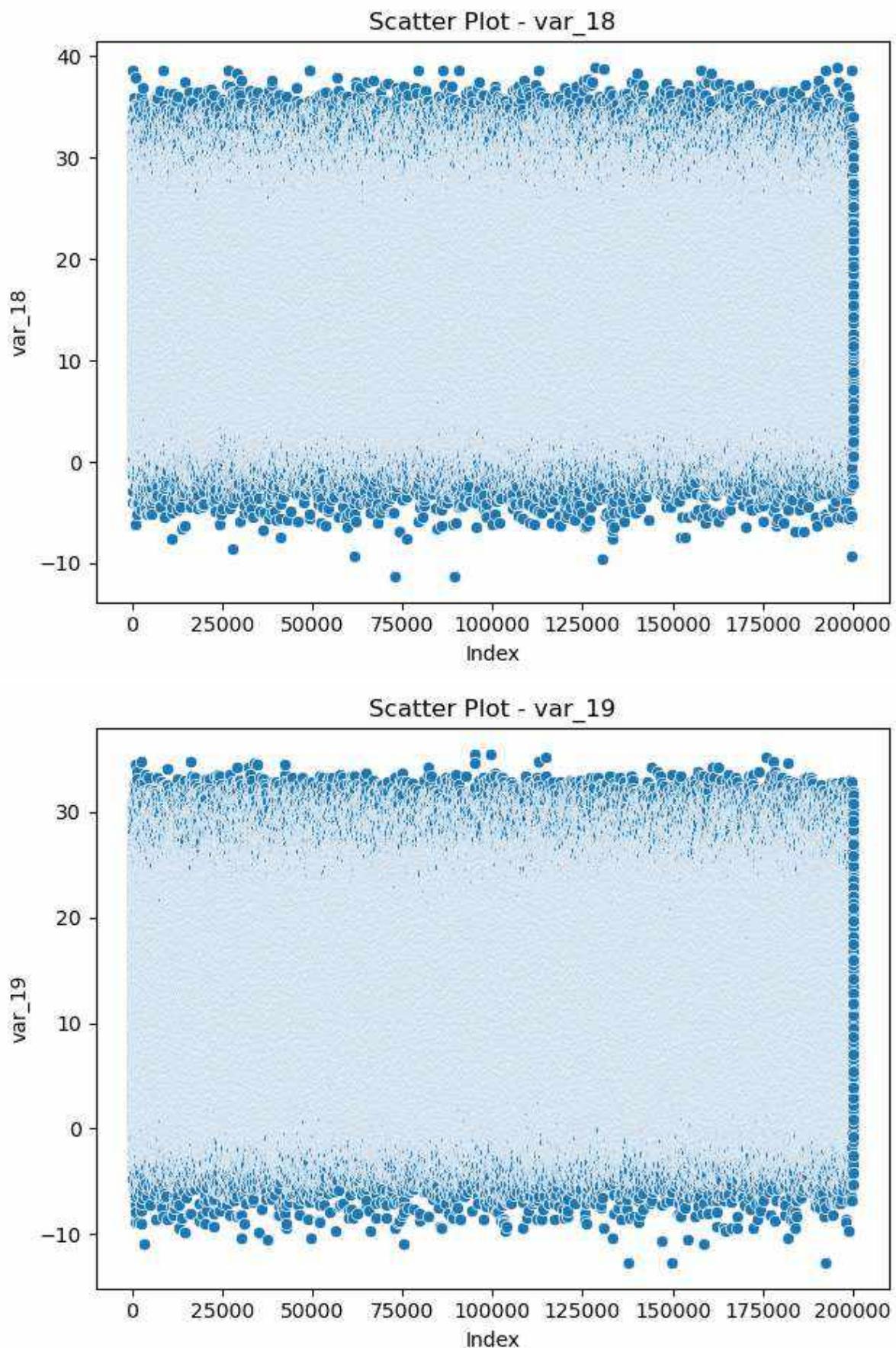
Scatter Plot - var\_14



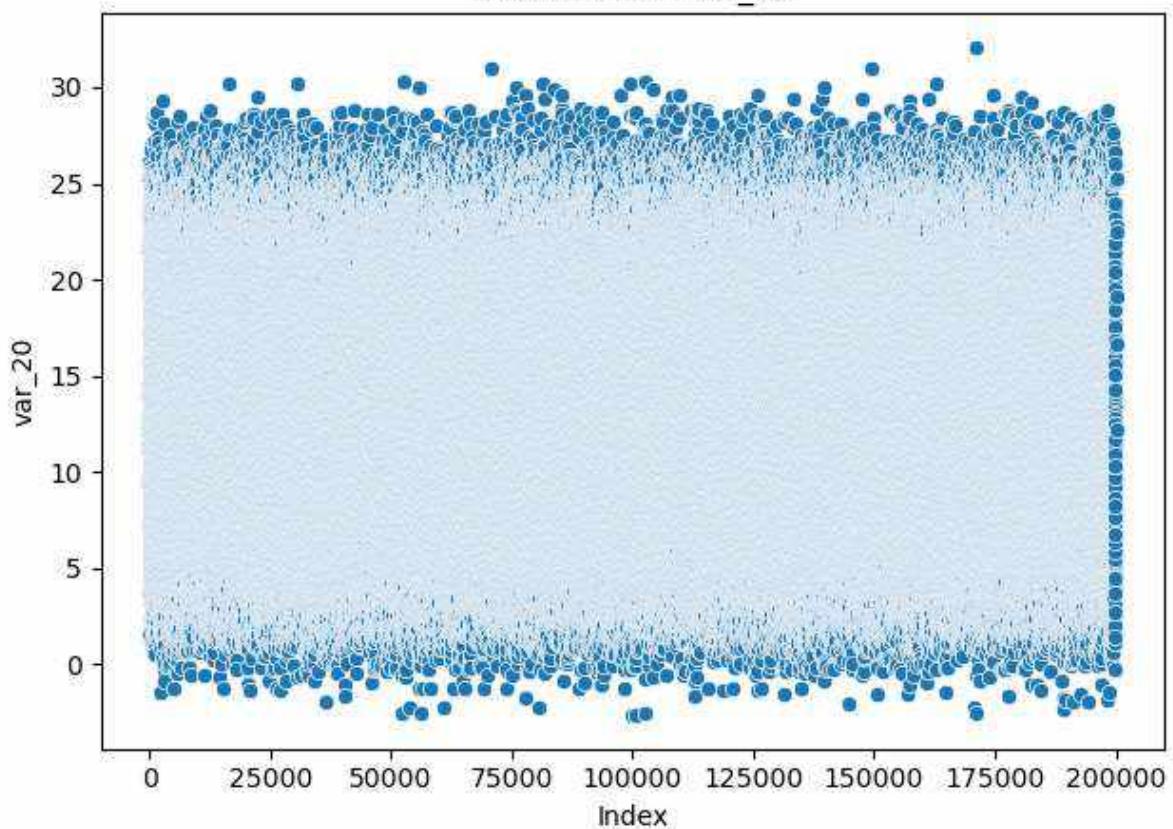
Scatter Plot - var\_15



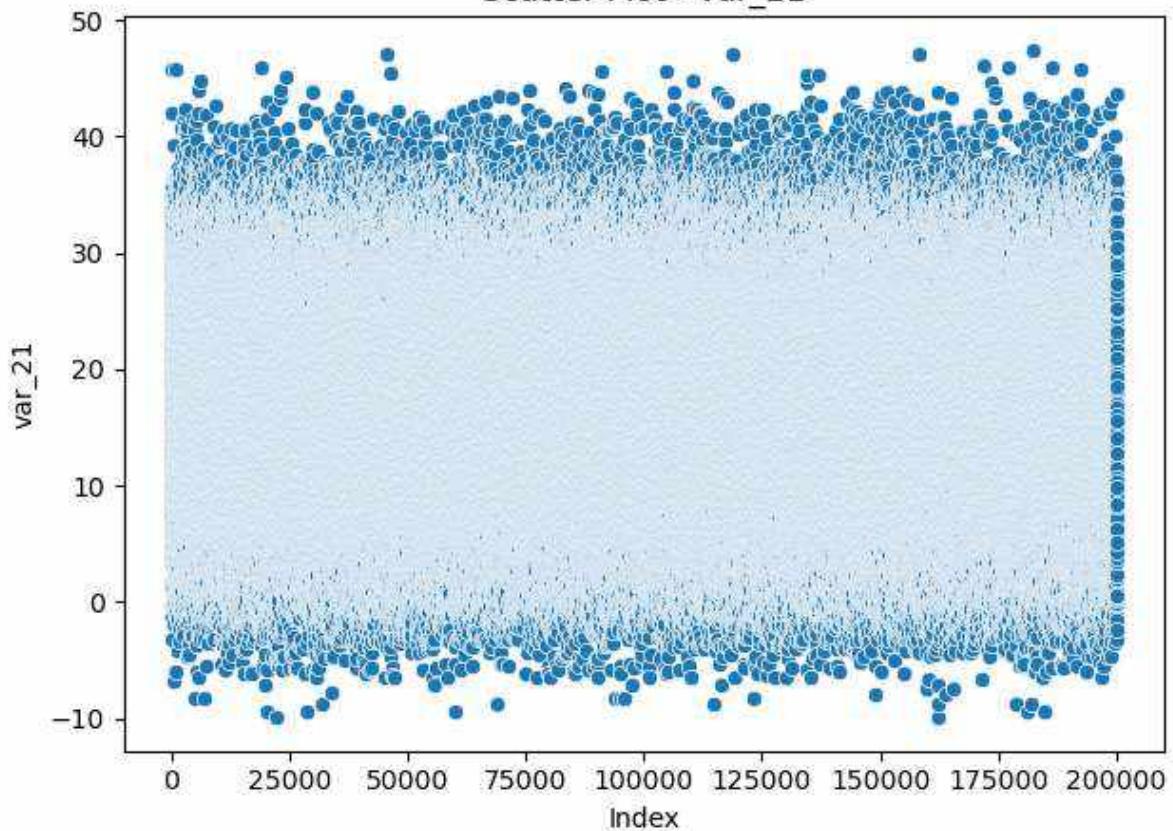


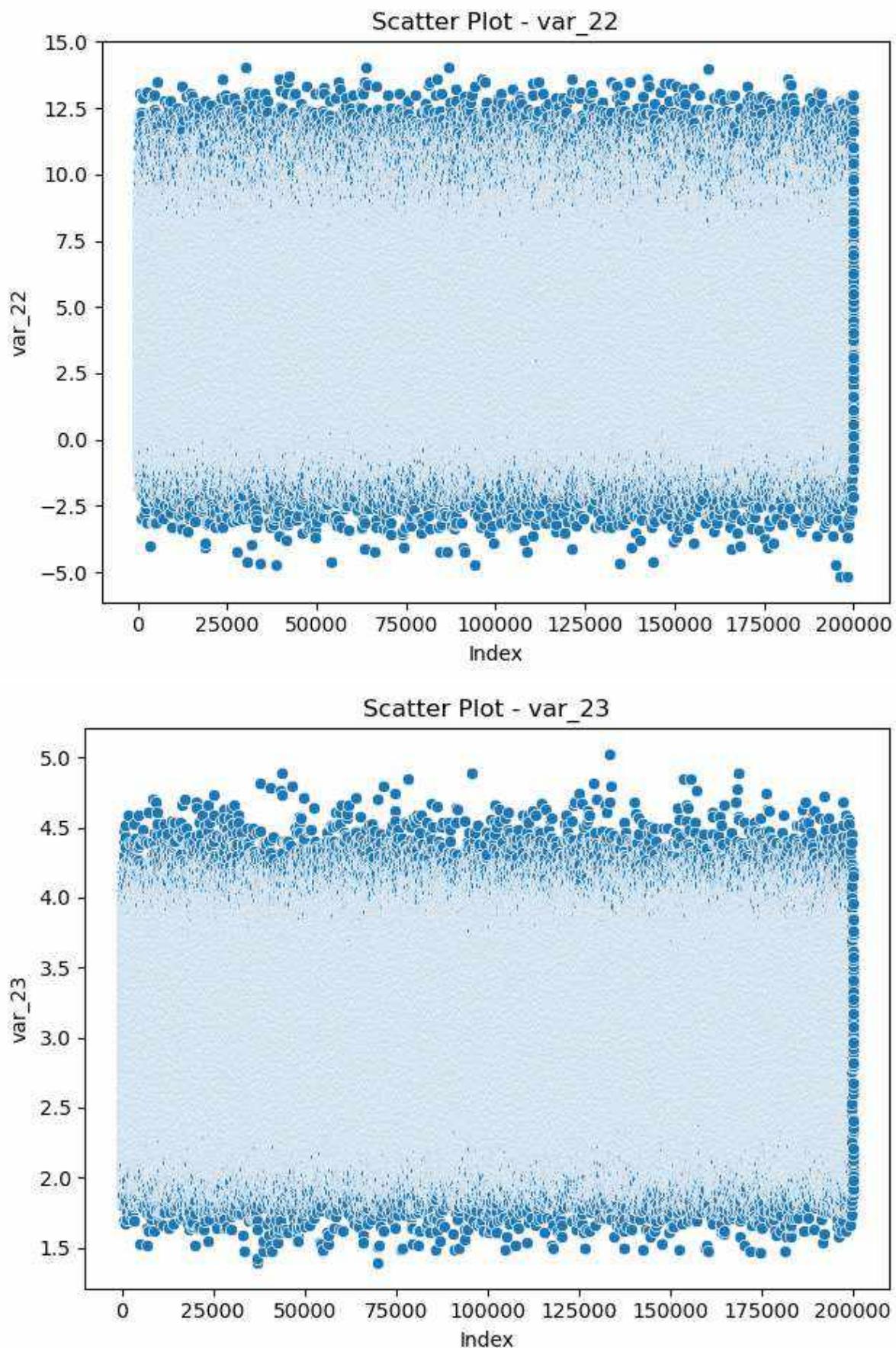


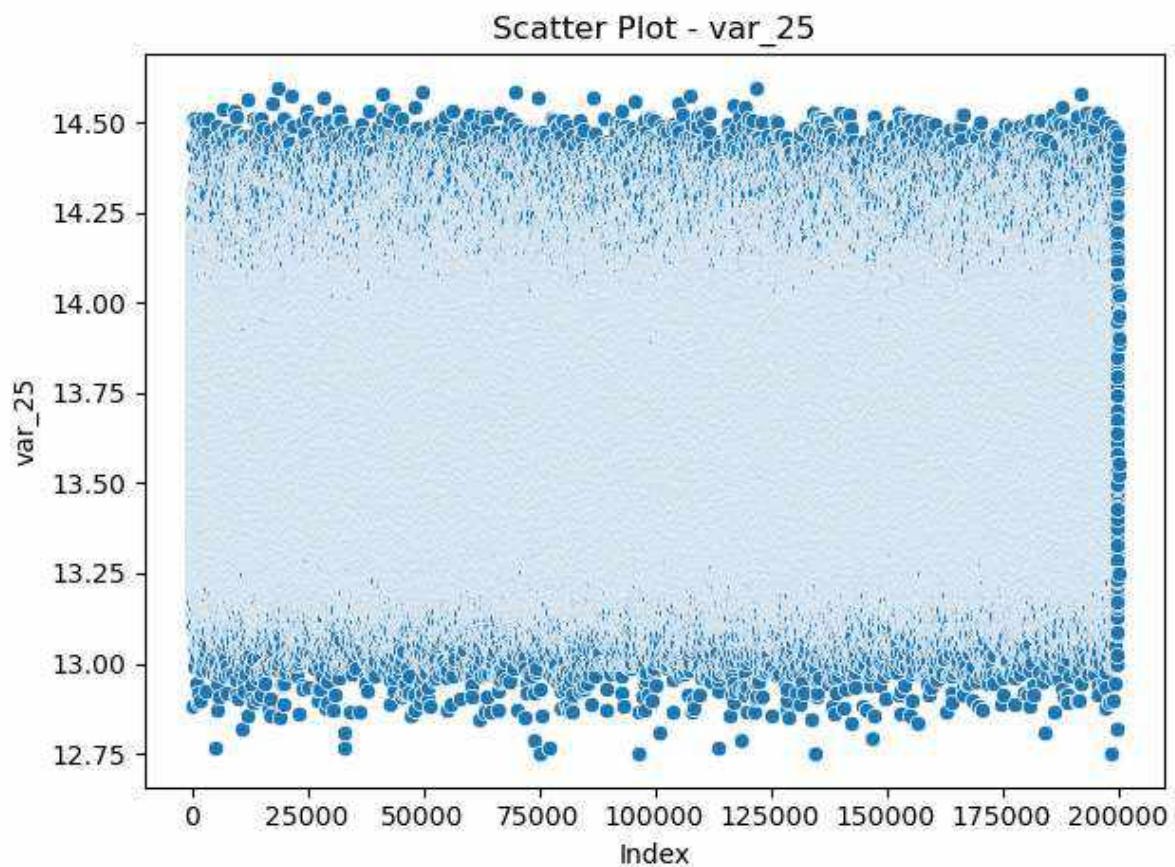
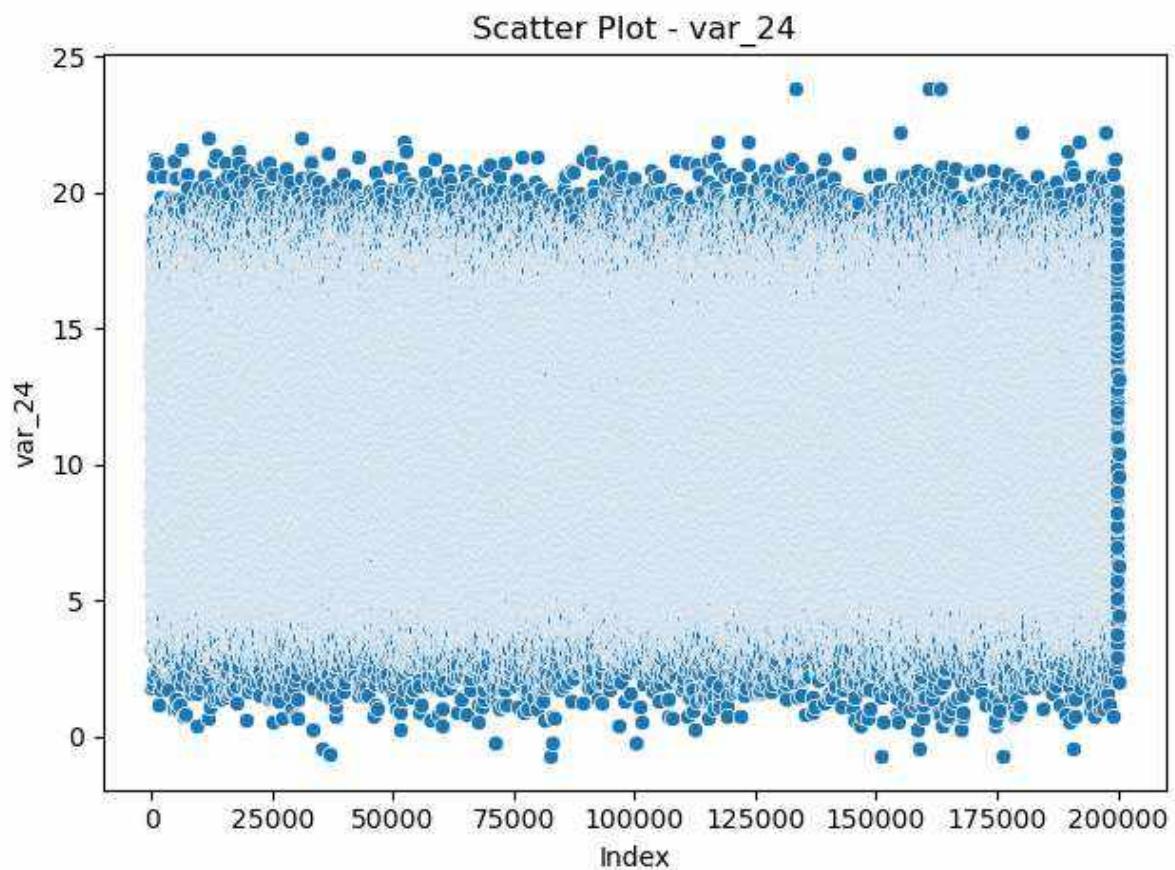
Scatter Plot - var\_20

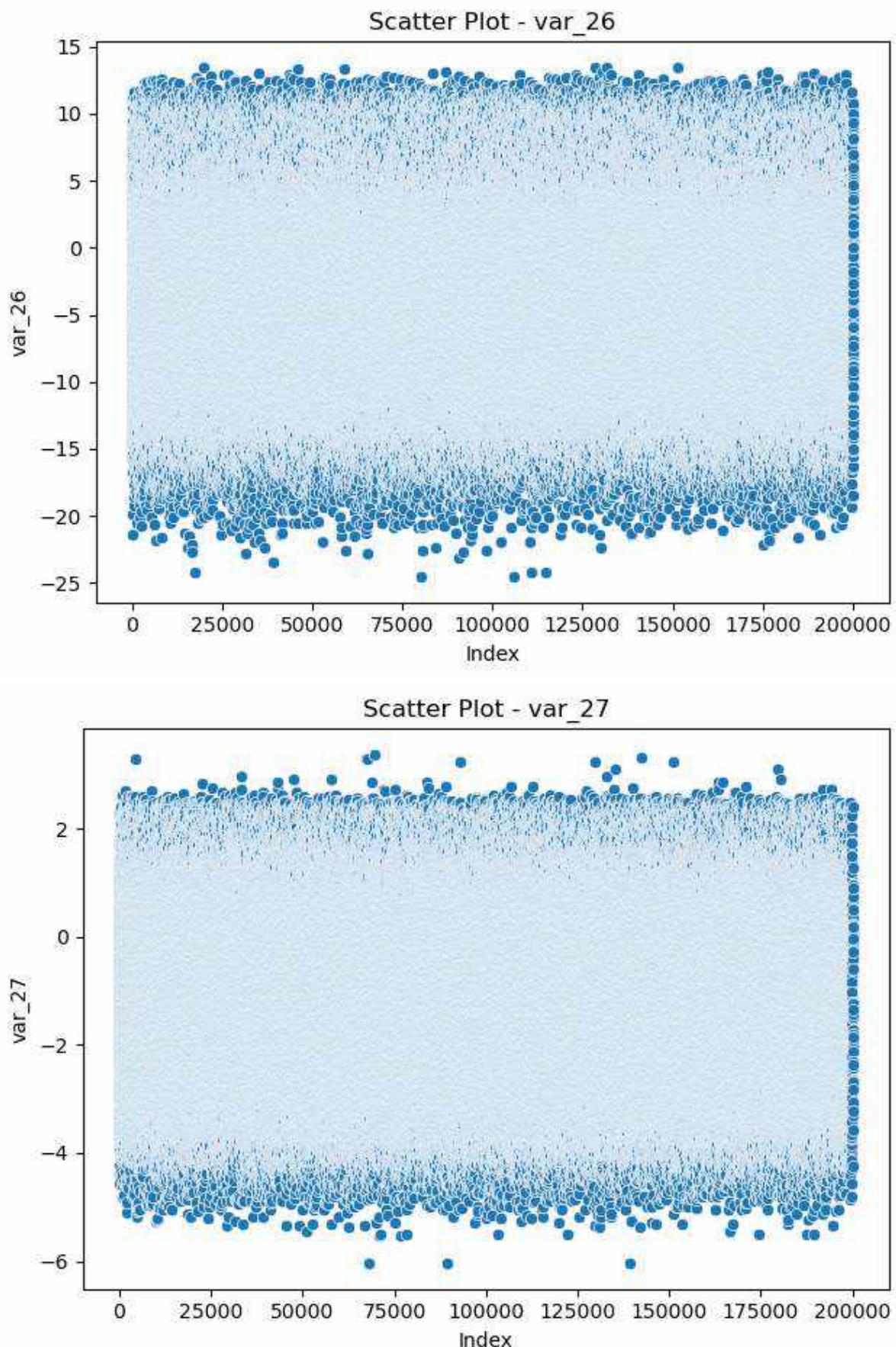


Scatter Plot - var\_21

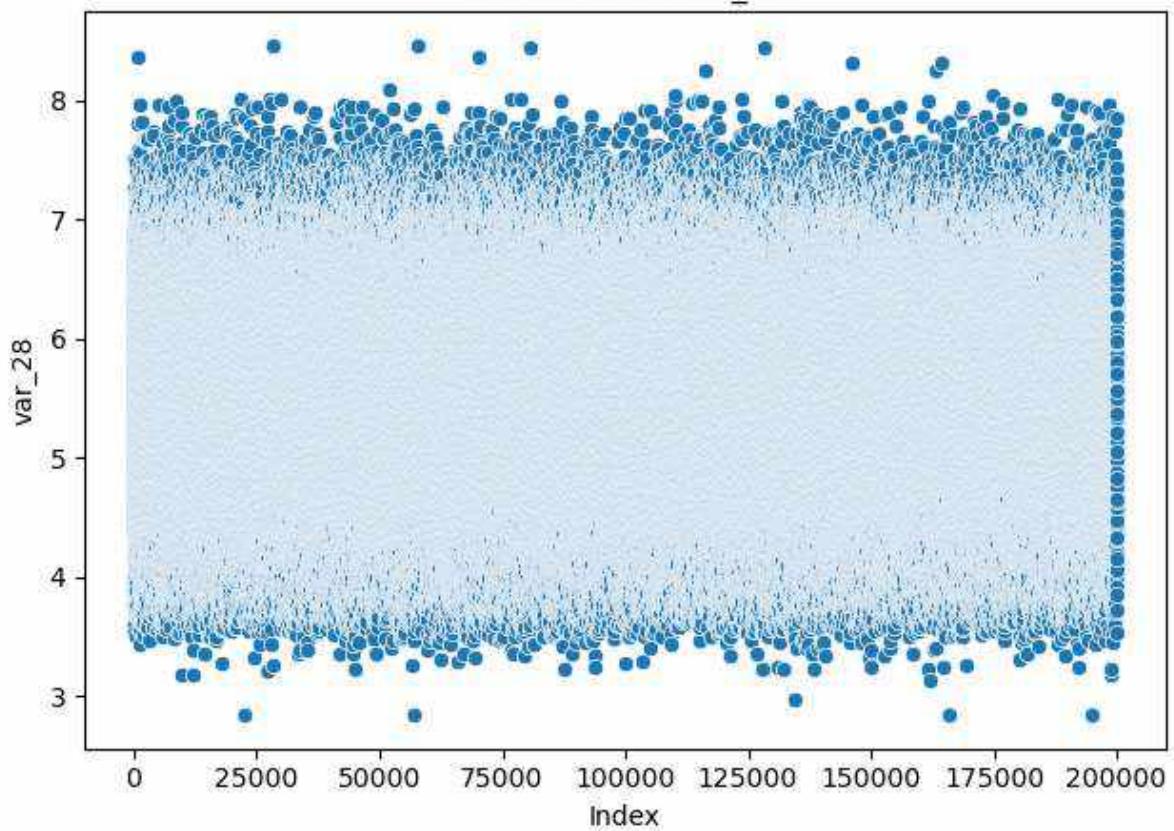




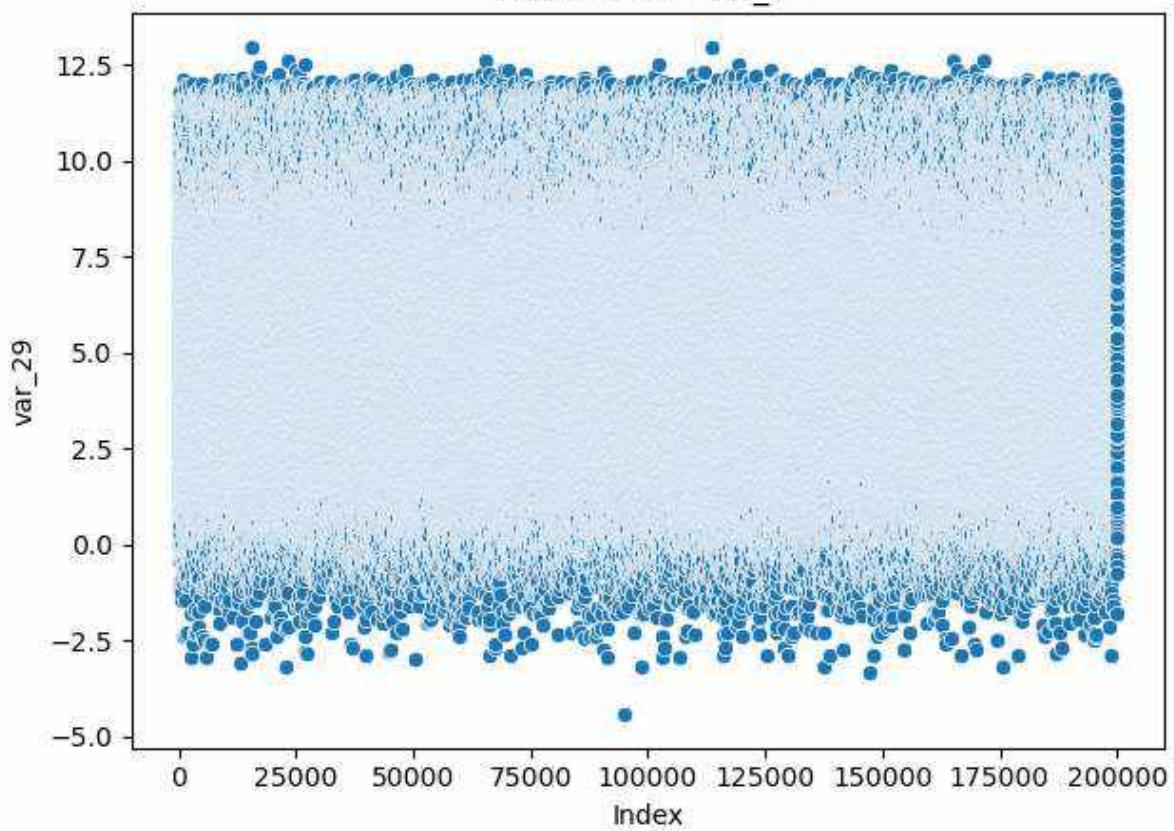




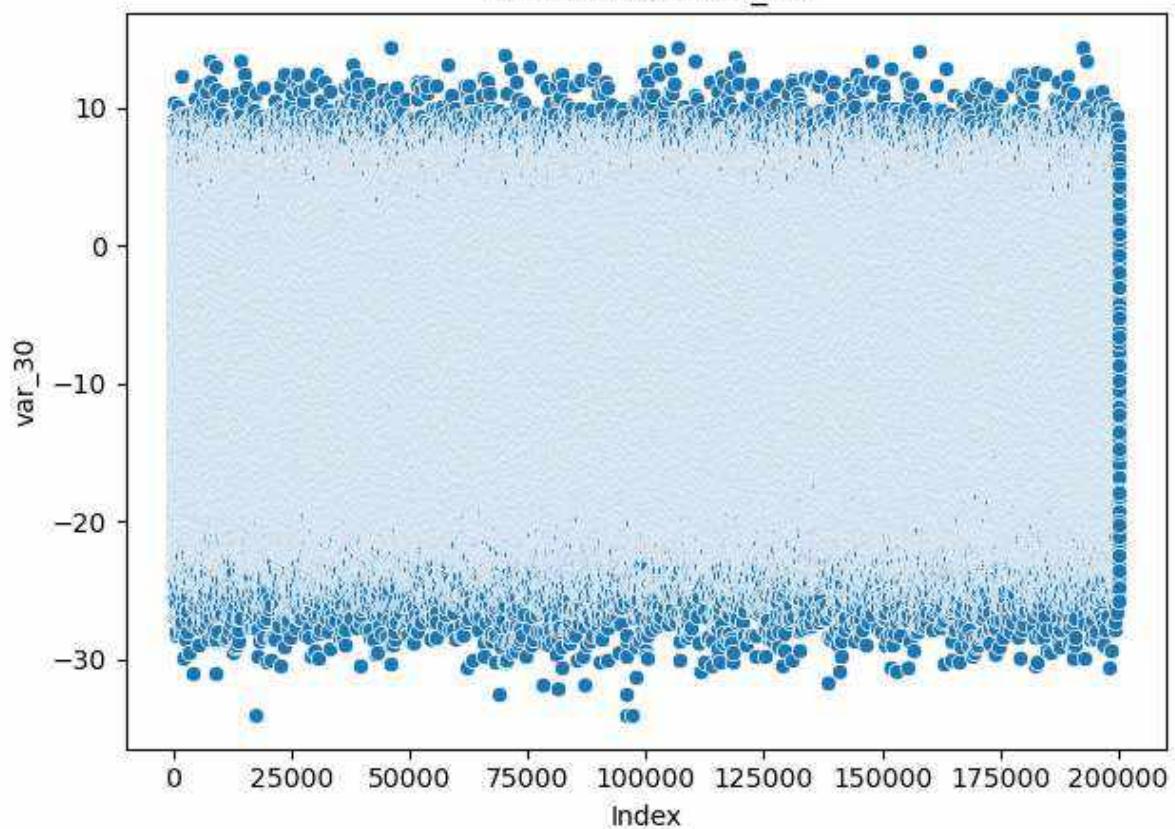
Scatter Plot - var\_28



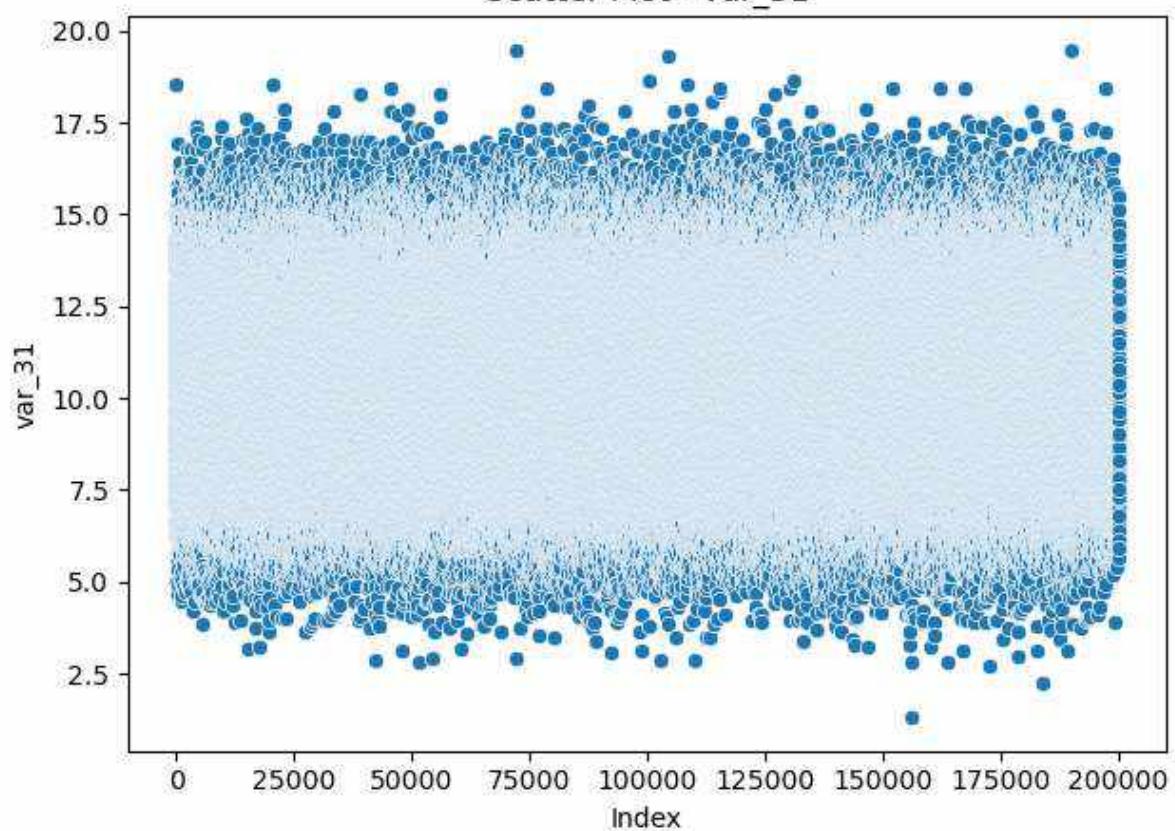
Scatter Plot - var\_29



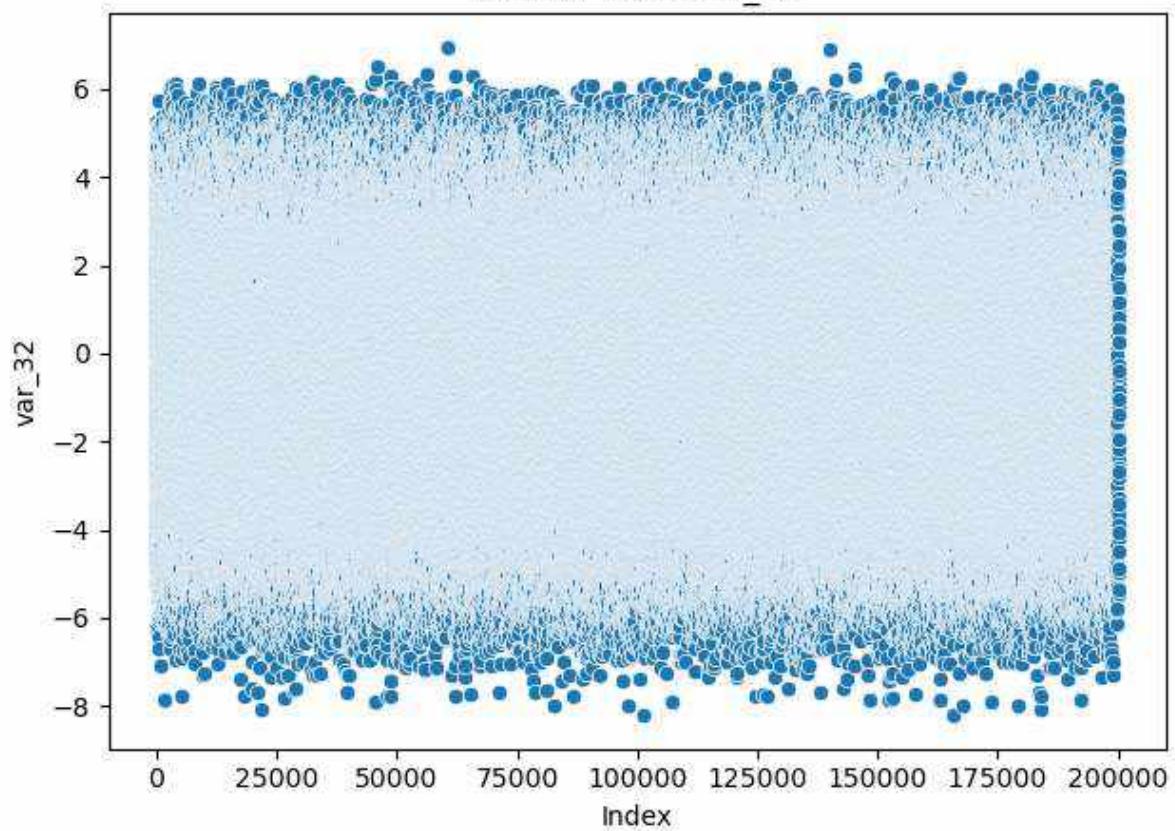
Scatter Plot - var\_30



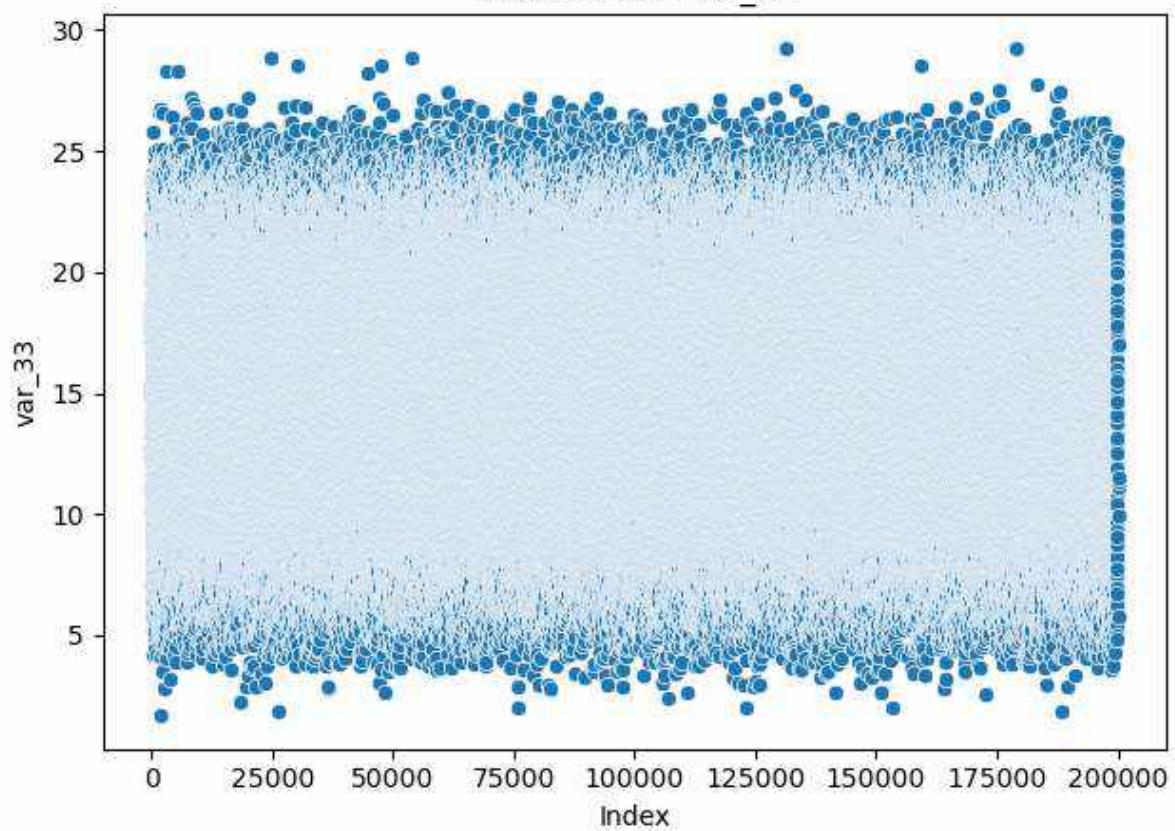
Scatter Plot - var\_31

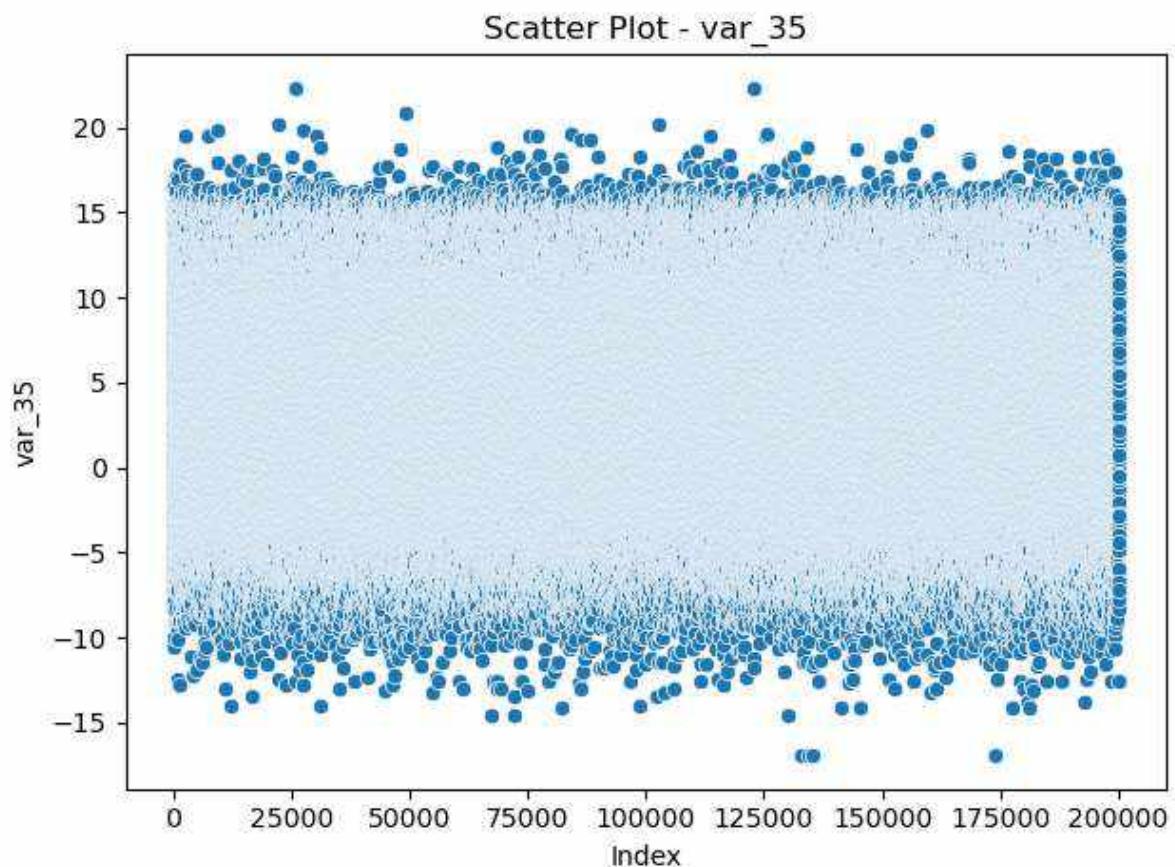
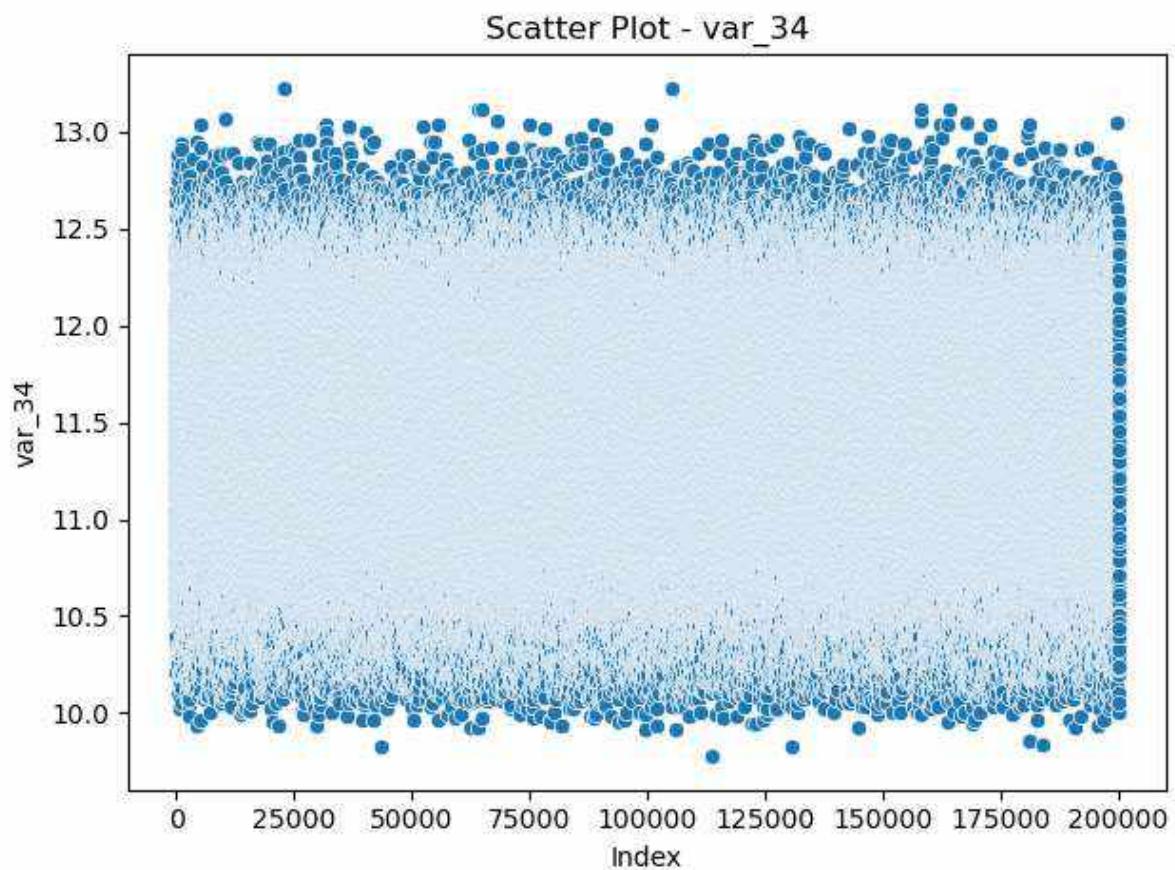


Scatter Plot - var\_32

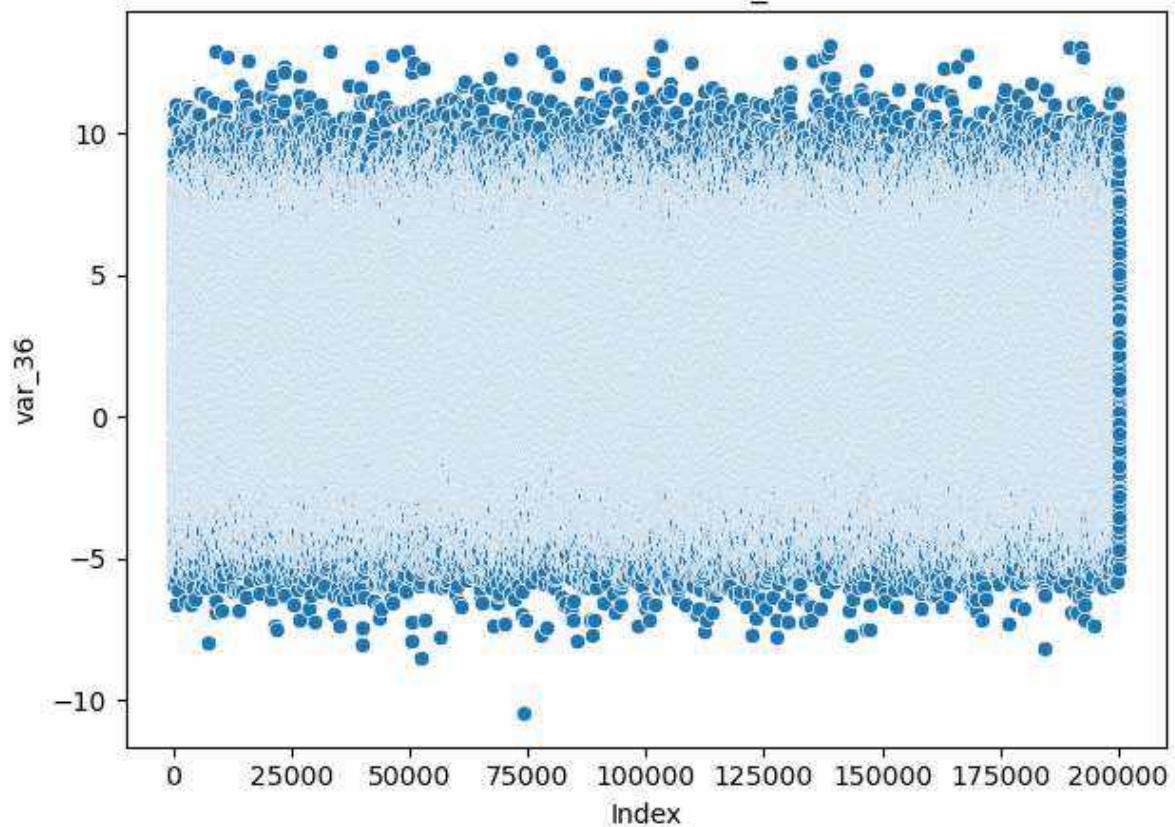


Scatter Plot - var\_33

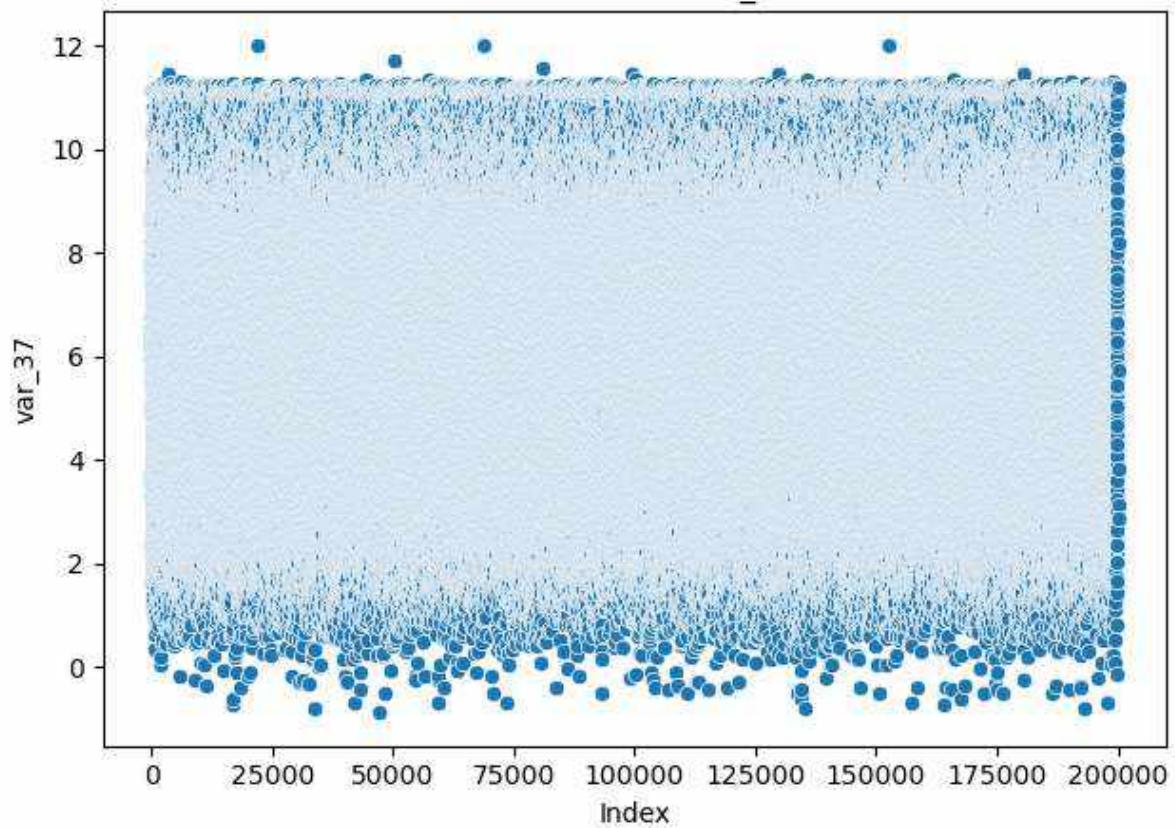




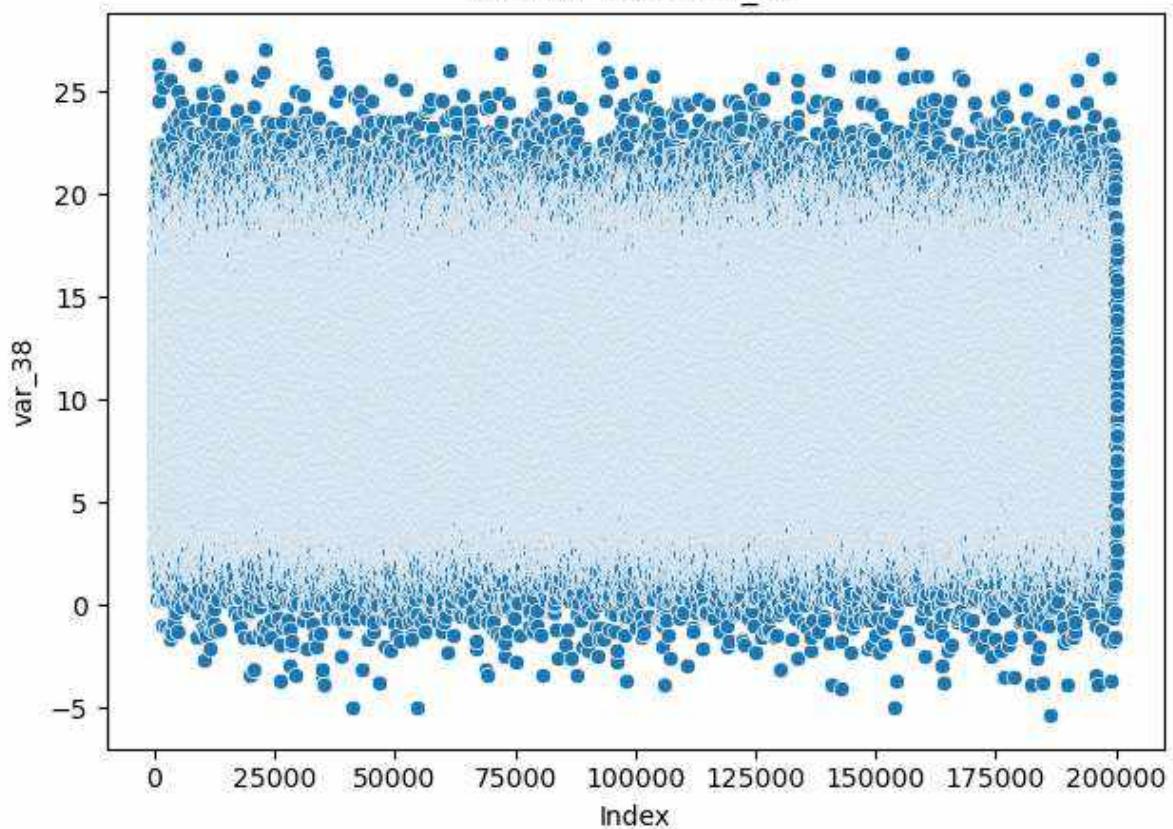
Scatter Plot - var\_36



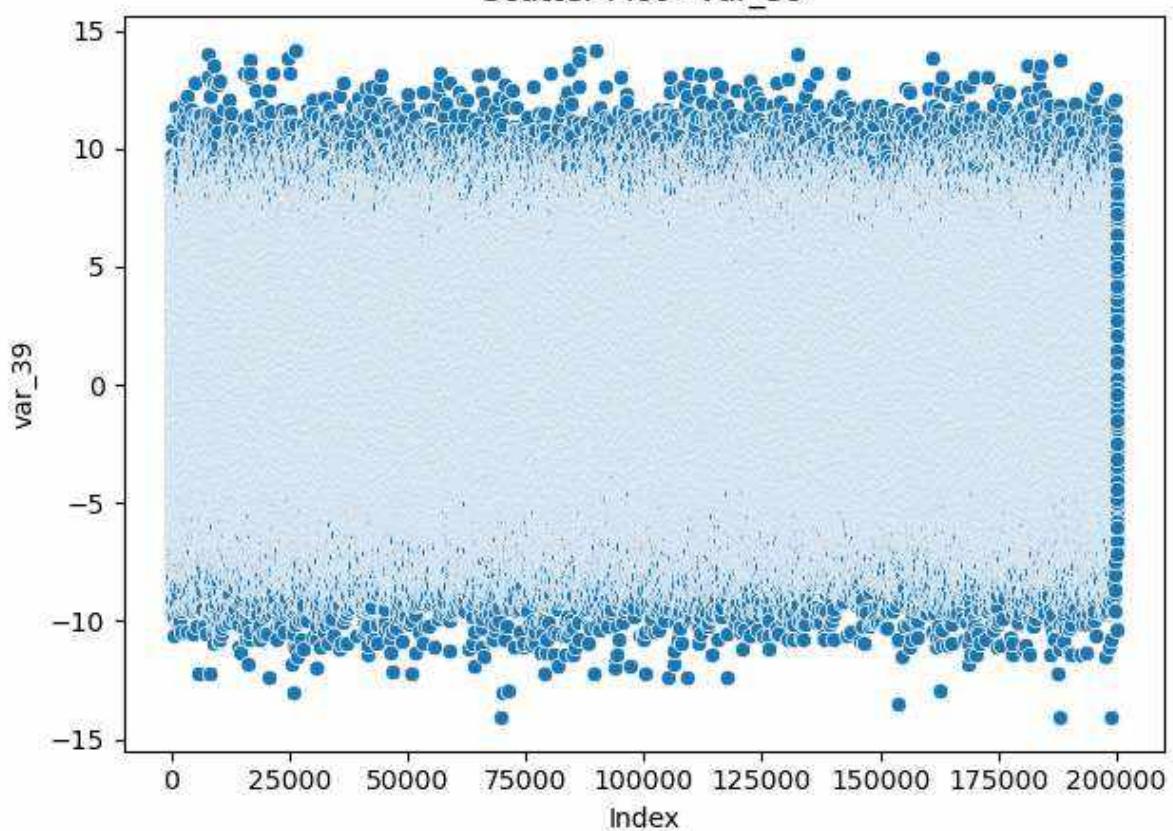
Scatter Plot - var\_37



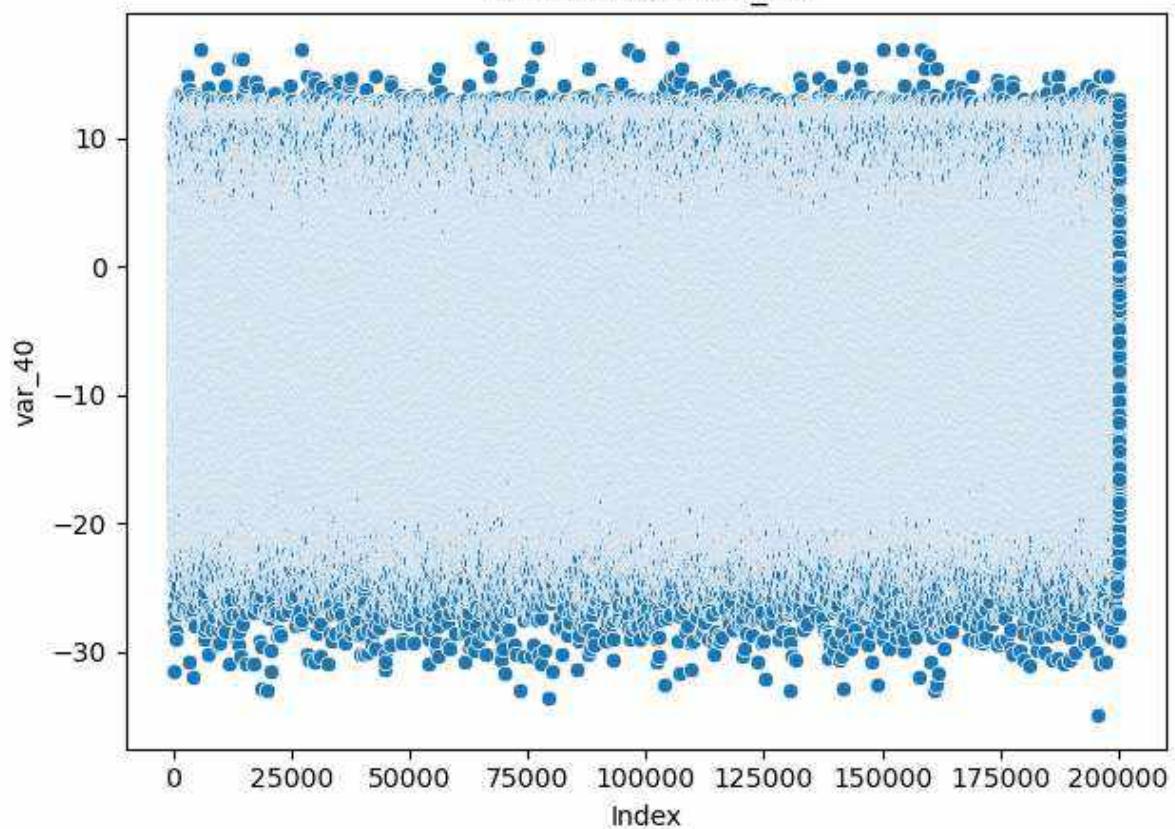
Scatter Plot - var\_38



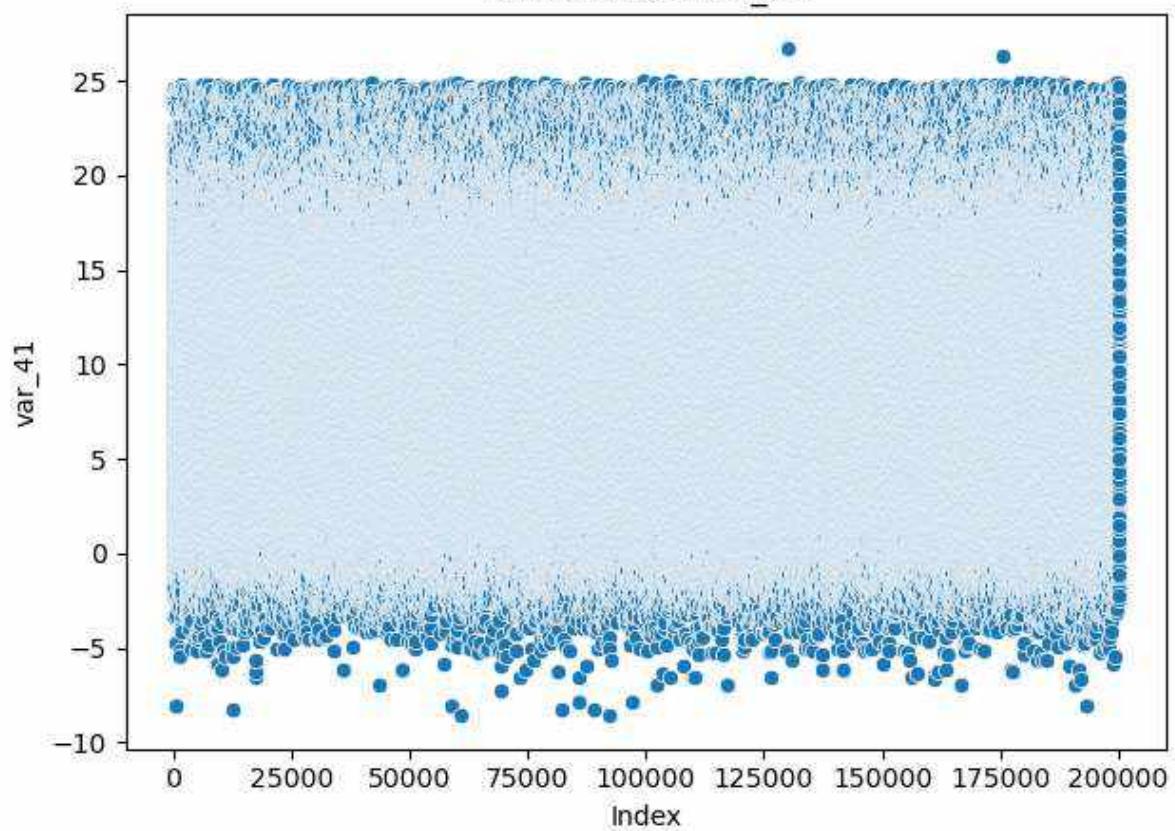
Scatter Plot - var\_39

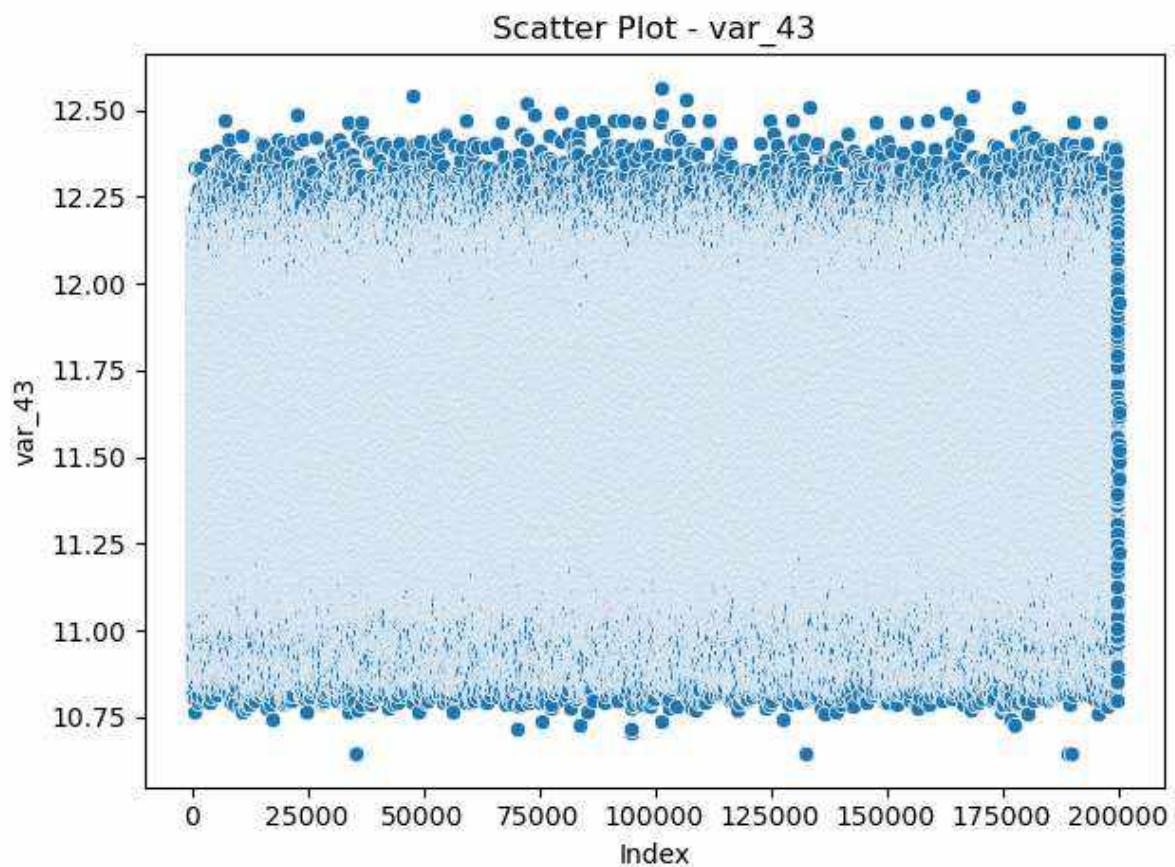
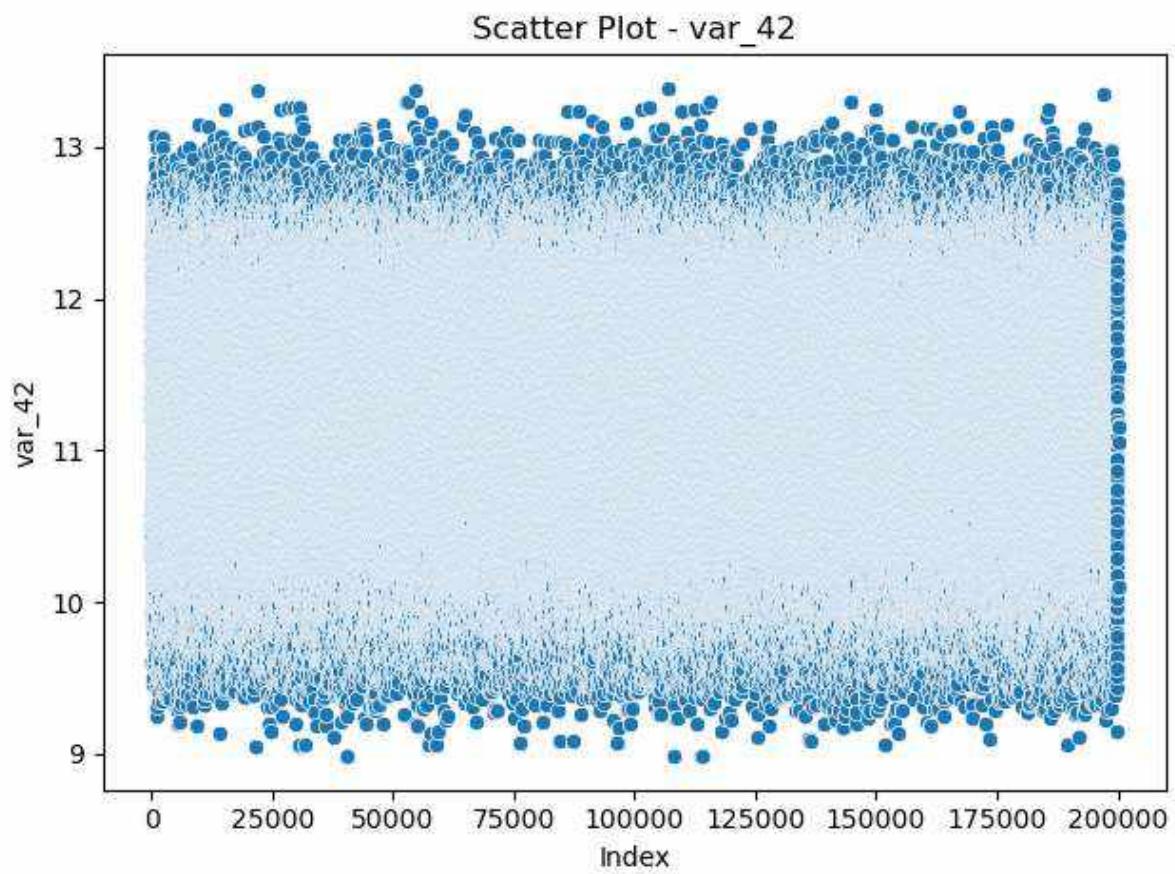


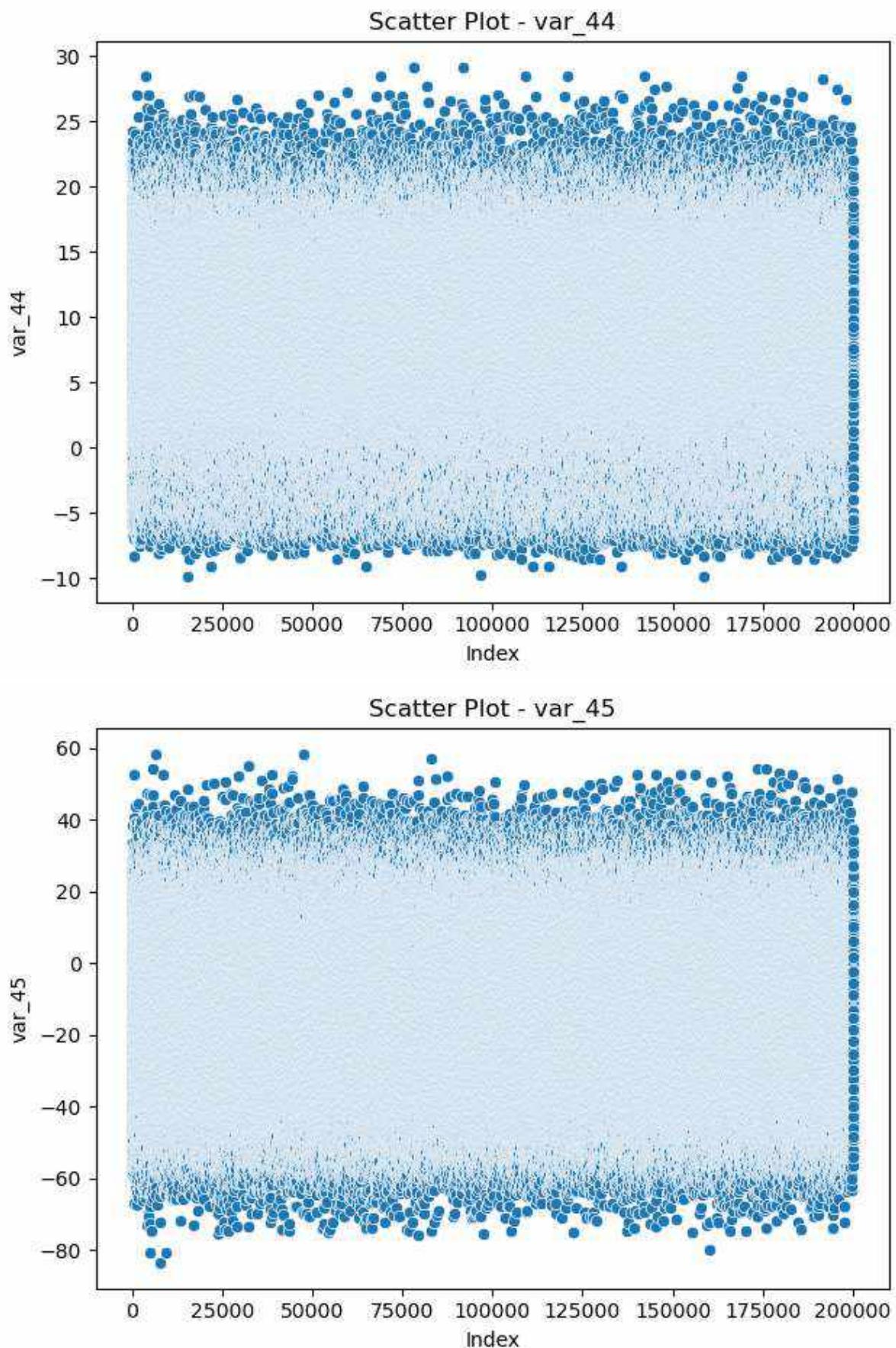
Scatter Plot - var\_40

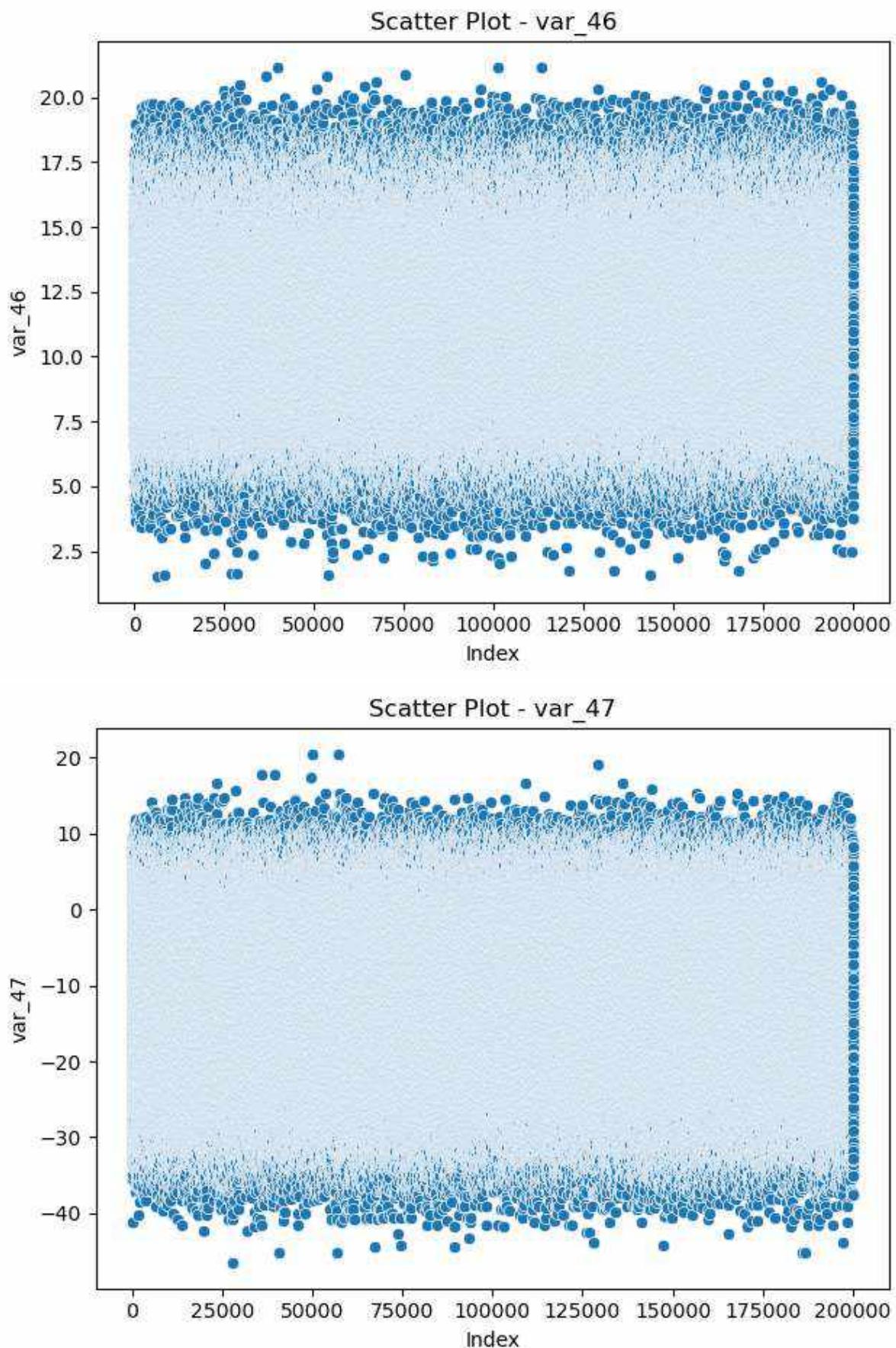


Scatter Plot - var\_41

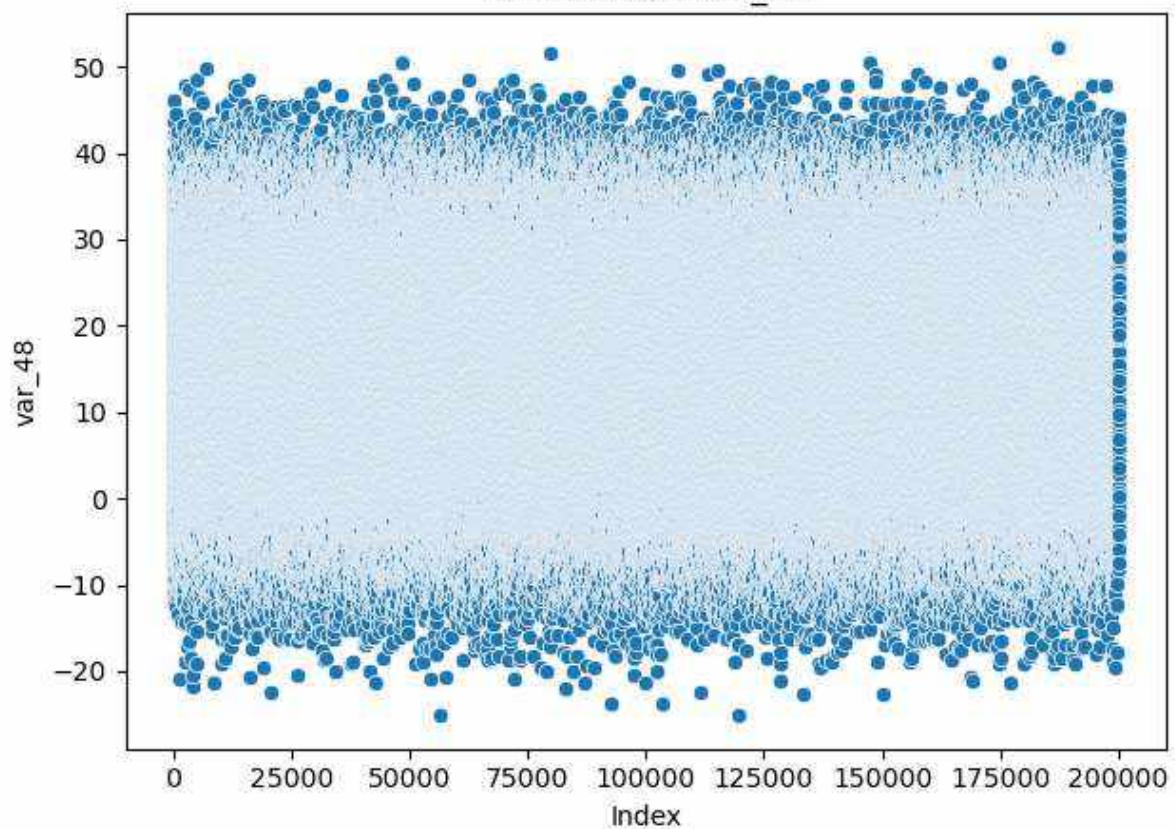




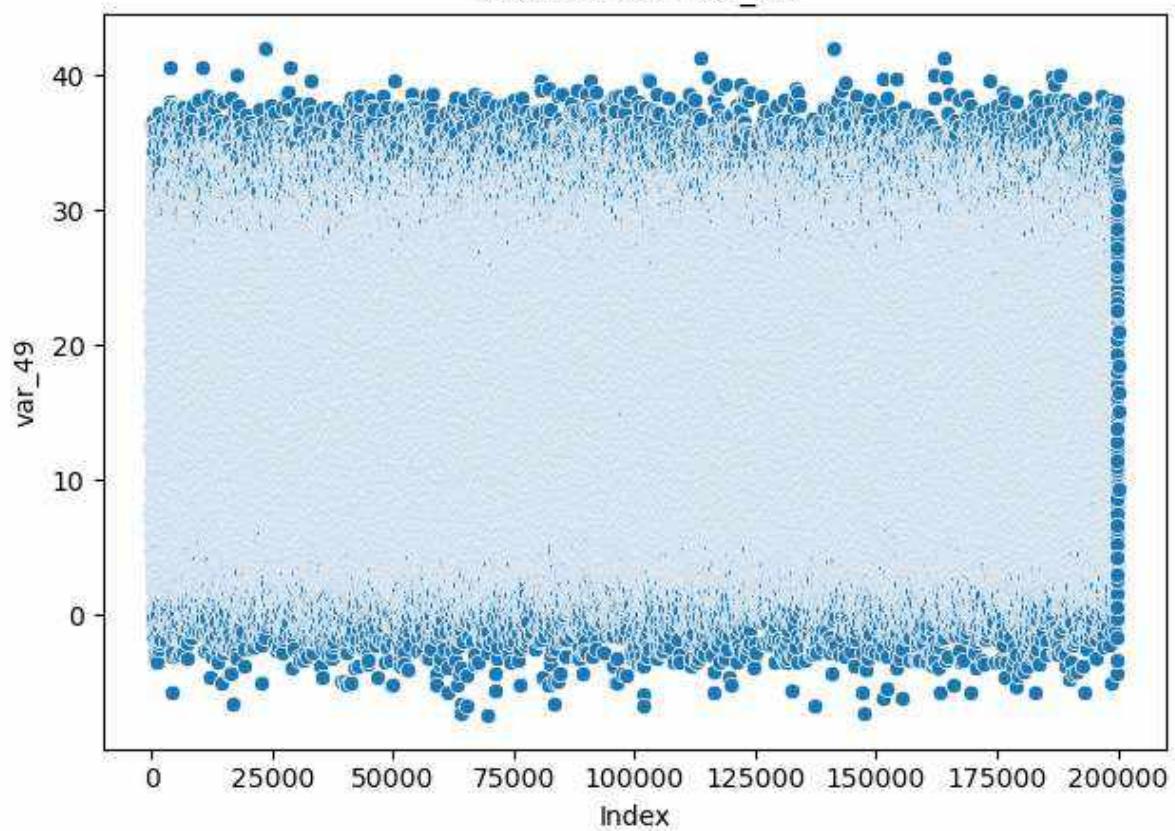


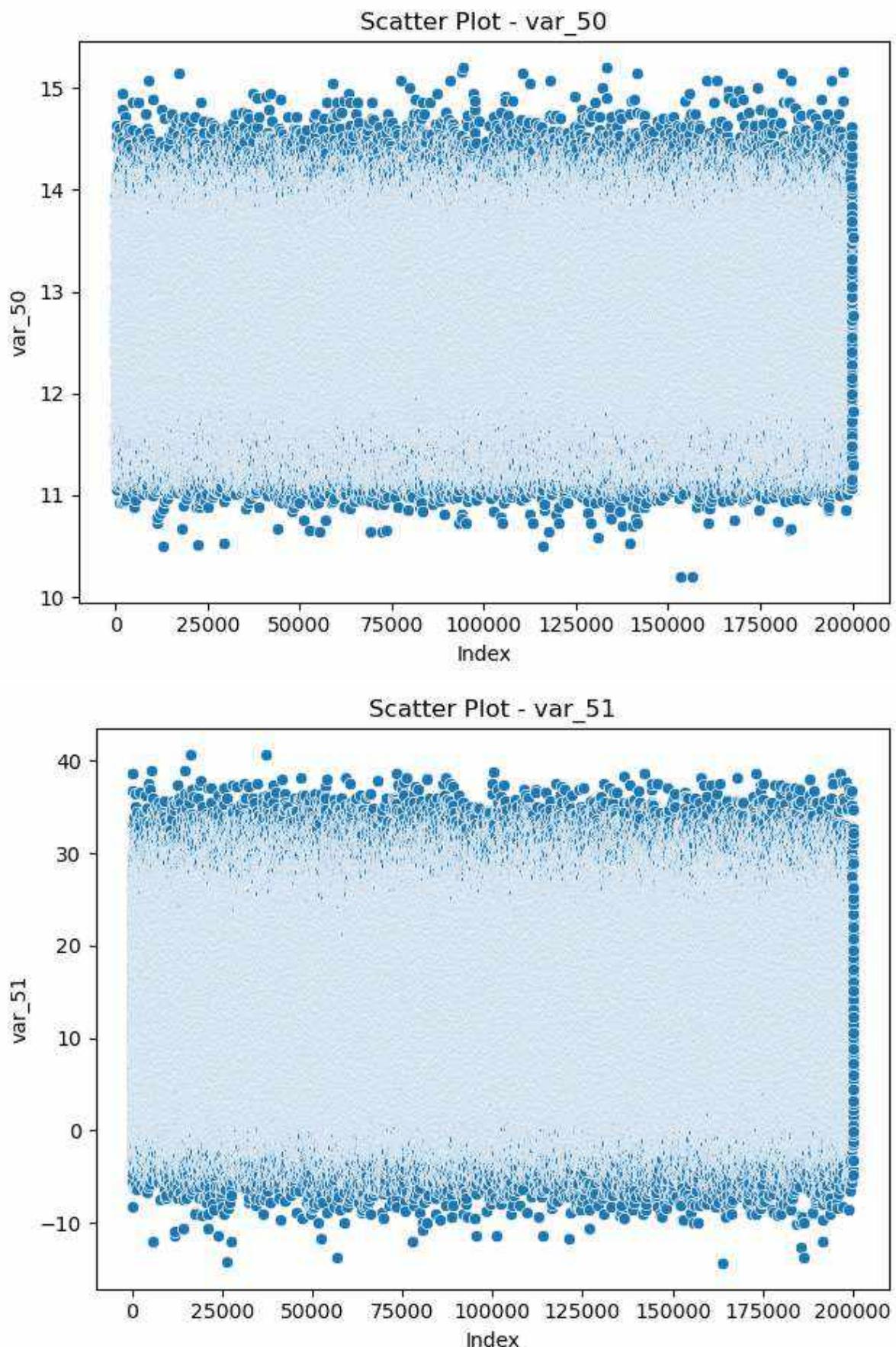


Scatter Plot - var\_48

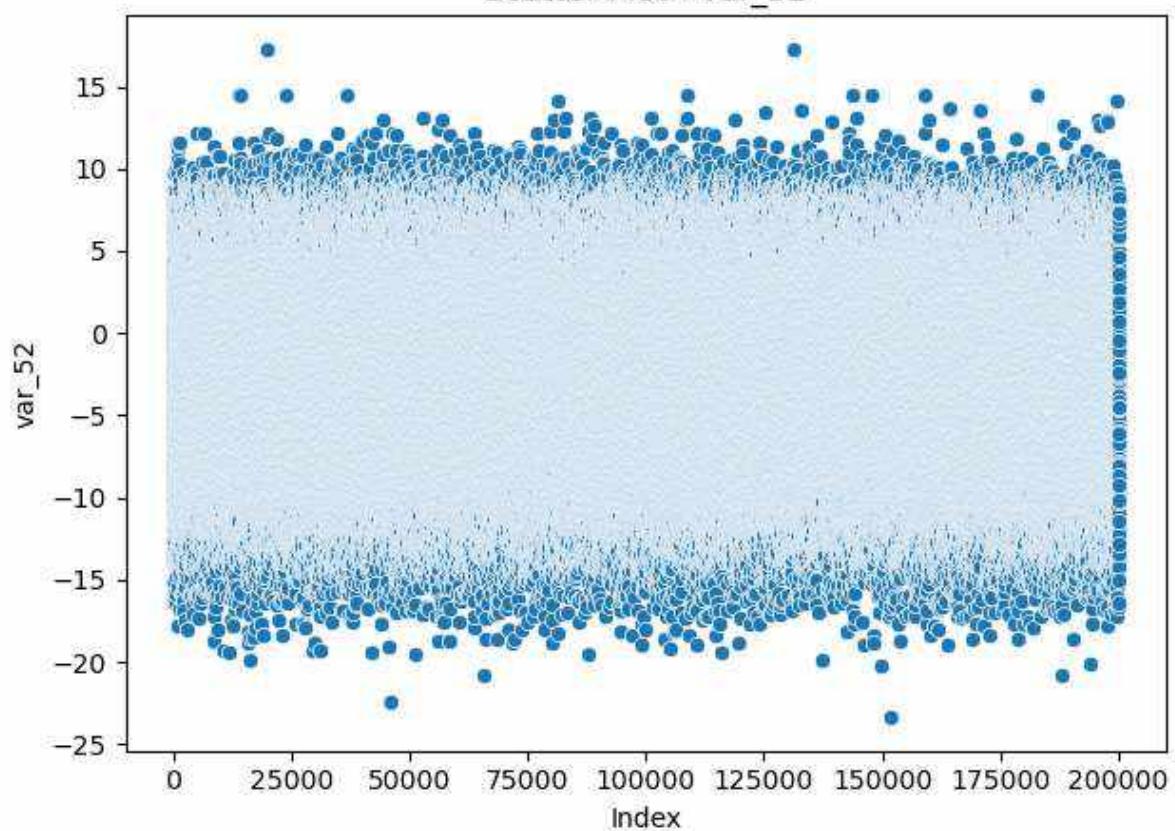


Scatter Plot - var\_49

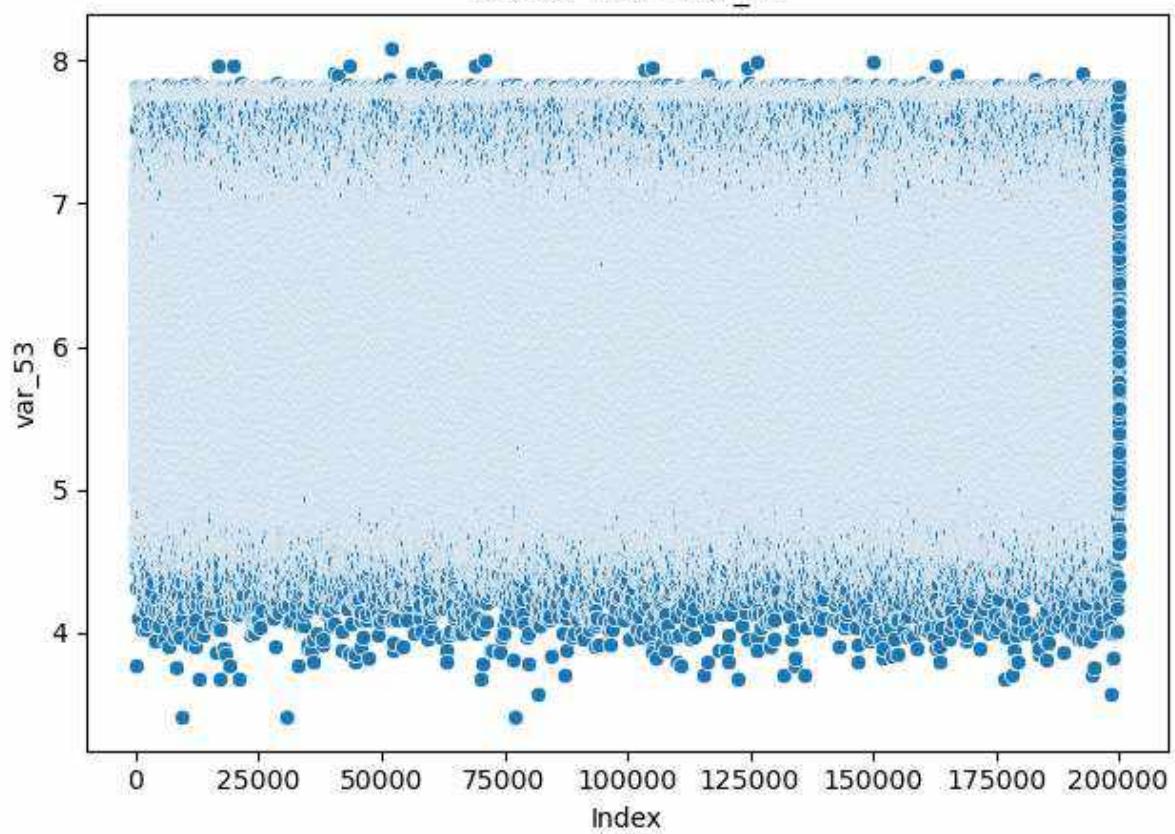




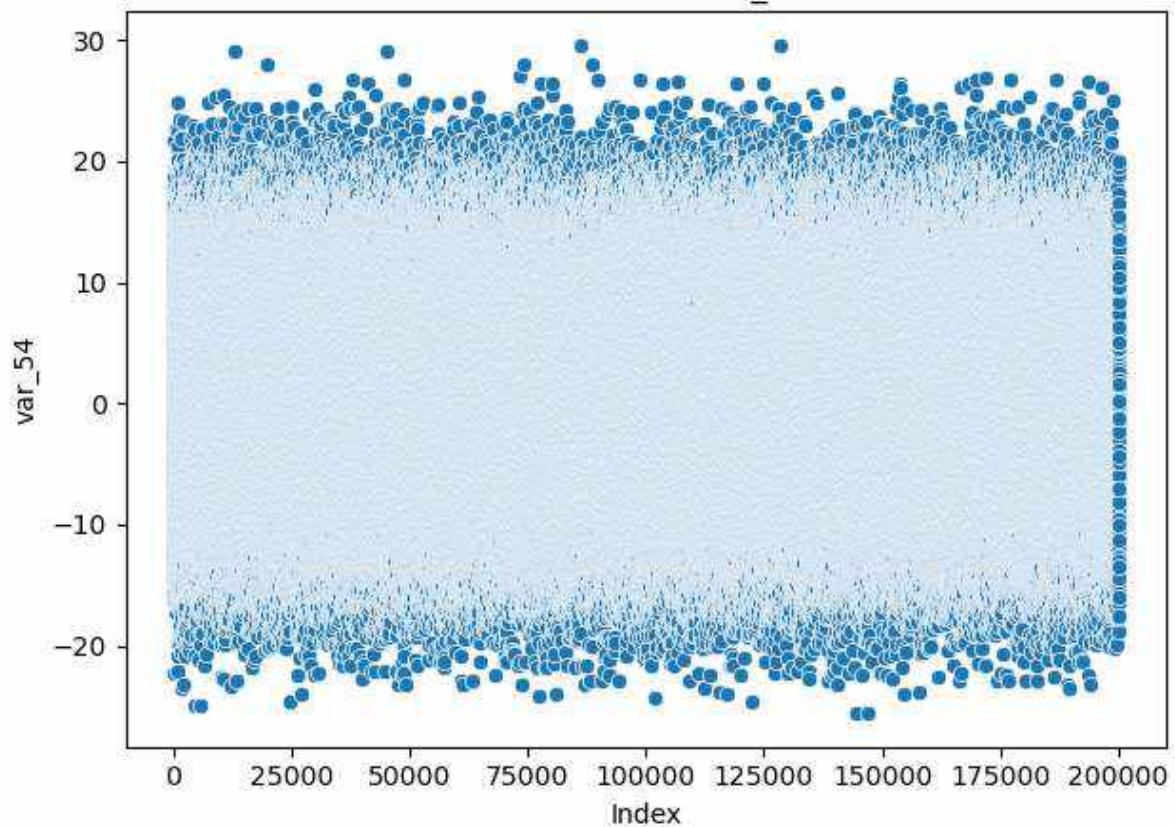
Scatter Plot - var\_52



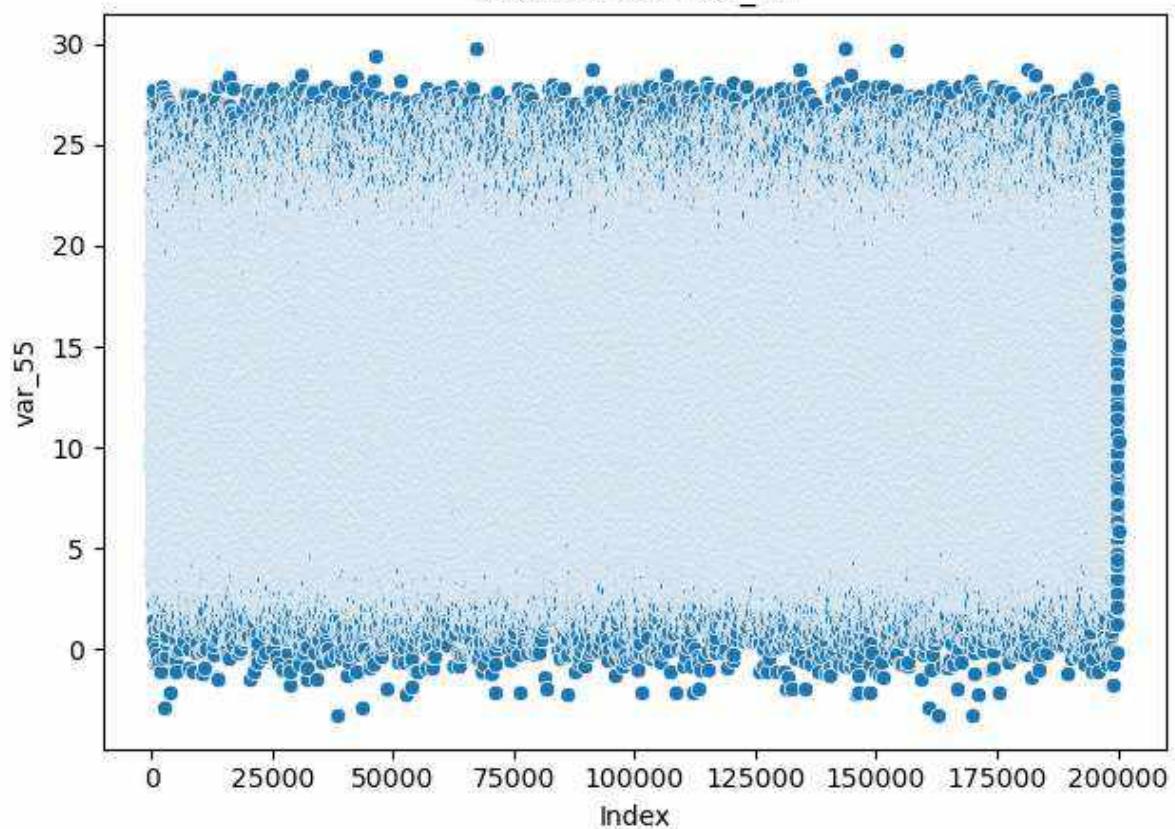
Scatter Plot - var\_53



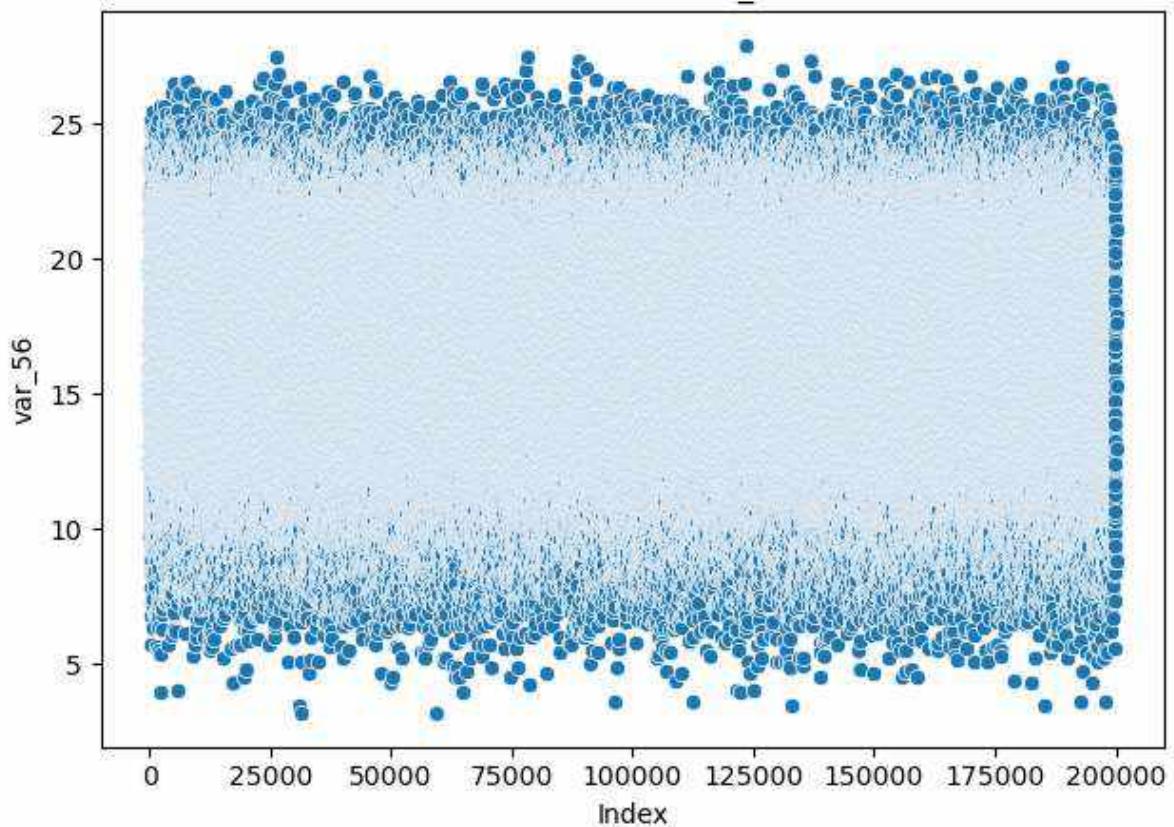
Scatter Plot - var\_54



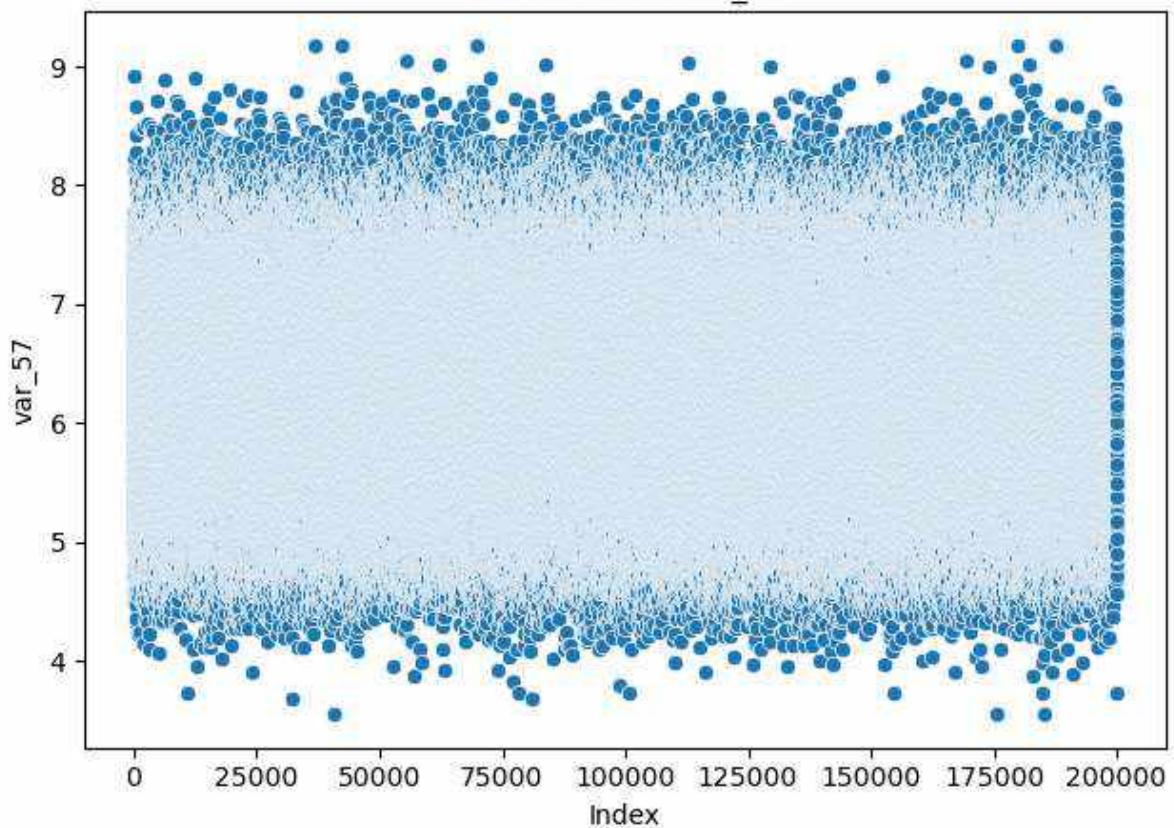
Scatter Plot - var\_55

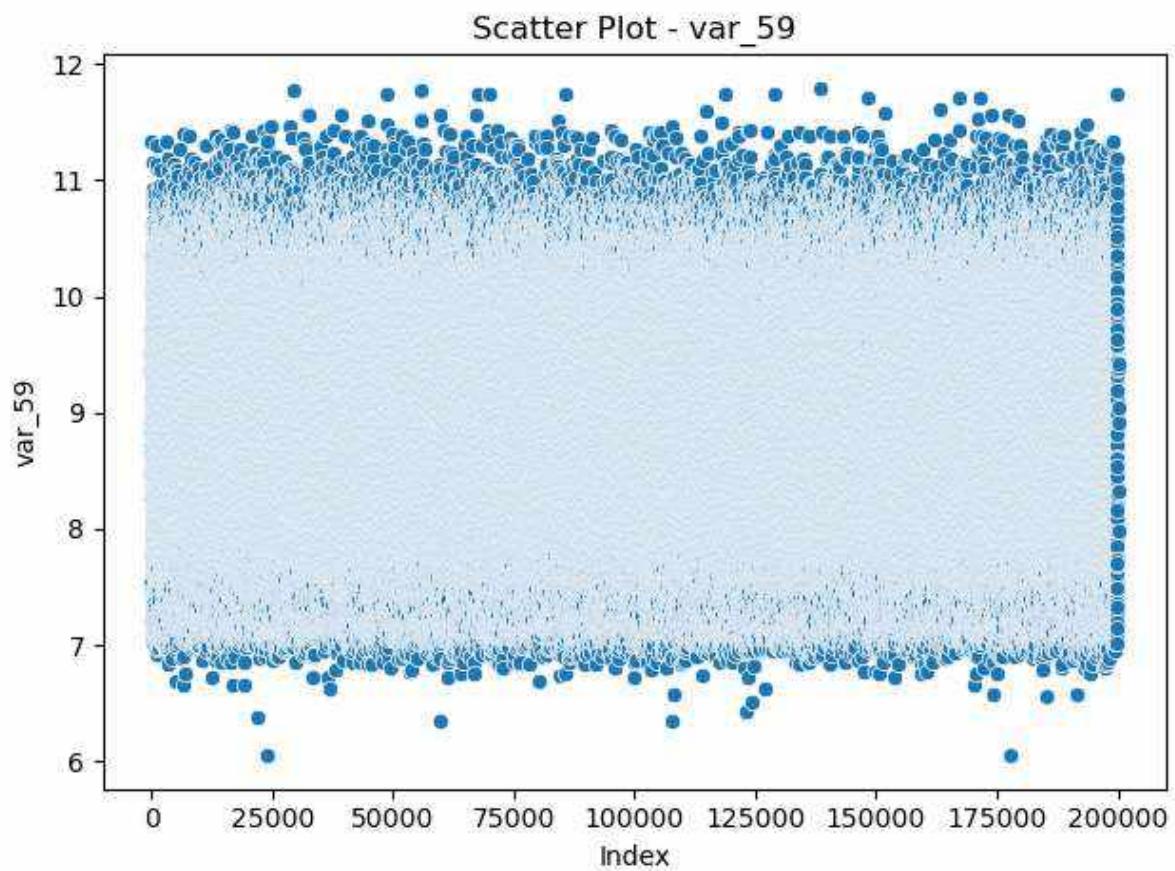
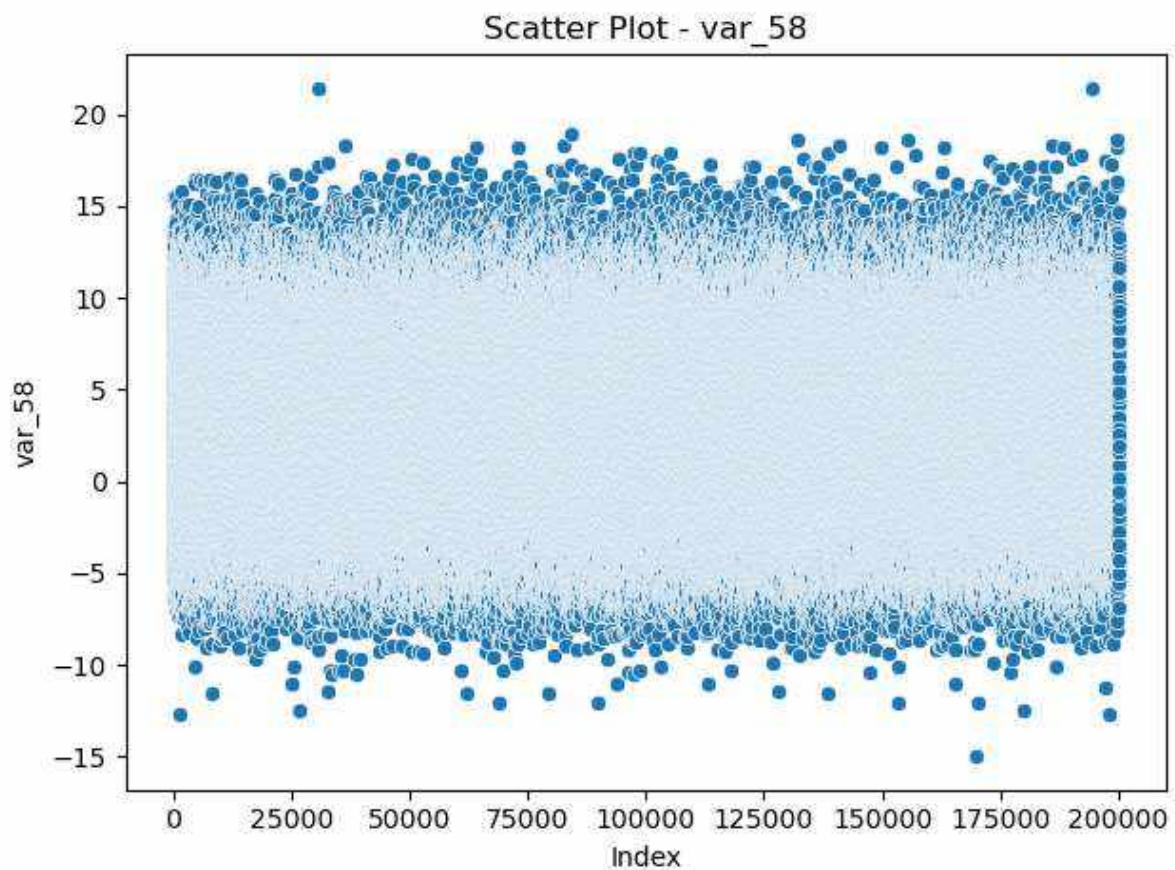


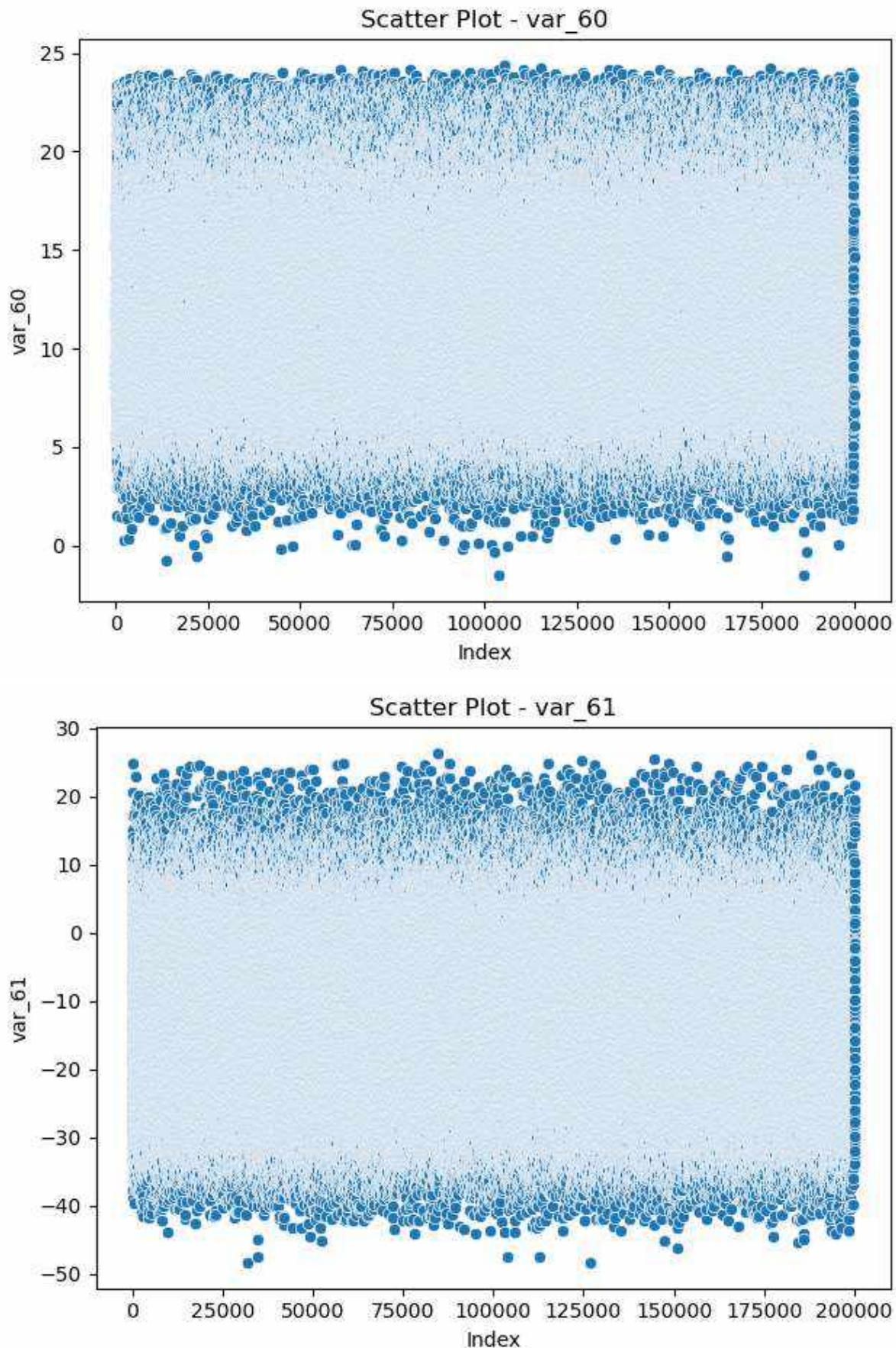
Scatter Plot - var\_56

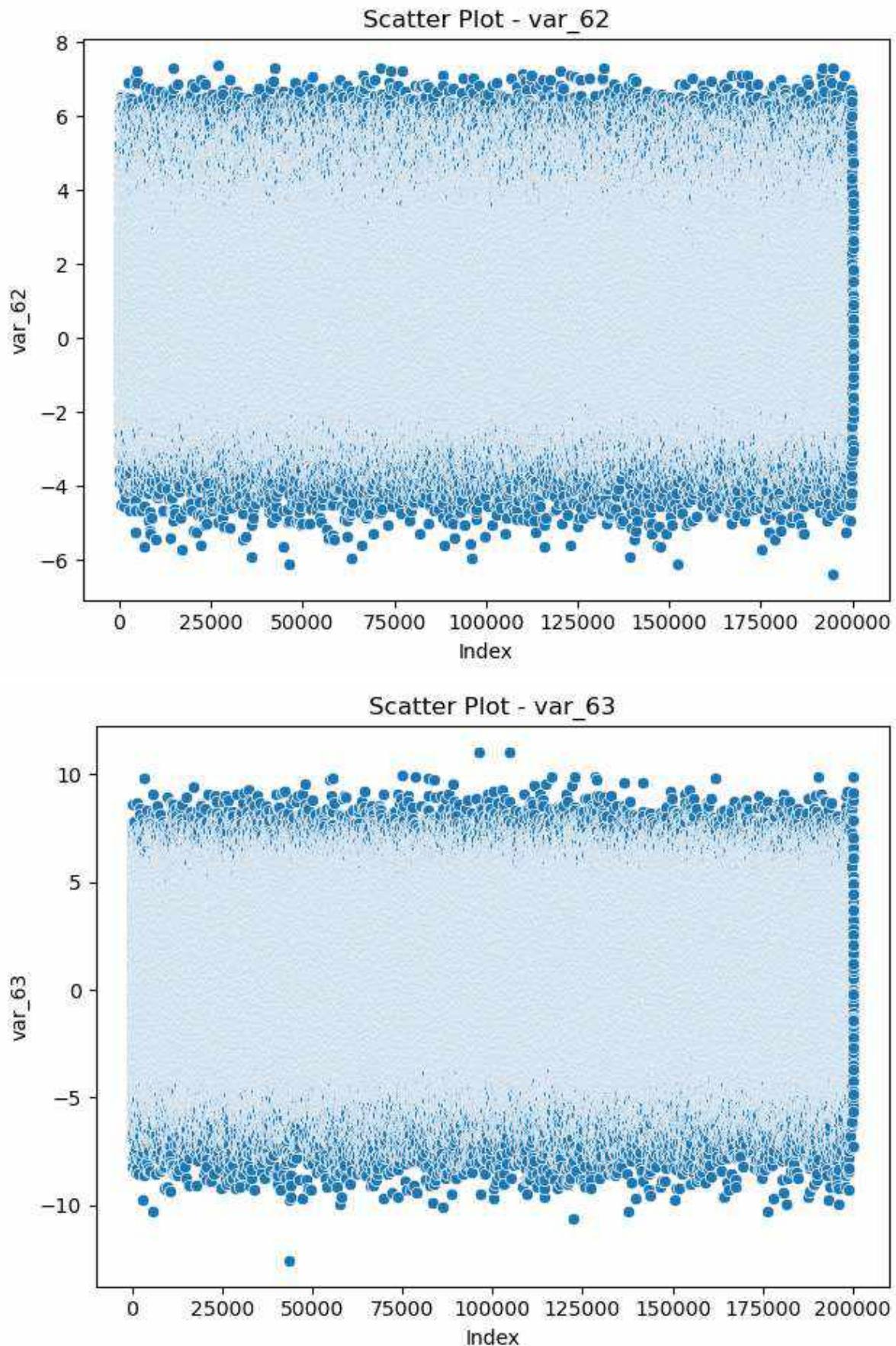


Scatter Plot - var\_57

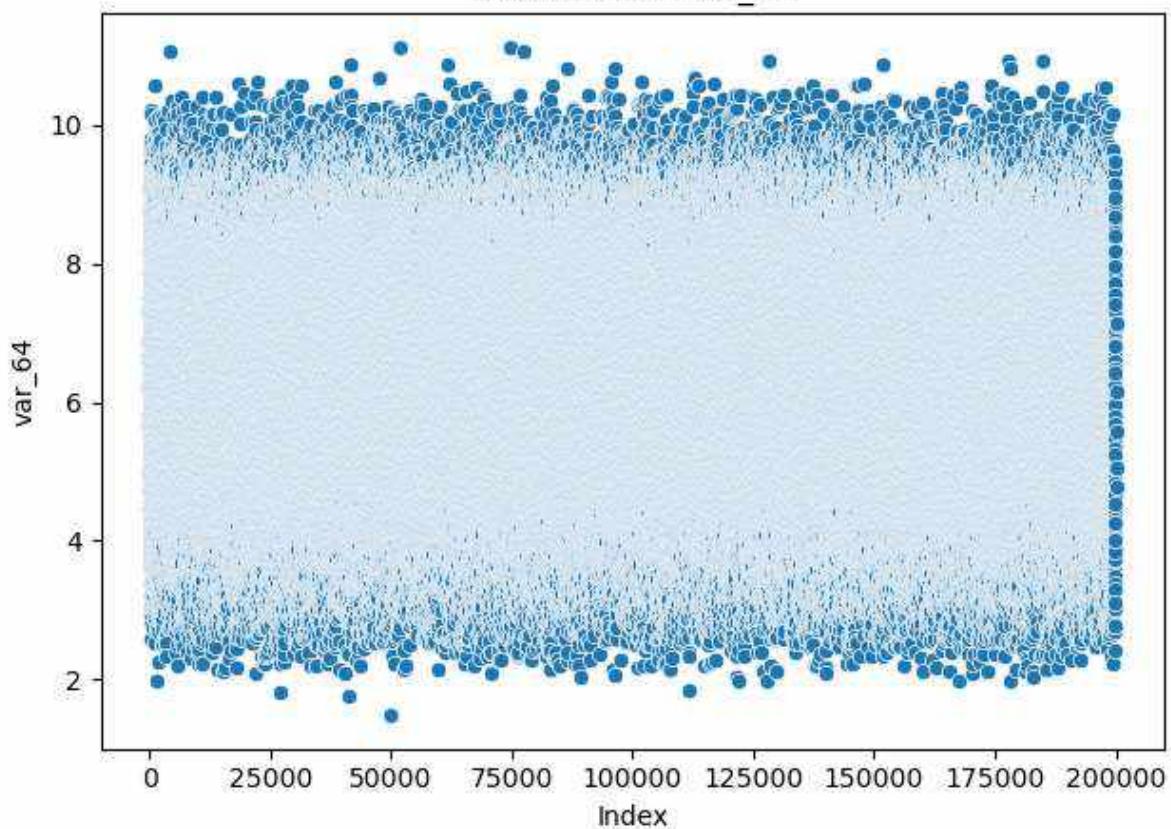




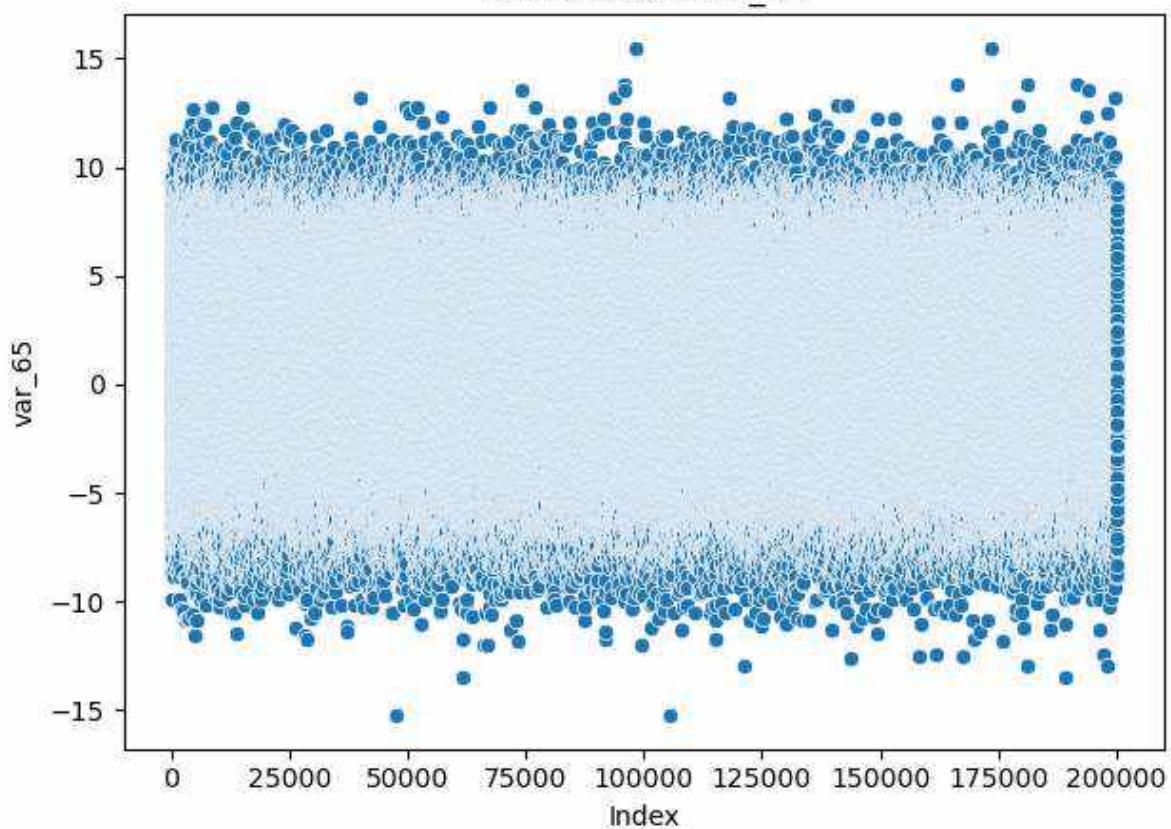




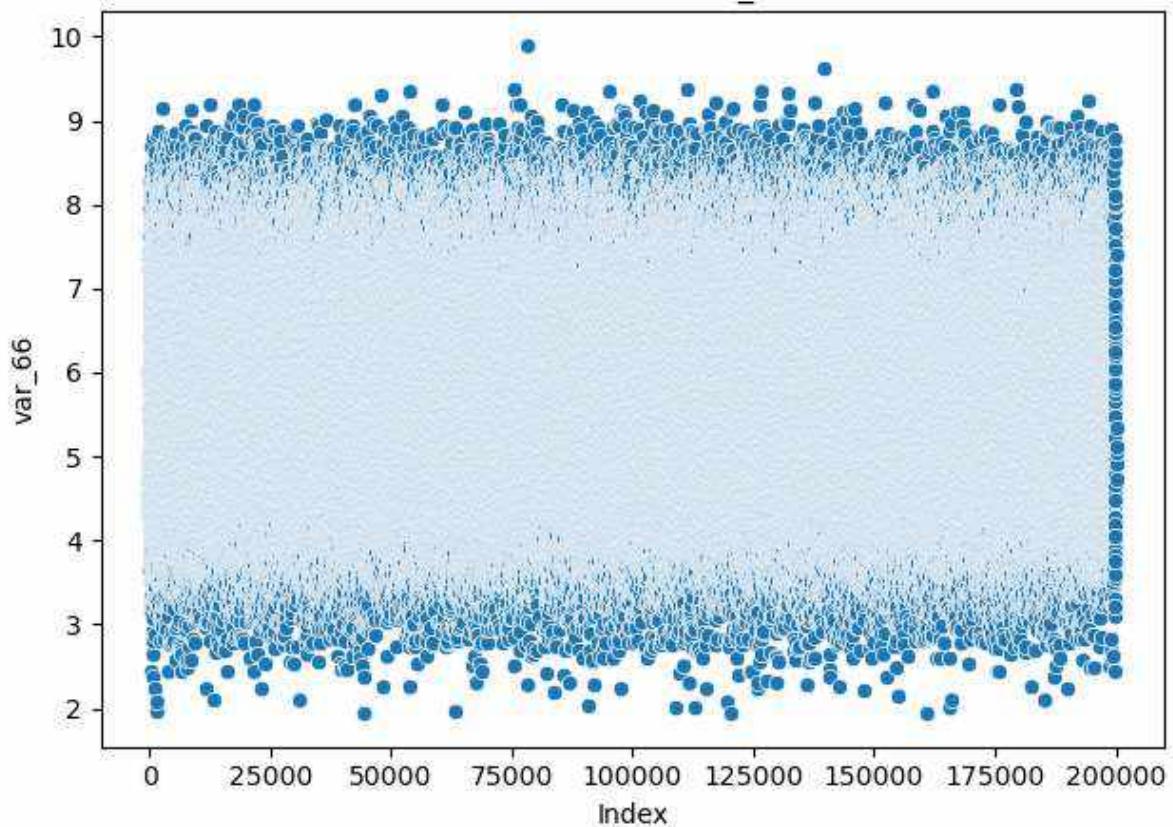
Scatter Plot - var\_64



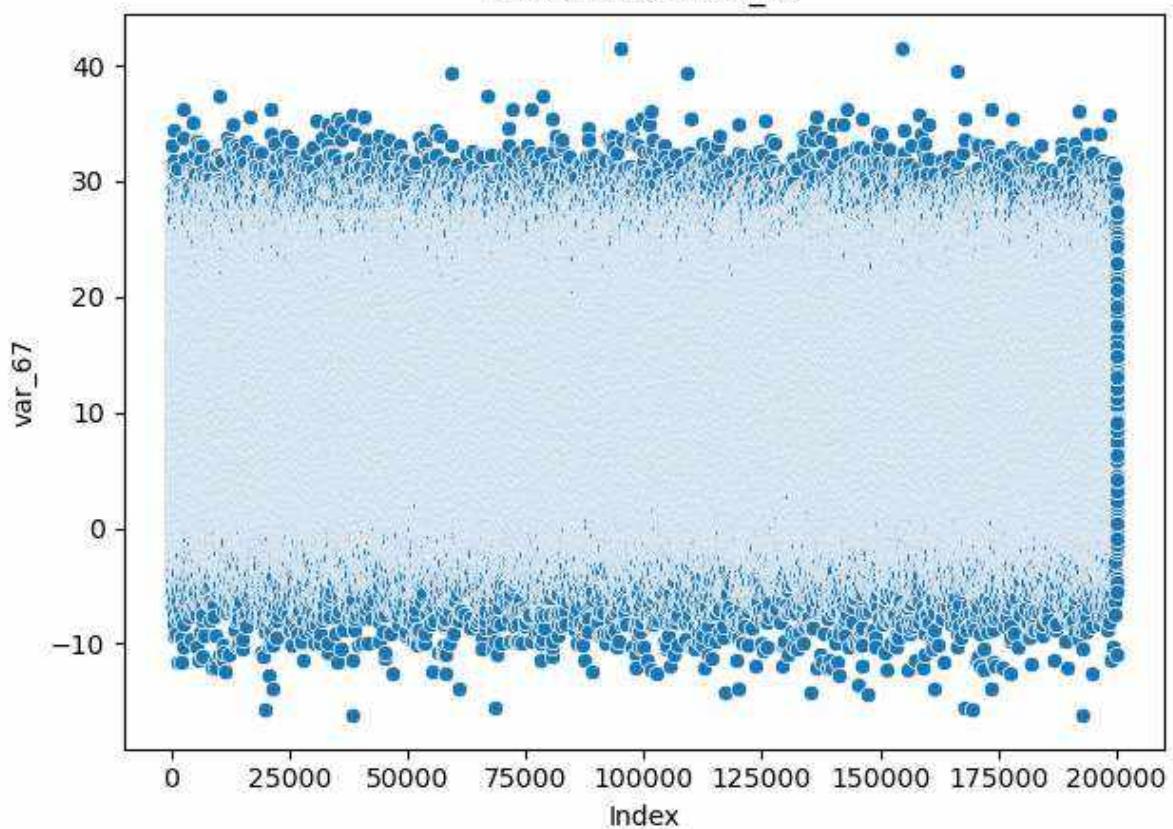
Scatter Plot - var\_65



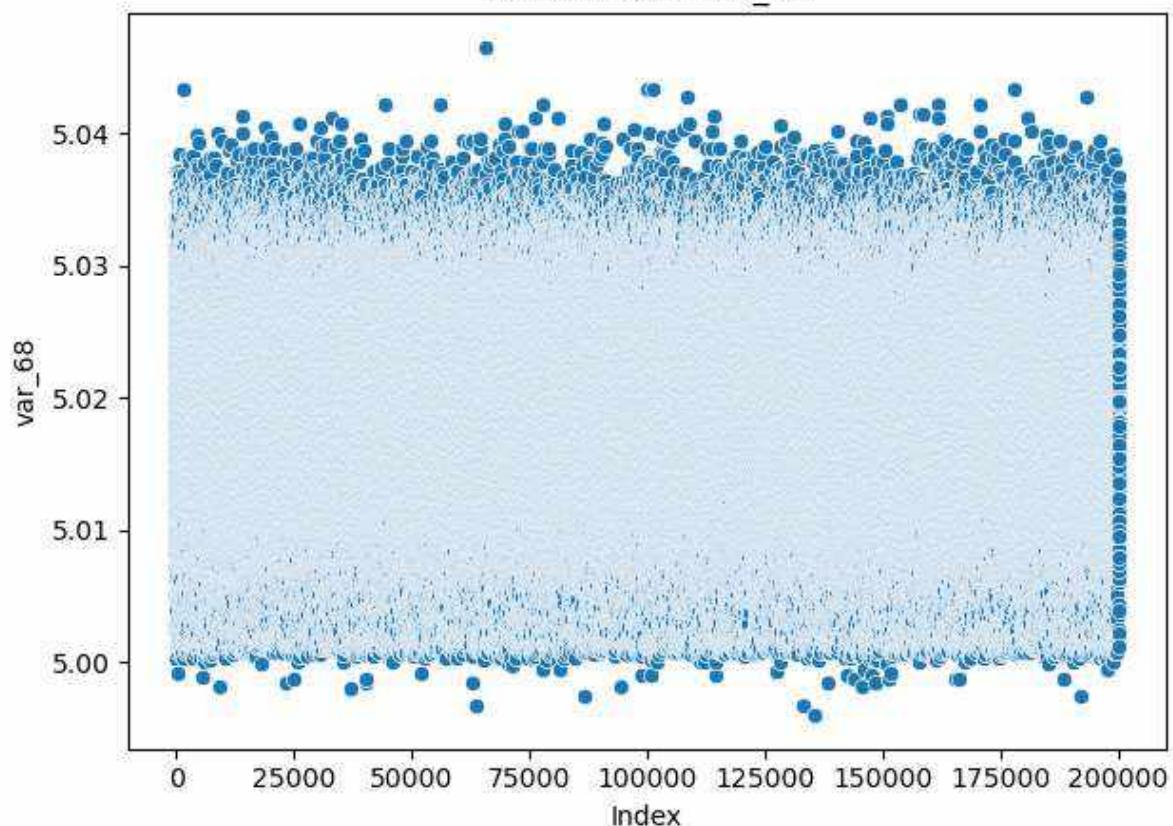
Scatter Plot - var\_66



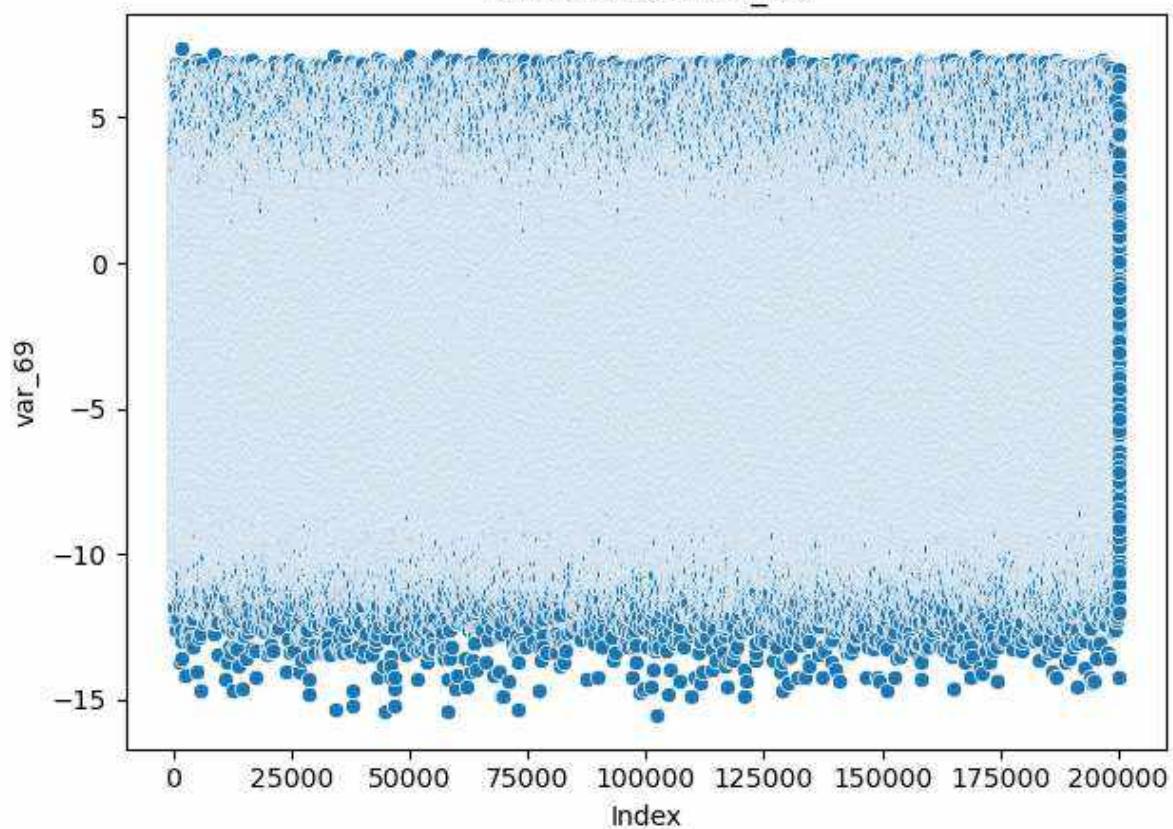
Scatter Plot - var\_67



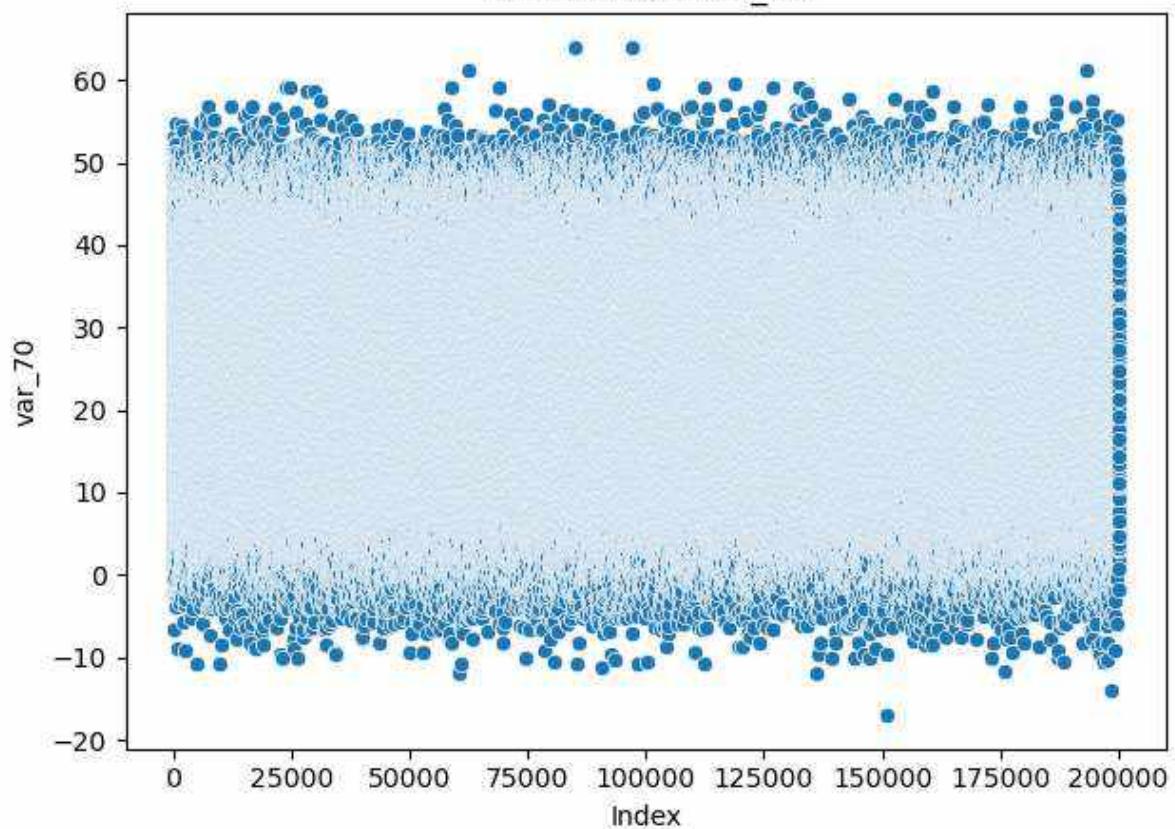
Scatter Plot - var\_68



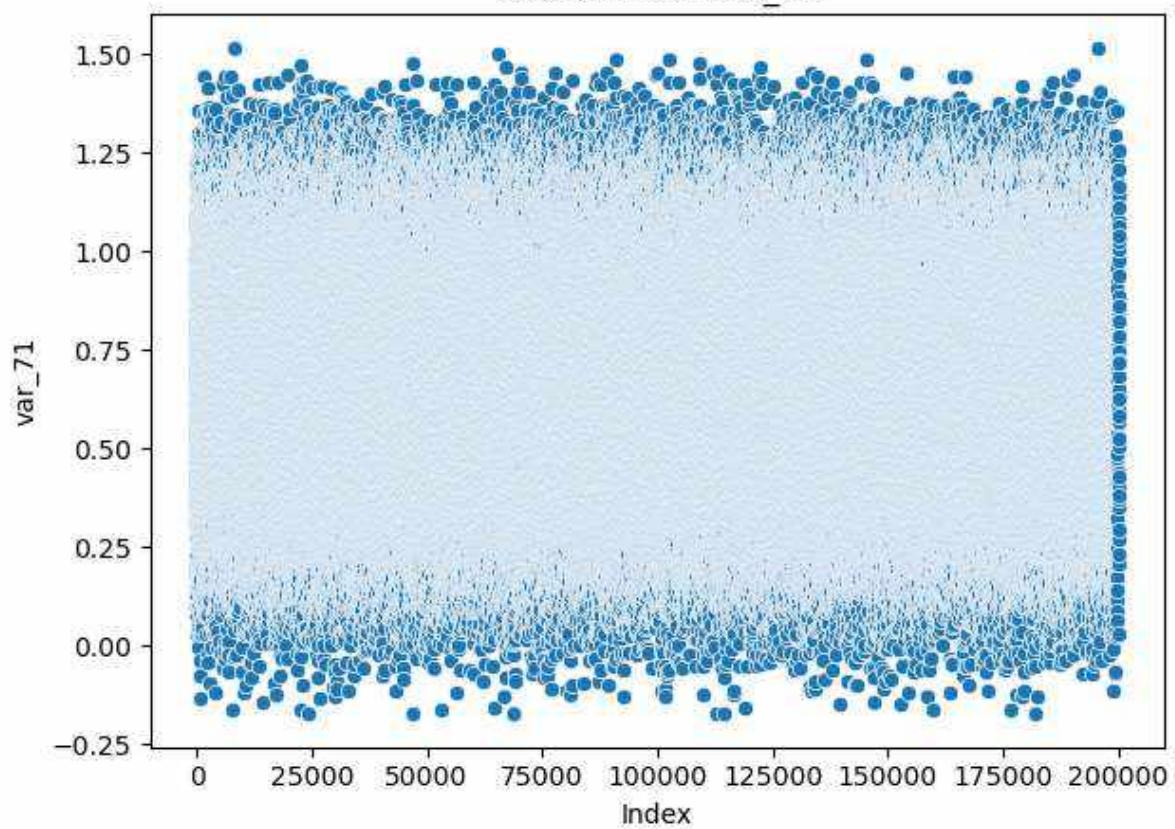
Scatter Plot - var\_69

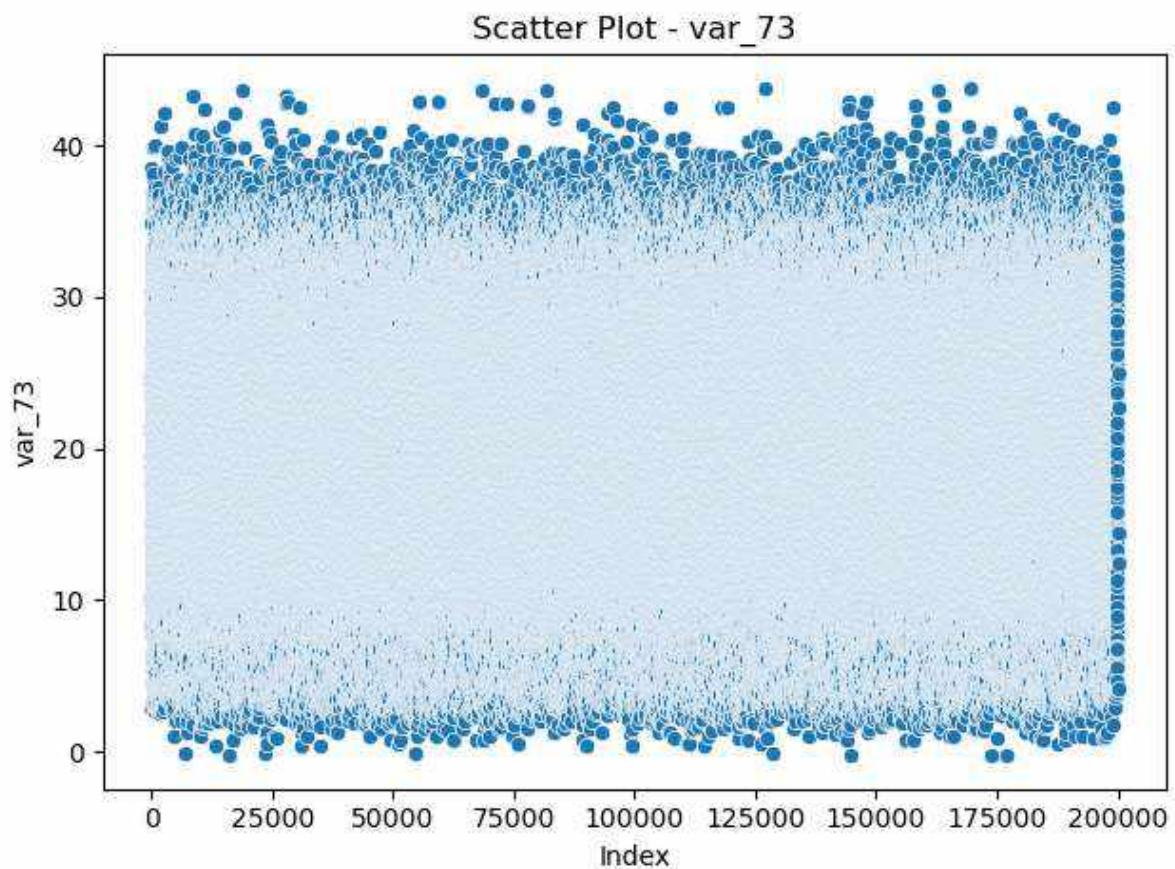
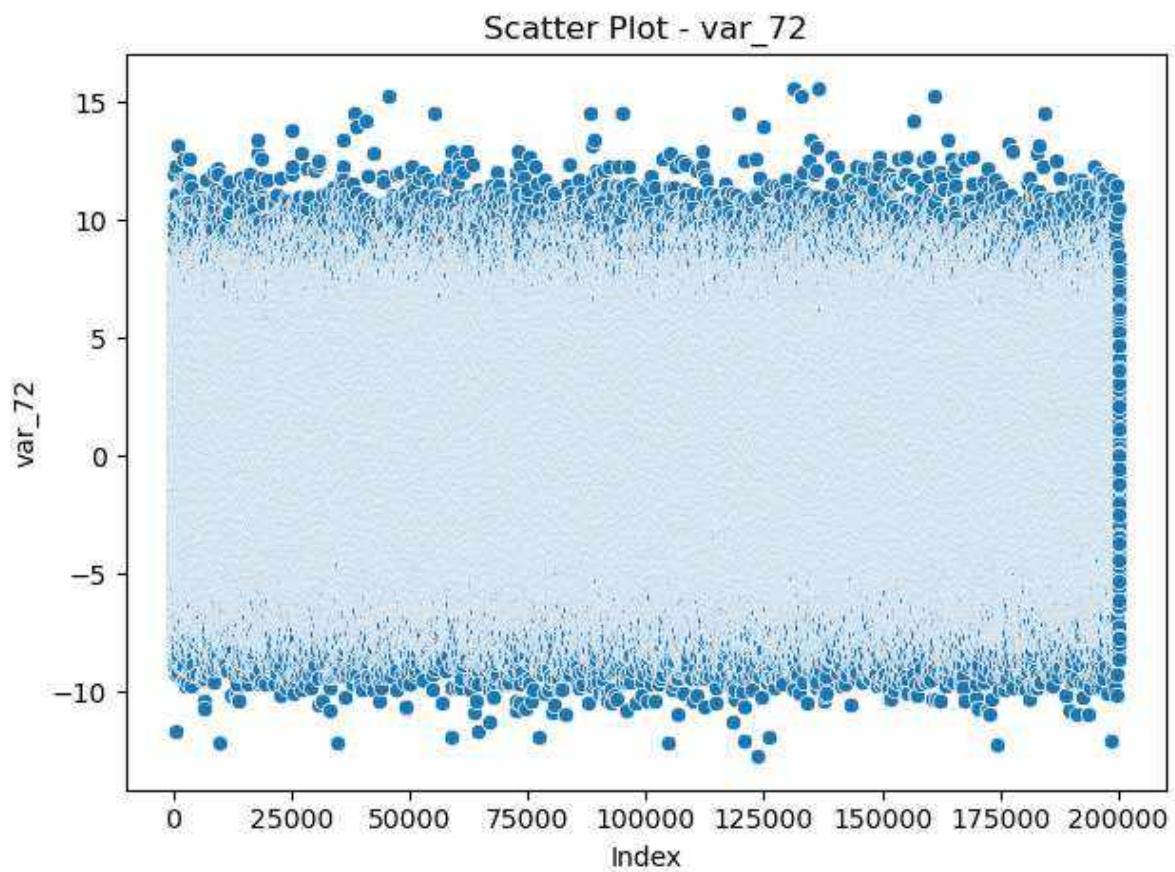


Scatter Plot - var\_70

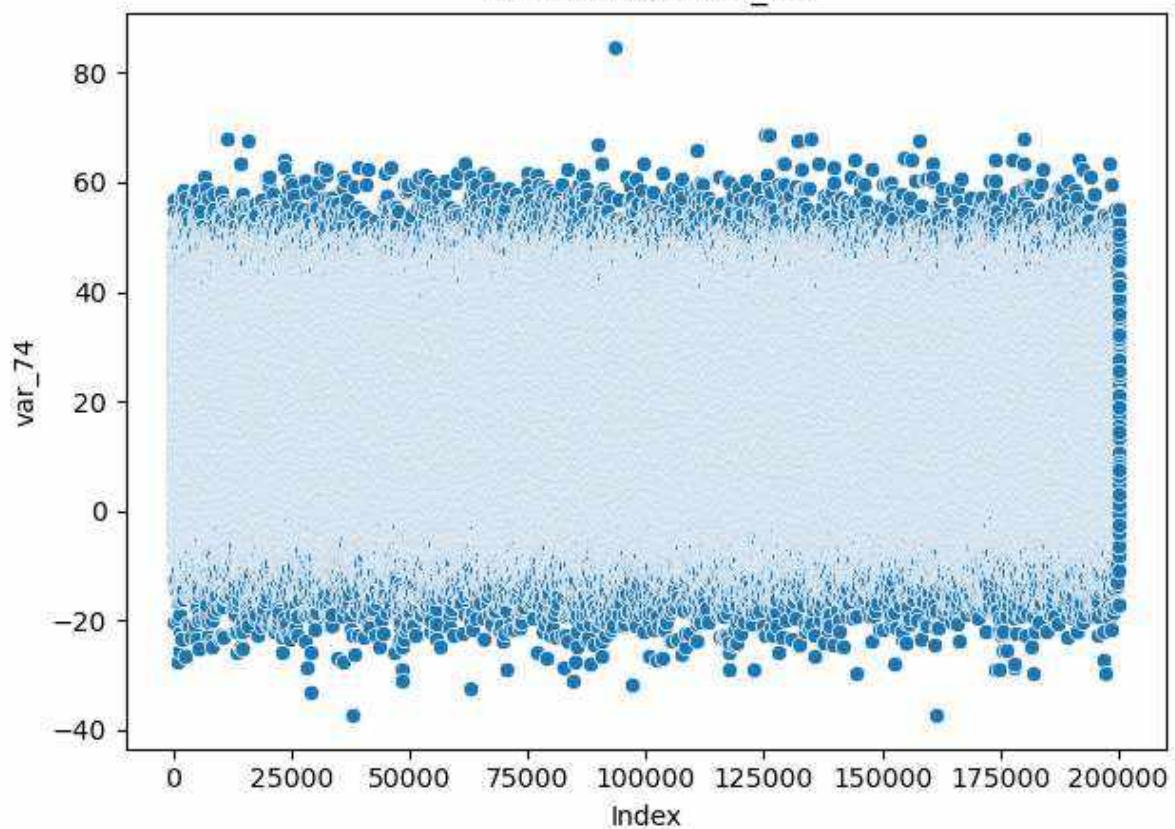


Scatter Plot - var\_71

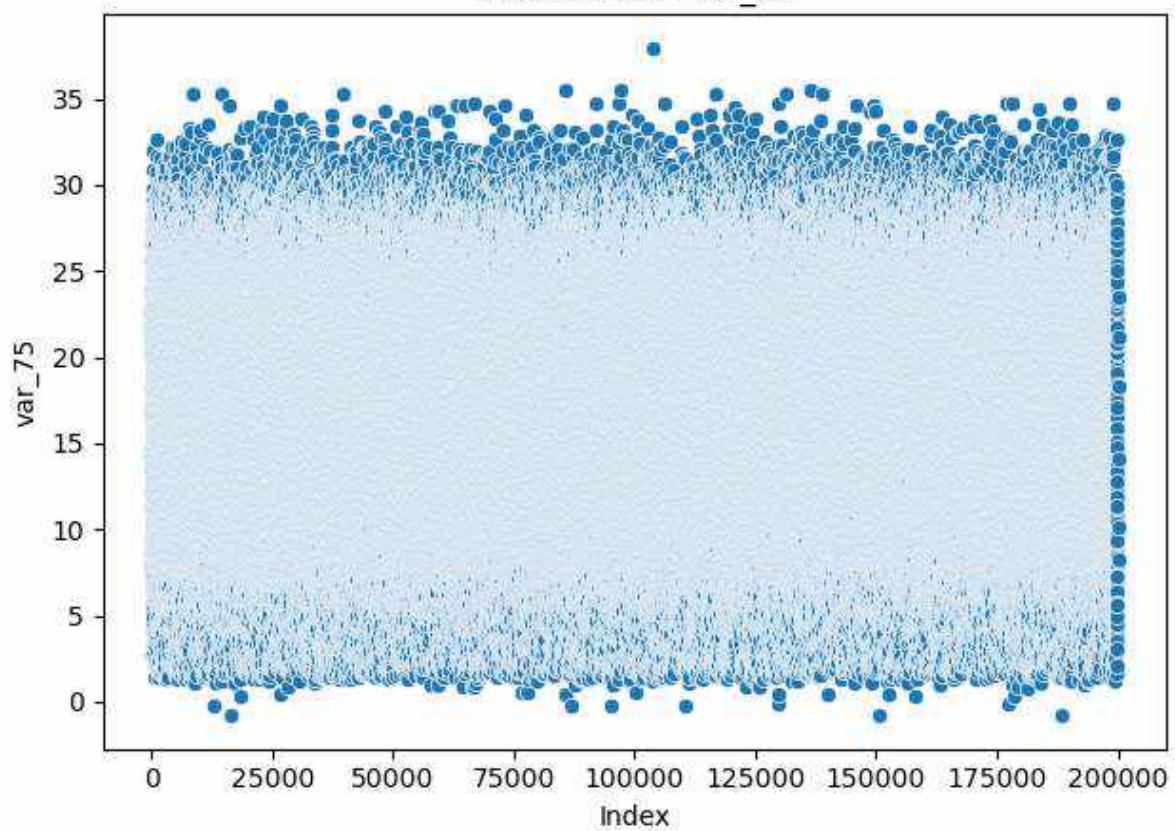




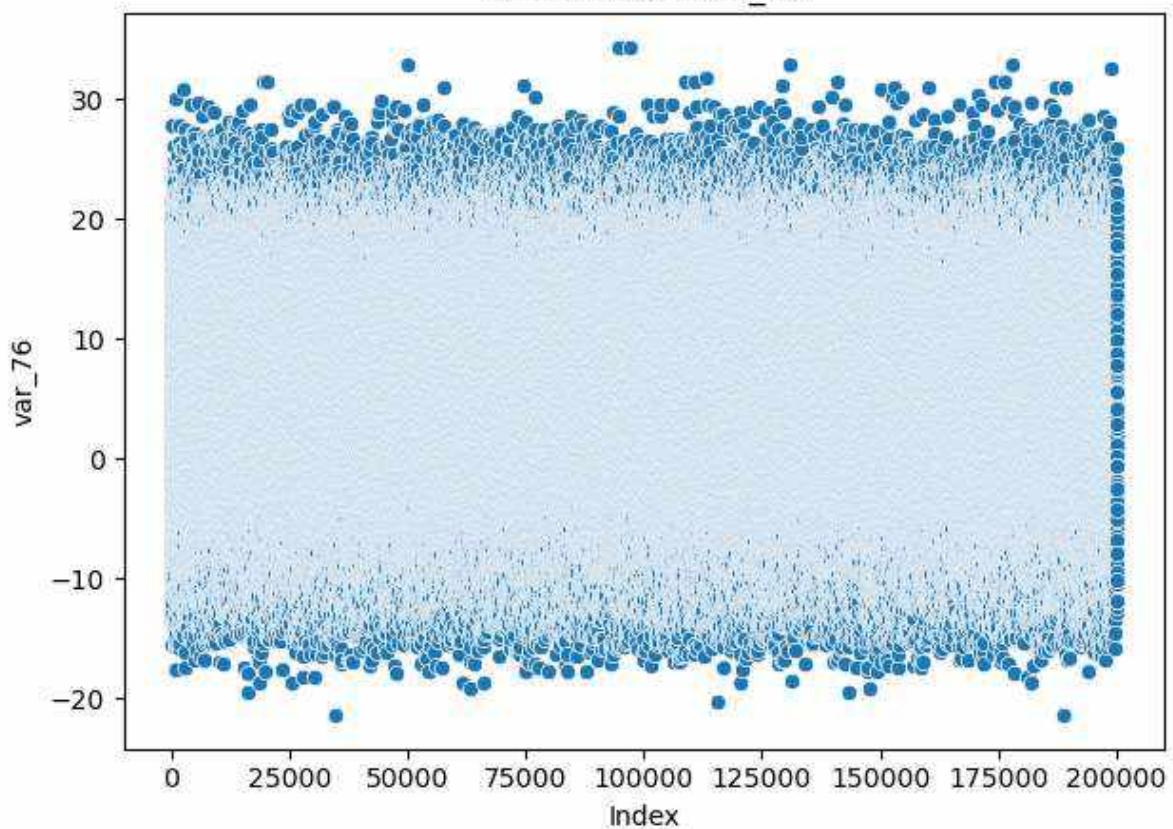
Scatter Plot - var\_74



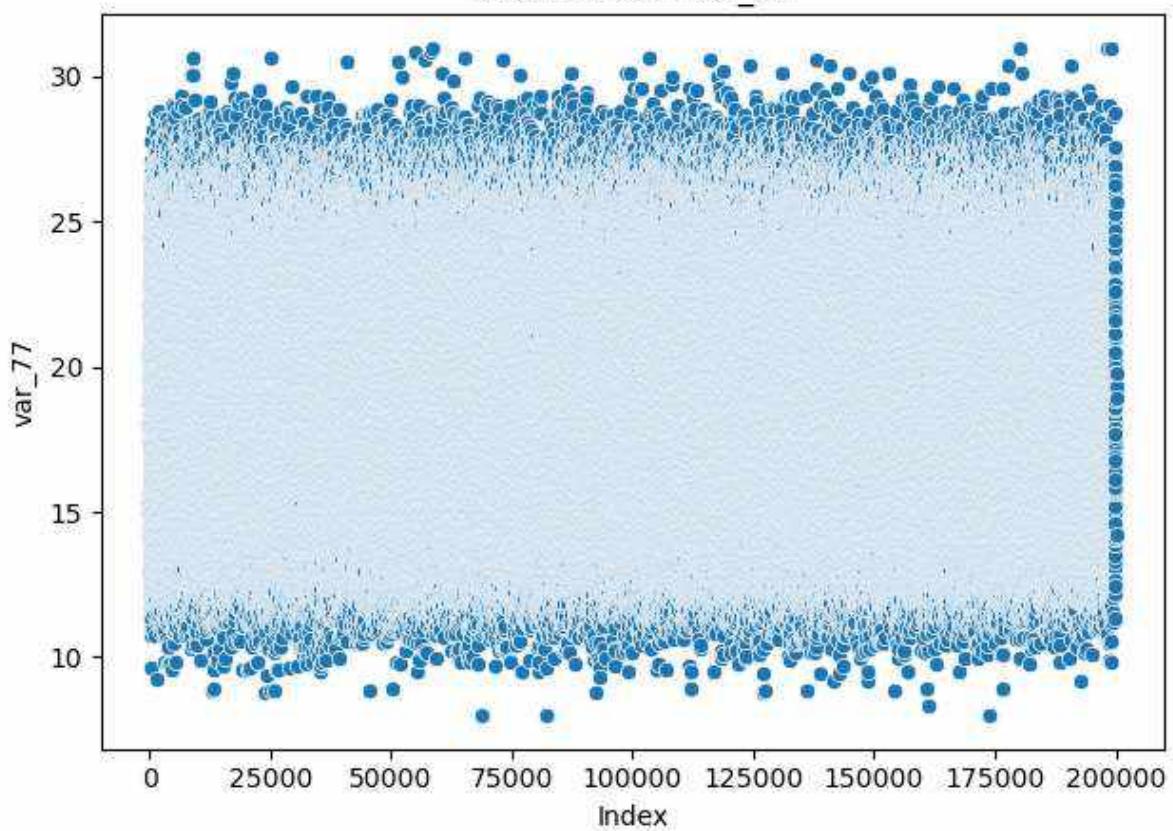
Scatter Plot - var\_75



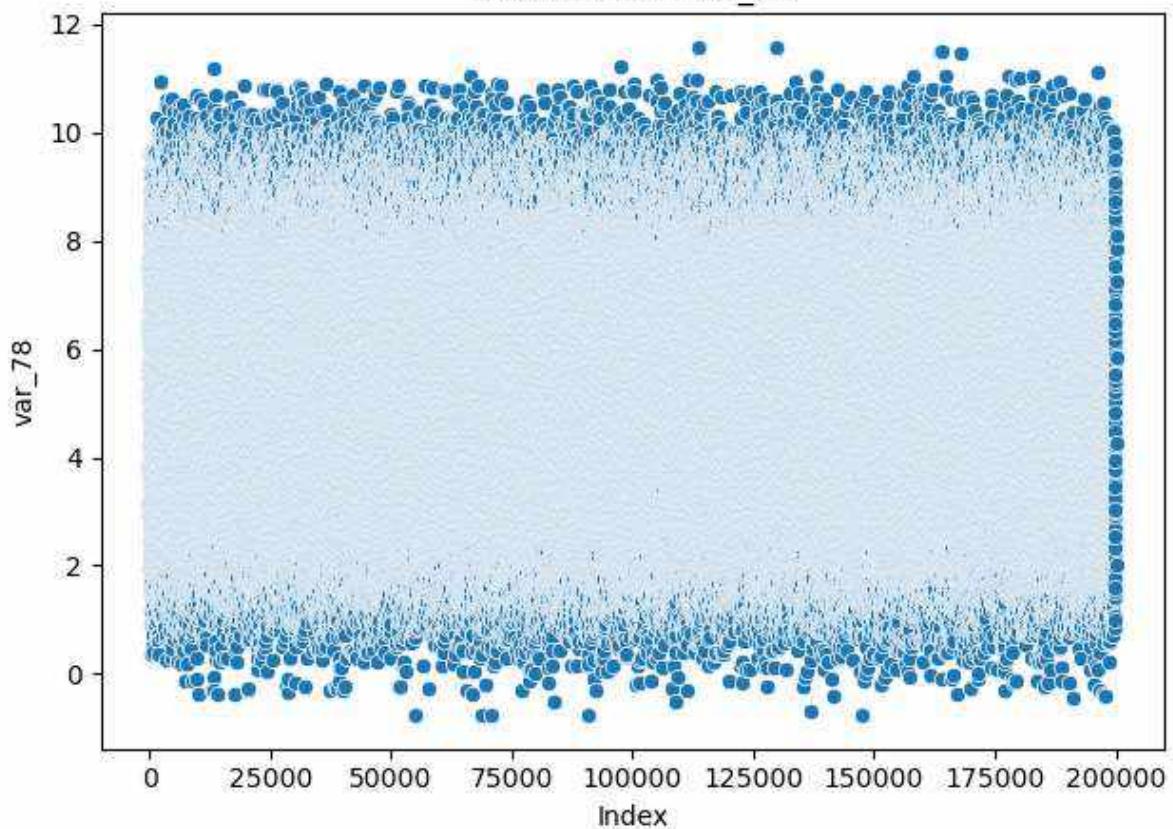
Scatter Plot - var\_76



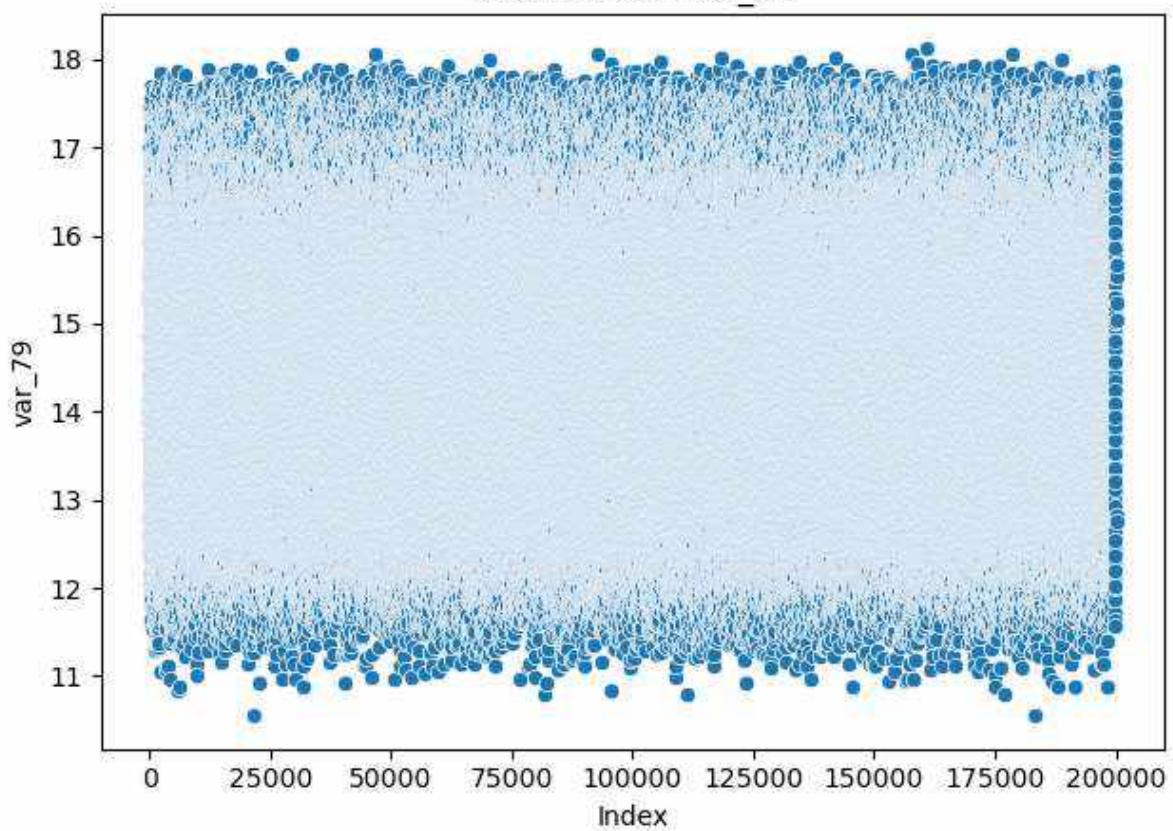
Scatter Plot - var\_77

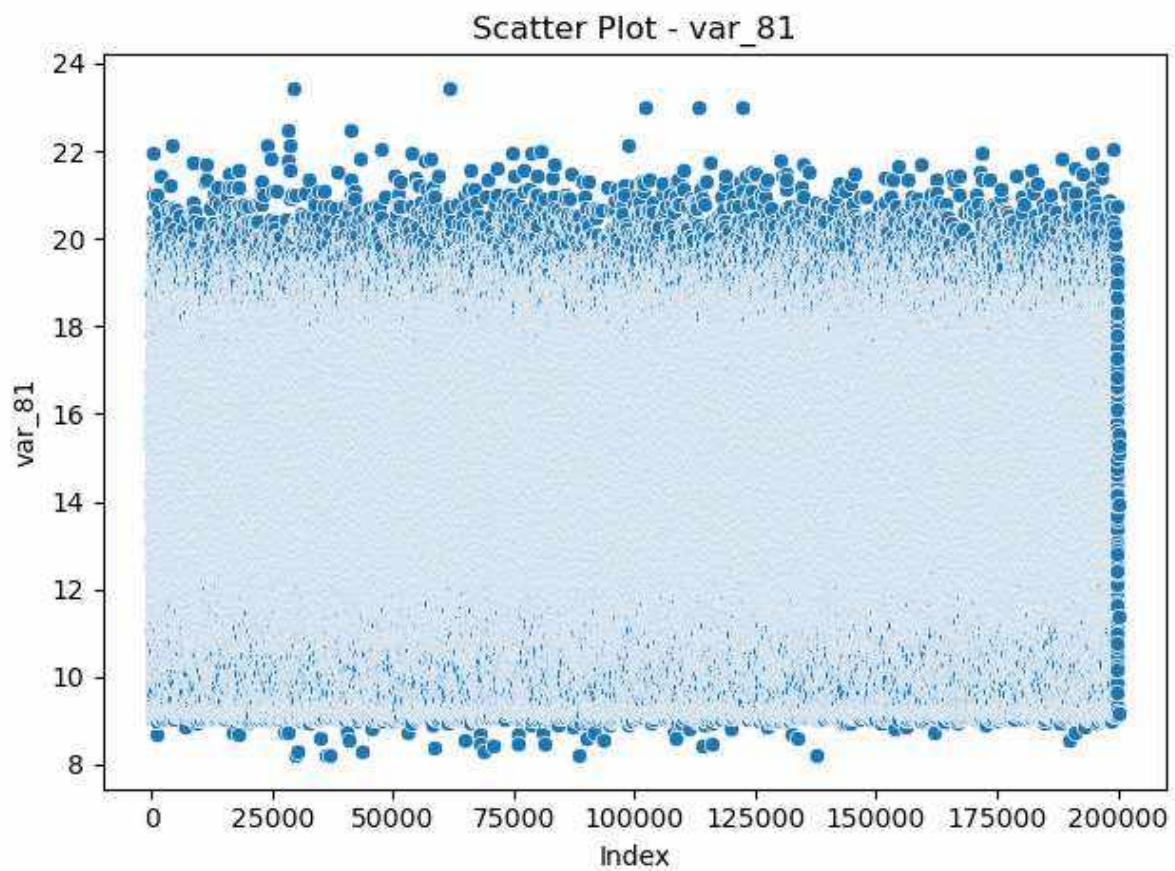
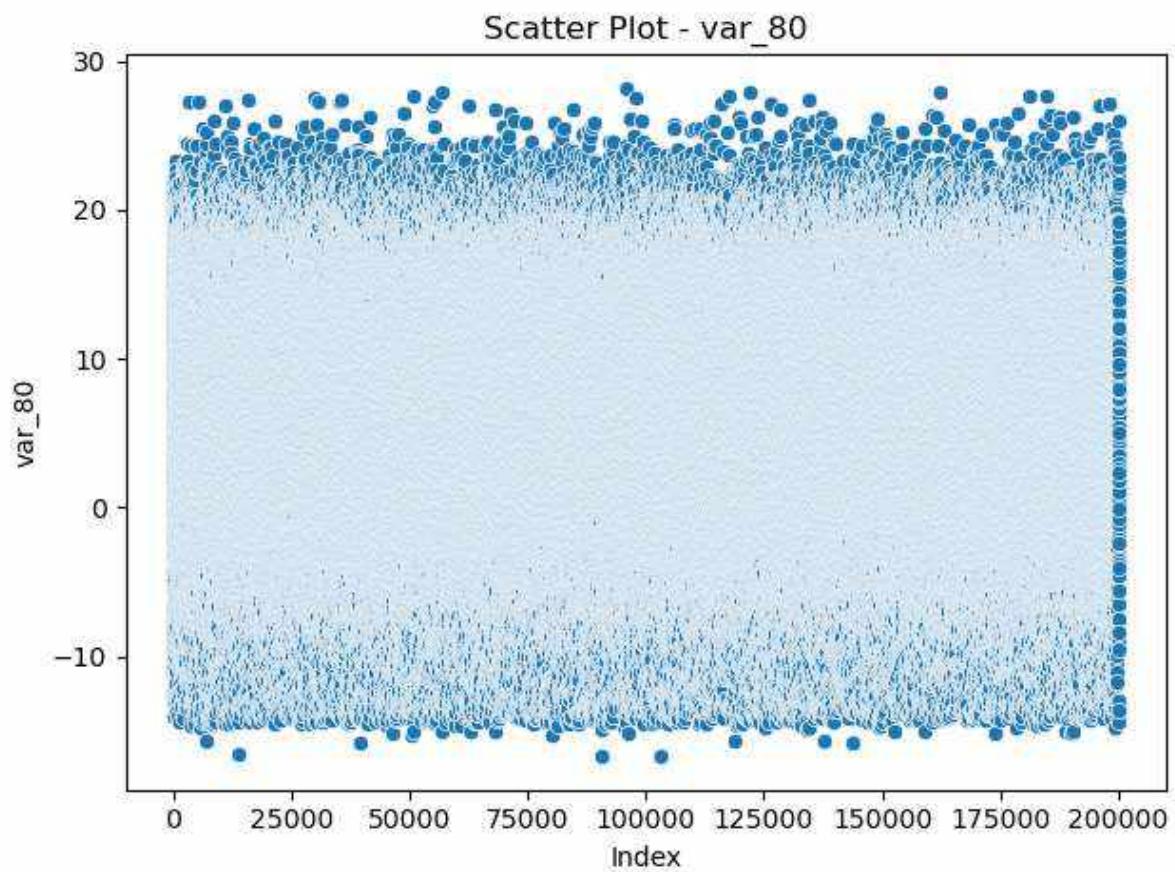


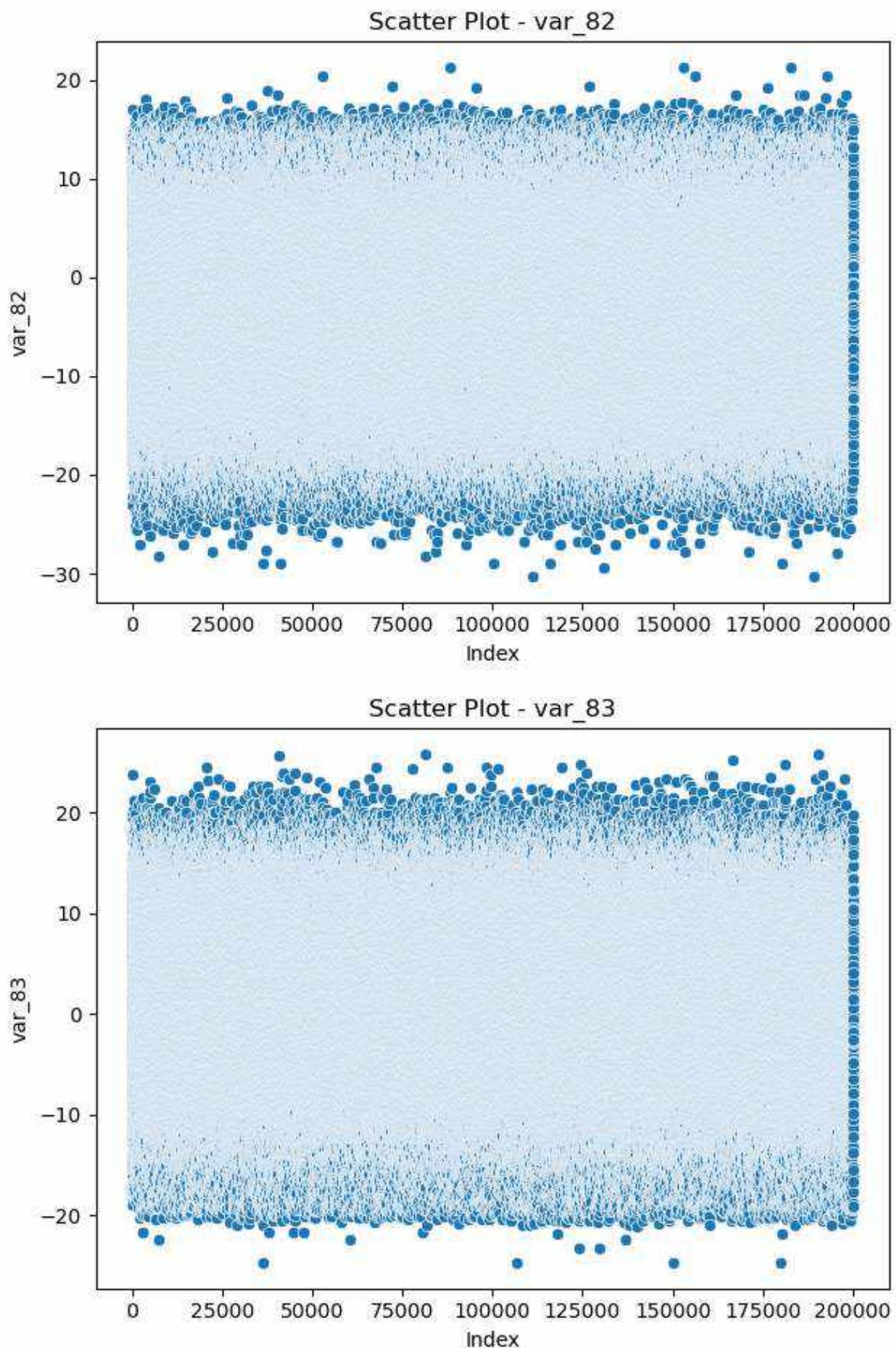
Scatter Plot - var\_78



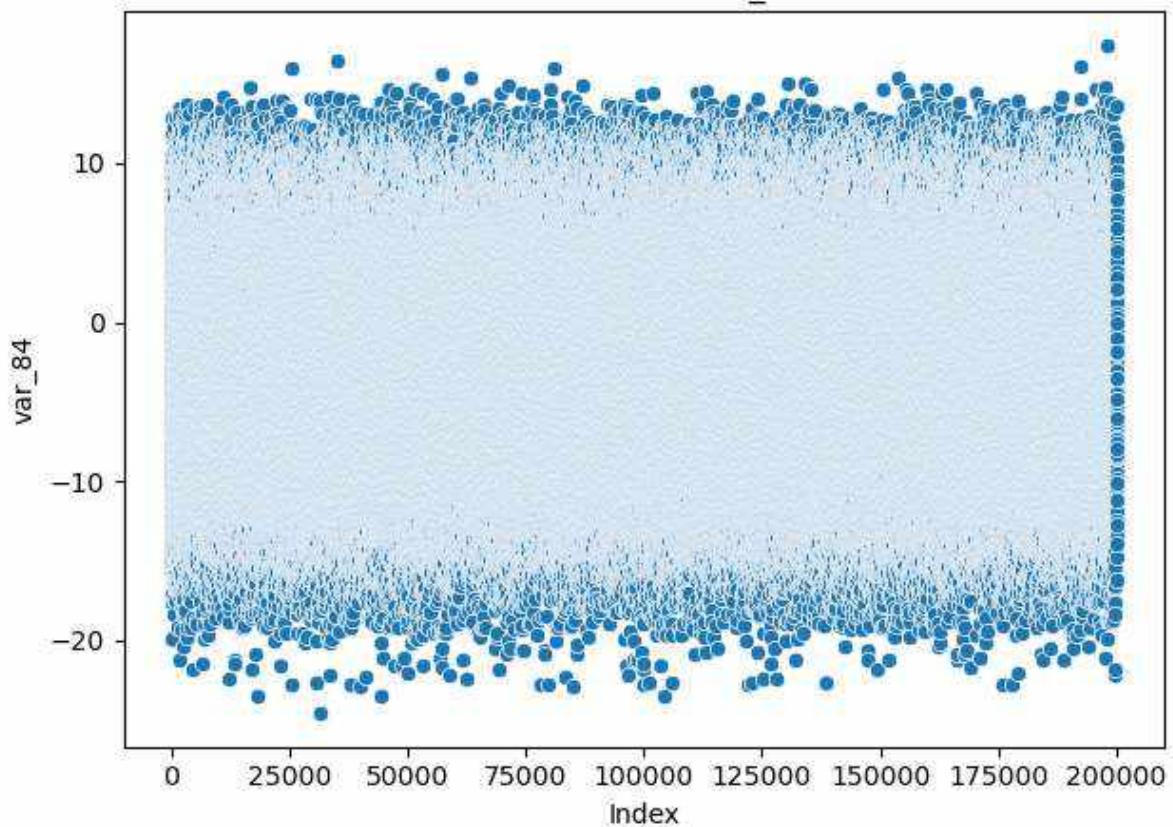
Scatter Plot - var\_79



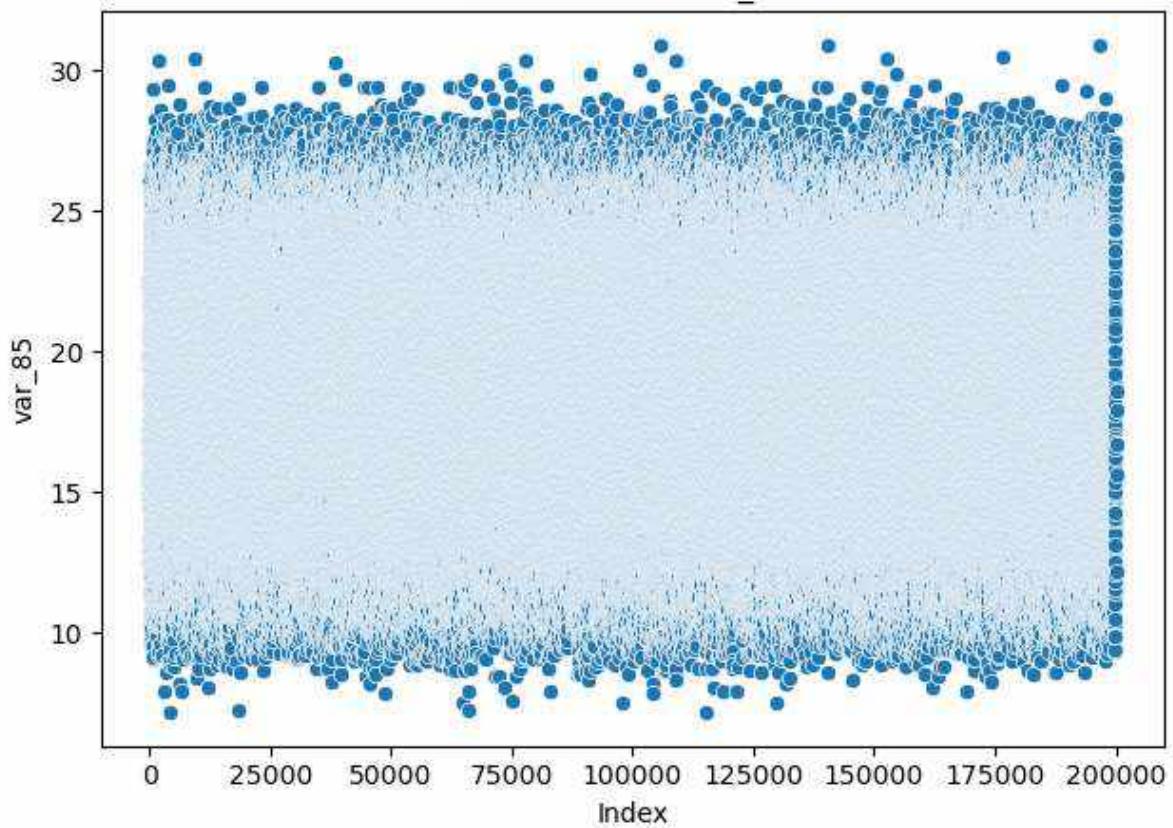


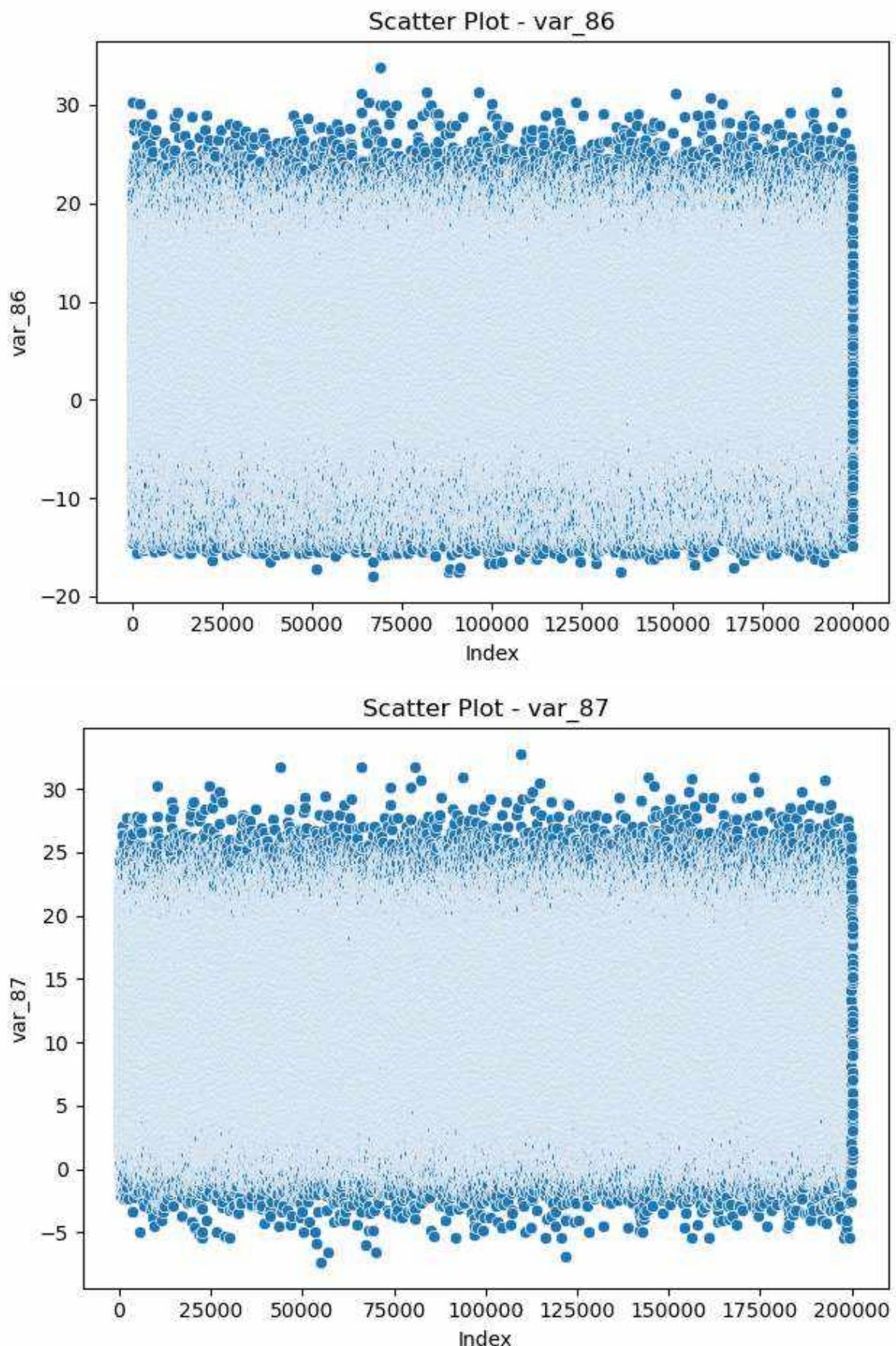


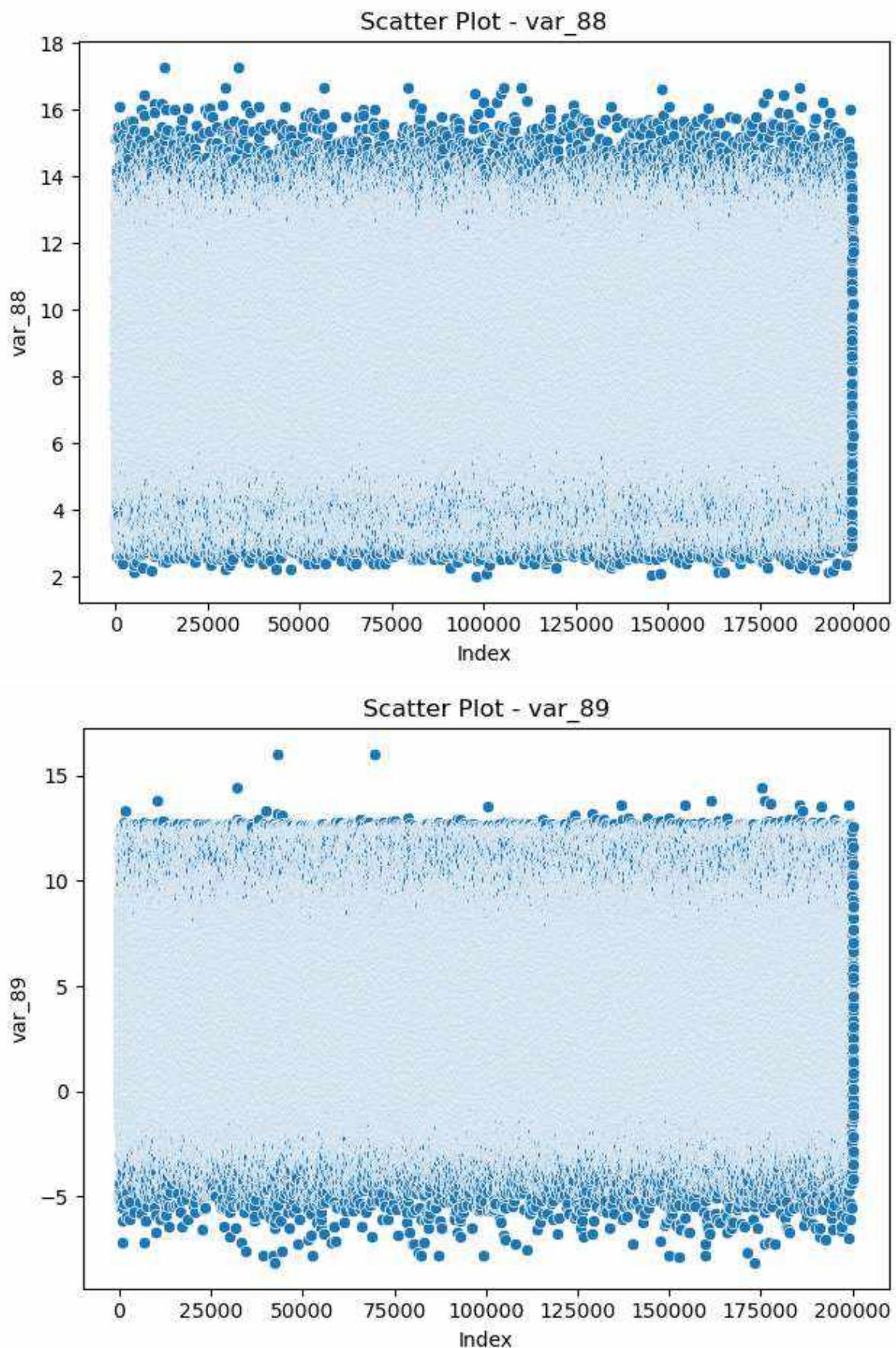
Scatter Plot - var\_84



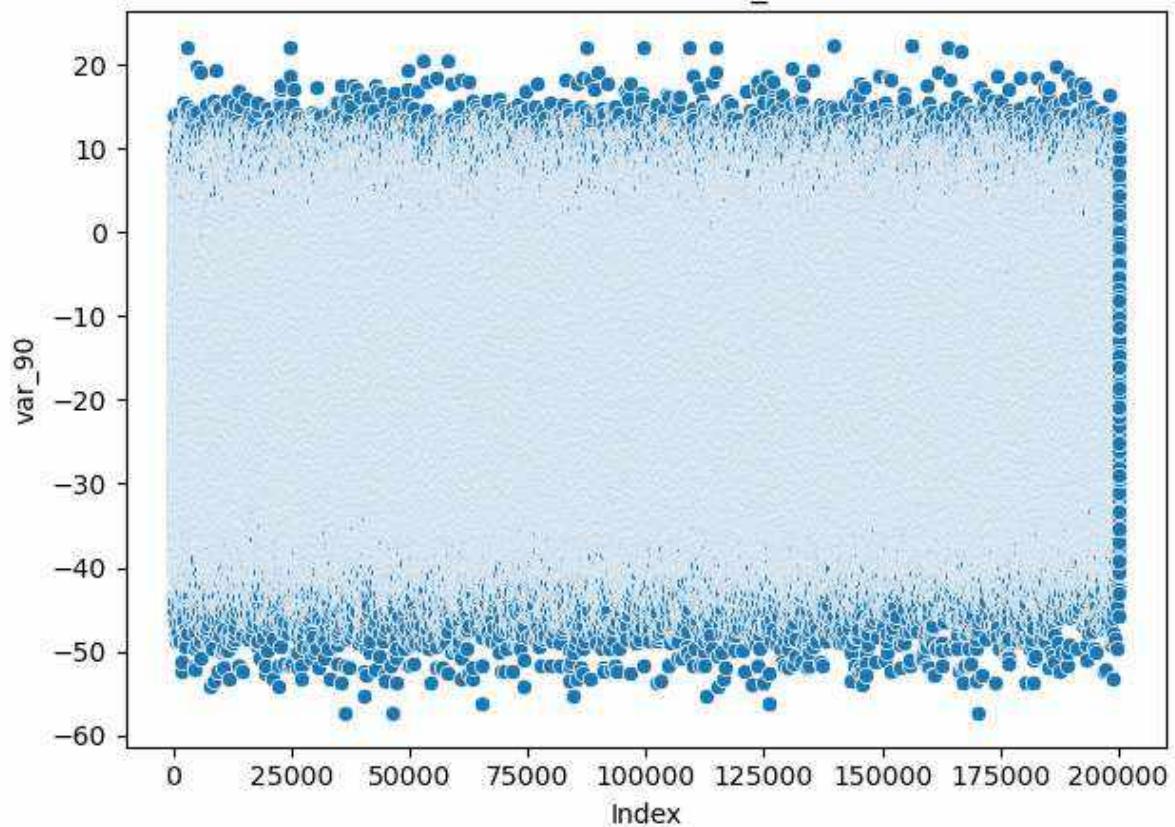
Scatter Plot - var\_85



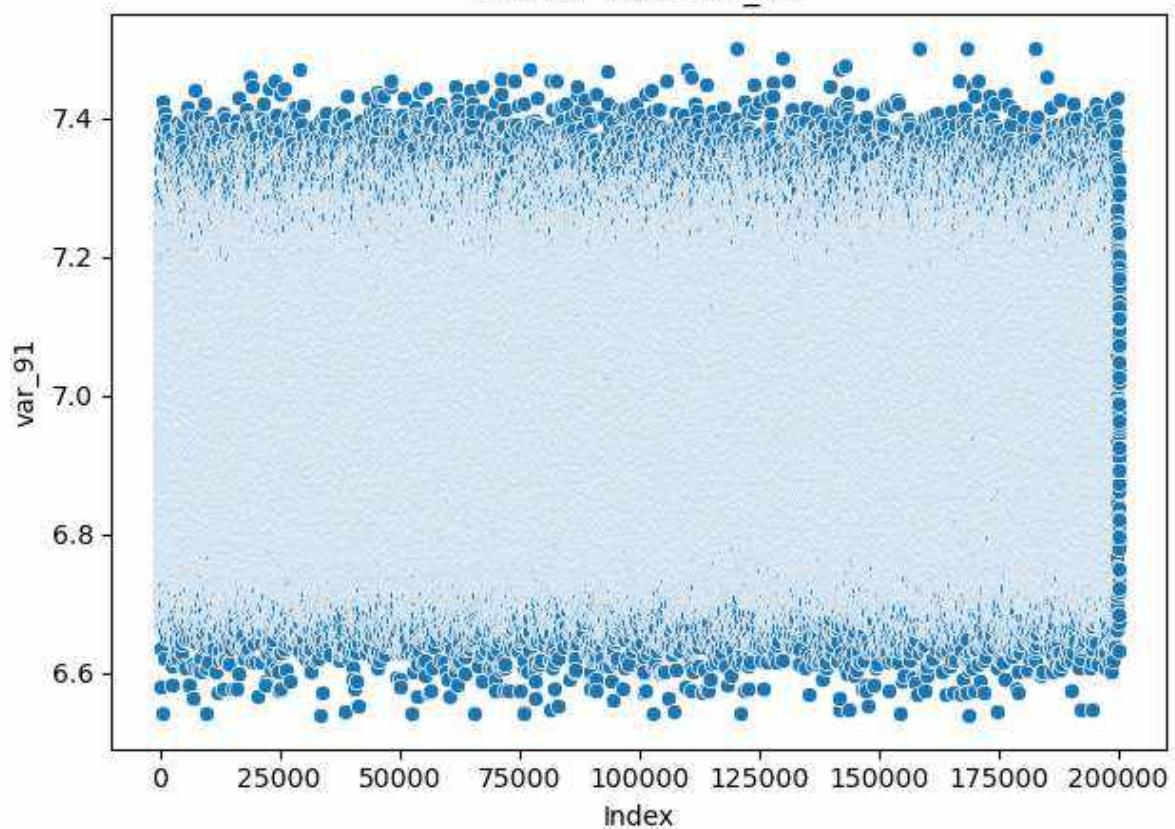




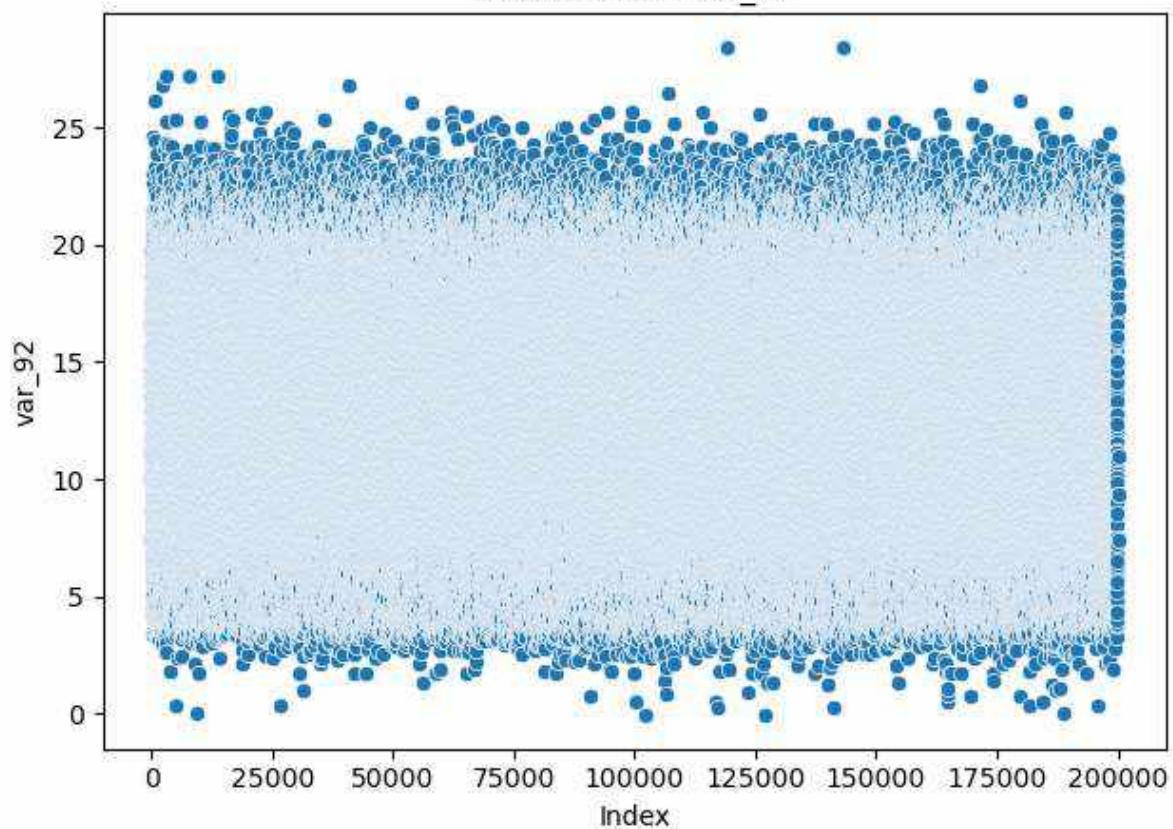
Scatter Plot - var\_90



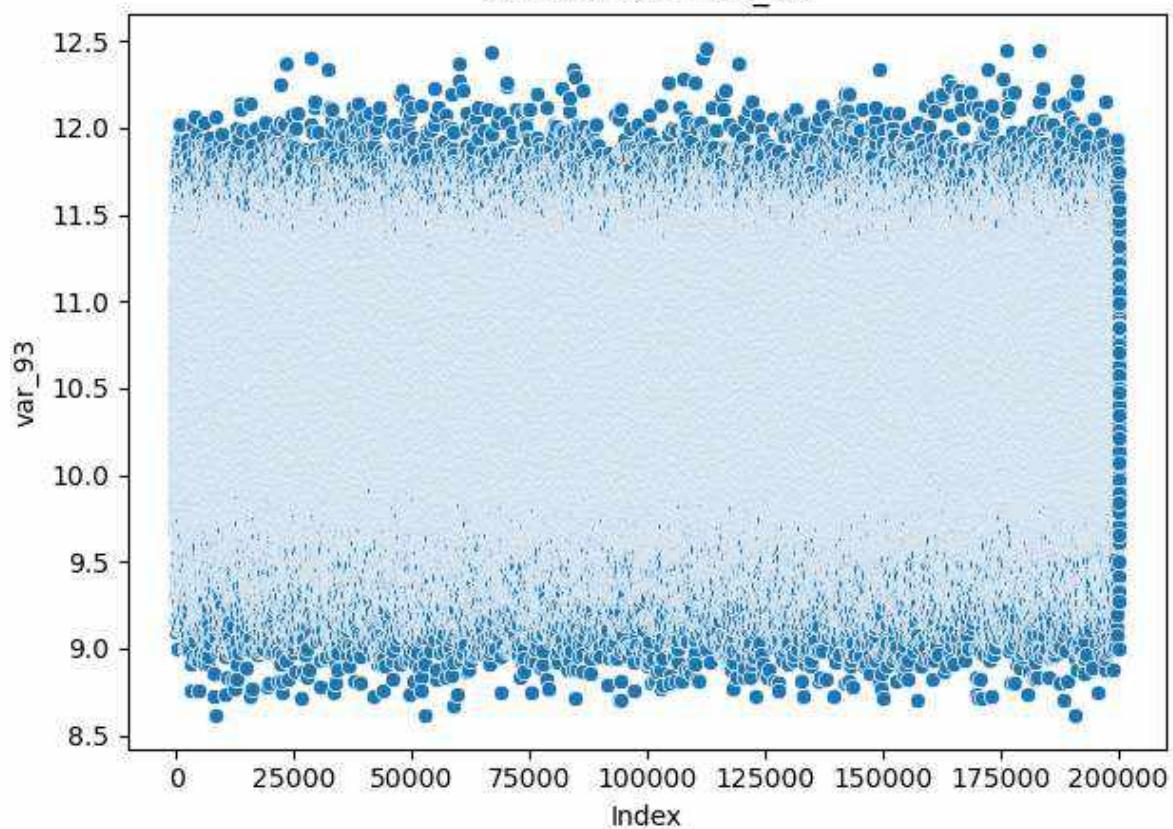
Scatter Plot - var\_91



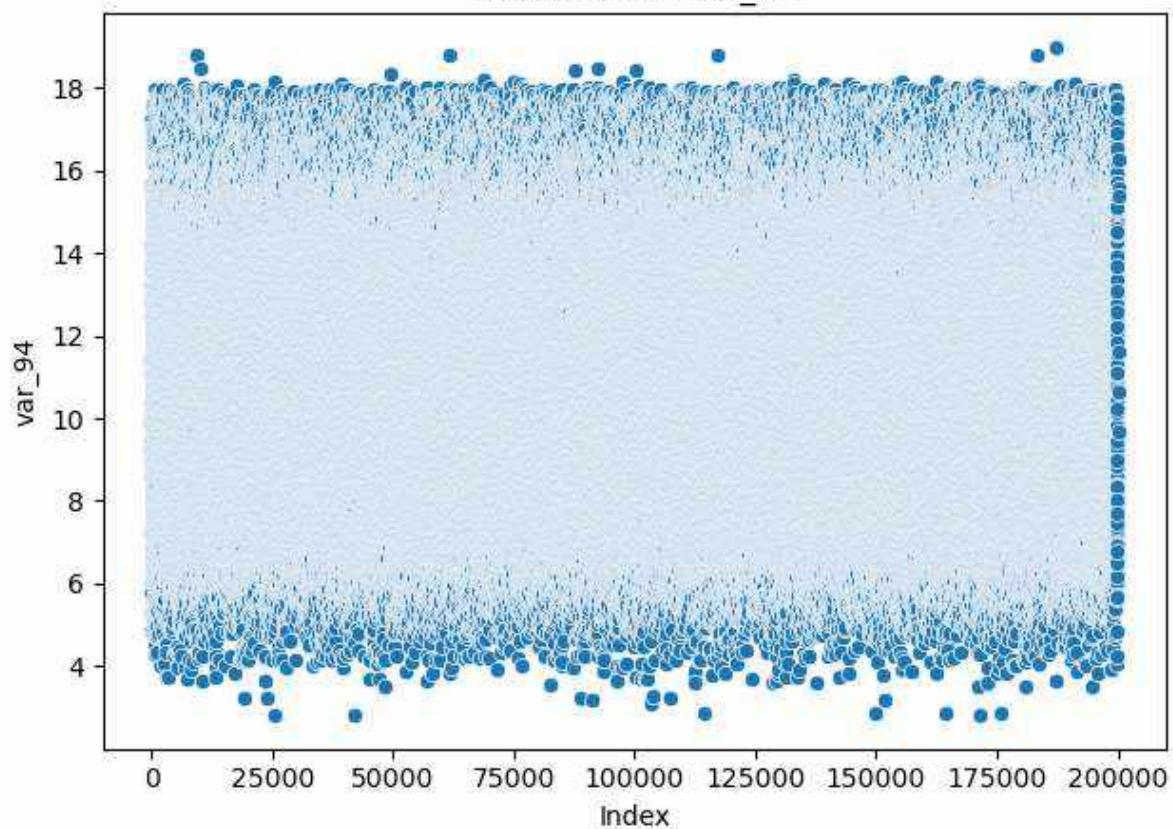
Scatter Plot - var\_92



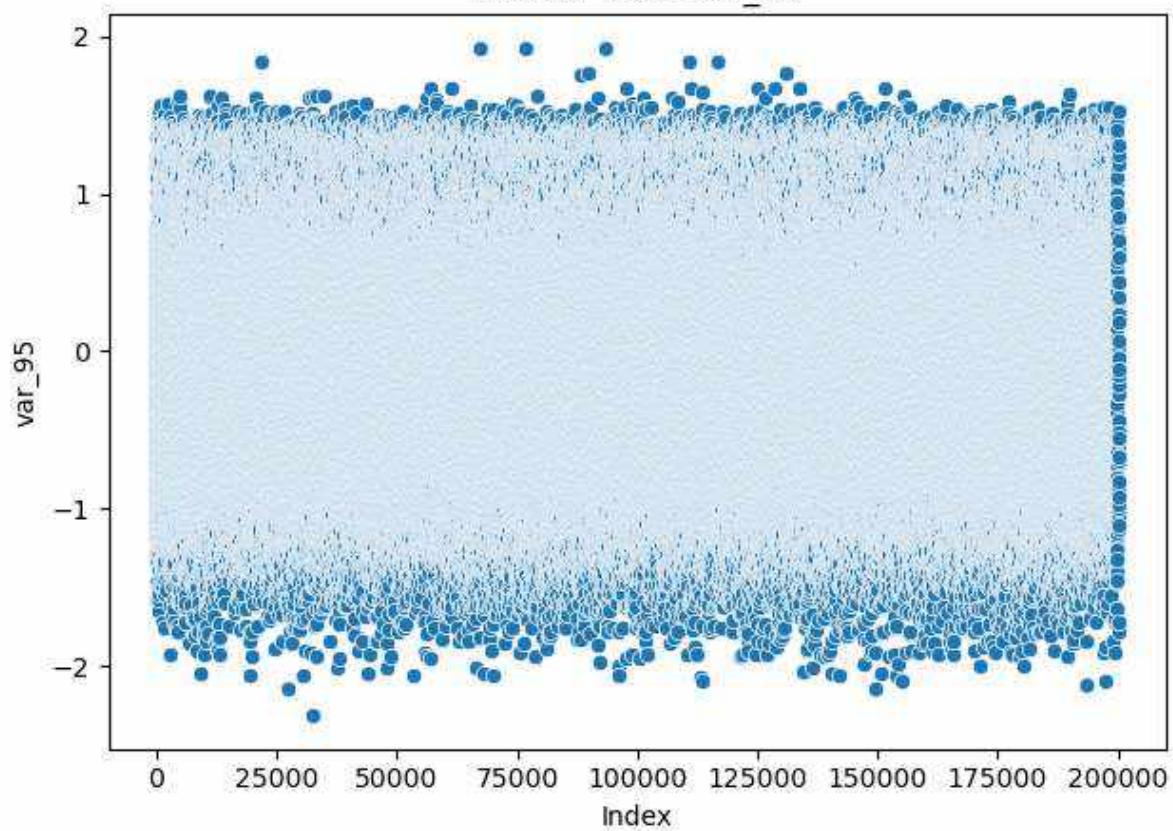
Scatter Plot - var\_93

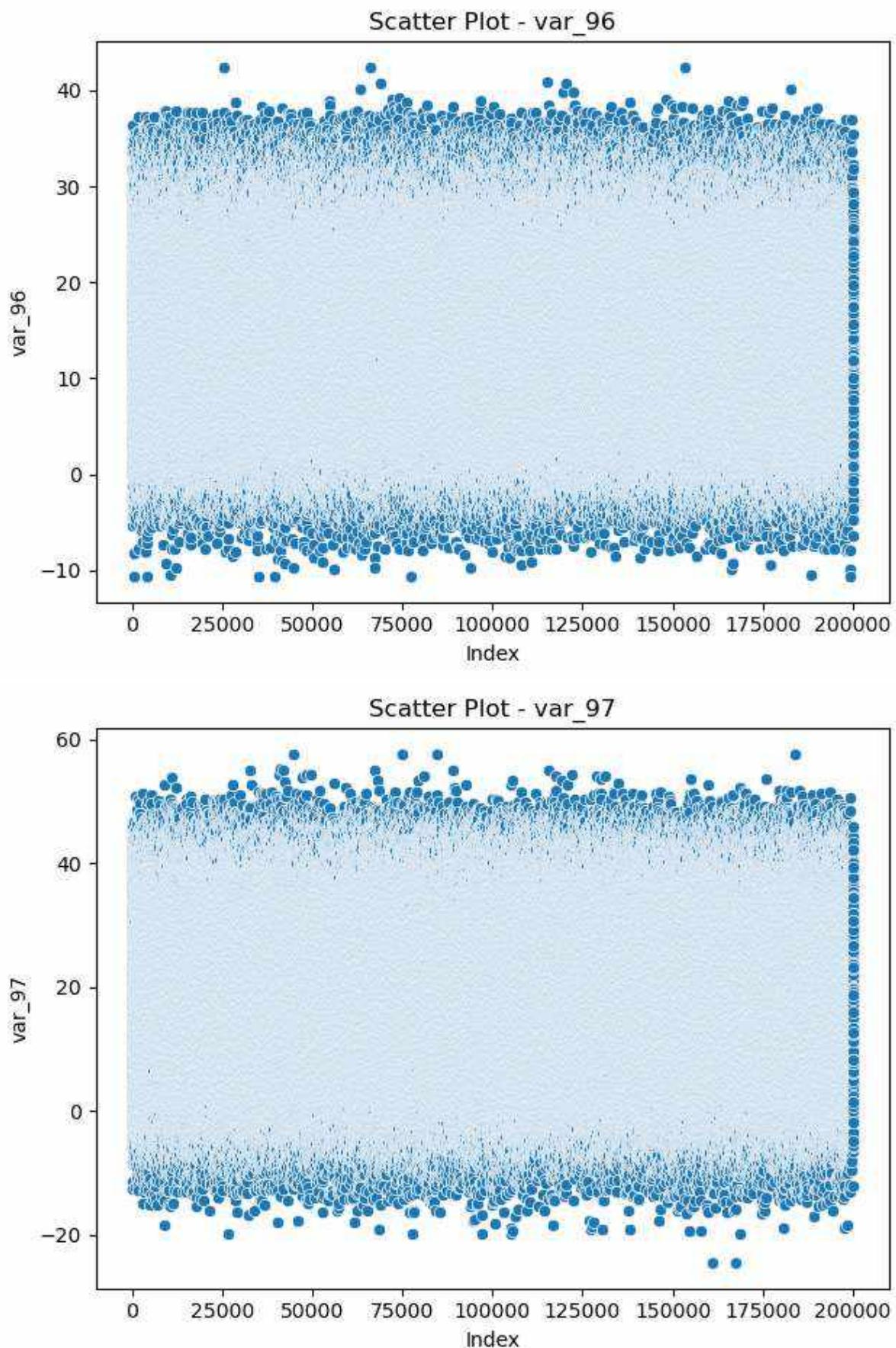


Scatter Plot - var\_94

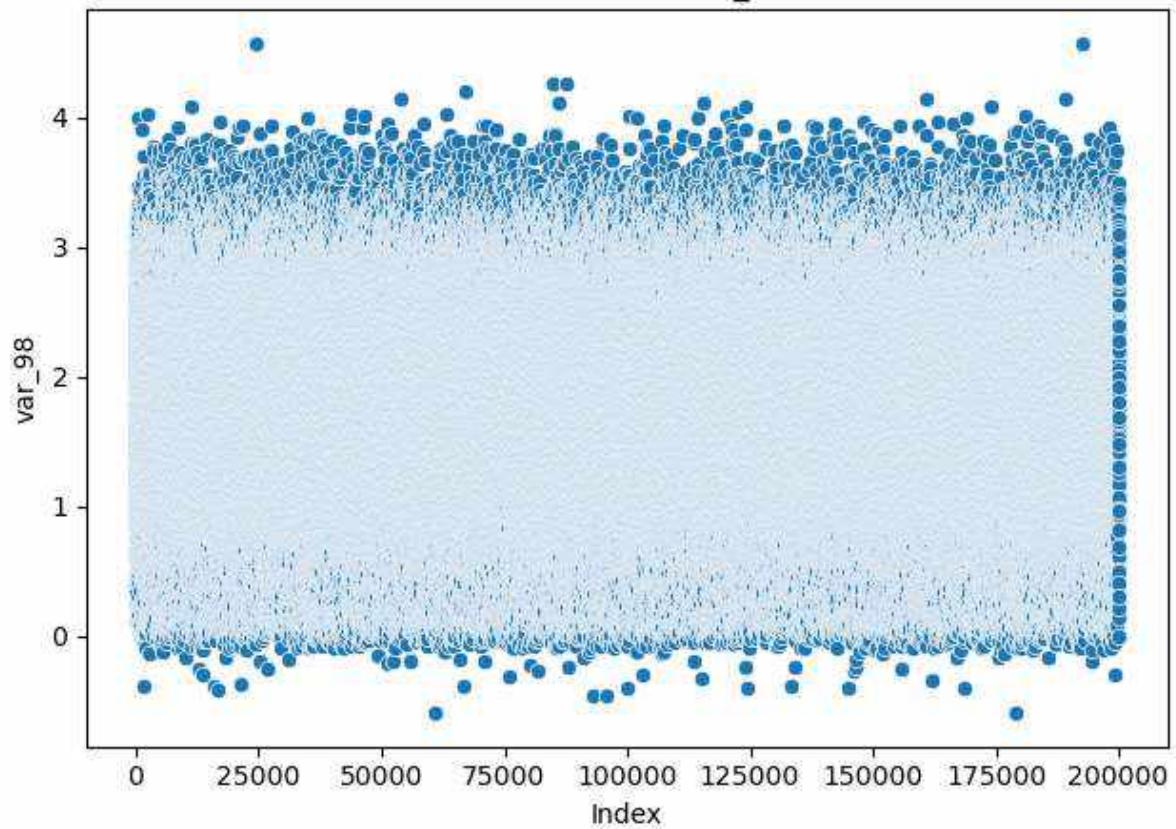


Scatter Plot - var\_95

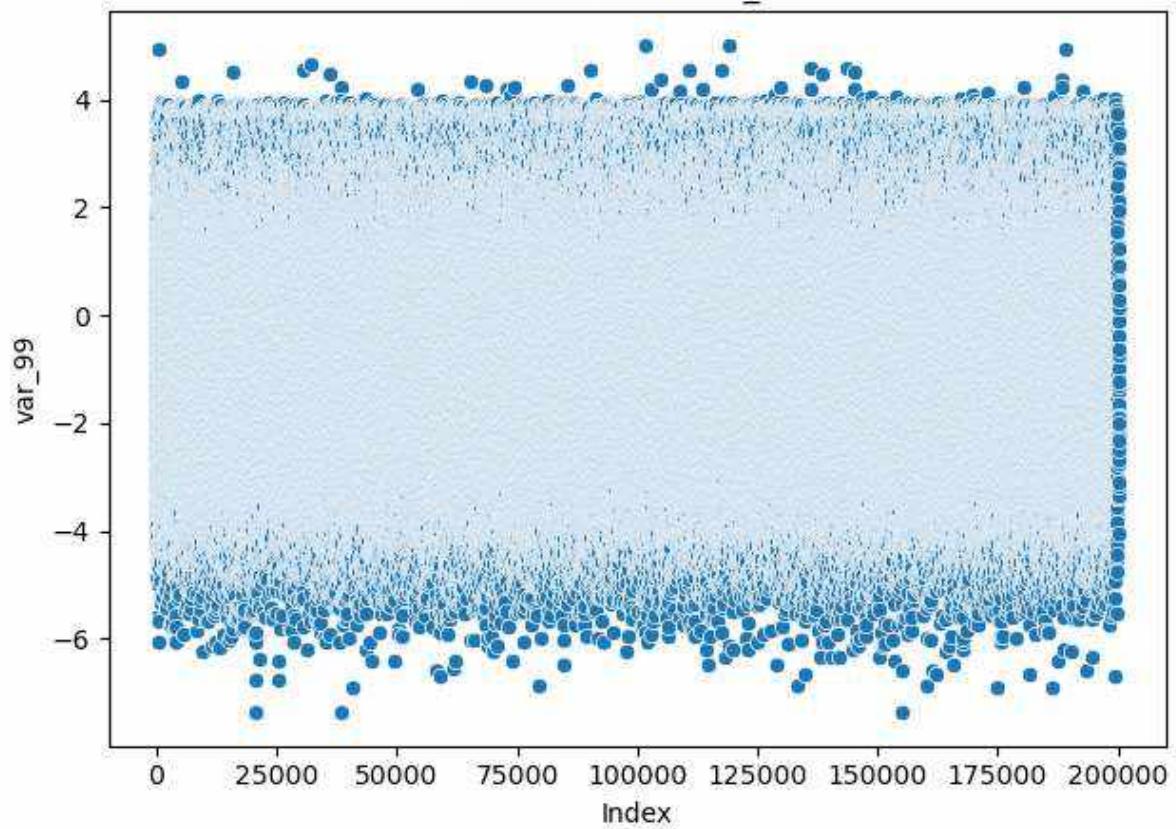




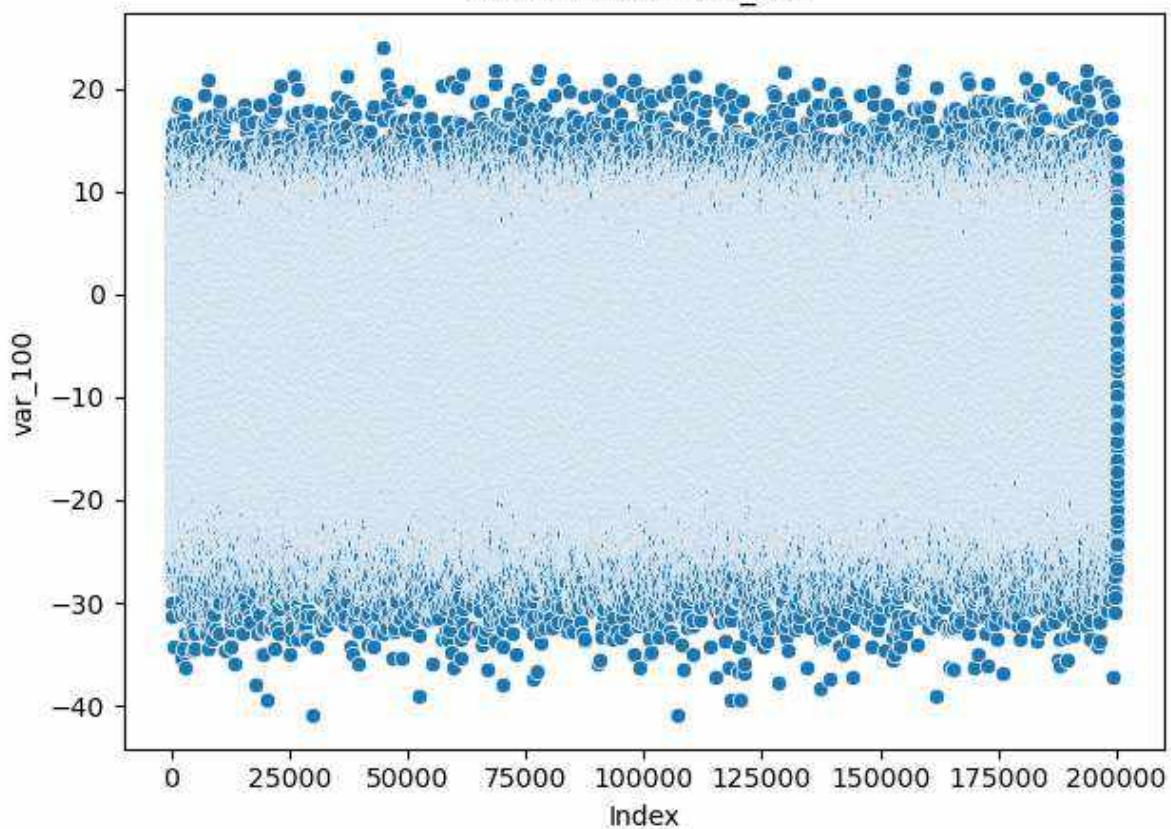
Scatter Plot - var\_98



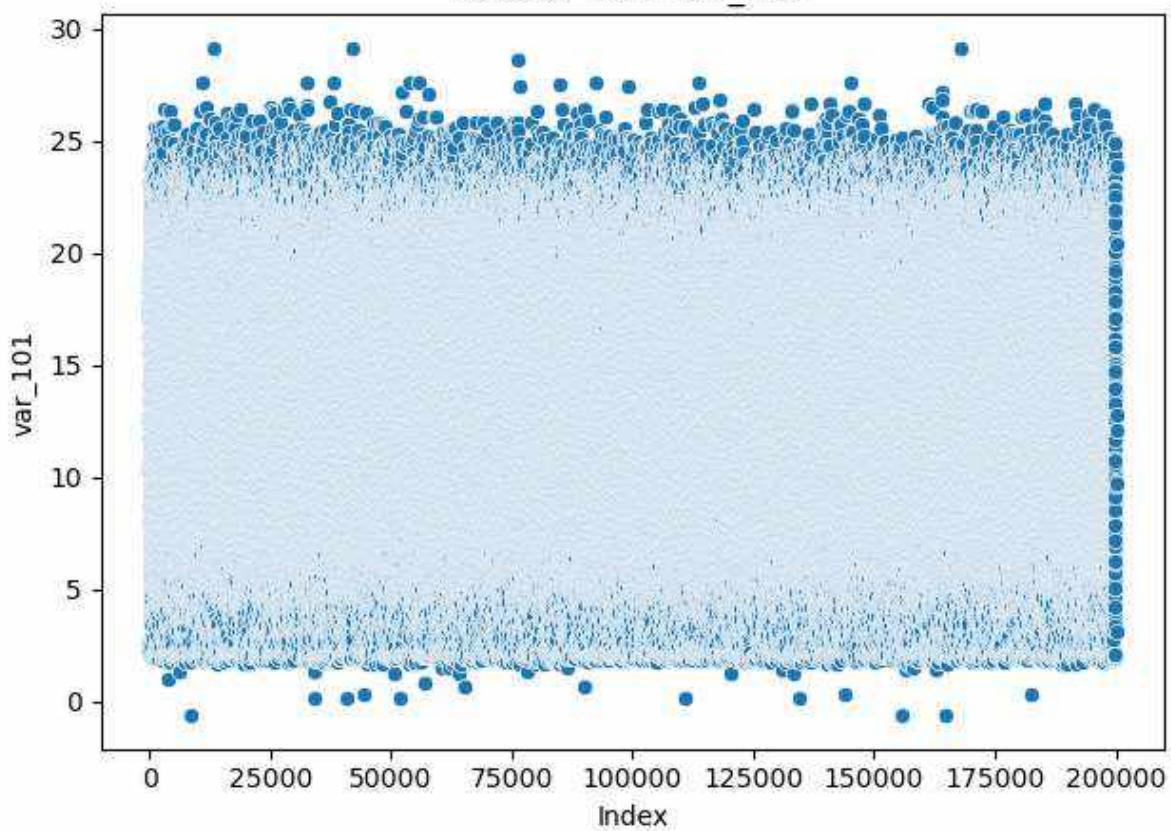
Scatter Plot - var\_99



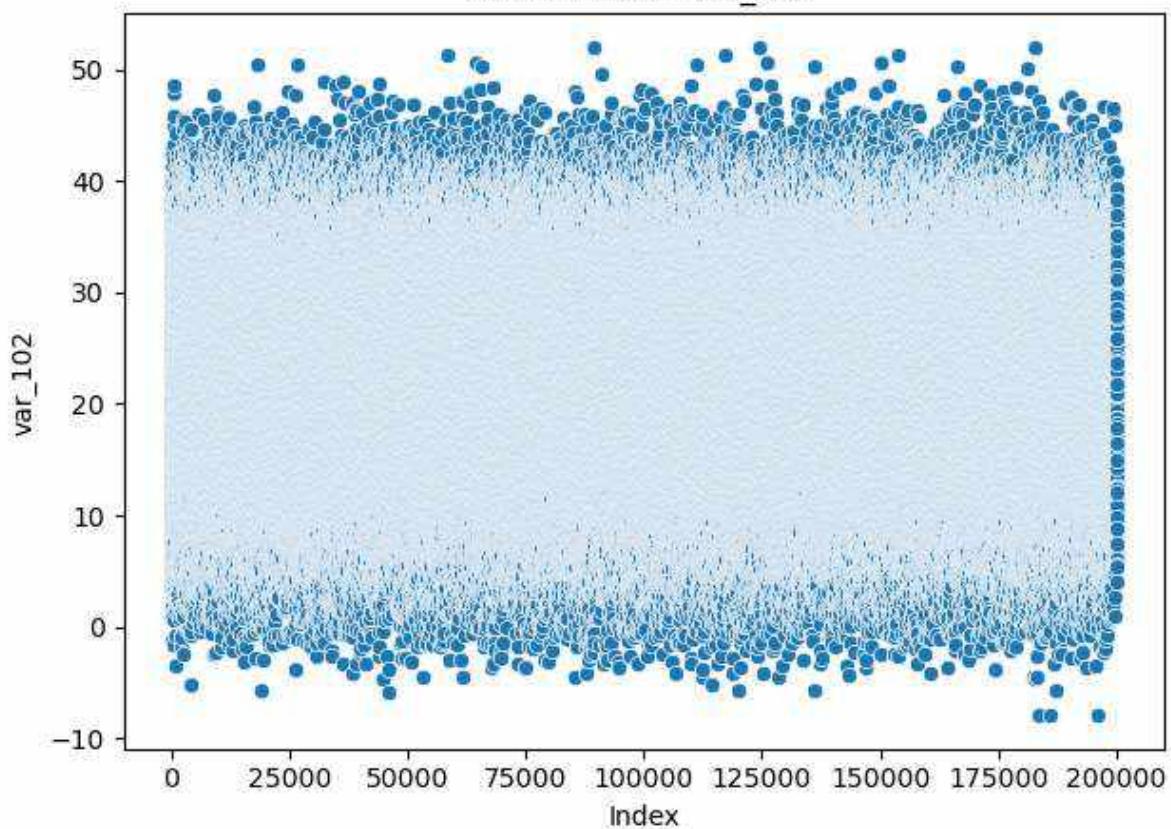
Scatter Plot - var\_100



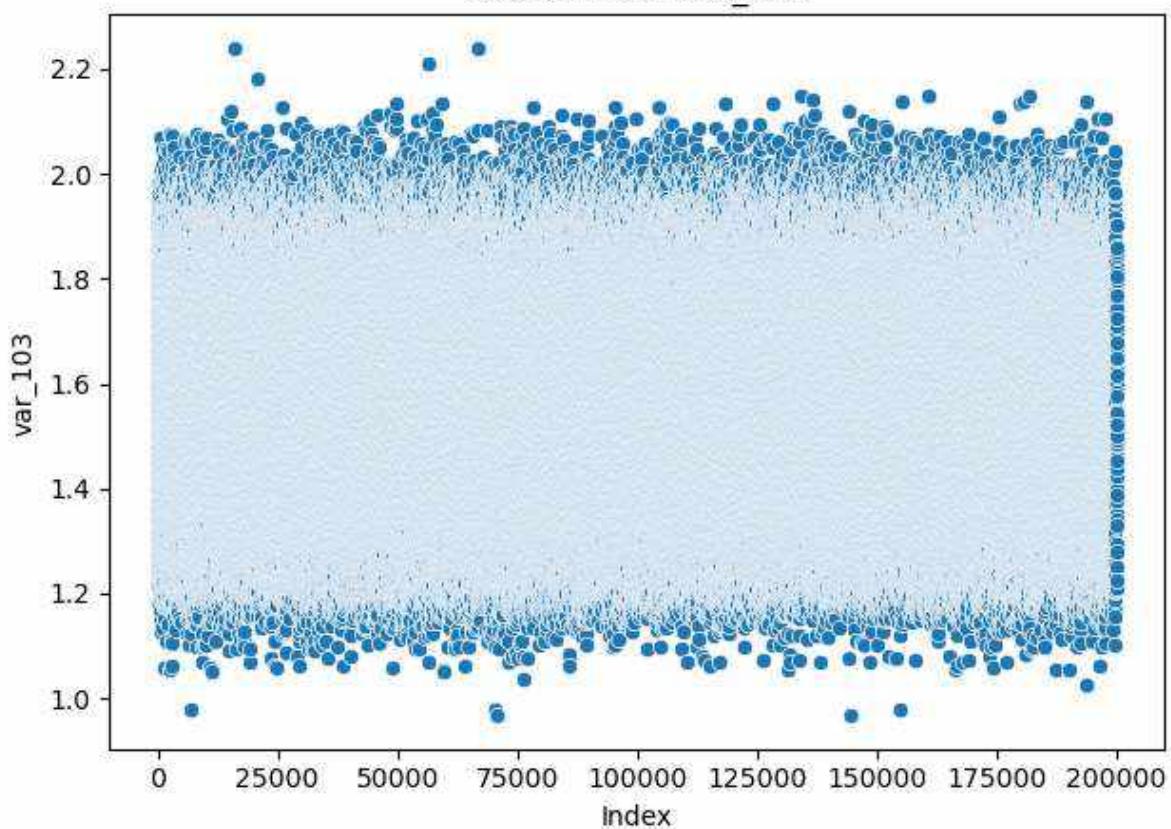
Scatter Plot - var\_101



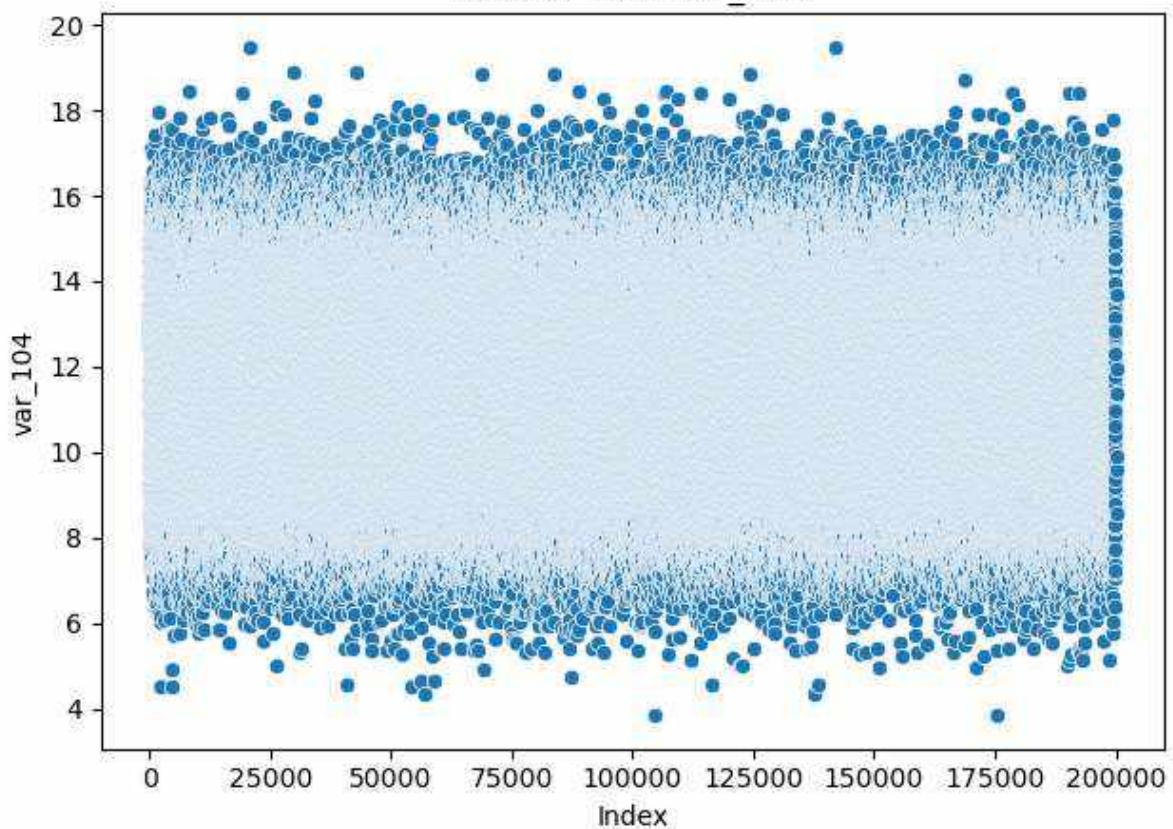
Scatter Plot - var\_102



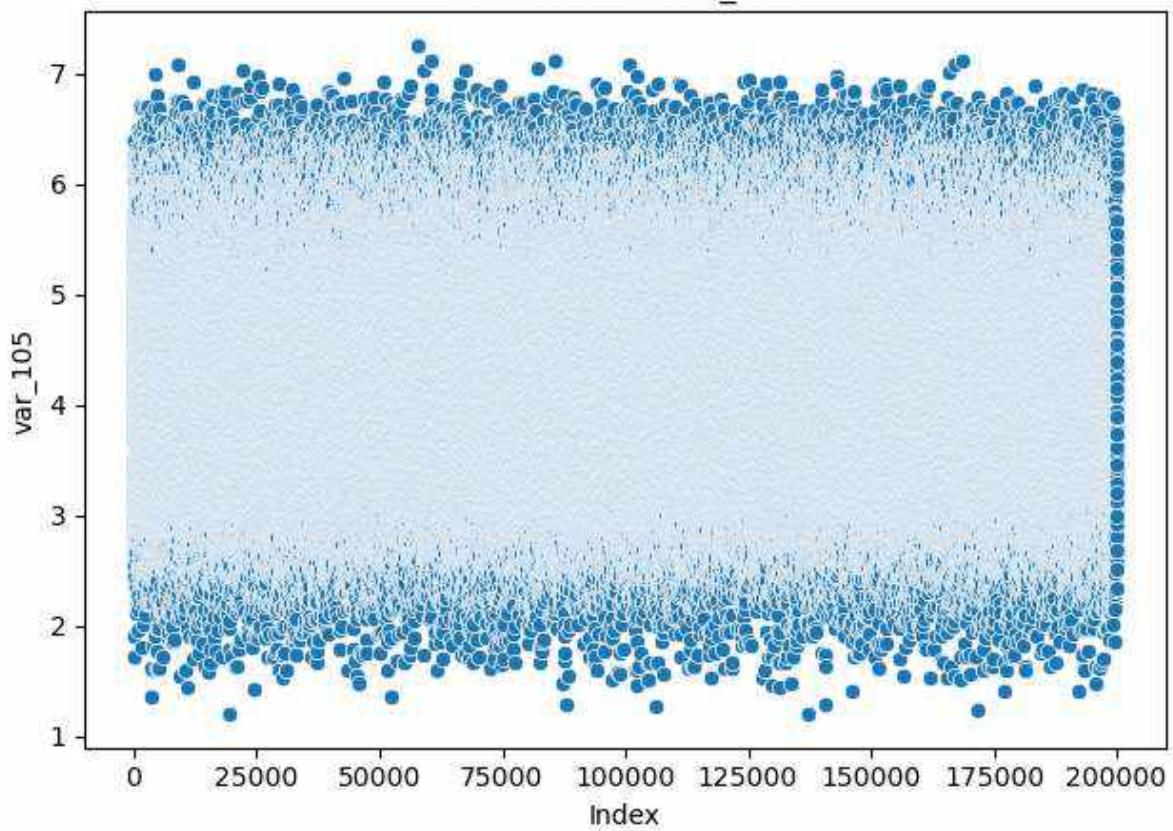
Scatter Plot - var\_103



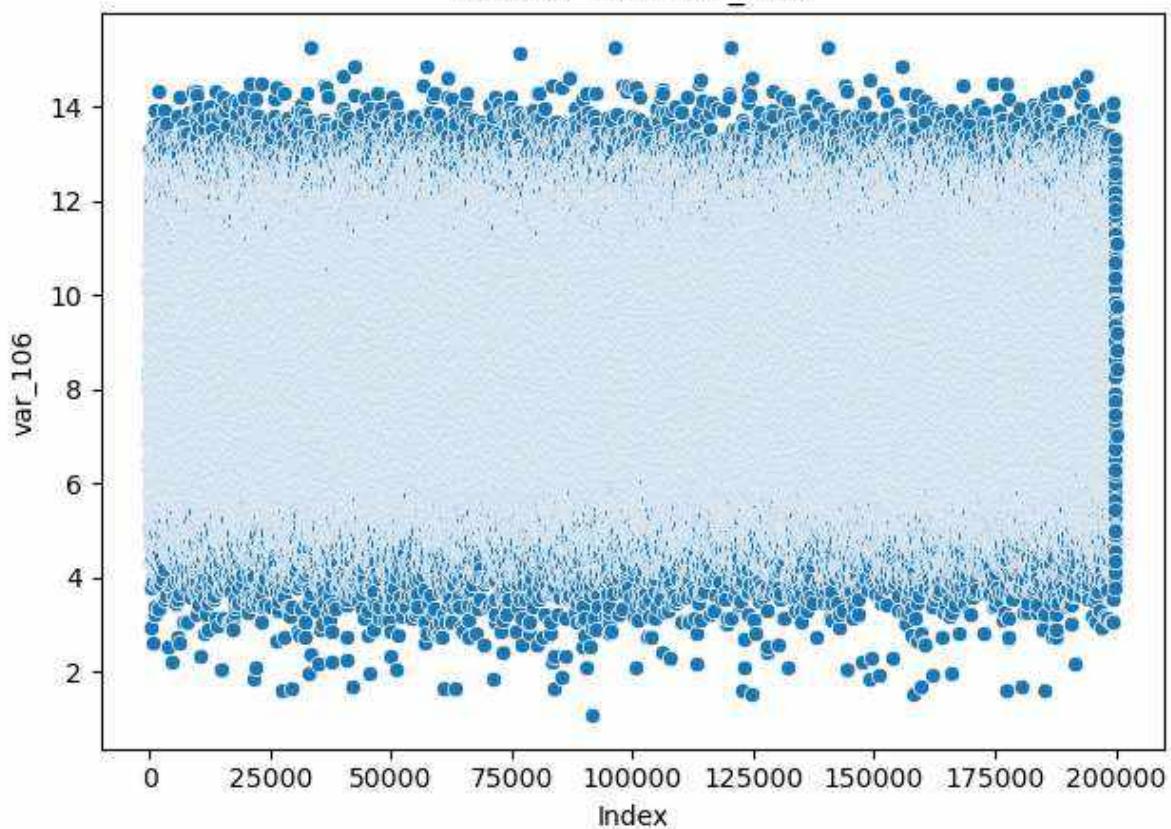
Scatter Plot - var\_104



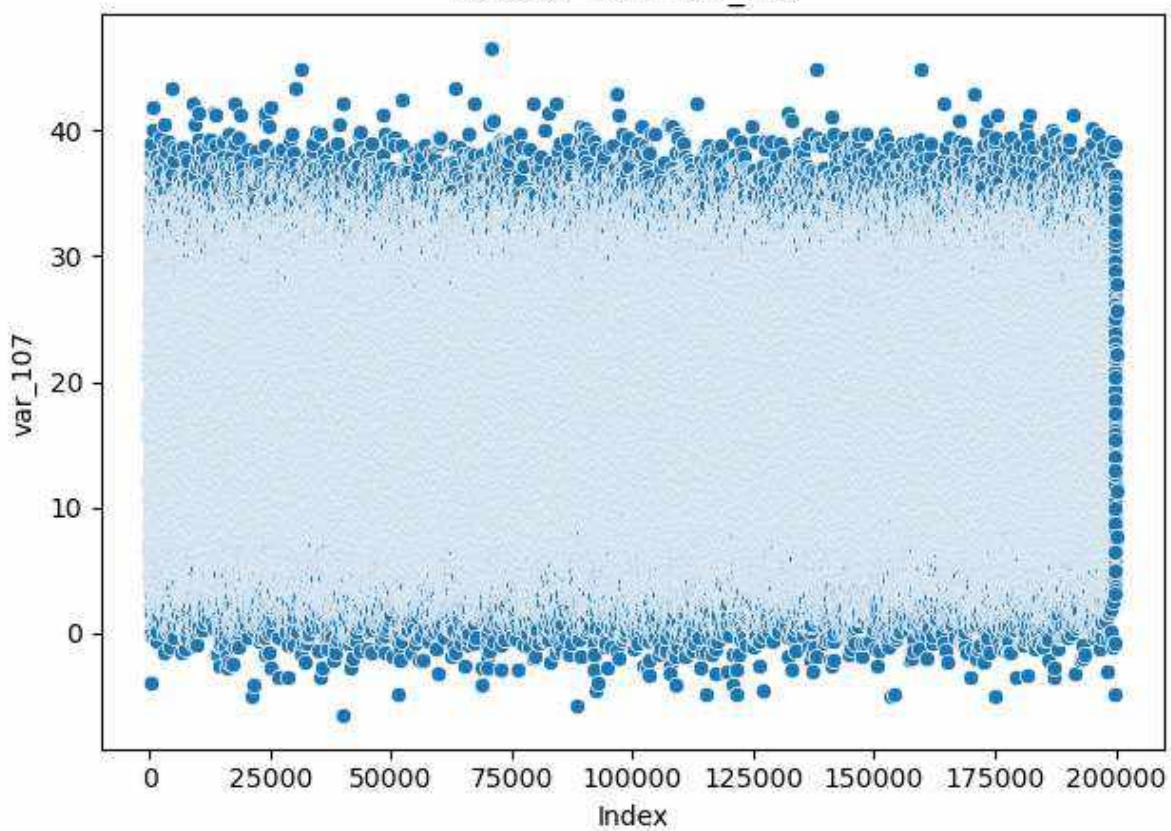
Scatter Plot - var\_105



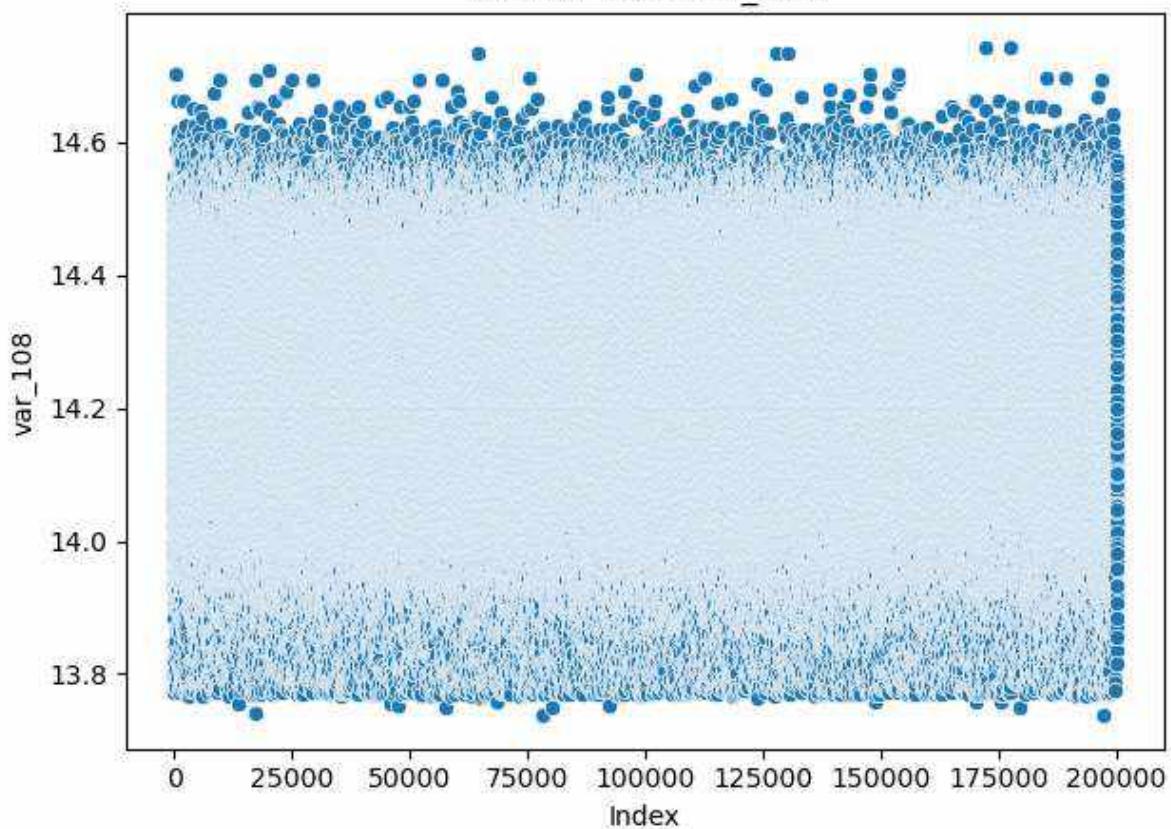
Scatter Plot - var\_106



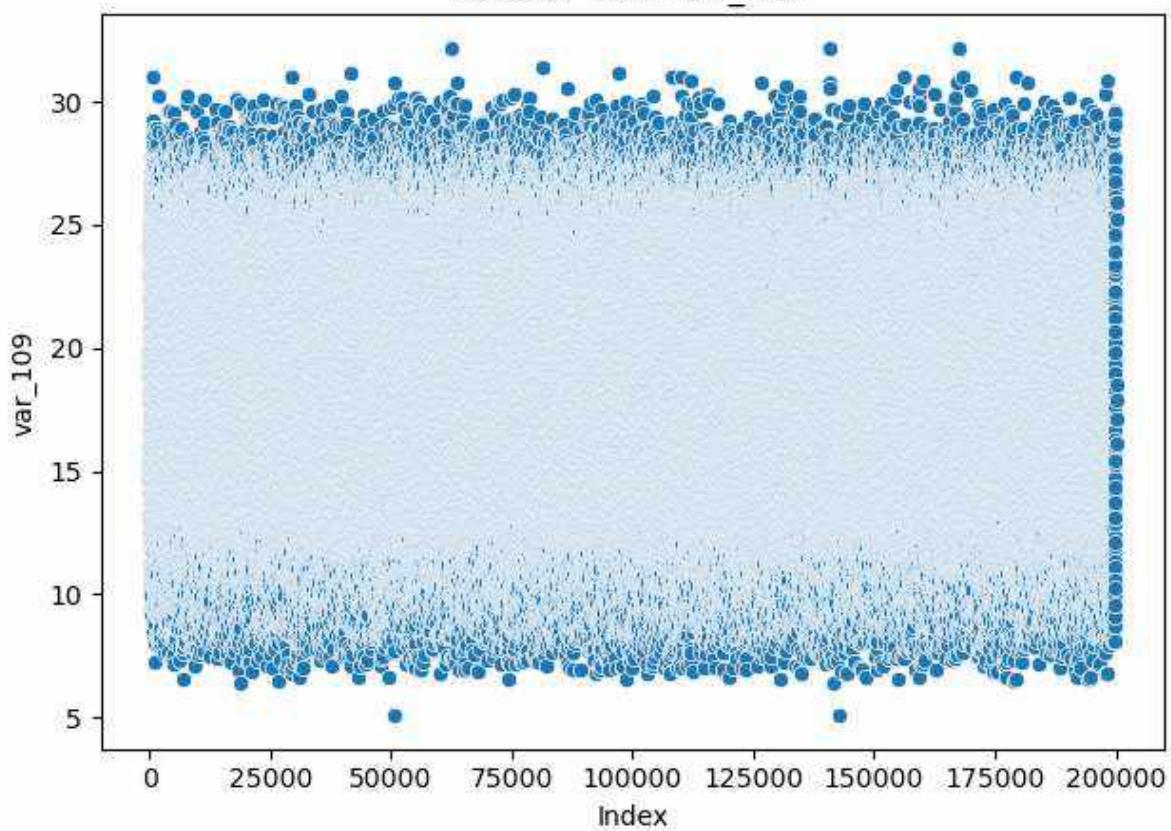
Scatter Plot - var\_107



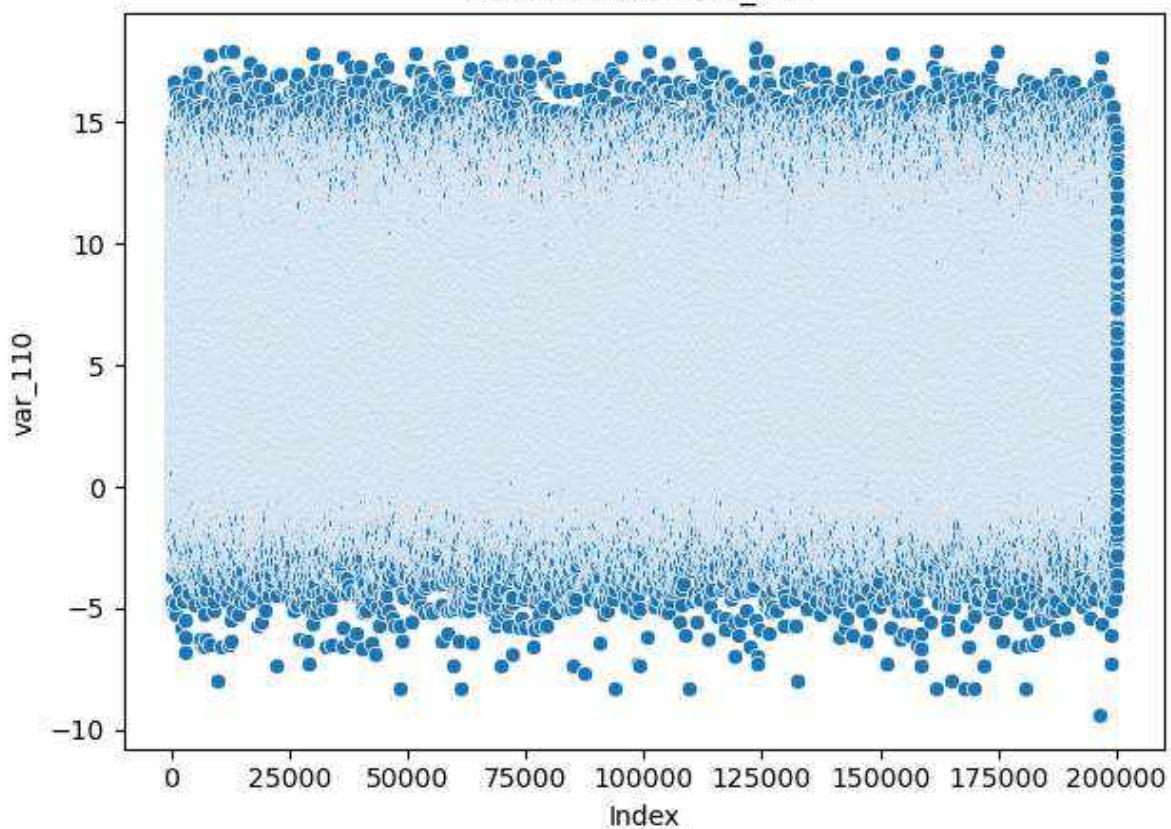
Scatter Plot - var\_108



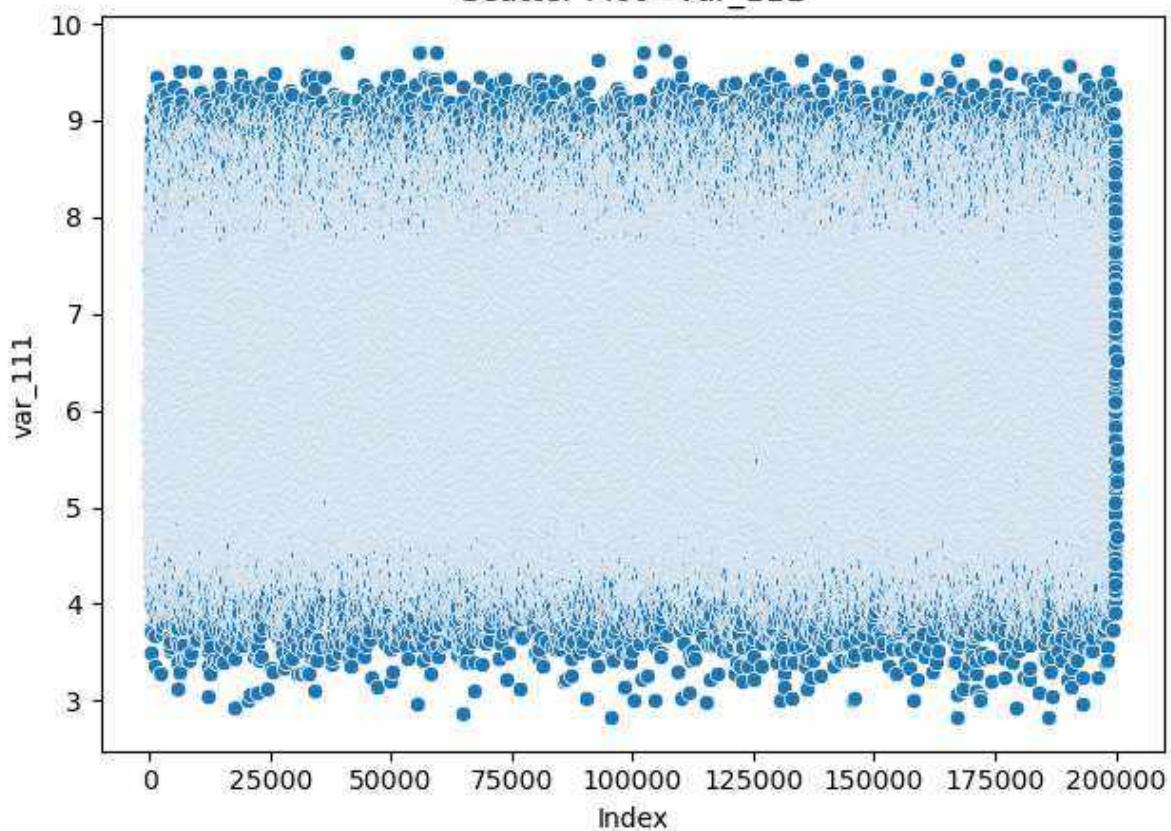
Scatter Plot - var\_109



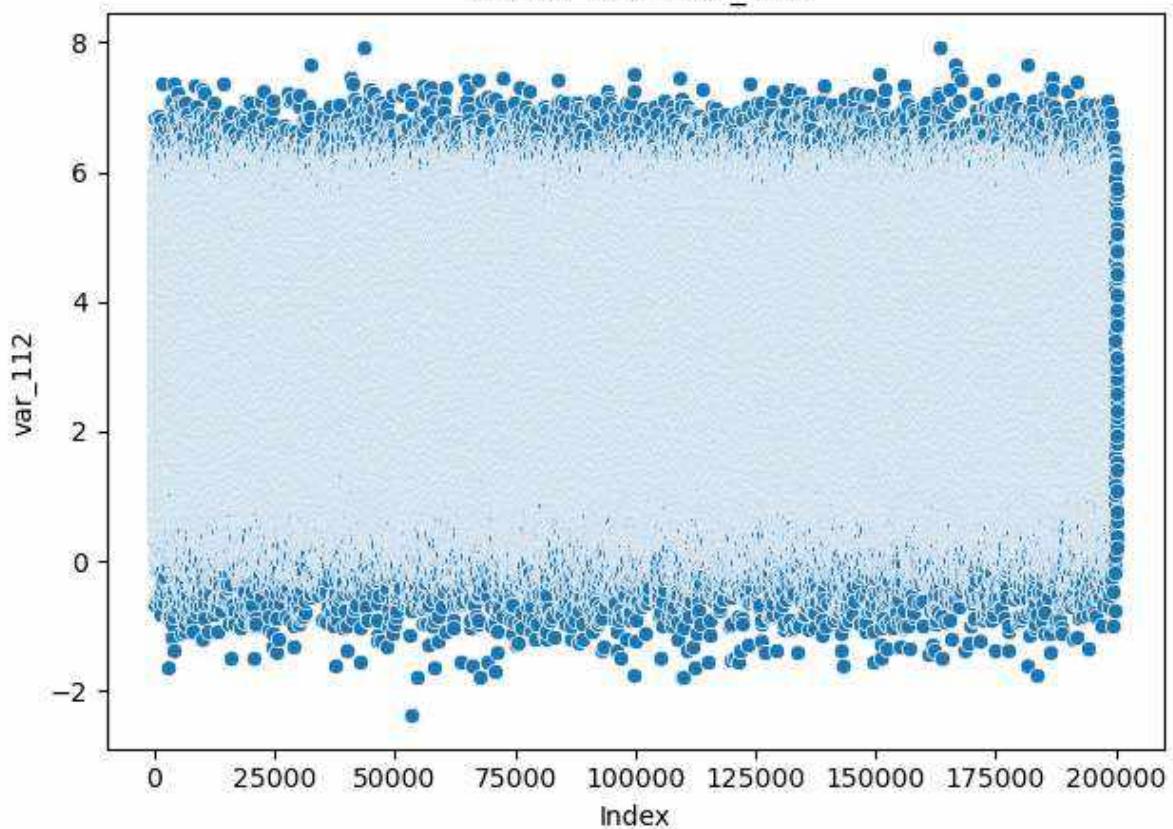
Scatter Plot - var\_110



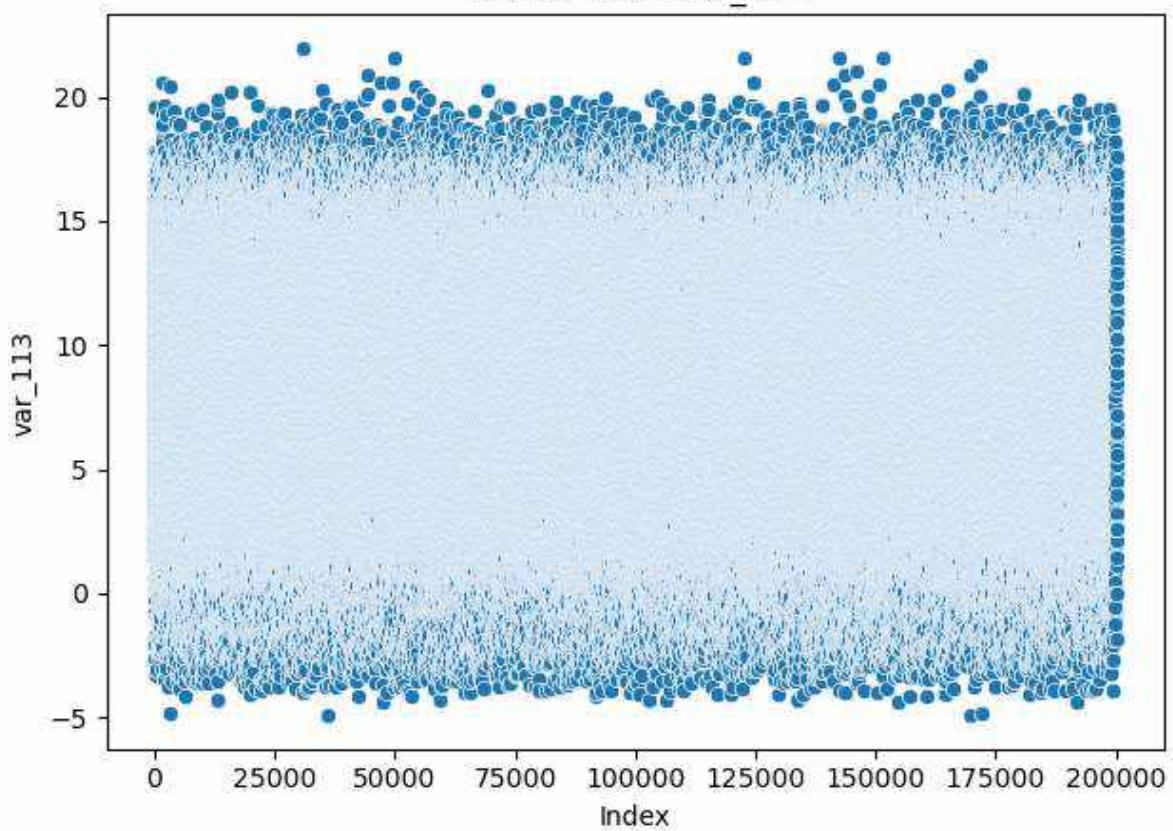
Scatter Plot - var\_111



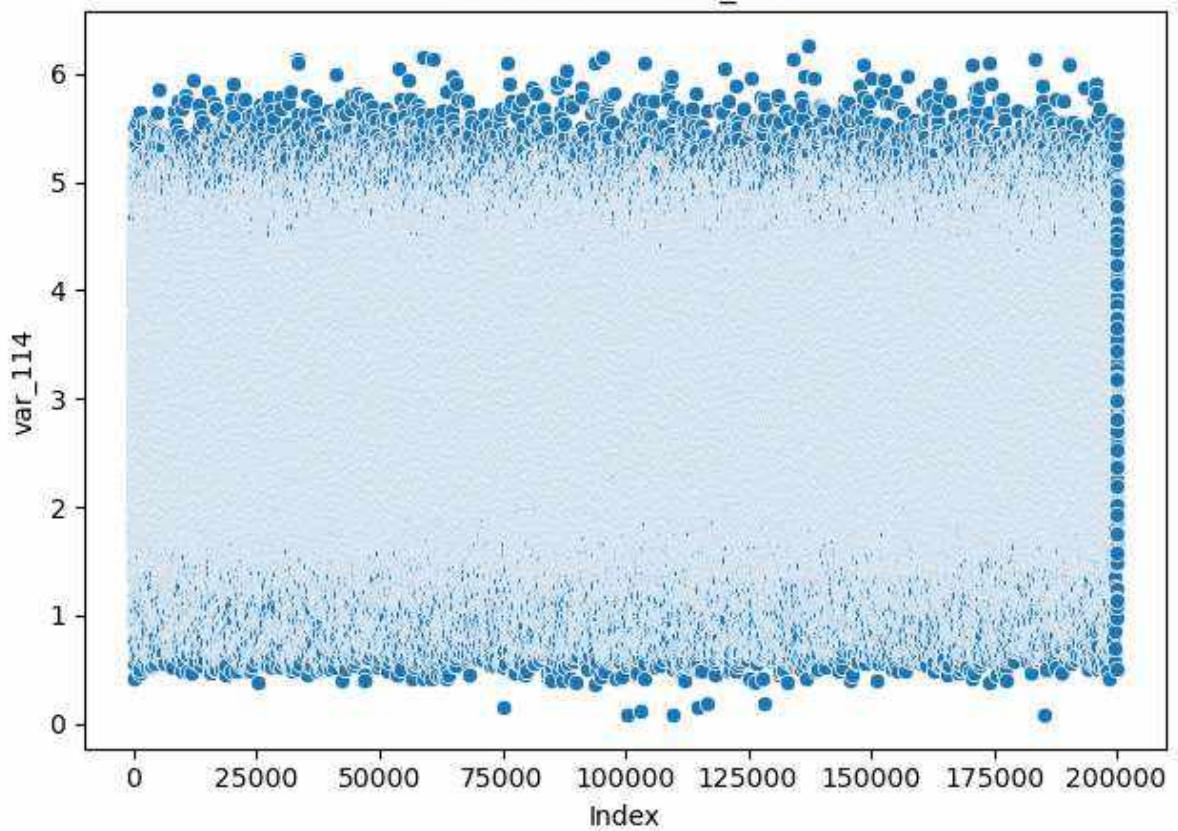
Scatter Plot - var\_112



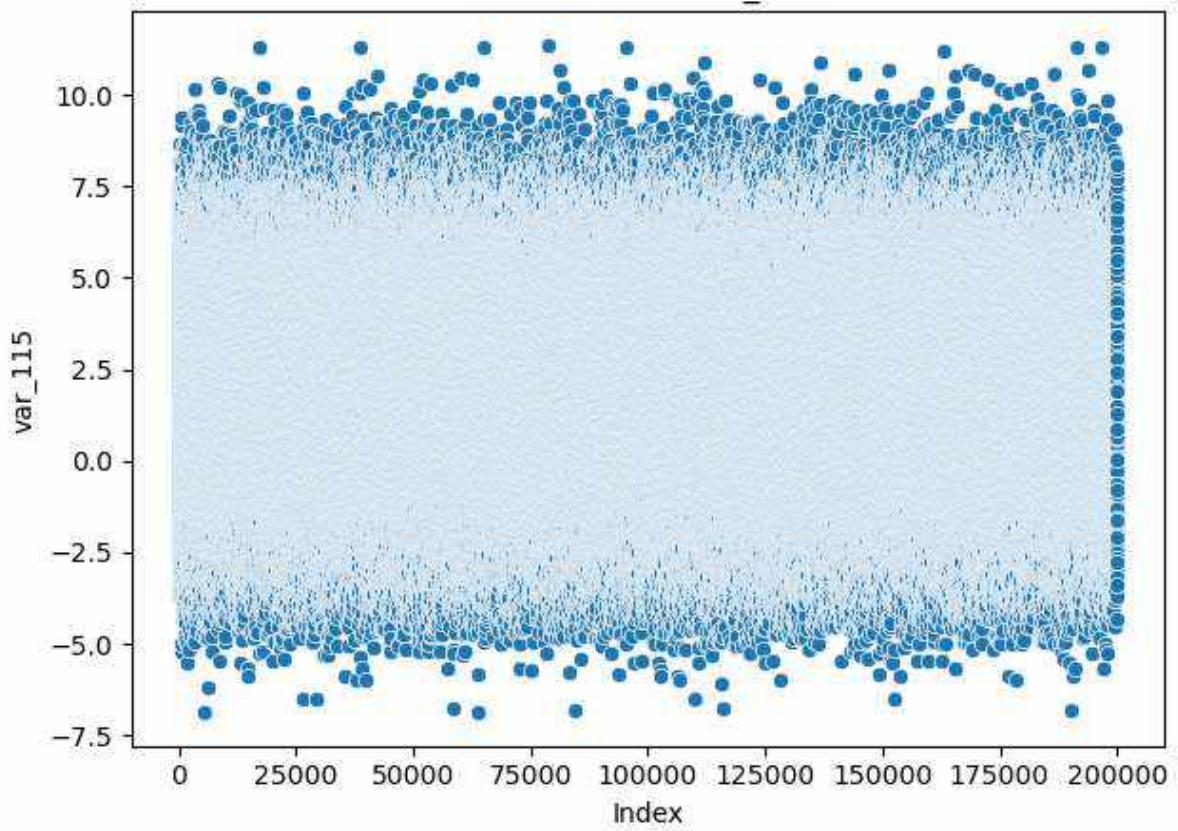
Scatter Plot - var\_113



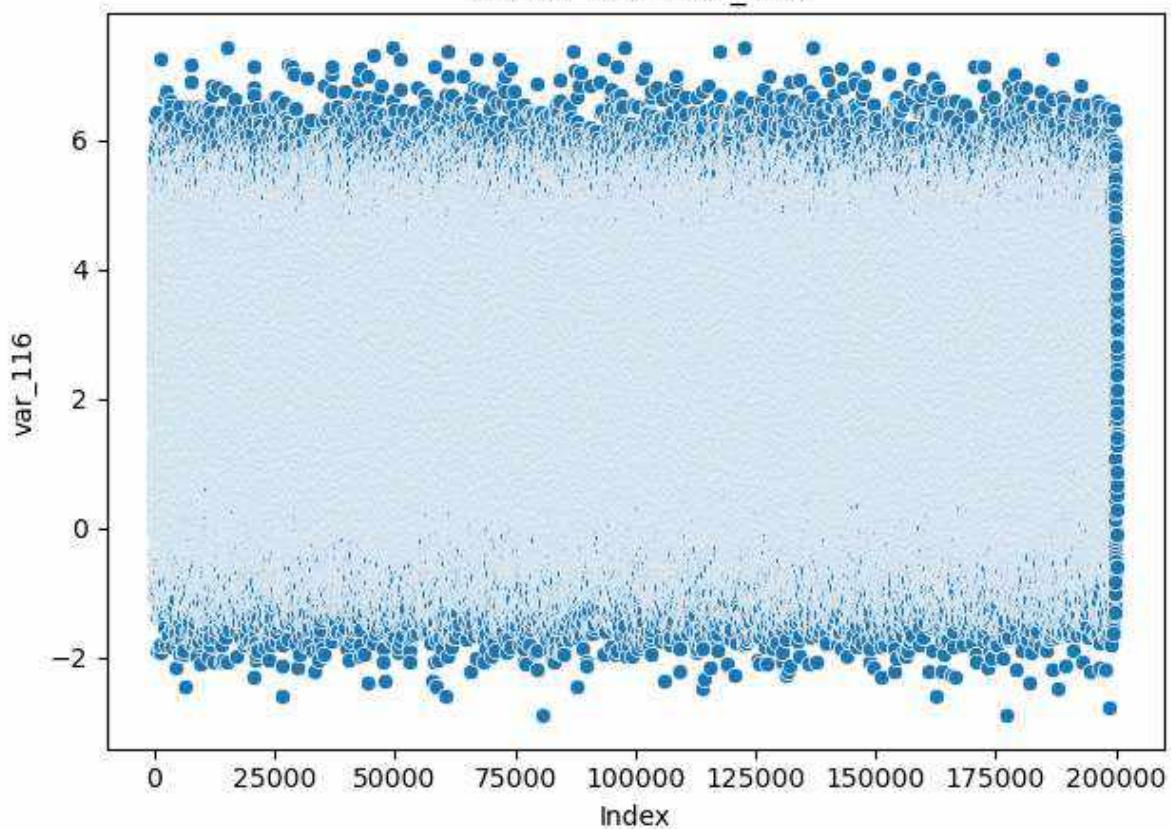
Scatter Plot - var\_114



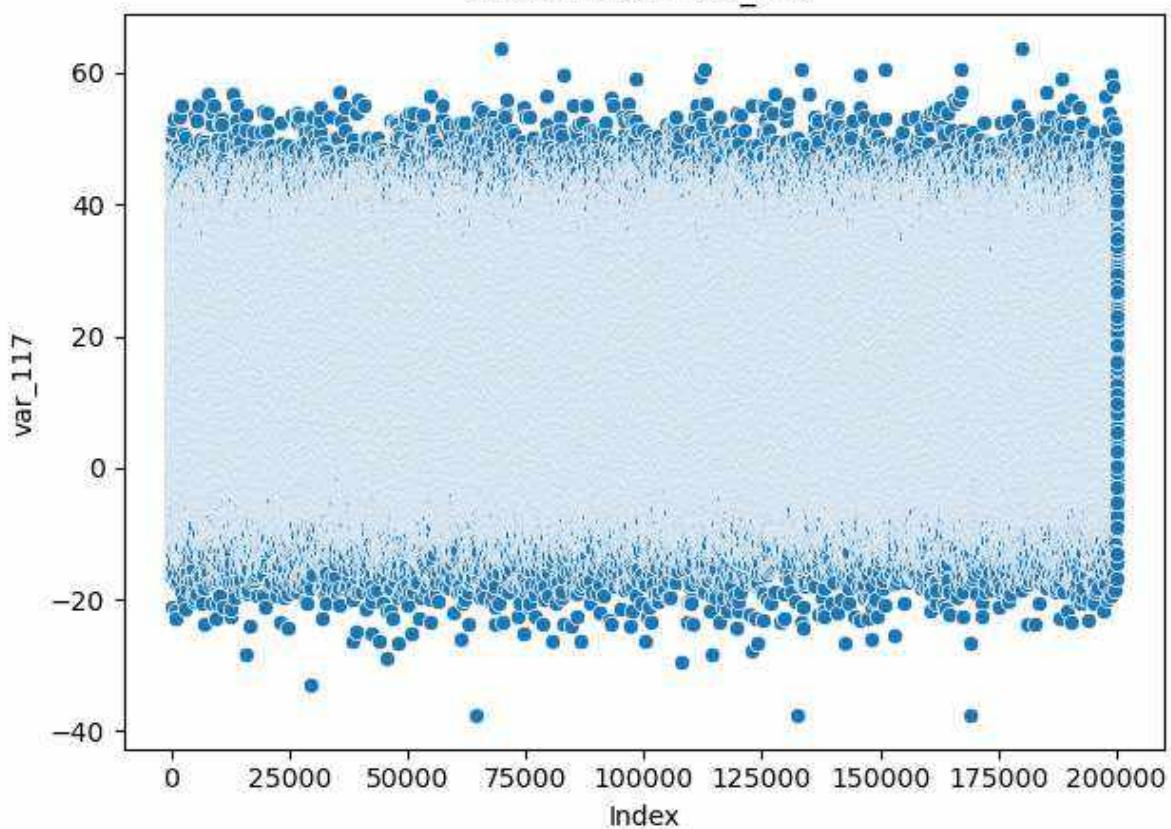
Scatter Plot - var\_115



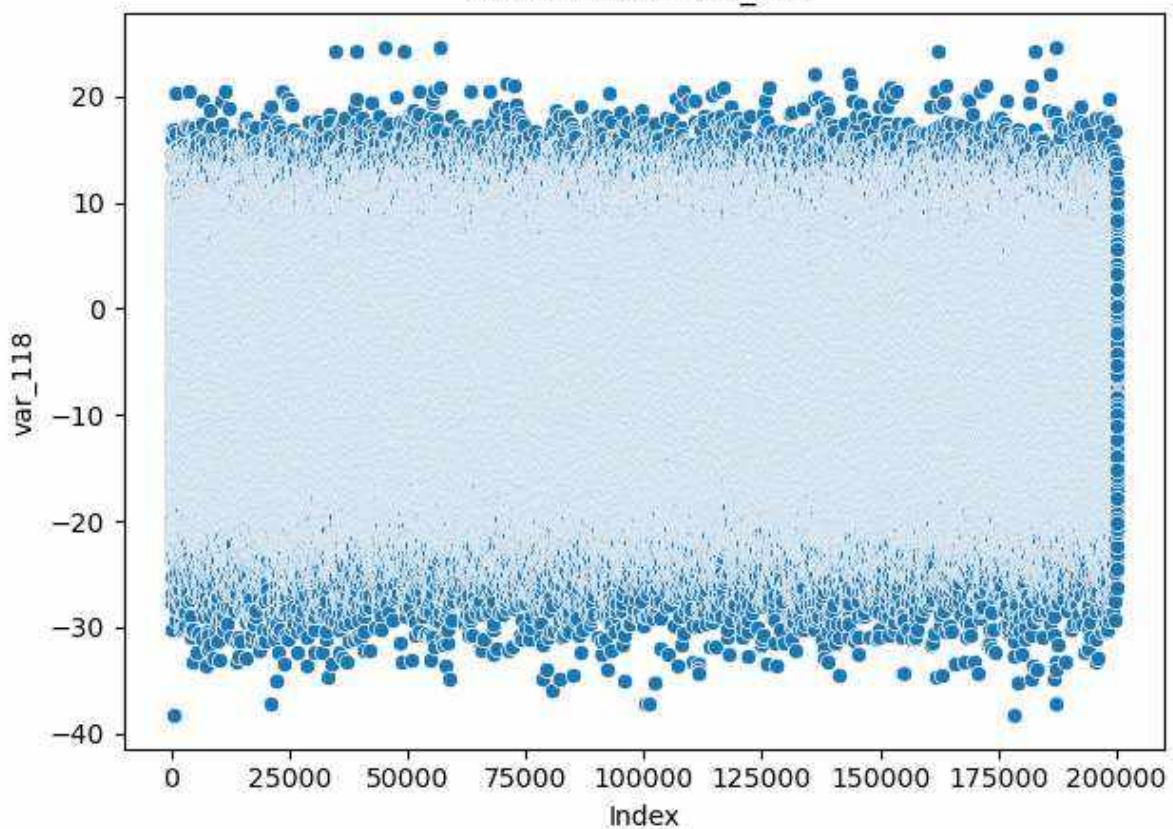
Scatter Plot - var\_116



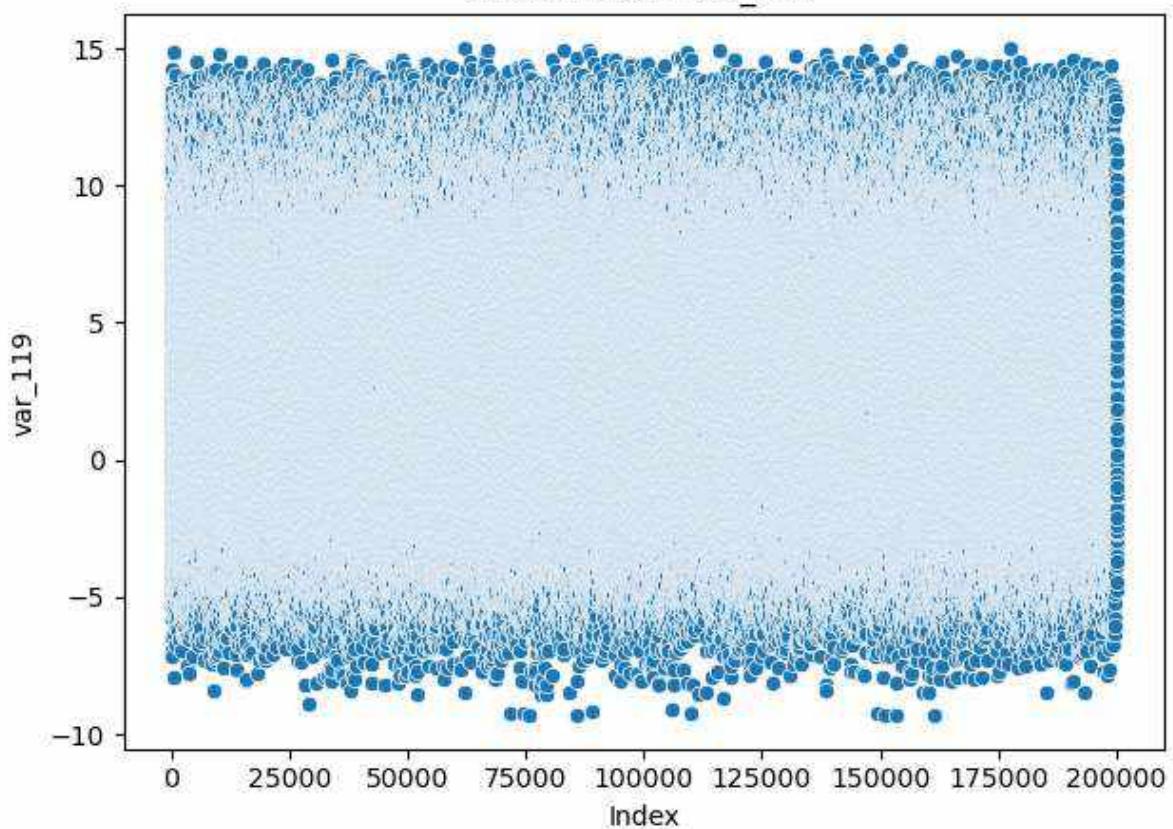
Scatter Plot - var\_117



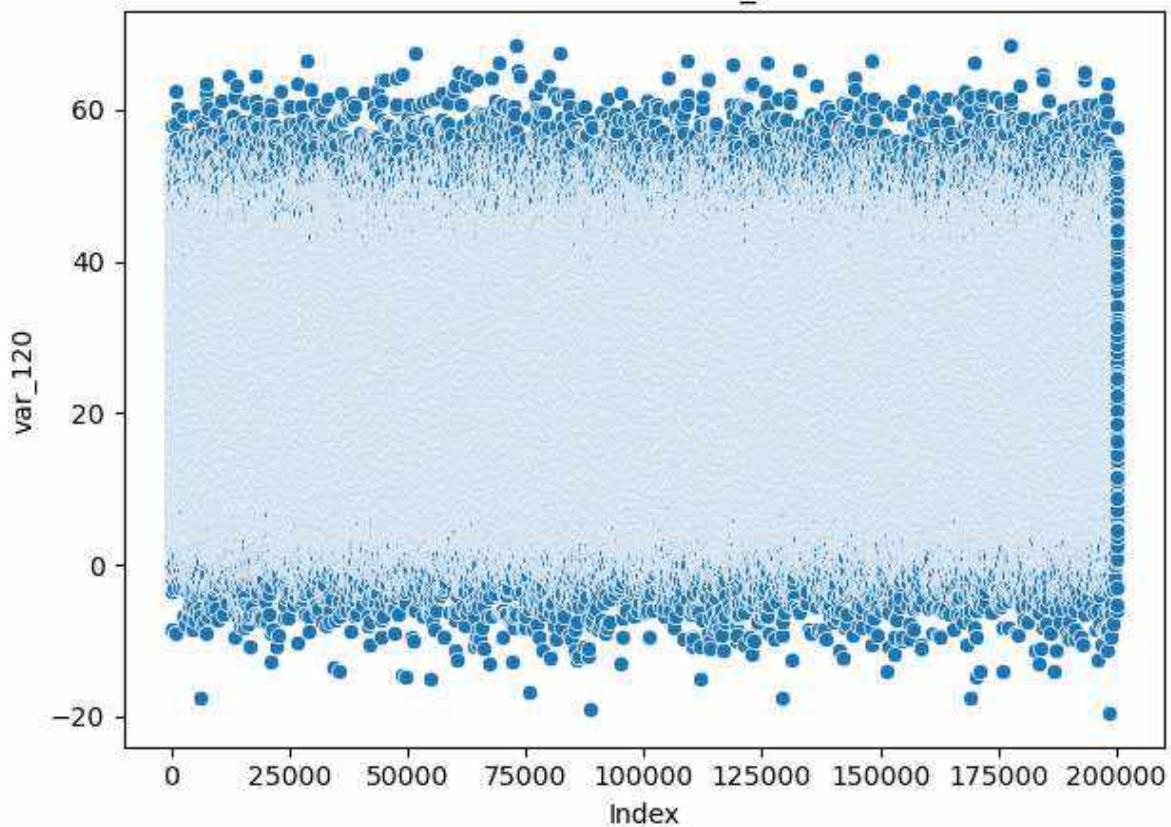
Scatter Plot - var\_118



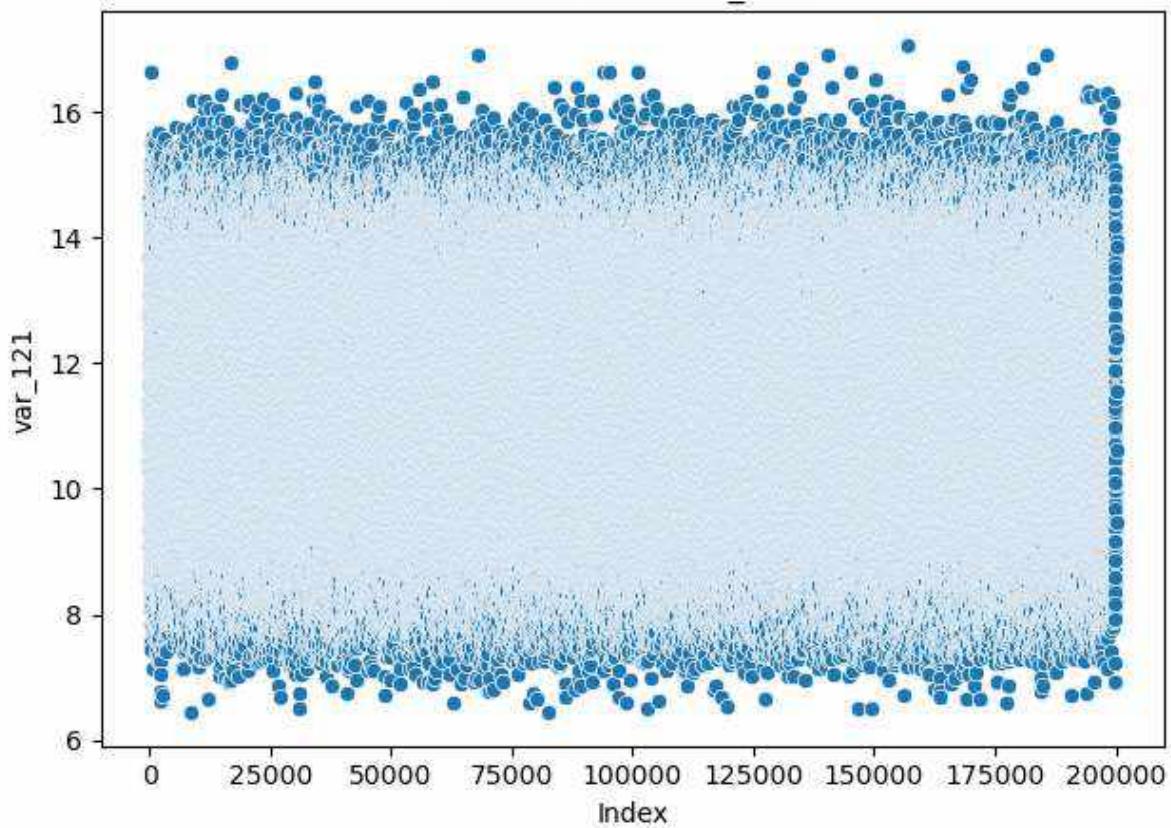
Scatter Plot - var\_119

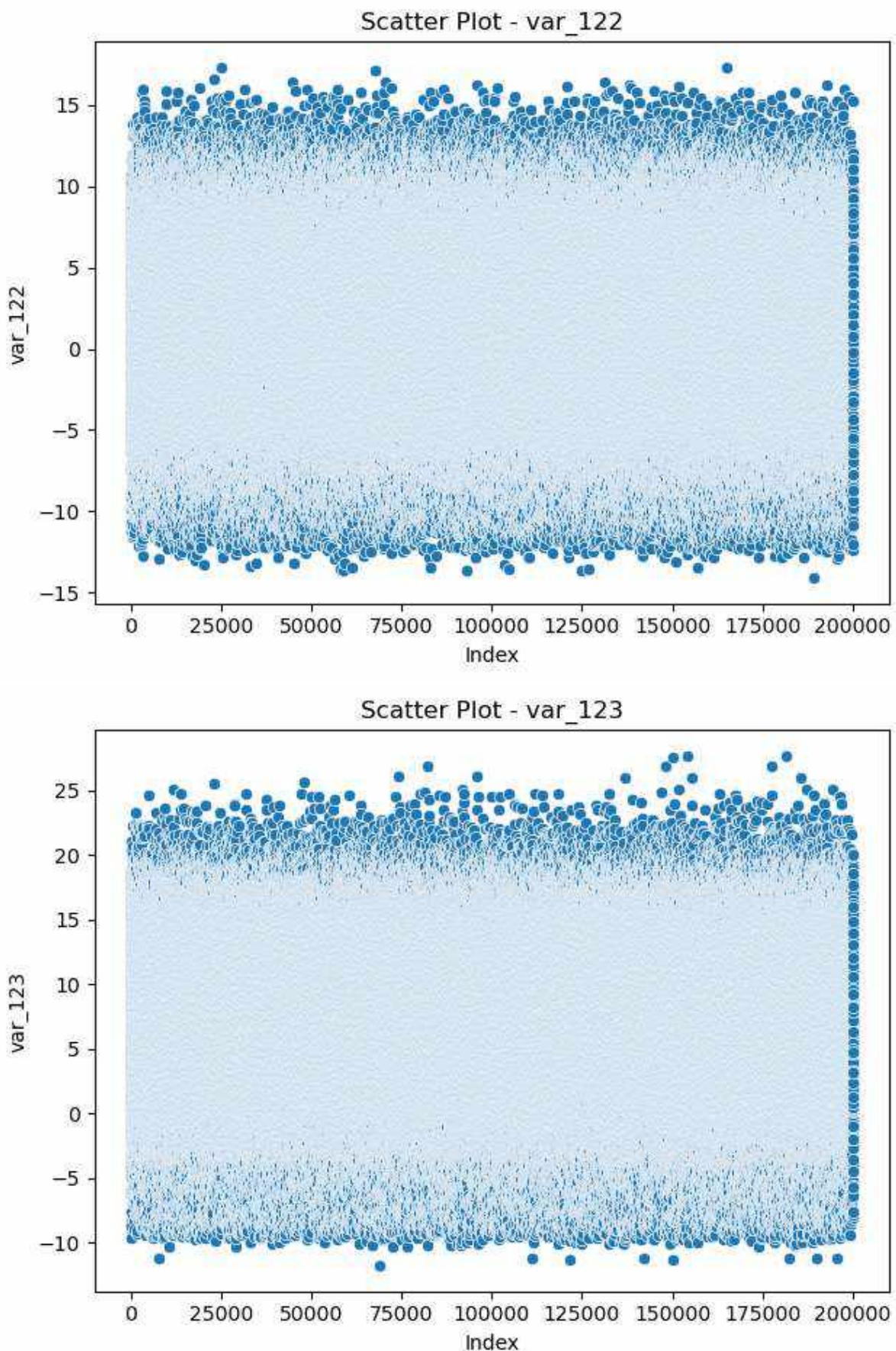


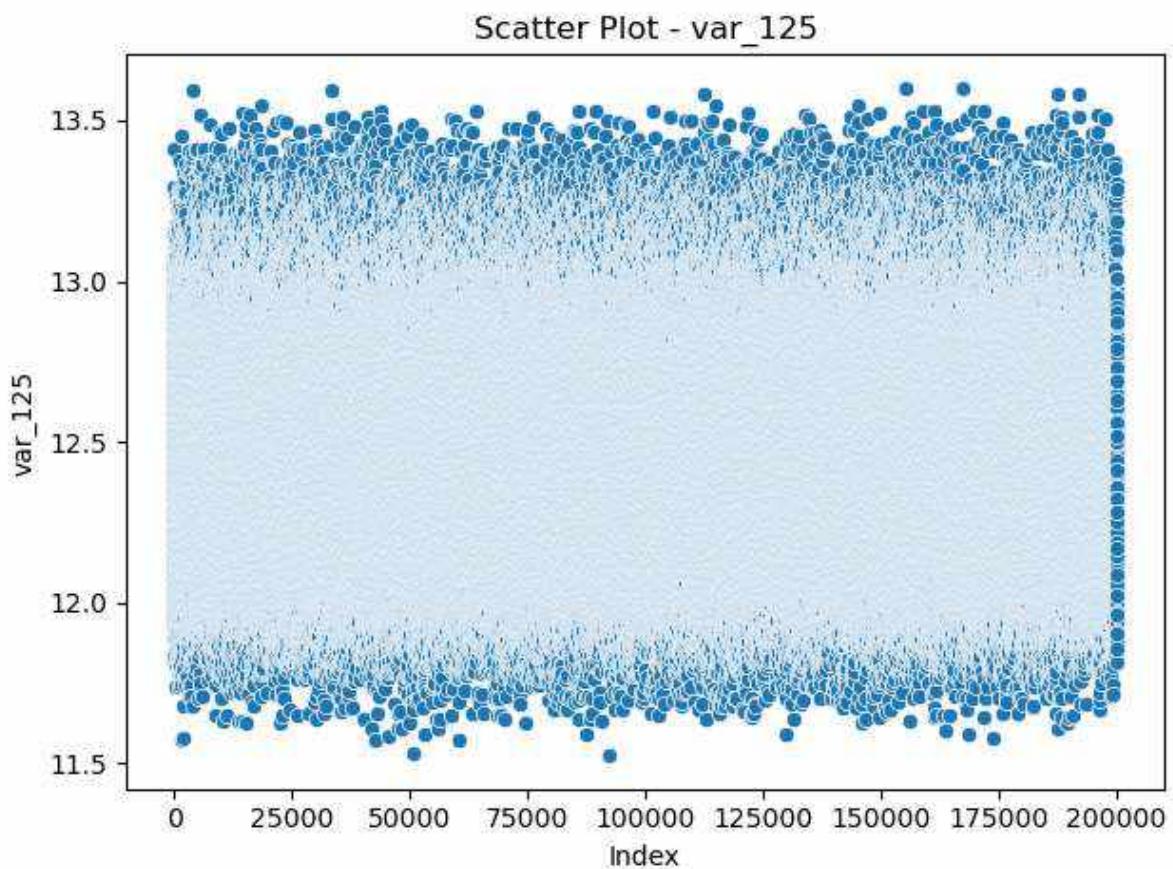
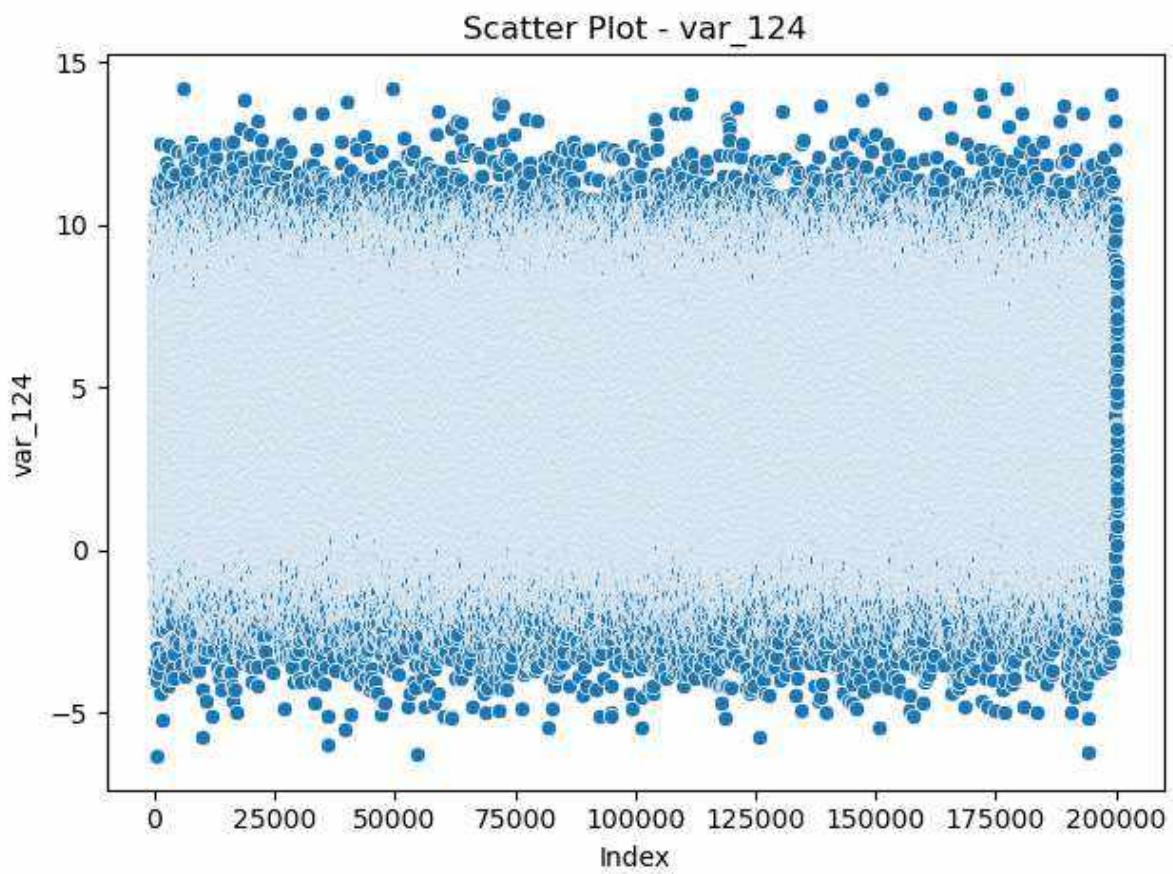
Scatter Plot - var\_120



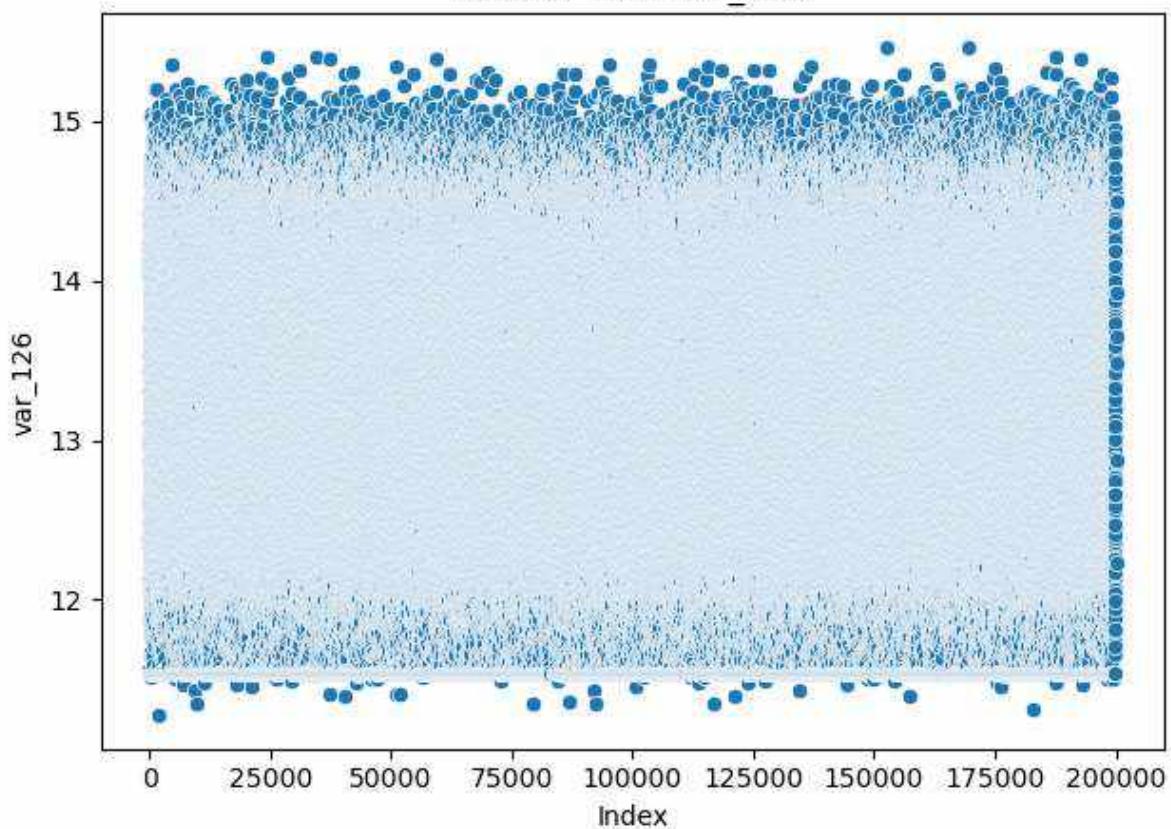
Scatter Plot - var\_121



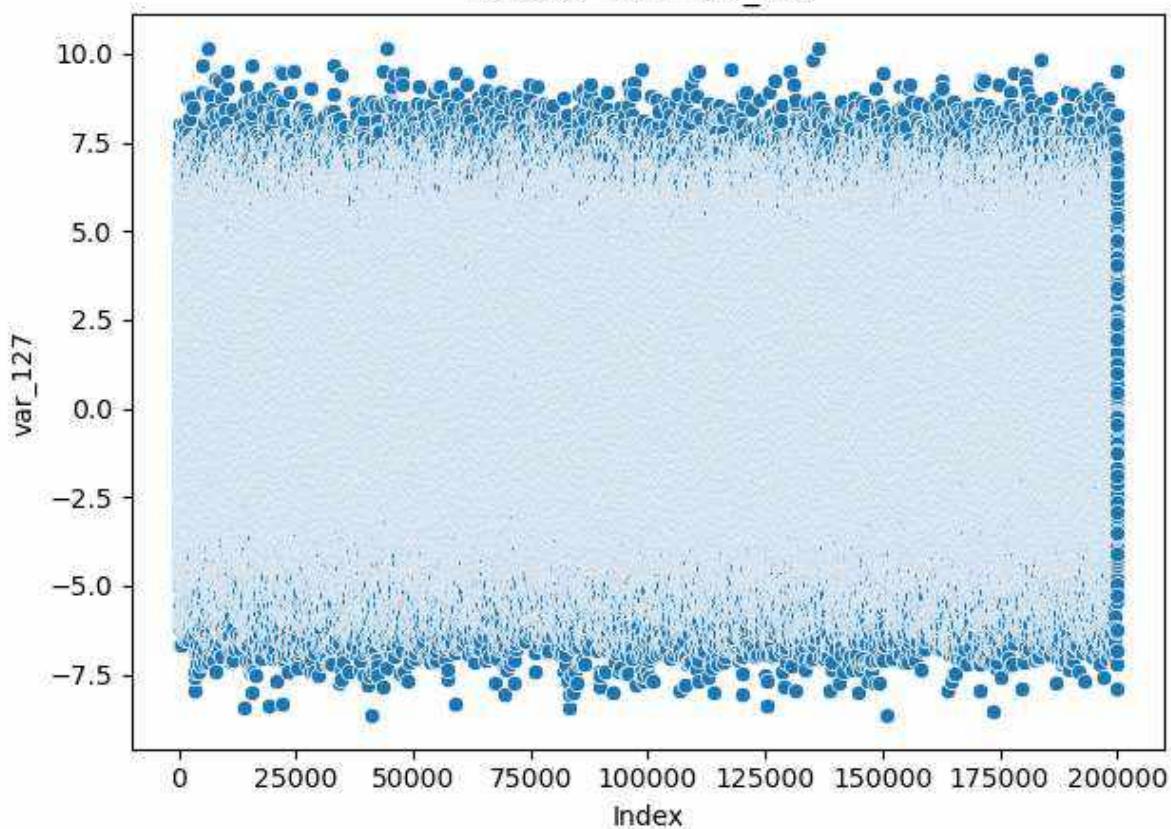


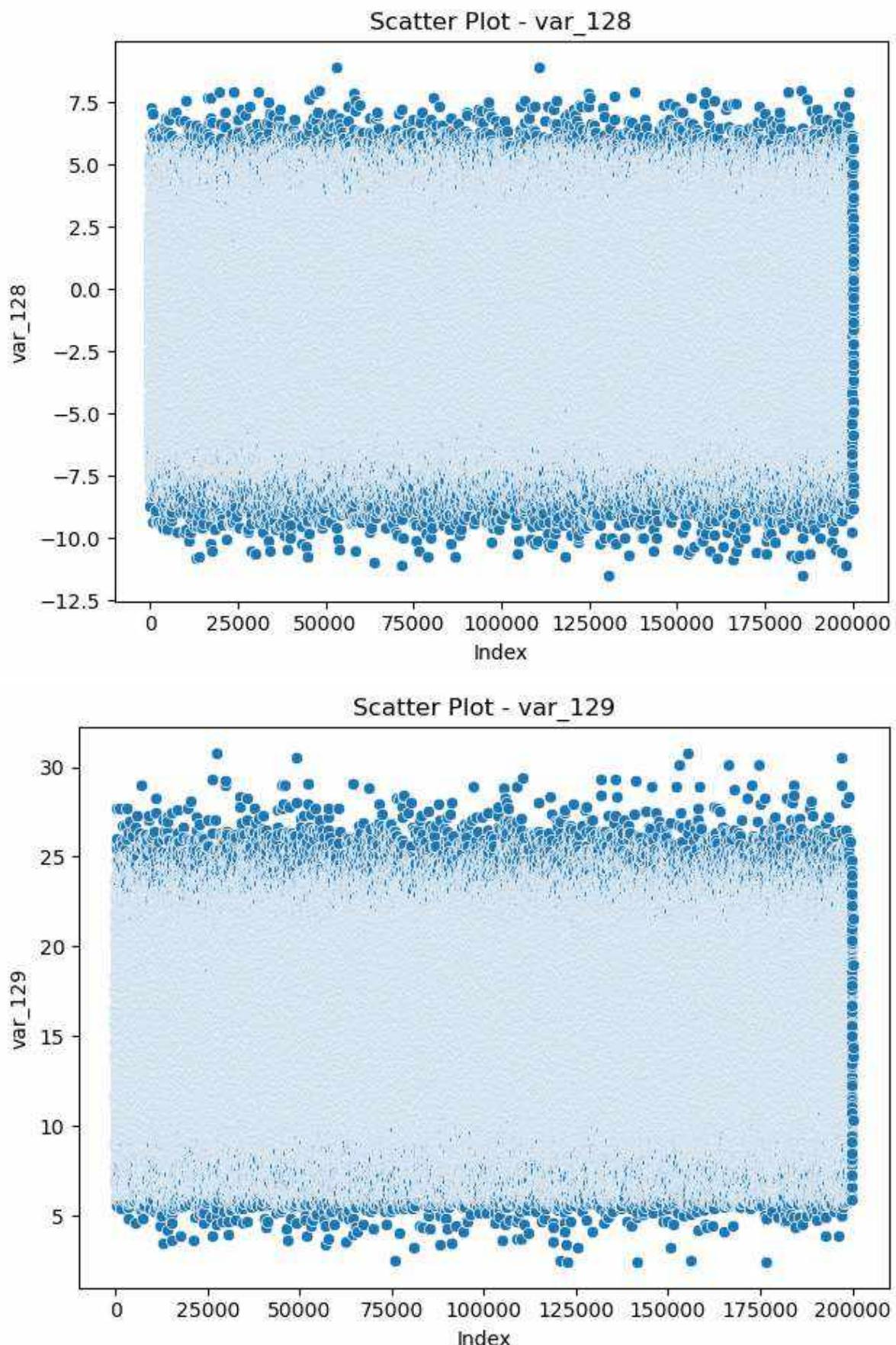


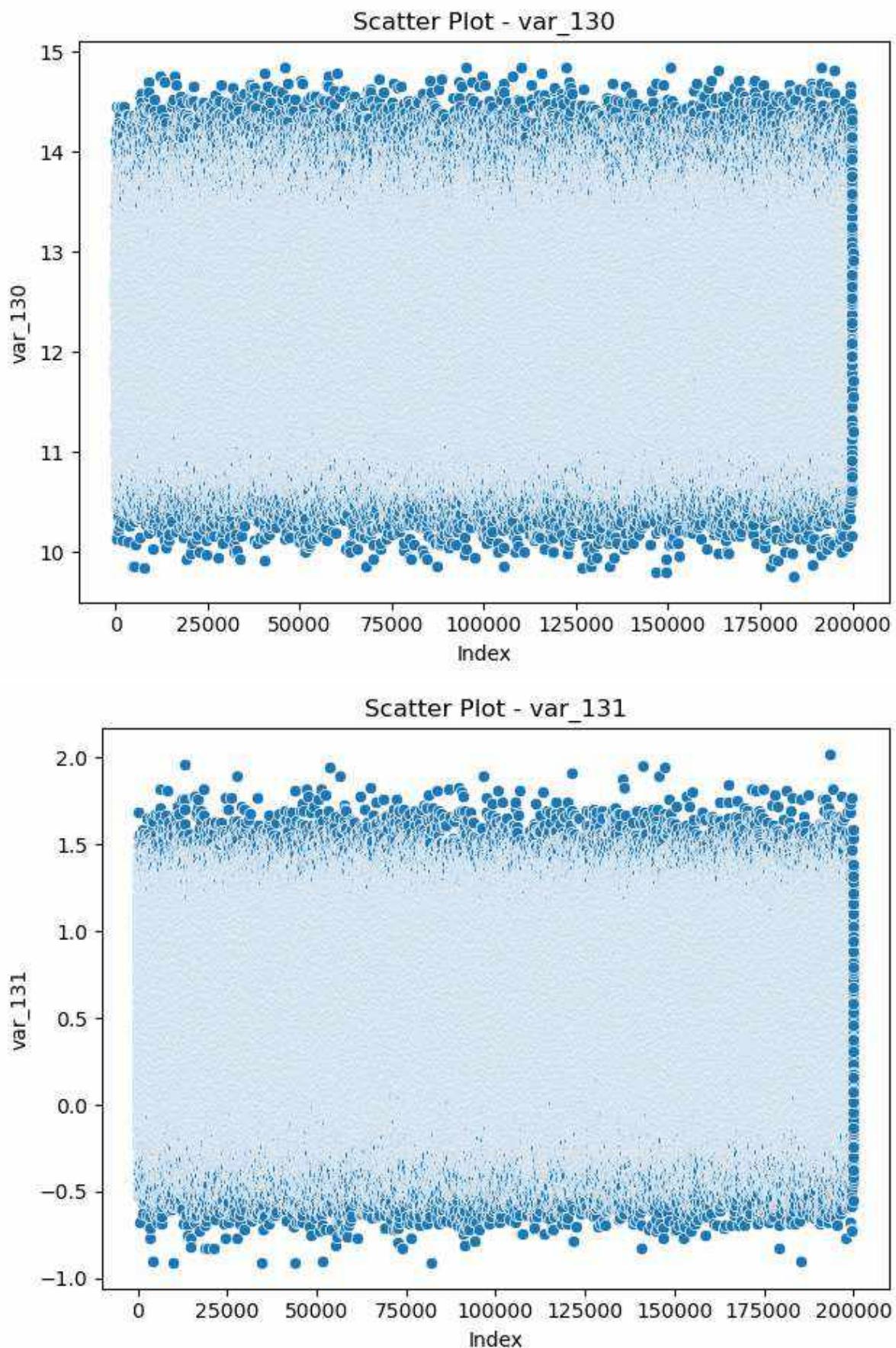
Scatter Plot - var\_126



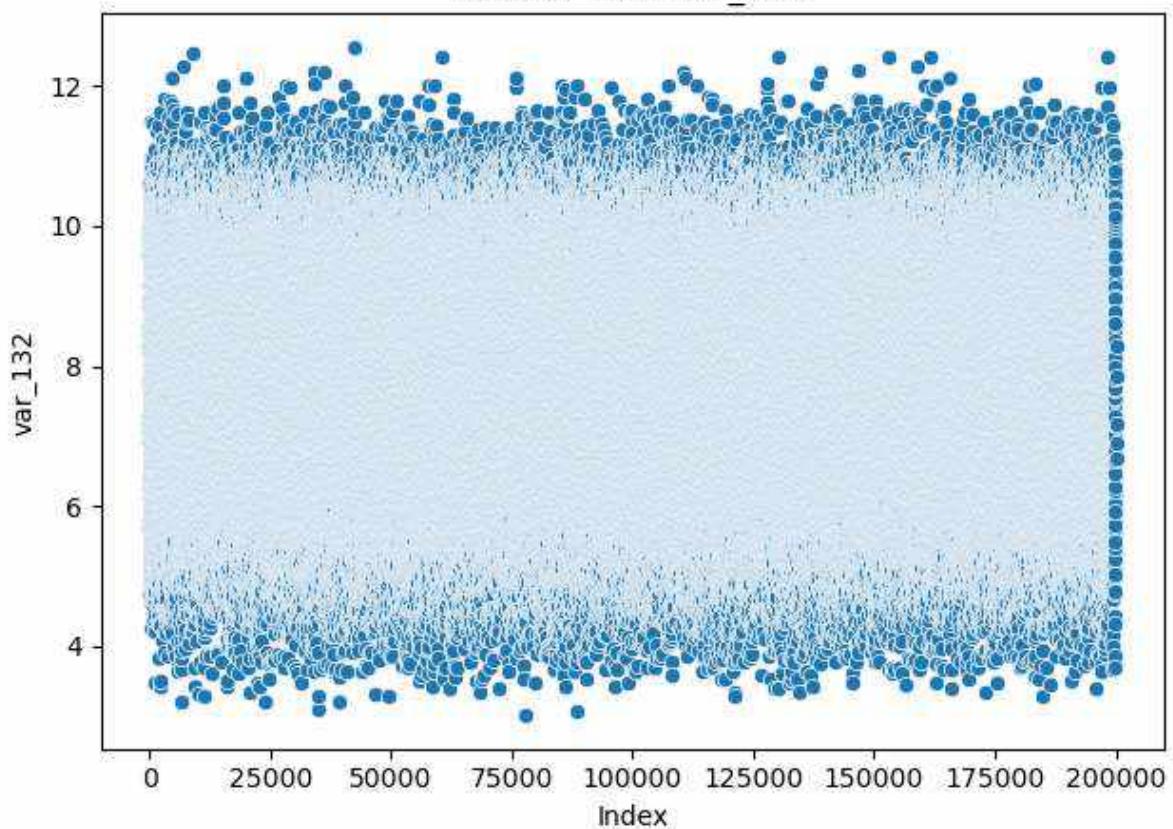
Scatter Plot - var\_127



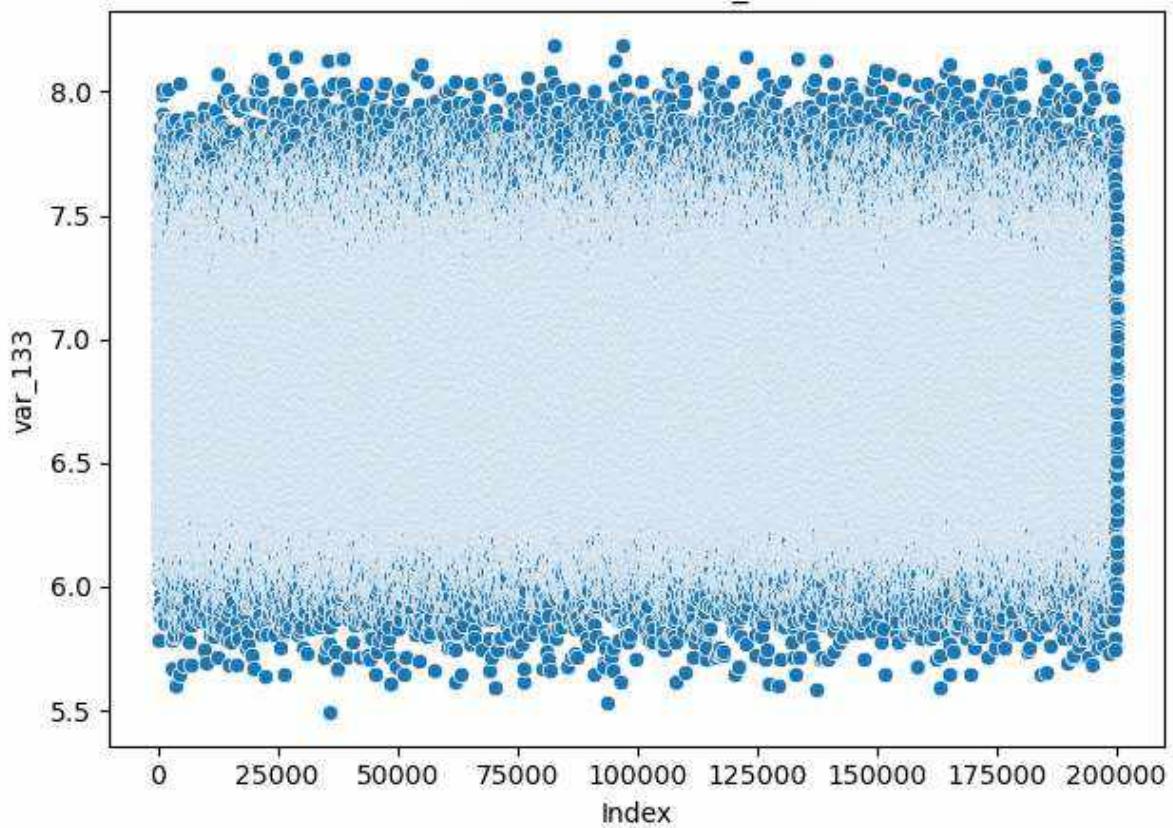


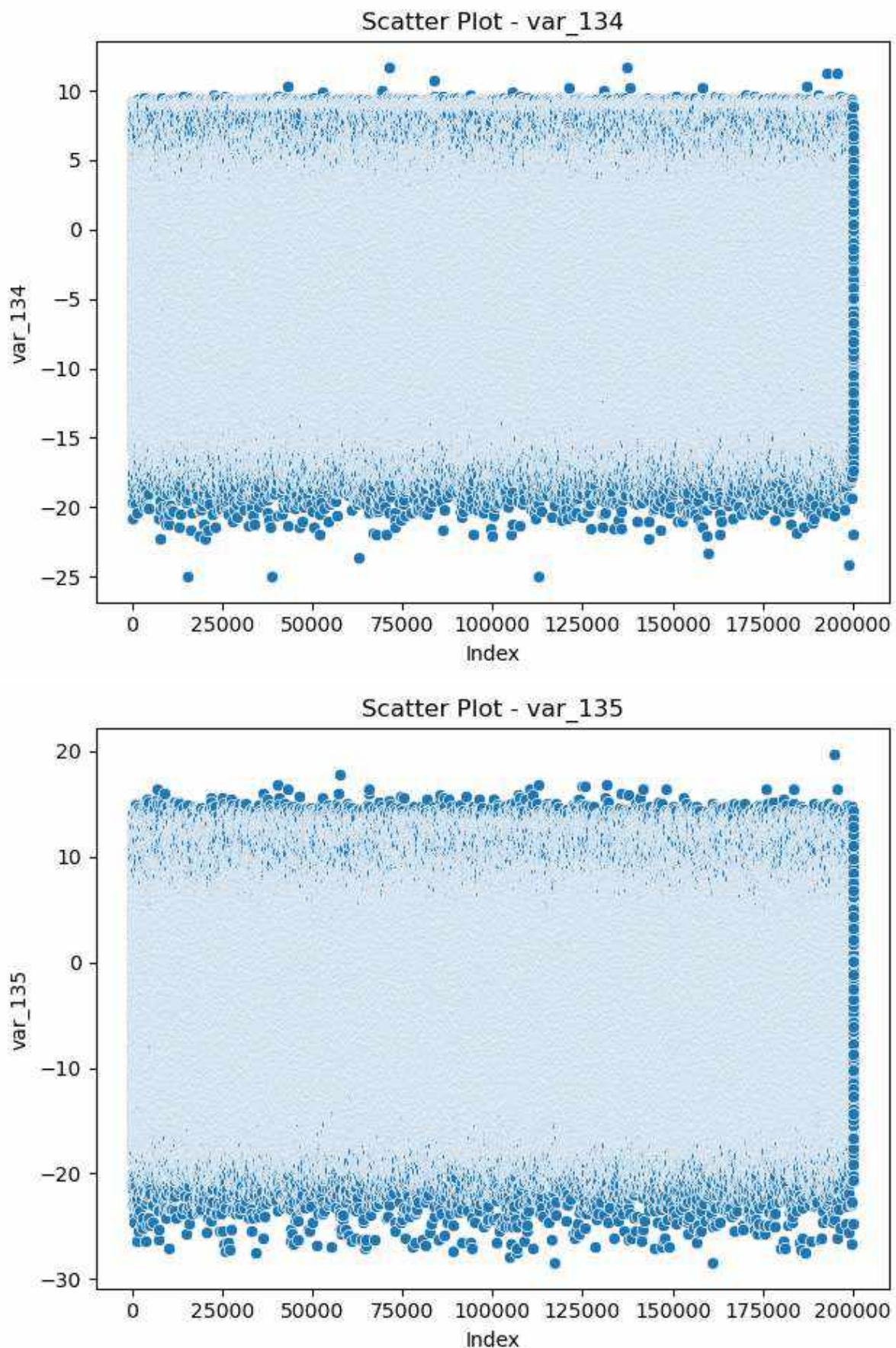


Scatter Plot - var\_132

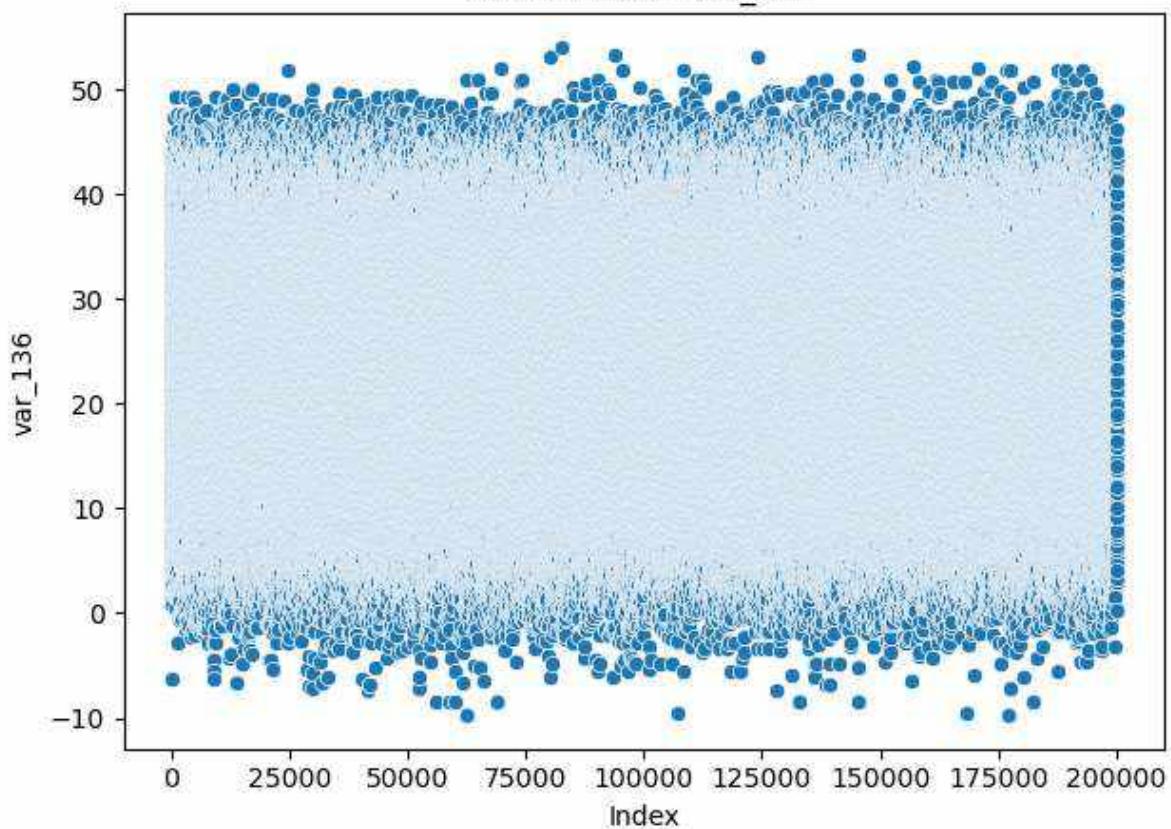


Scatter Plot - var\_133

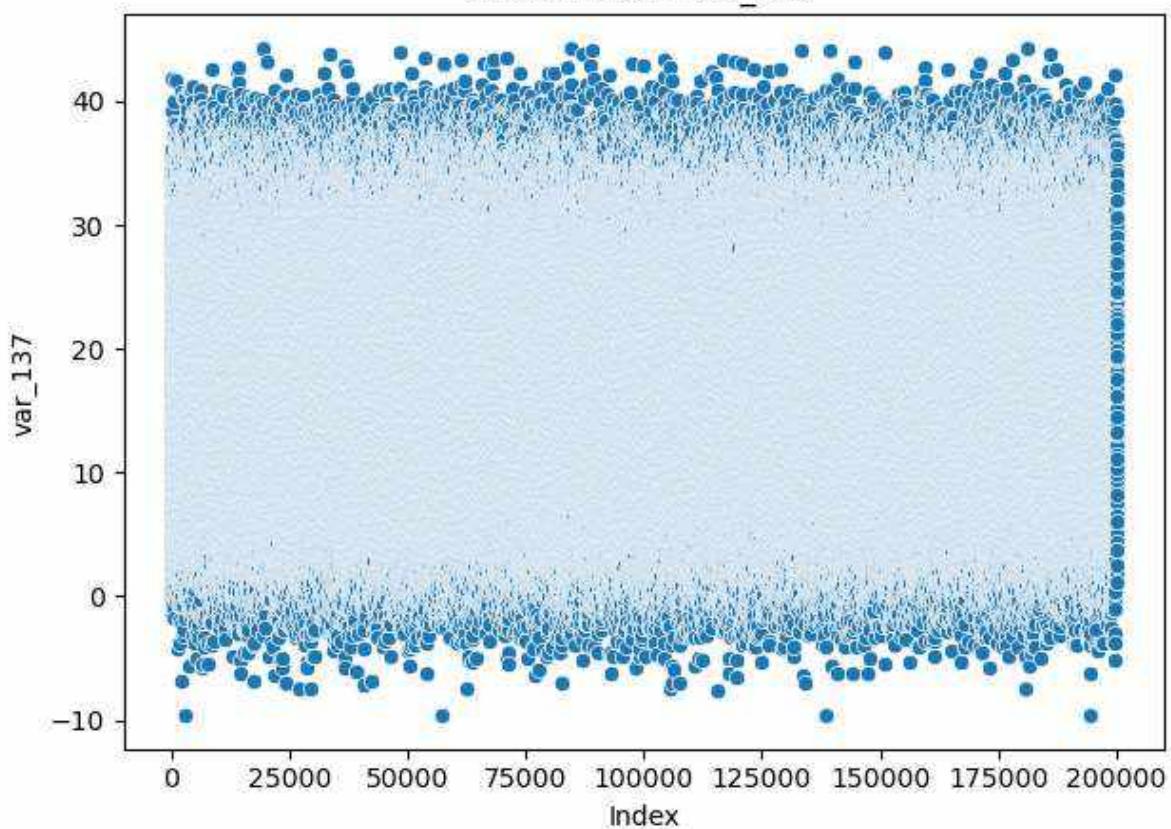


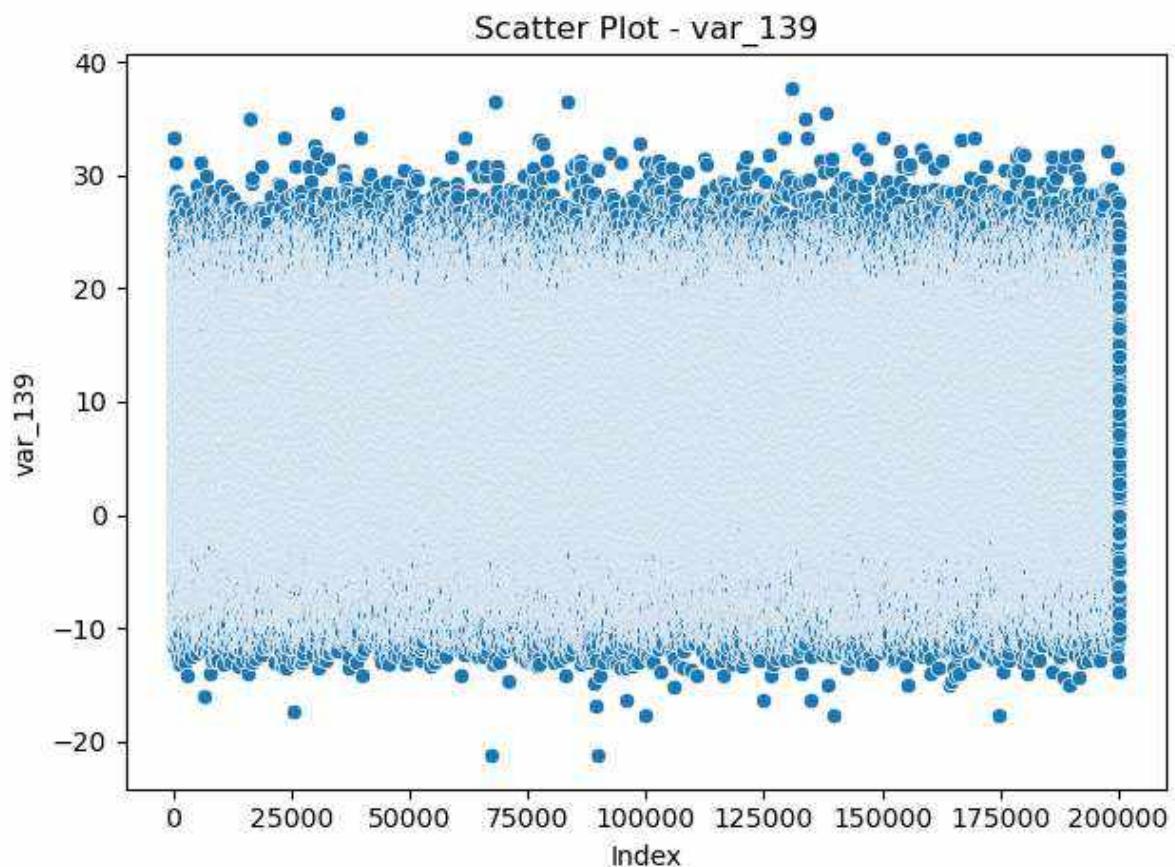
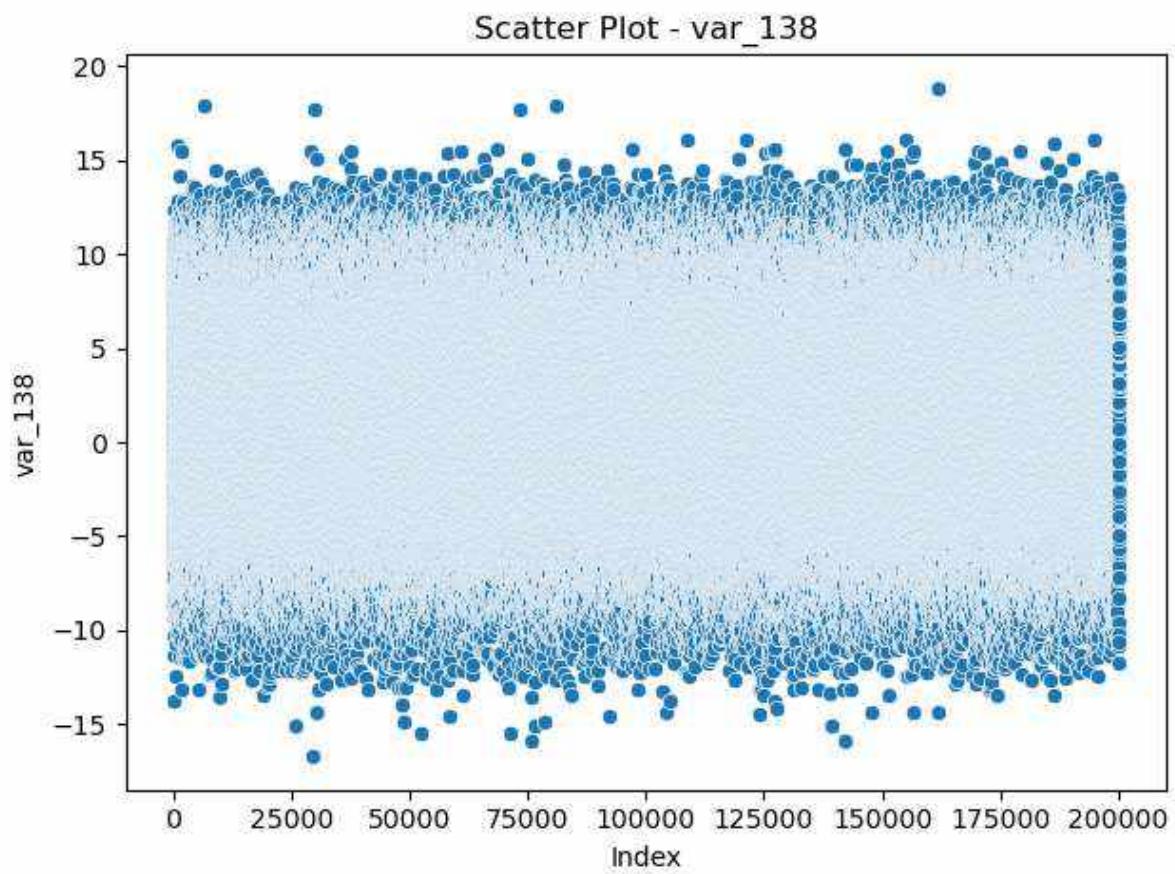


Scatter Plot - var\_136

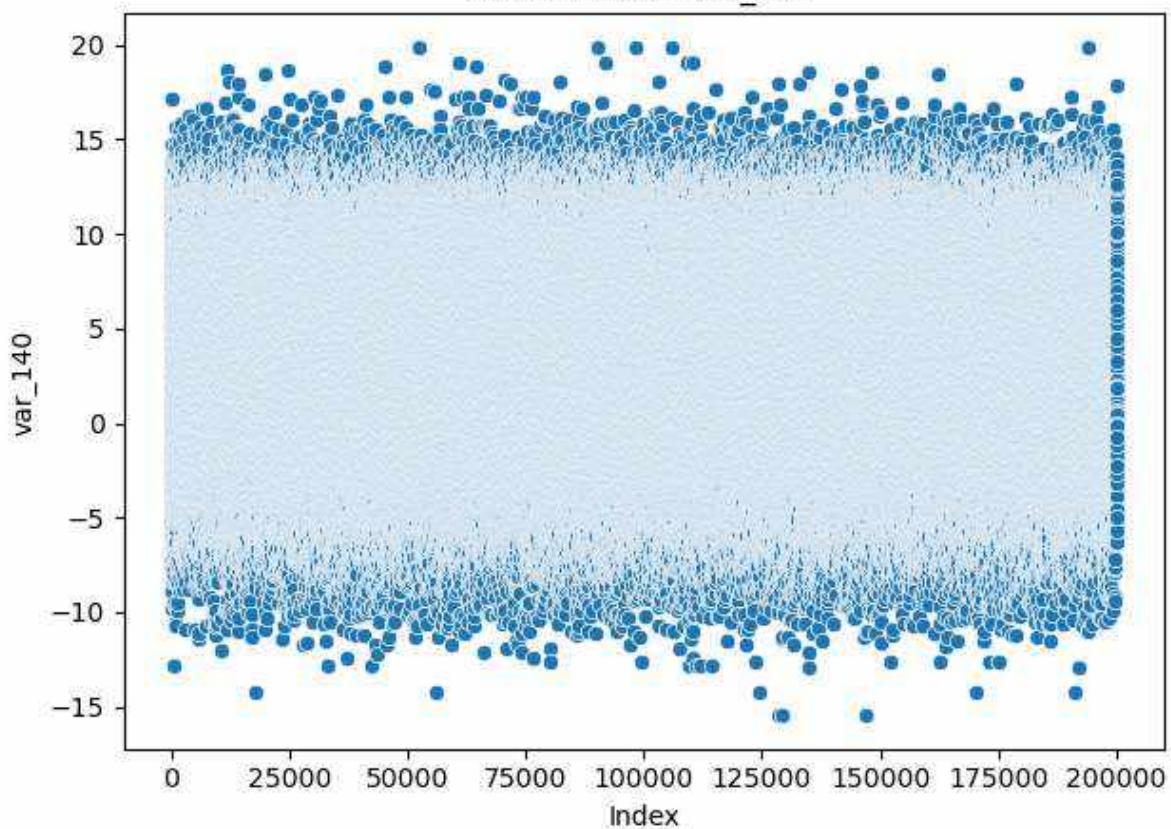


Scatter Plot - var\_137

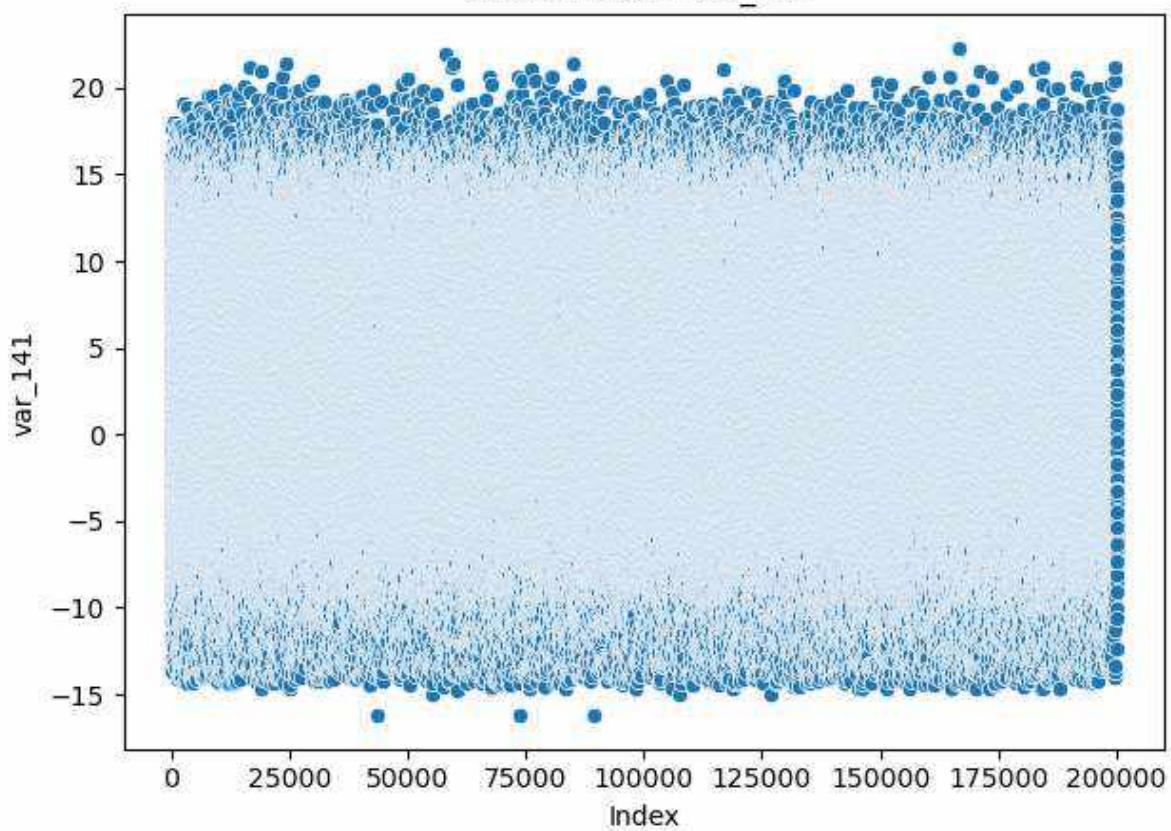




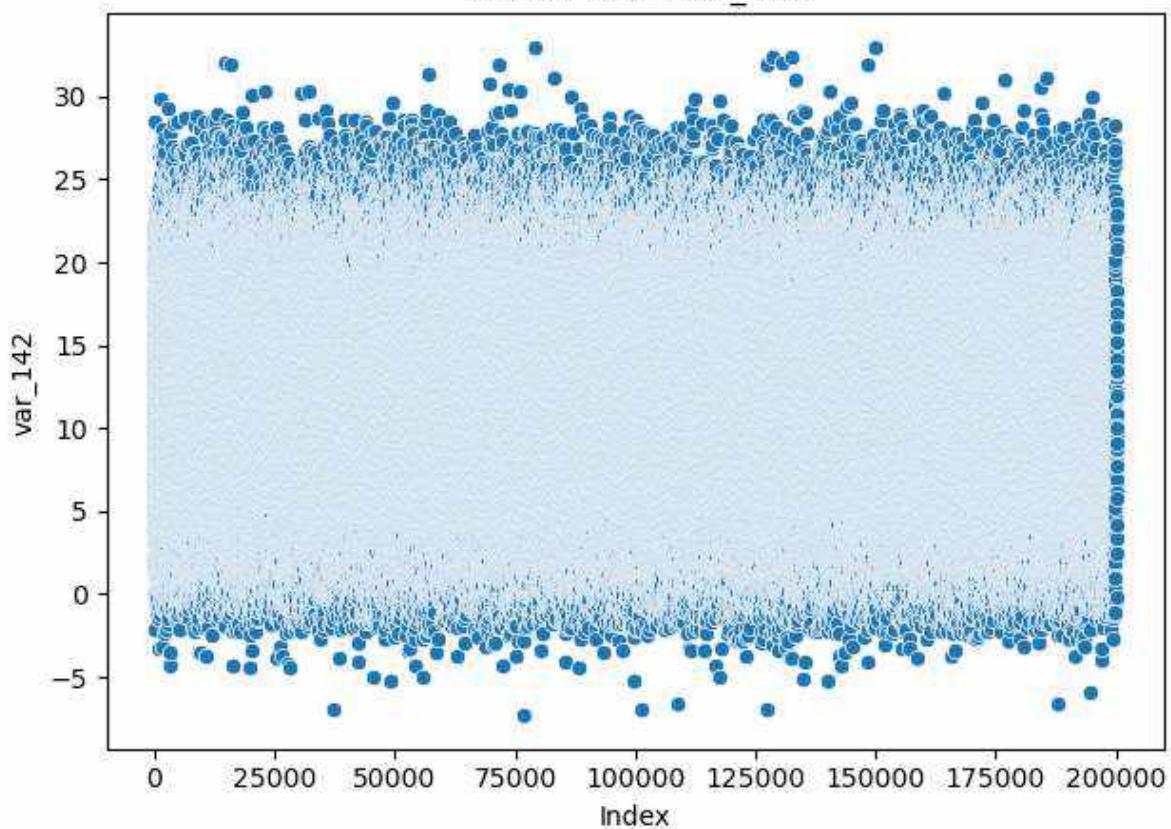
Scatter Plot - var\_140



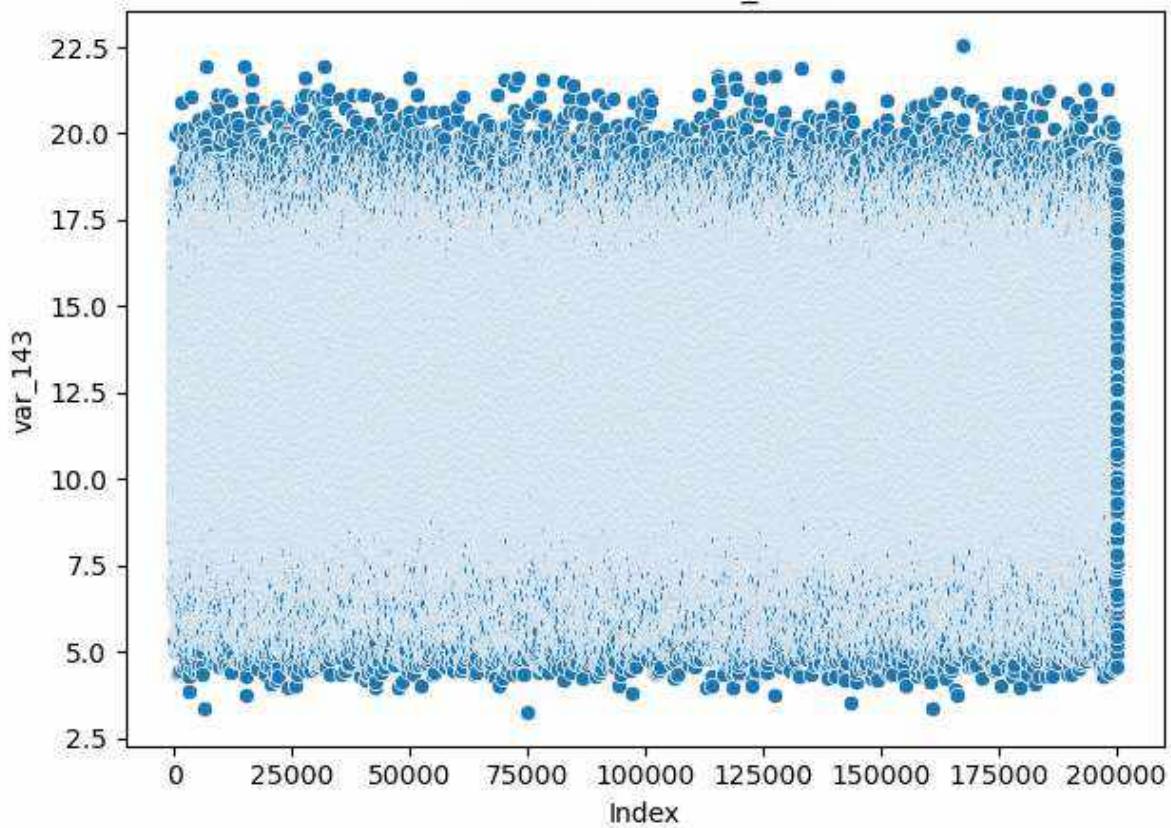
Scatter Plot - var\_141



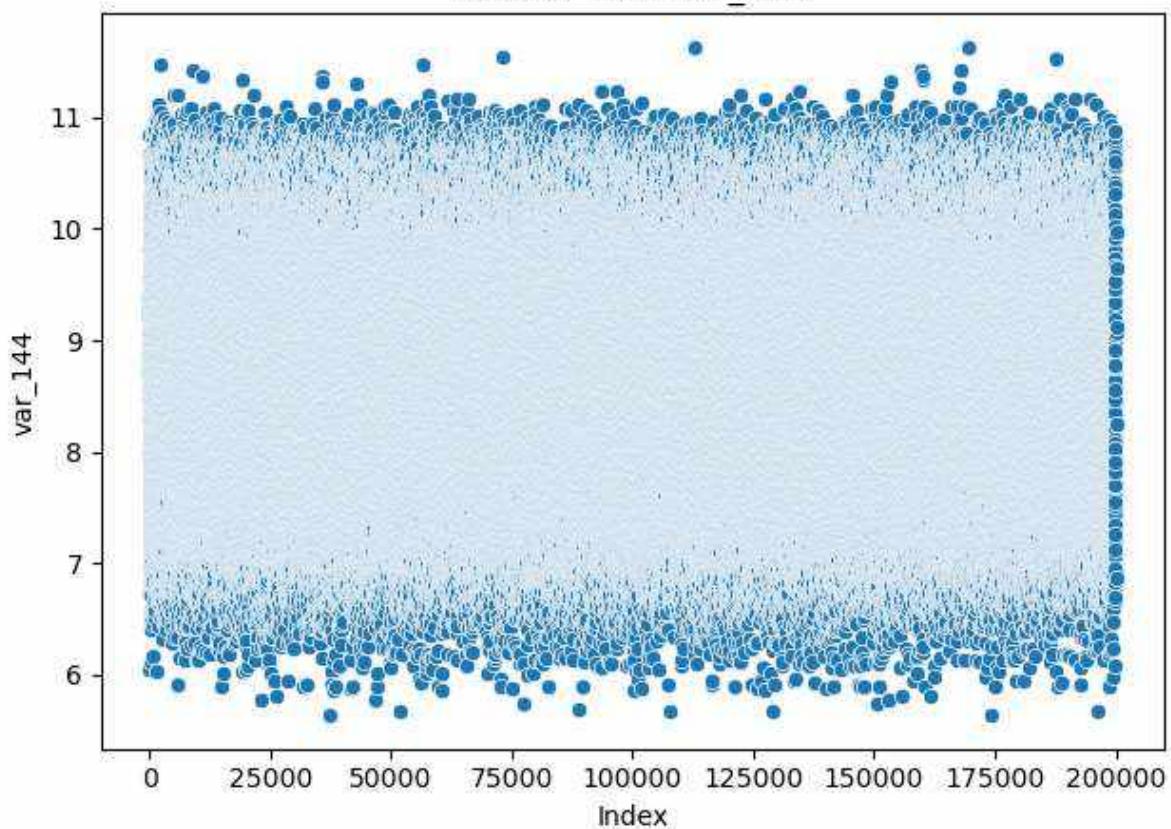
Scatter Plot - var\_142



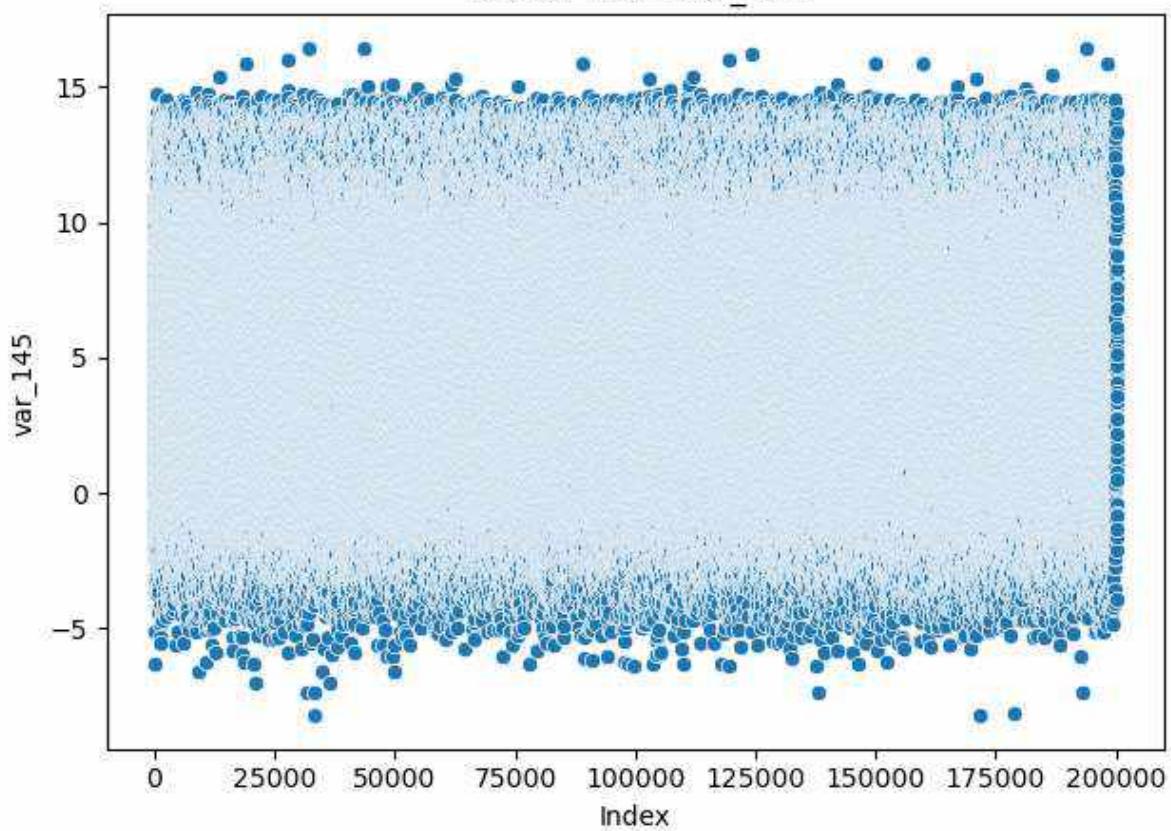
Scatter Plot - var\_143

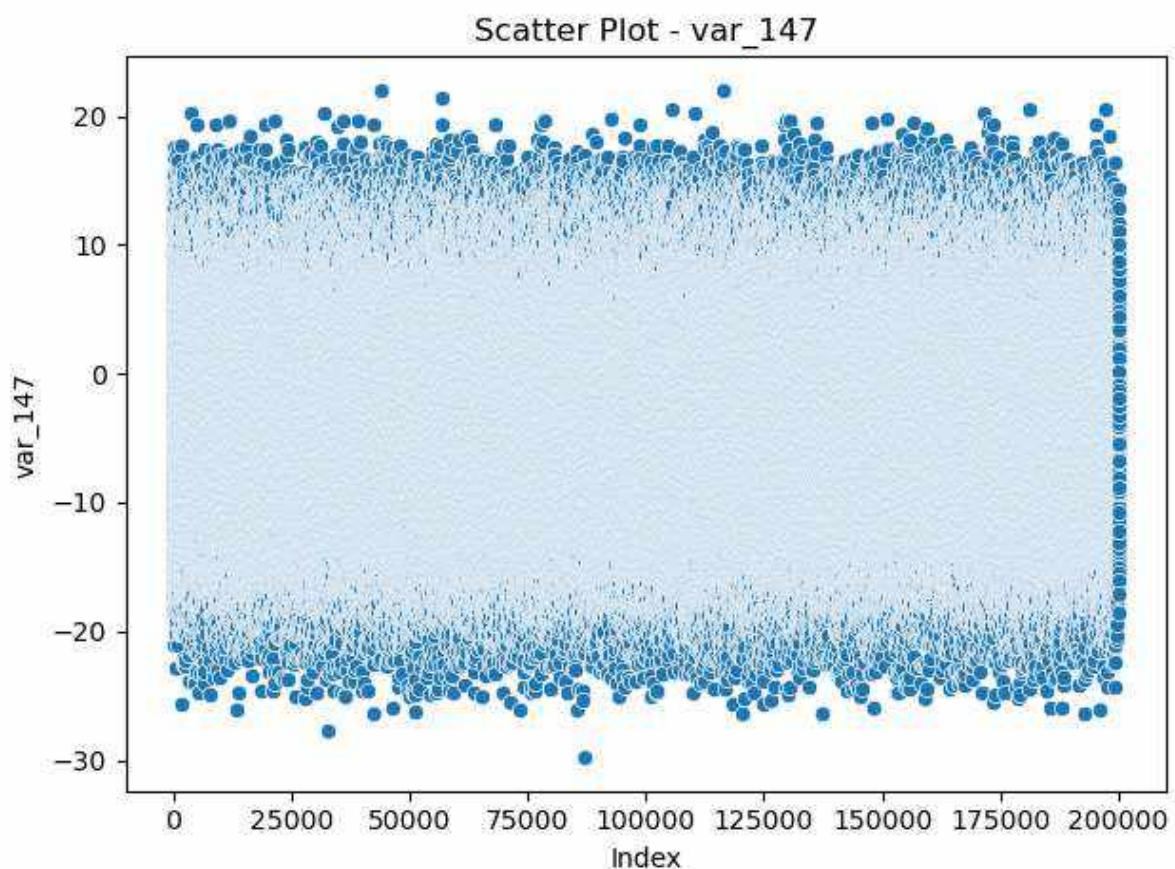
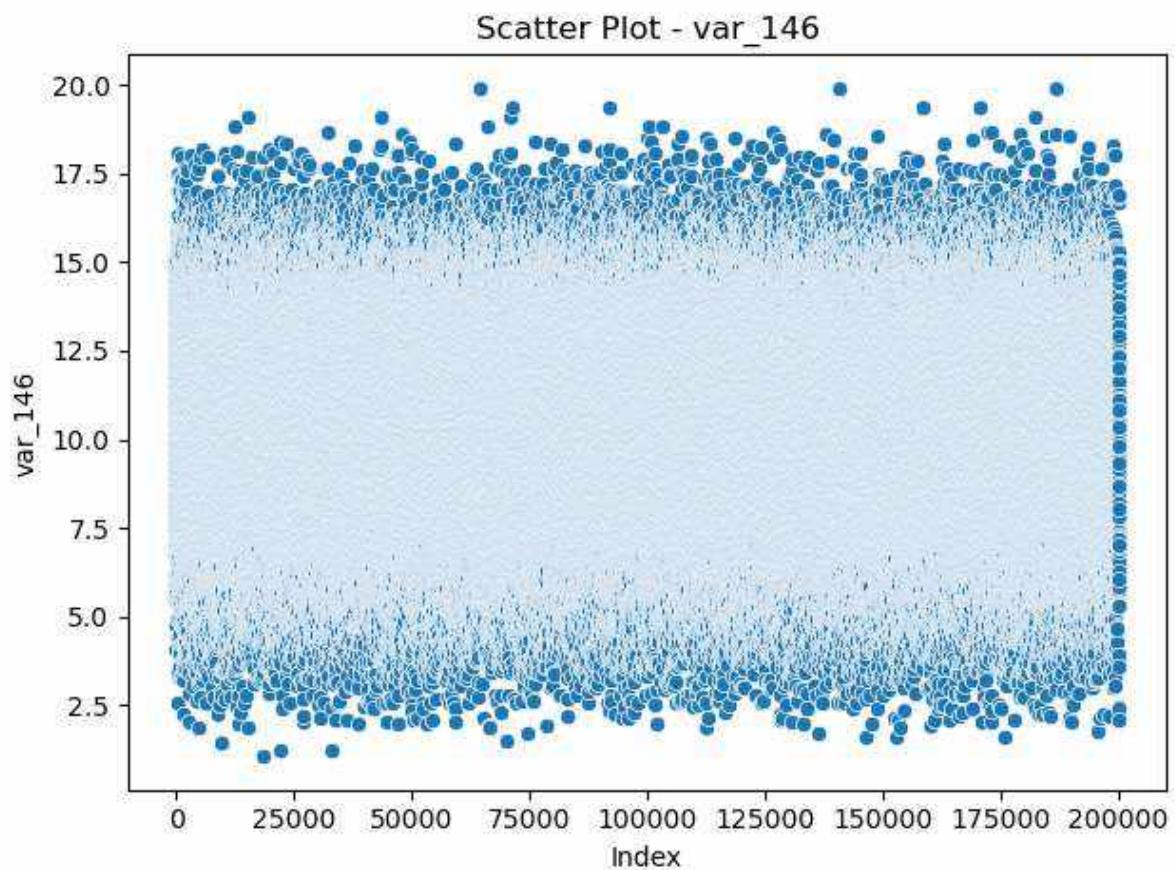


Scatter Plot - var\_144

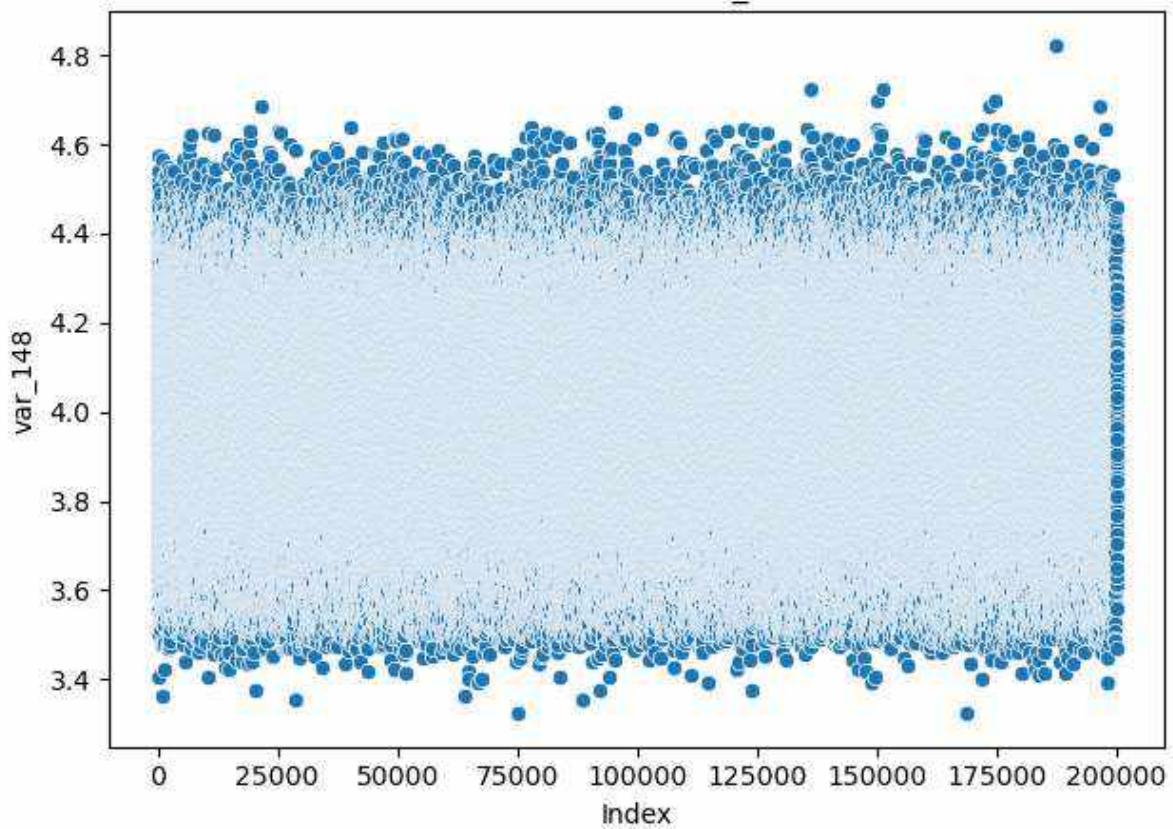


Scatter Plot - var\_145

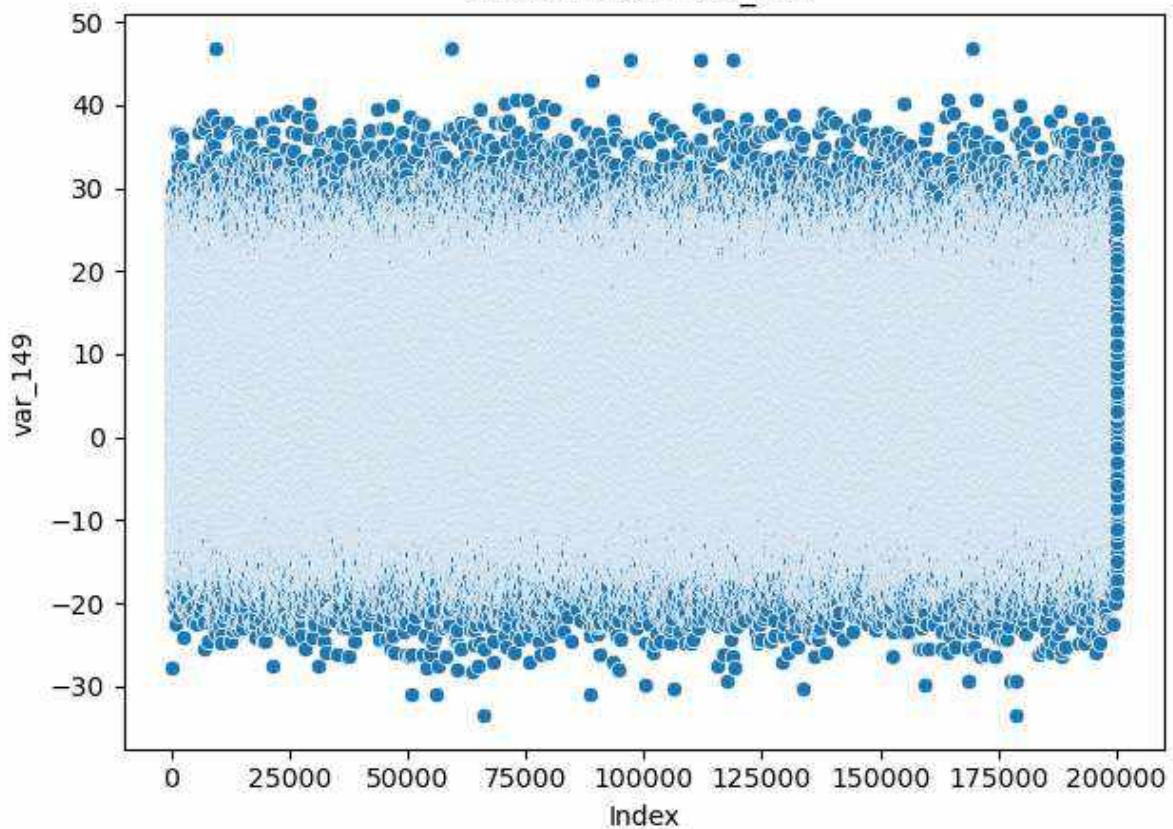




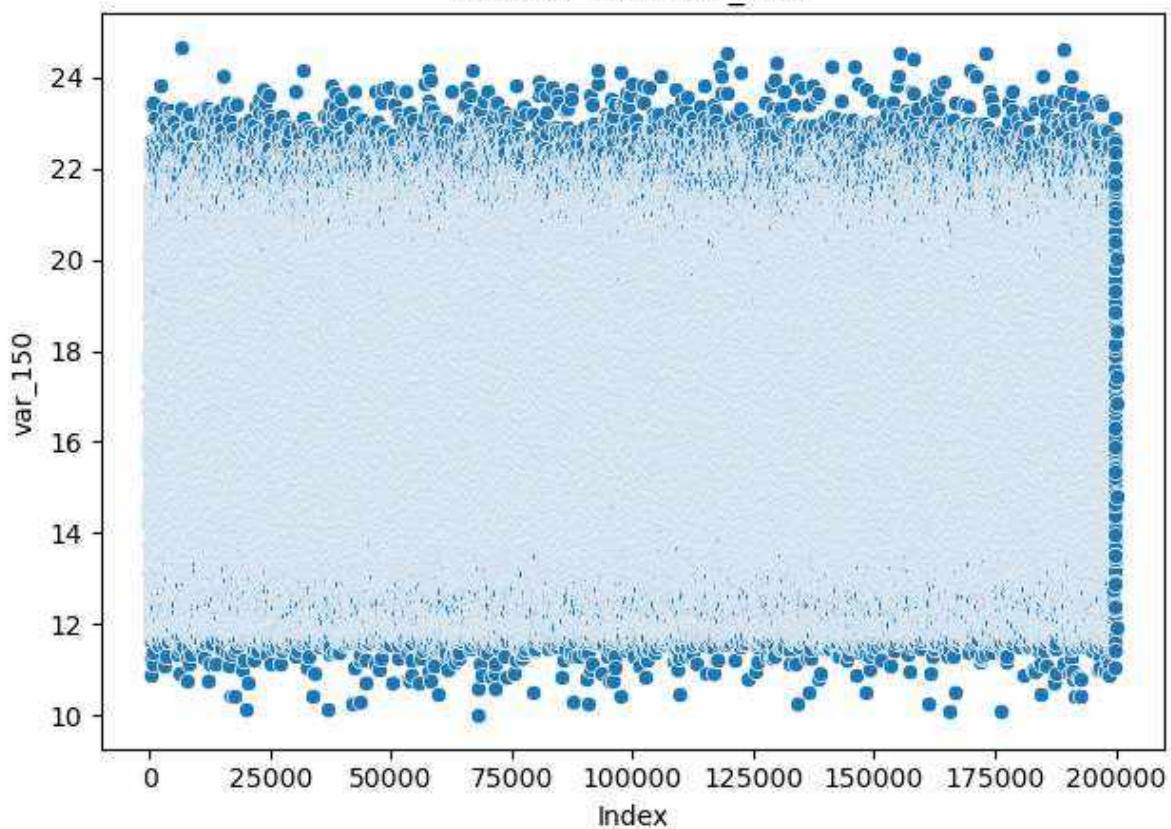
Scatter Plot - var\_148



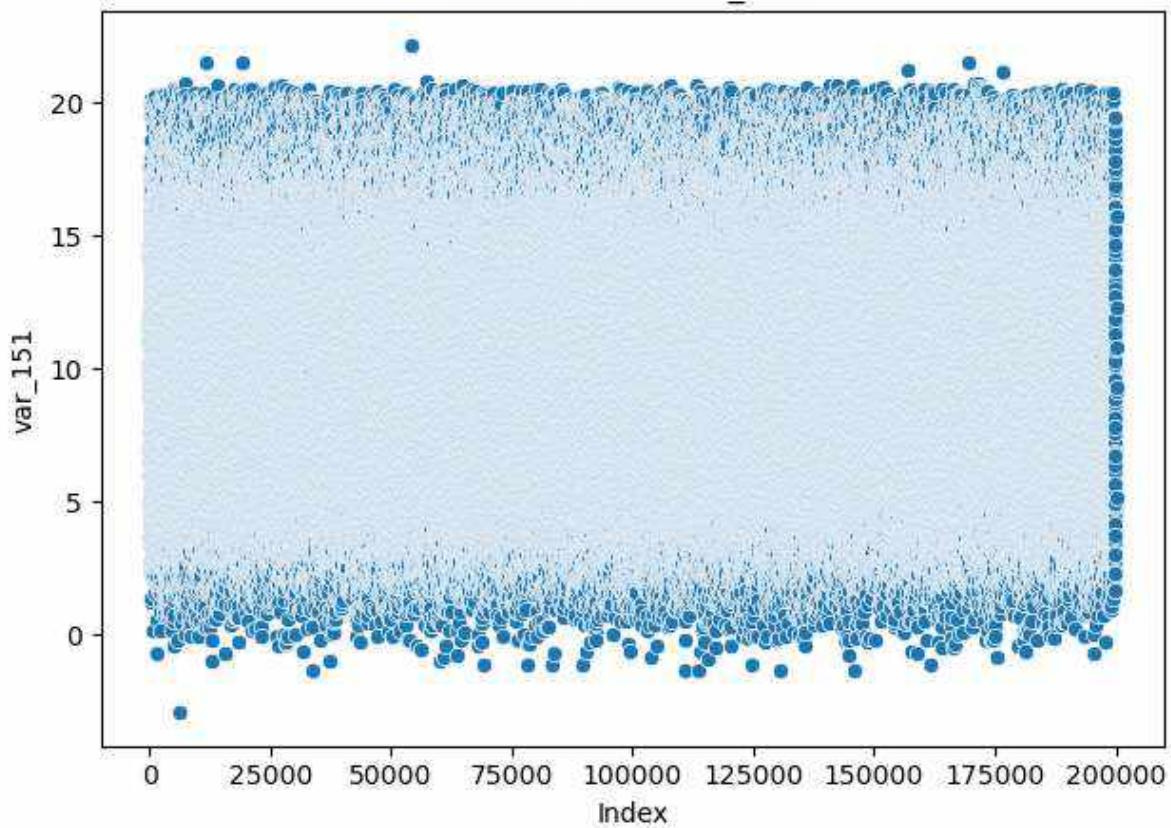
Scatter Plot - var\_149



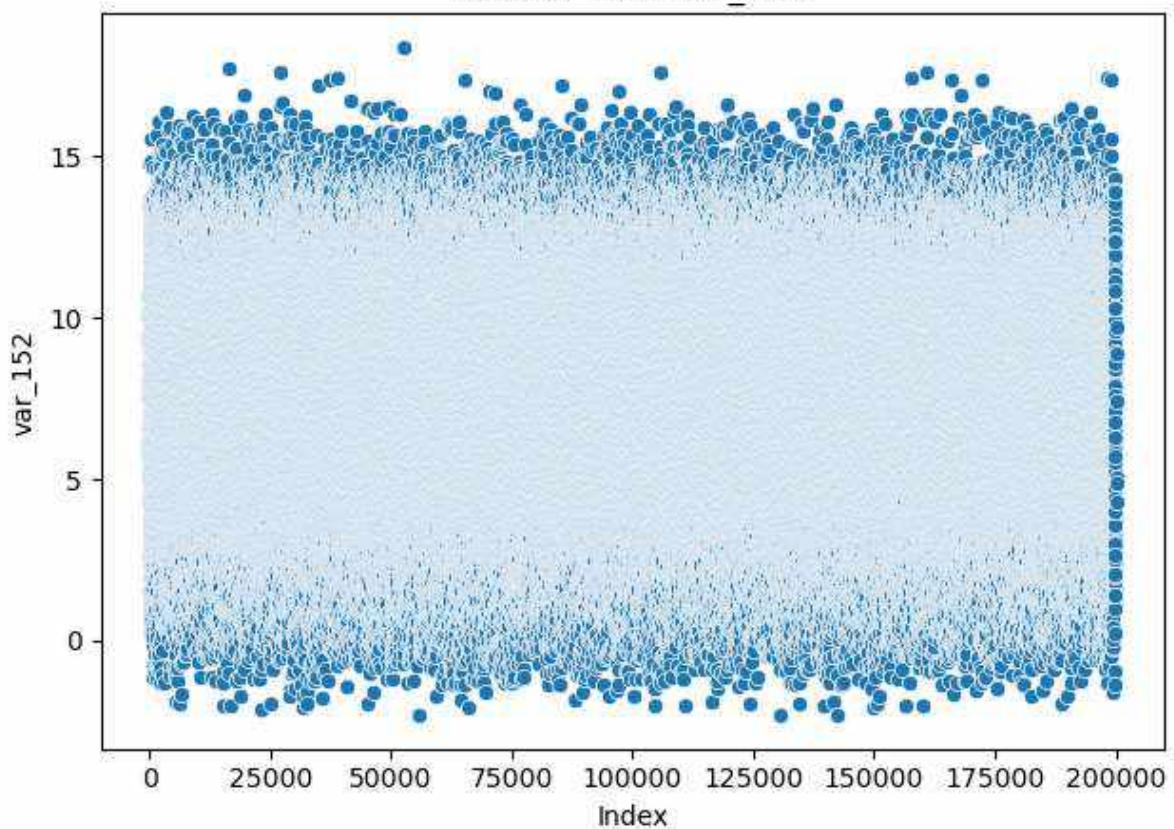
Scatter Plot - var\_150



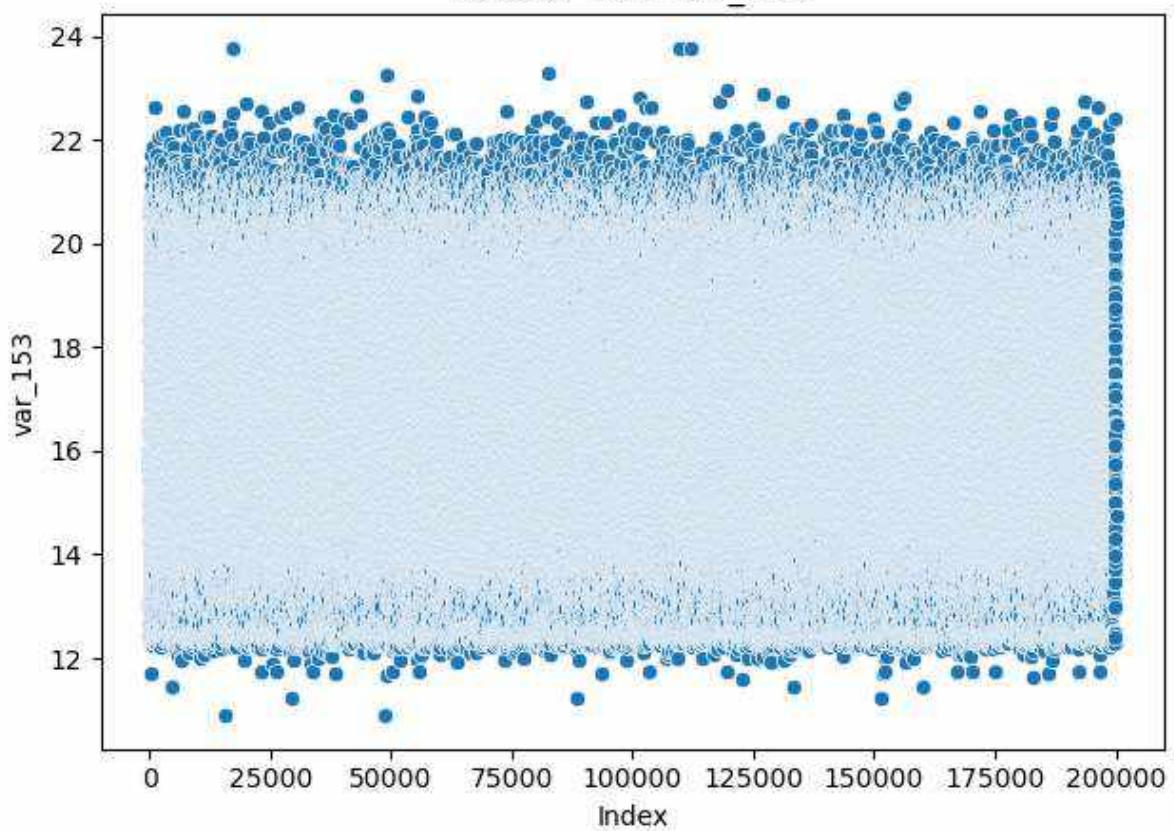
Scatter Plot - var\_151



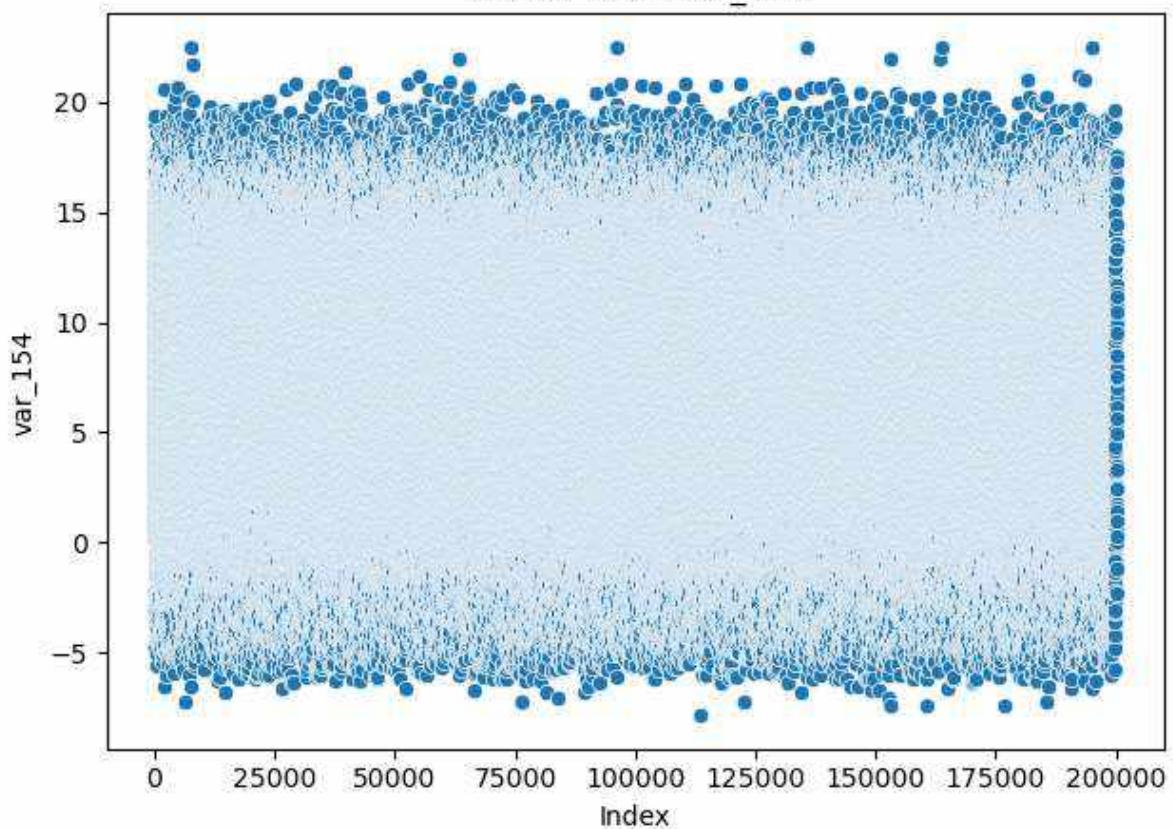
Scatter Plot - var\_152



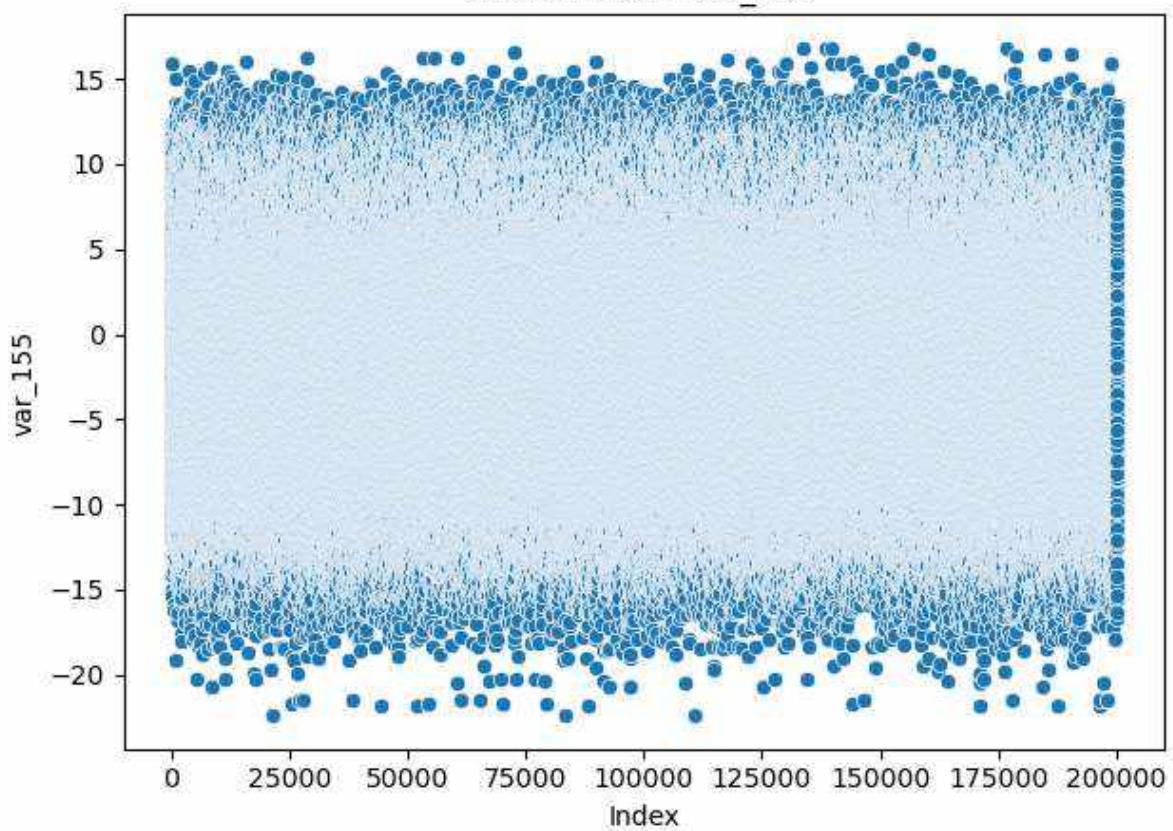
Scatter Plot - var\_153



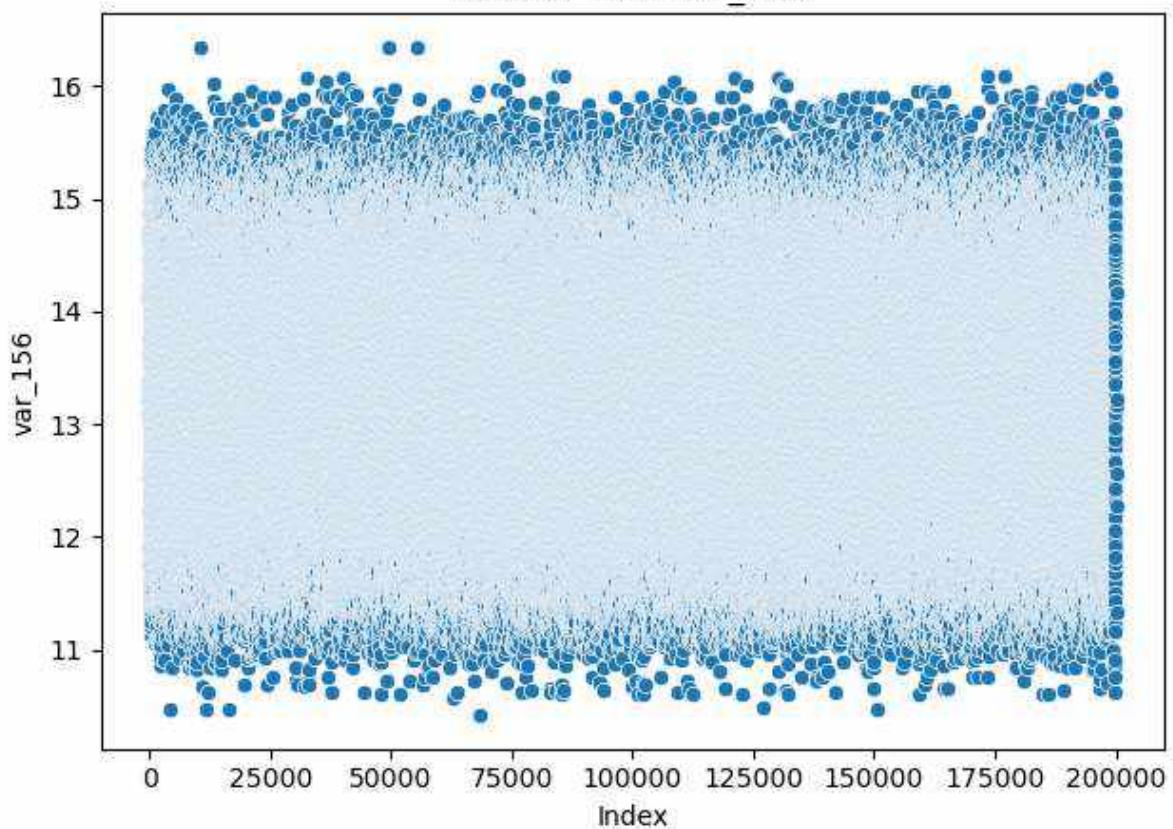
Scatter Plot - var\_154



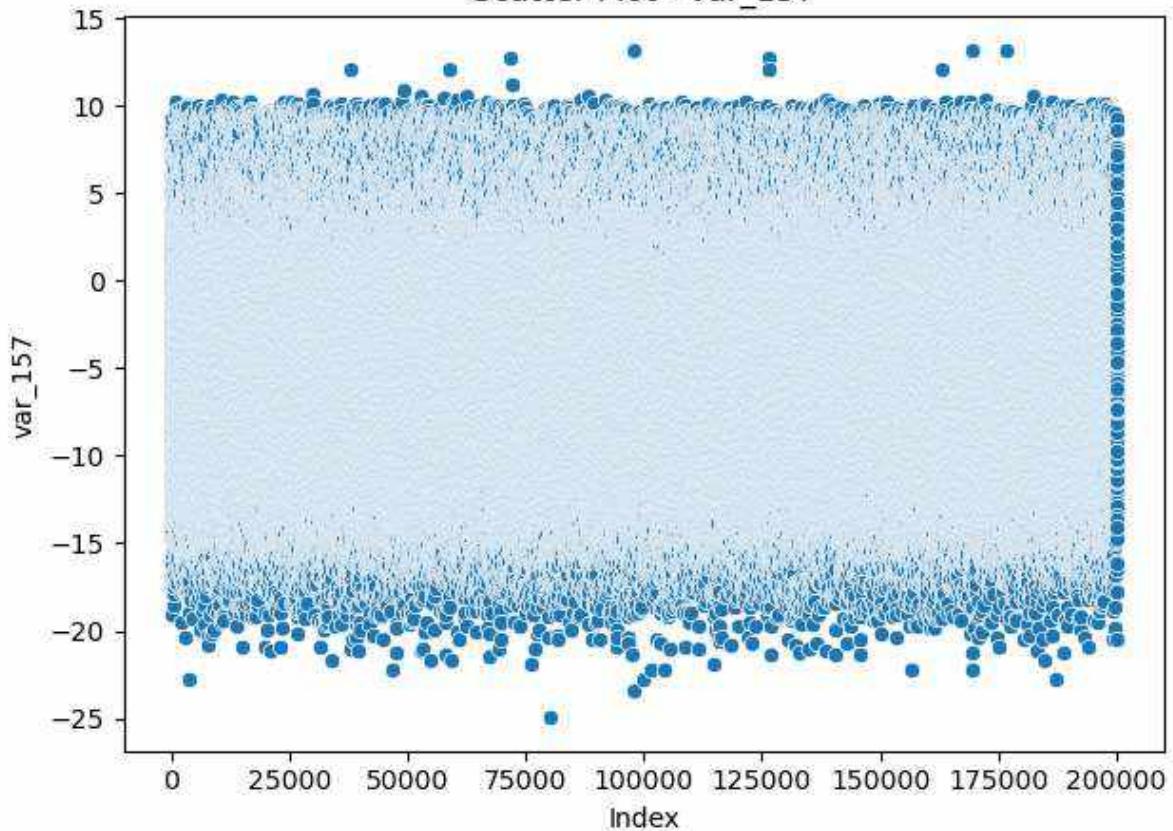
Scatter Plot - var\_155



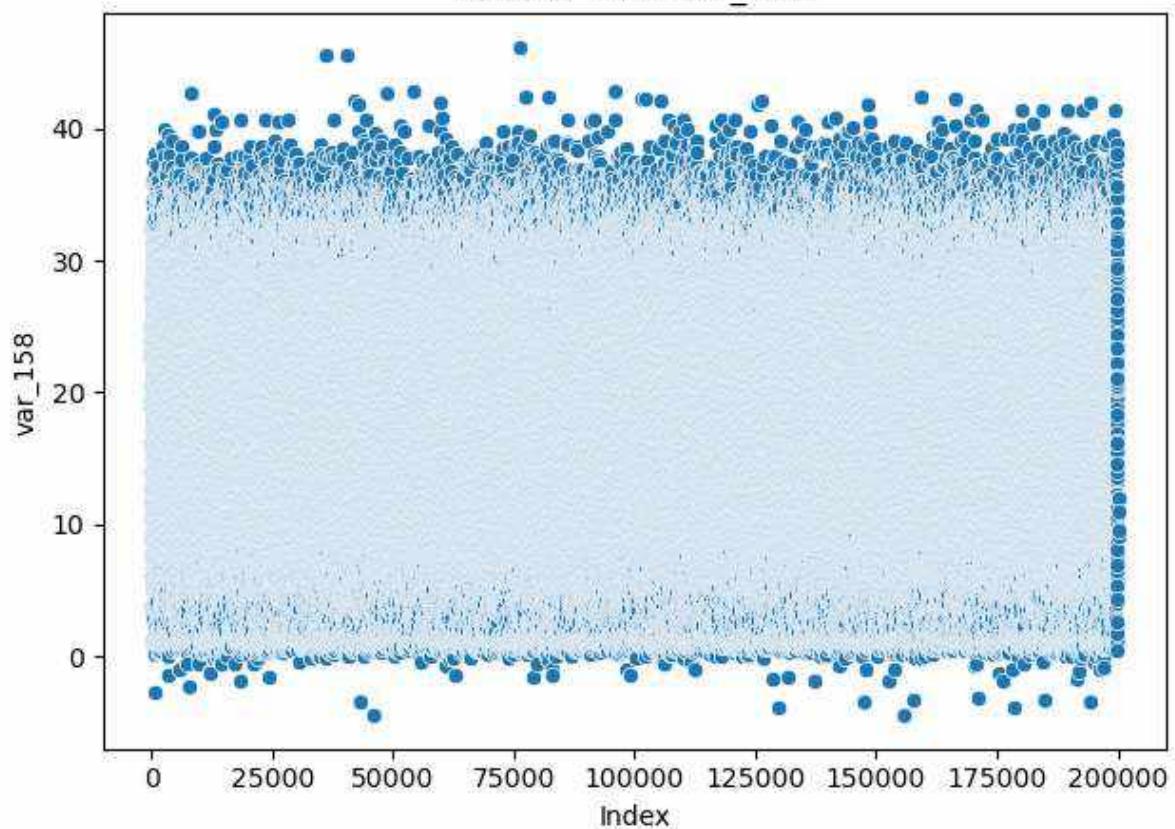
Scatter Plot - var\_156



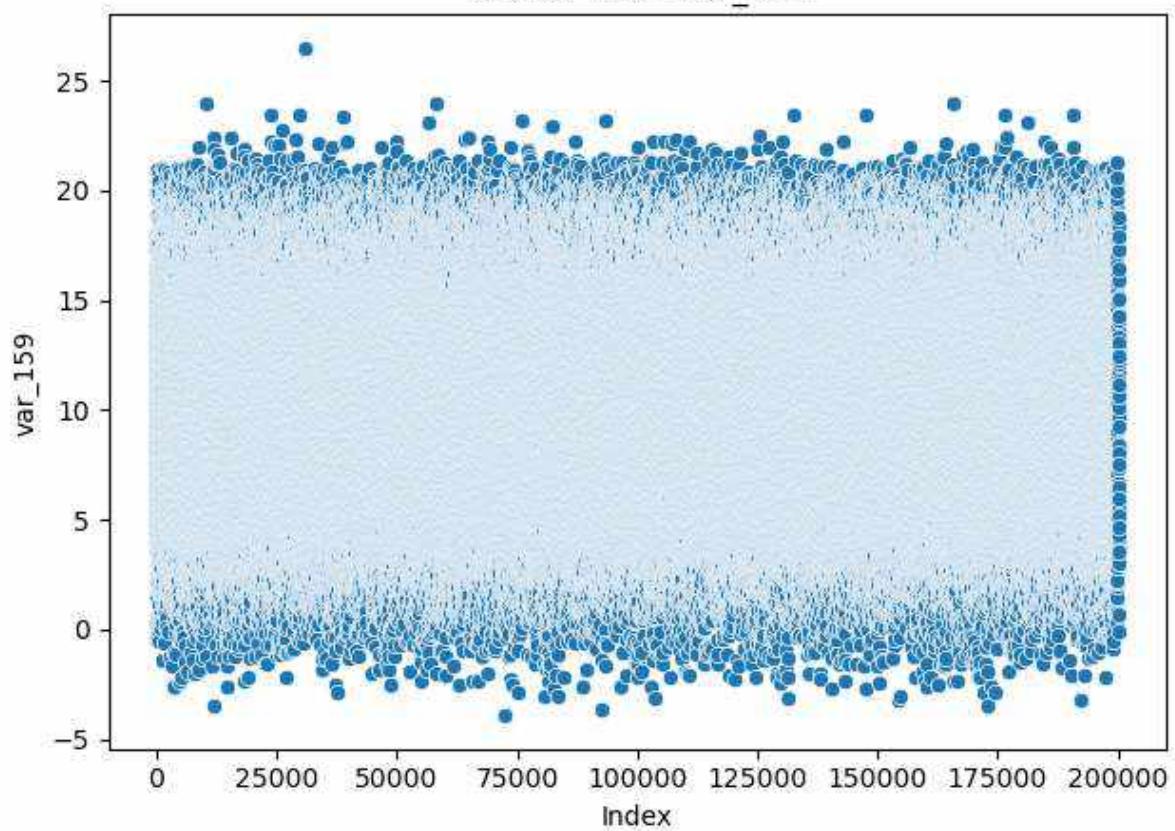
Scatter Plot - var\_157



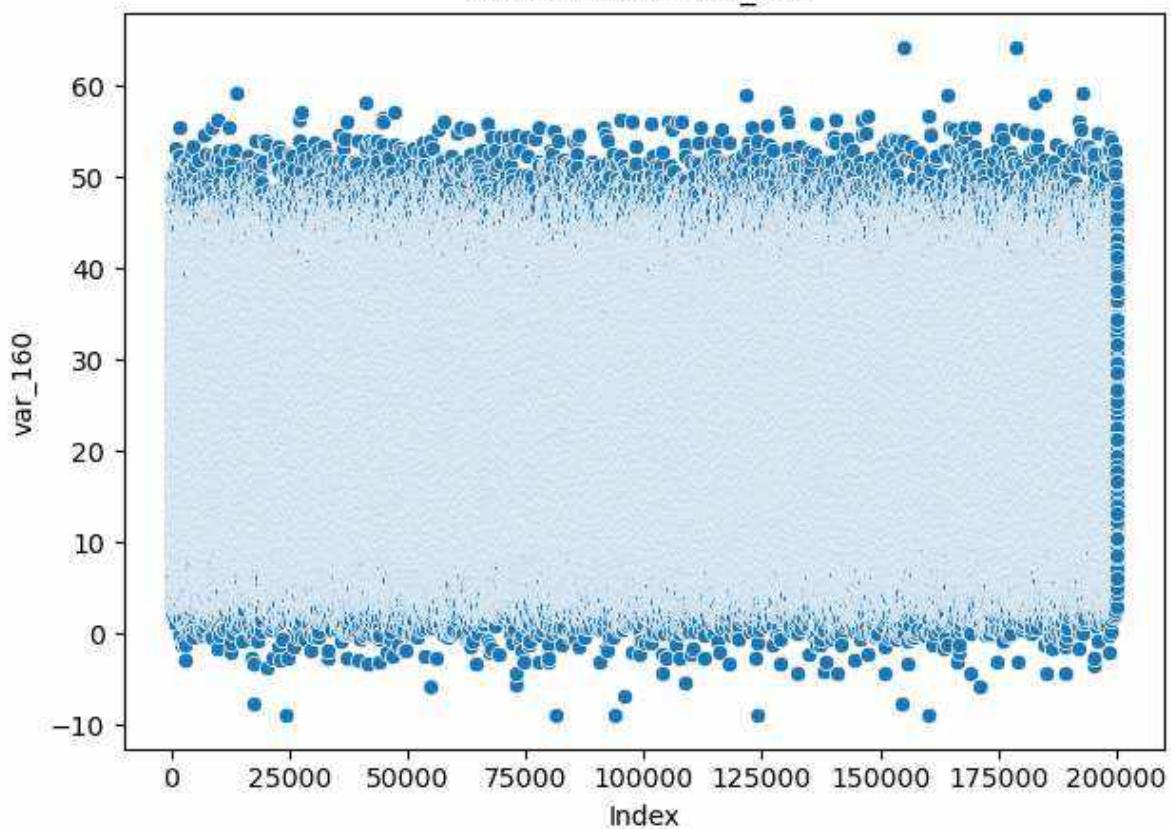
Scatter Plot - var\_158



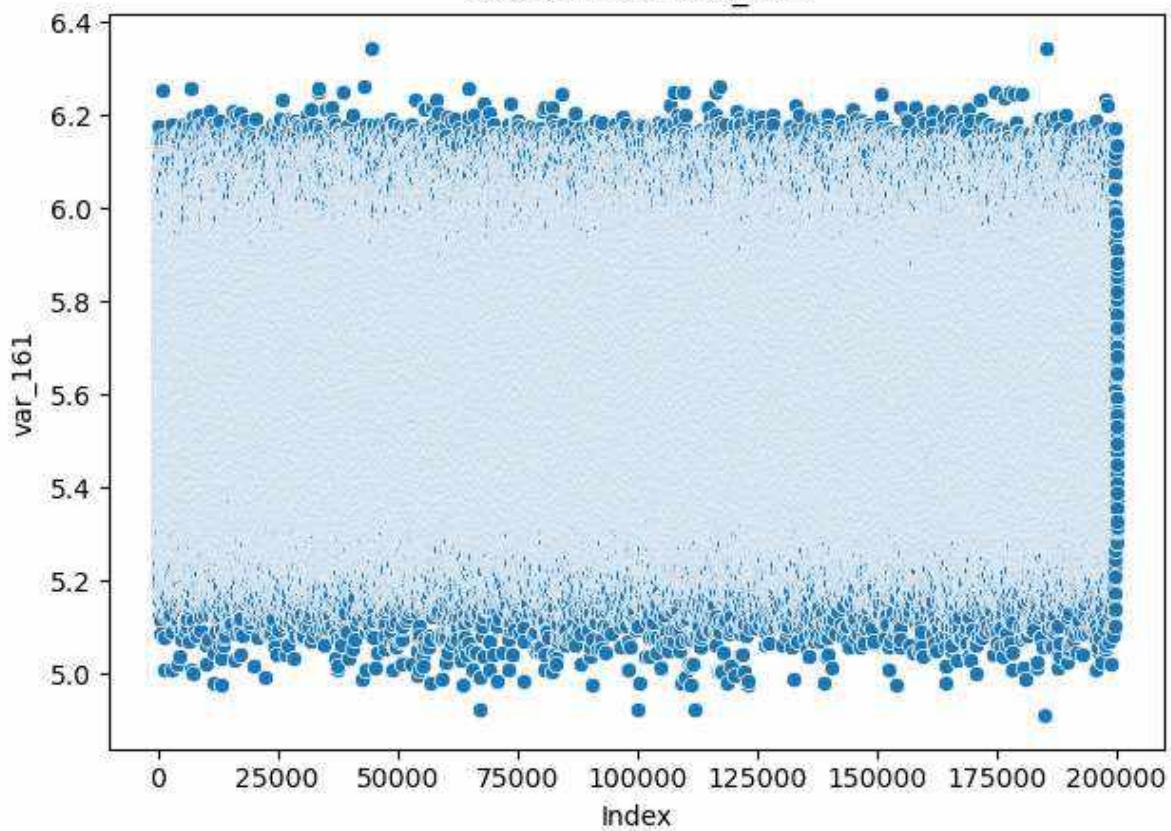
Scatter Plot - var\_159



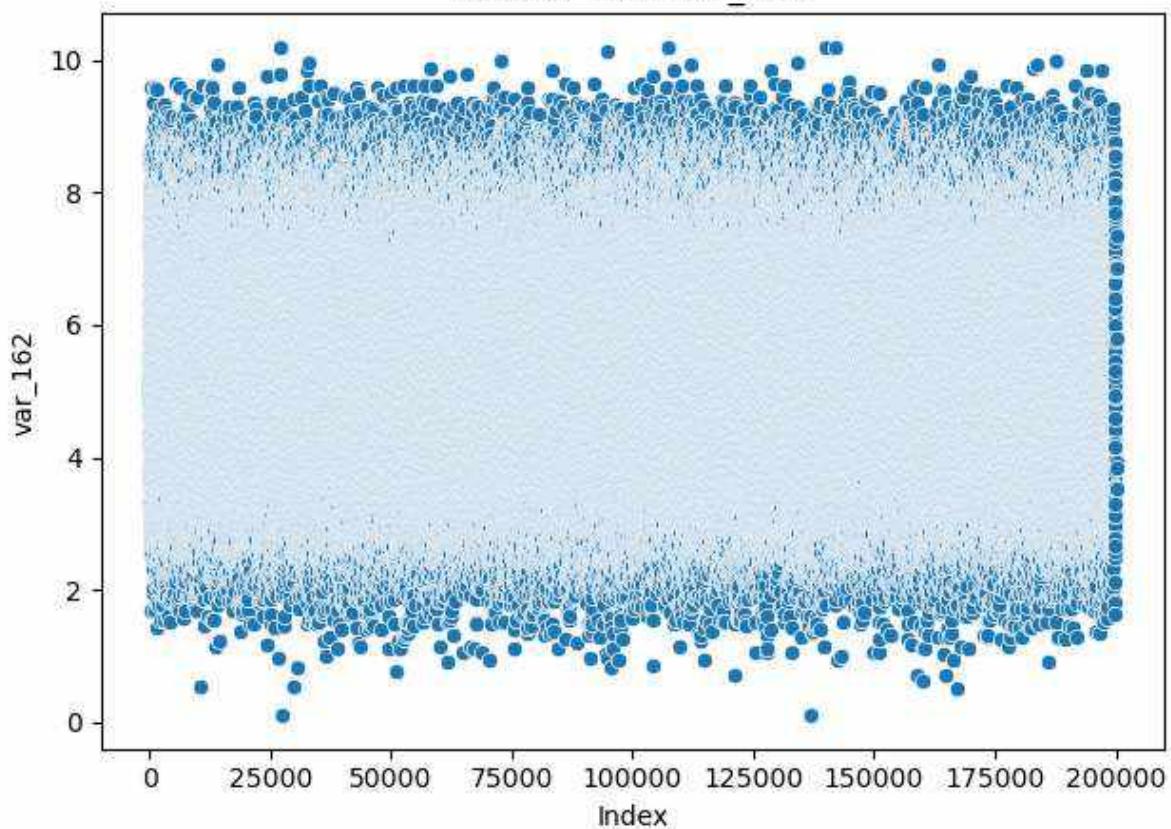
Scatter Plot - var\_160



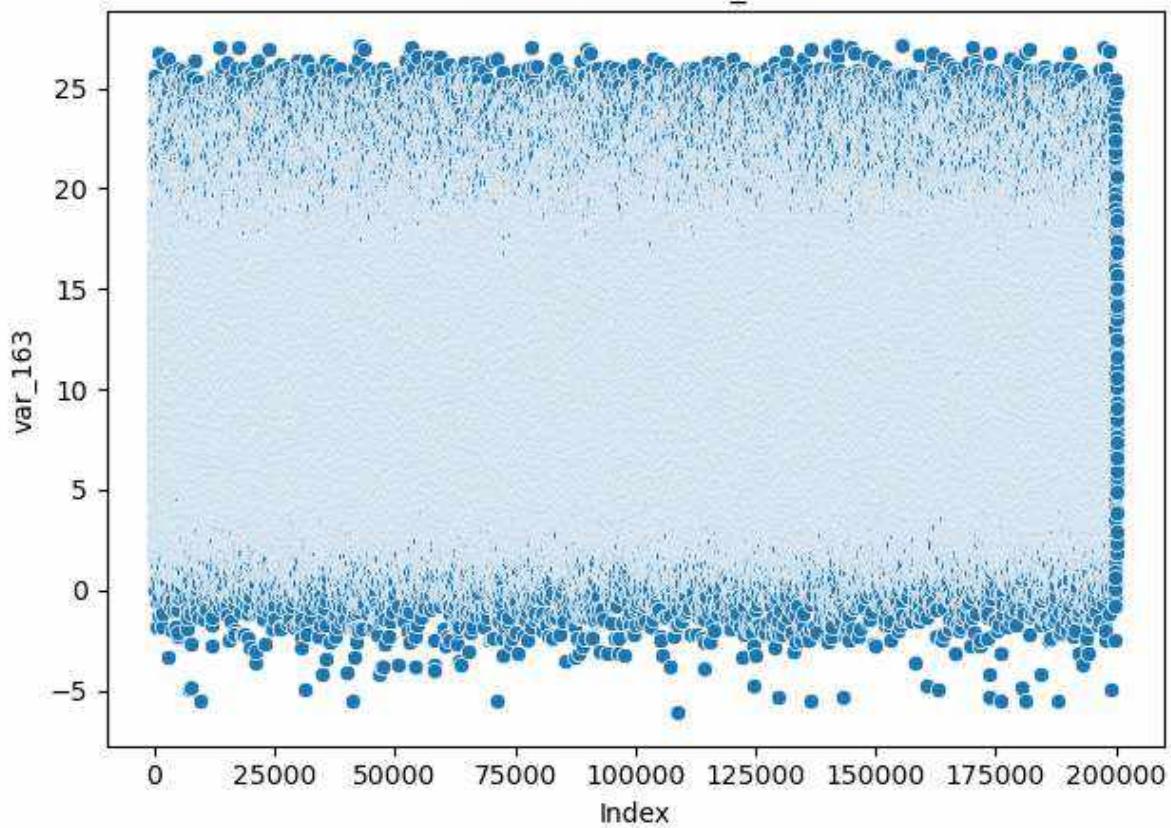
Scatter Plot - var\_161



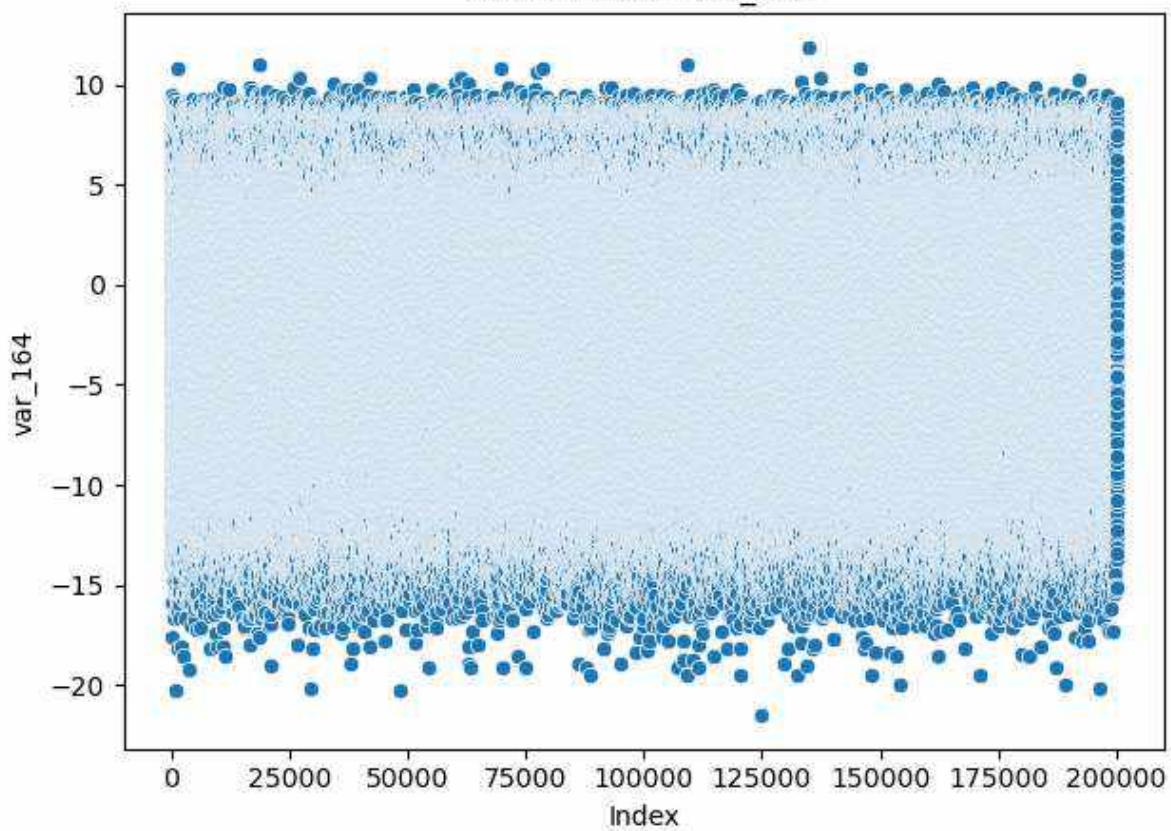
Scatter Plot - var\_162



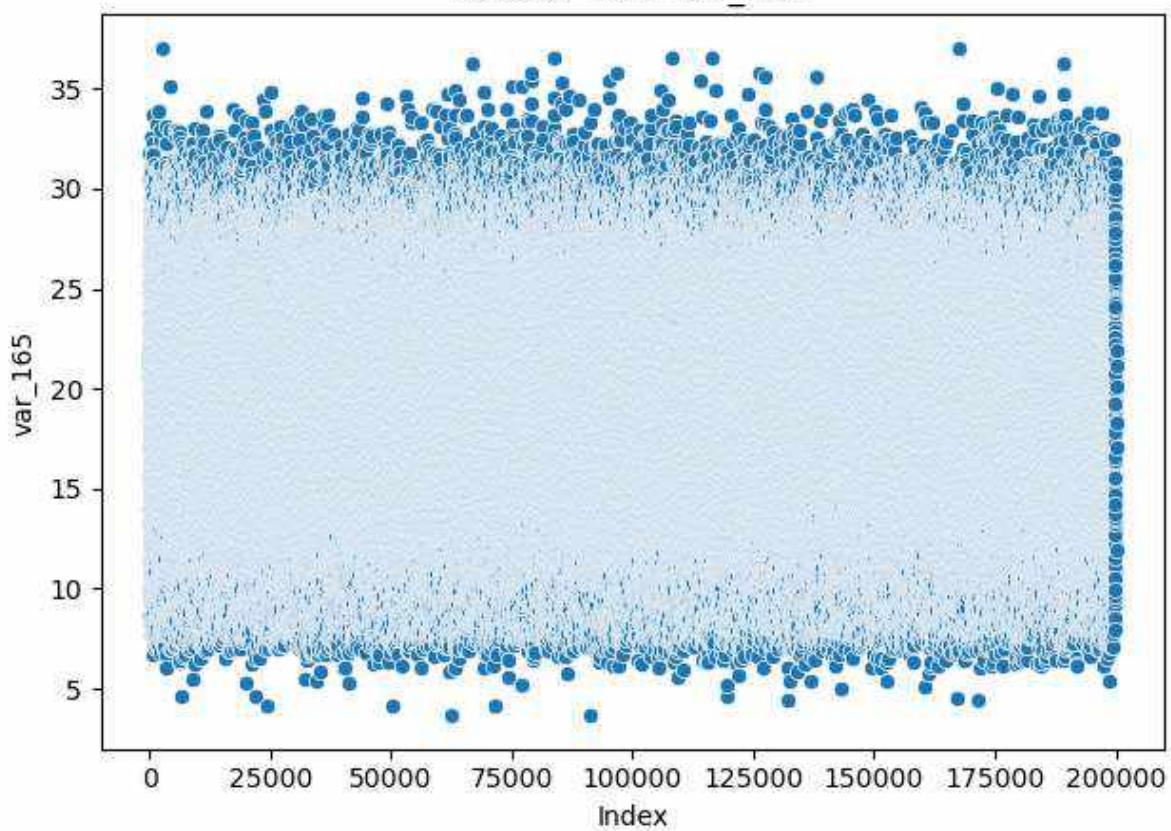
Scatter Plot - var\_163



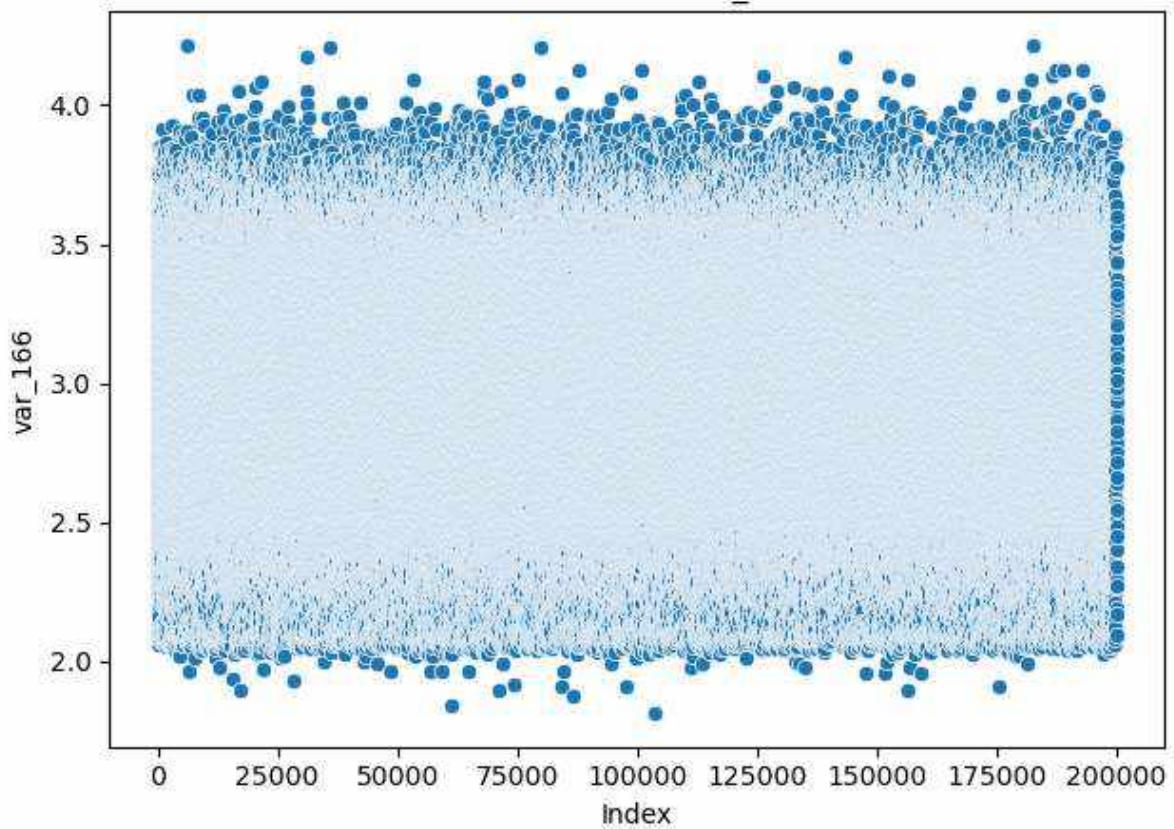
Scatter Plot - var\_164



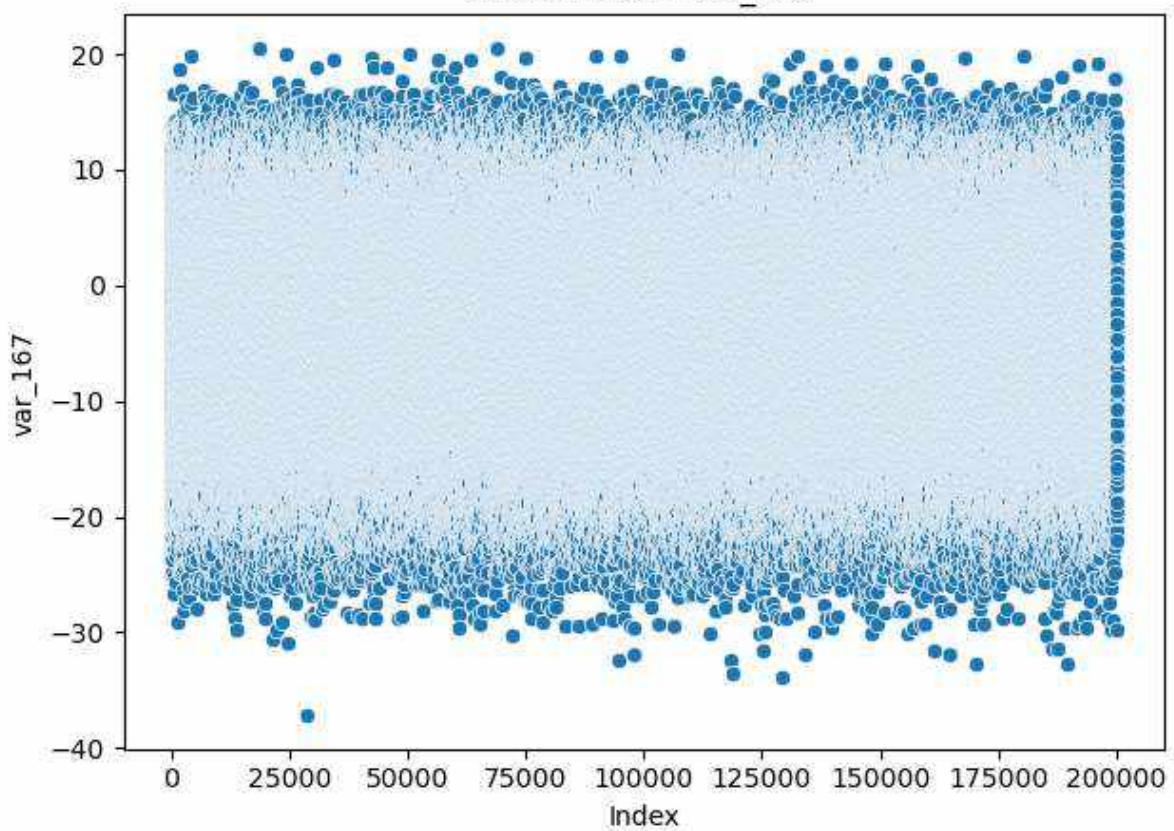
Scatter Plot - var\_165

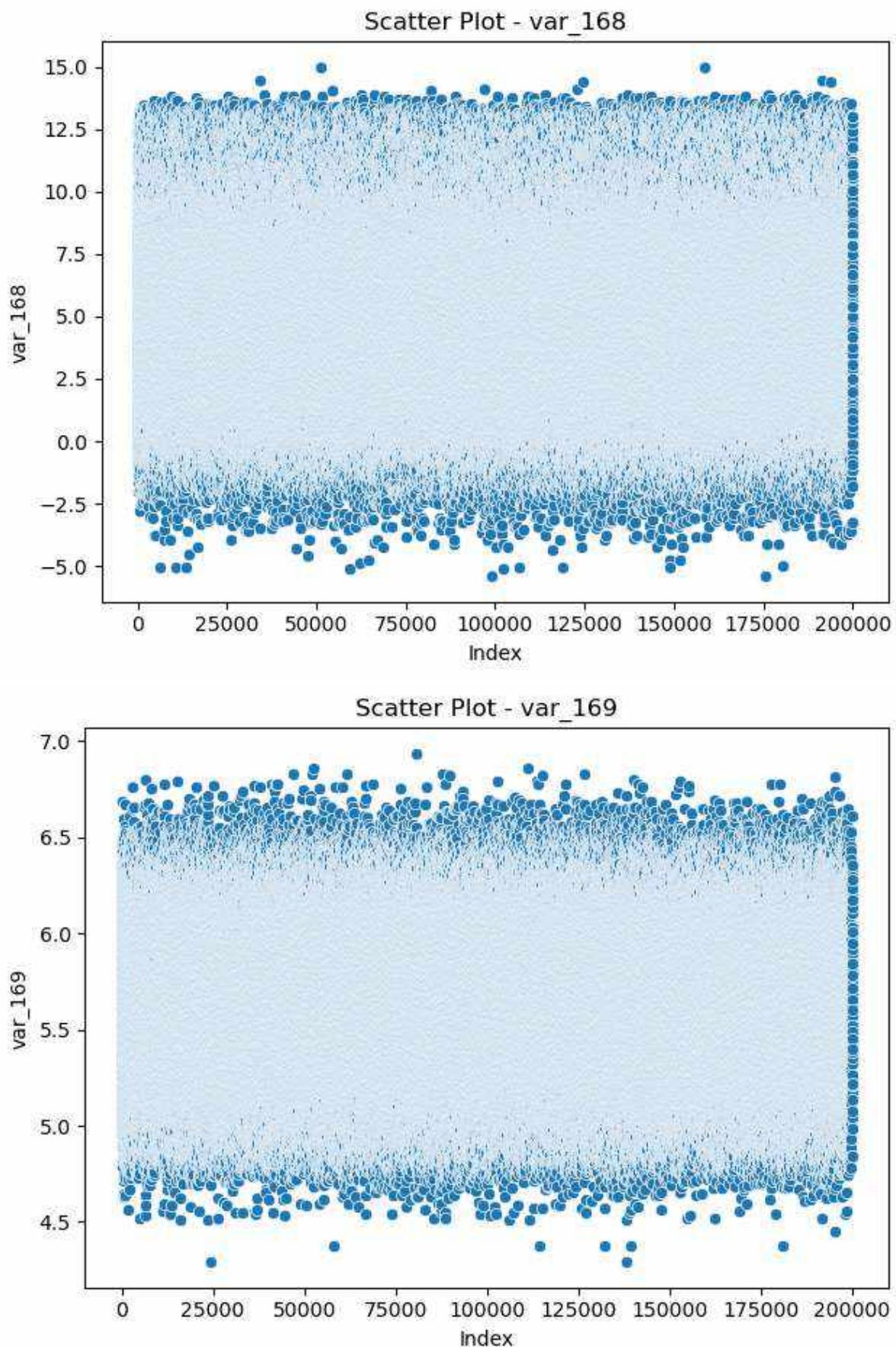


Scatter Plot - var\_166

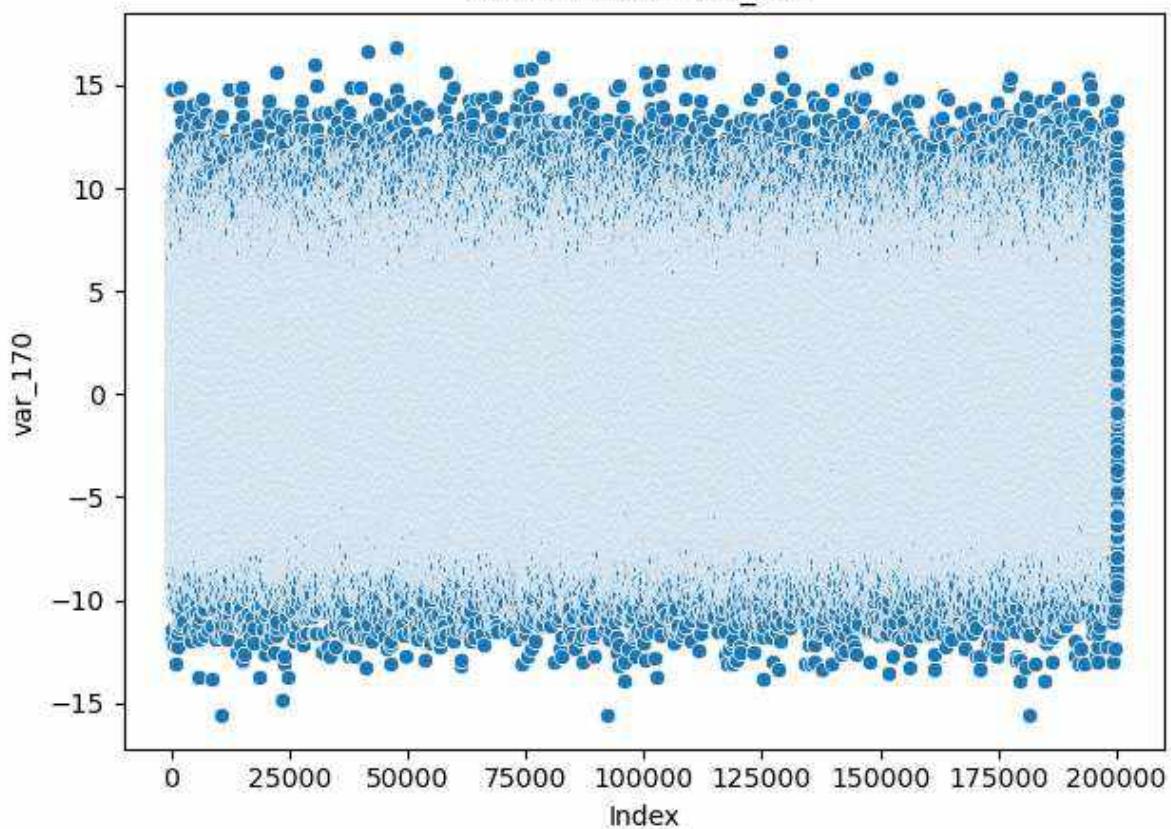


Scatter Plot - var\_167

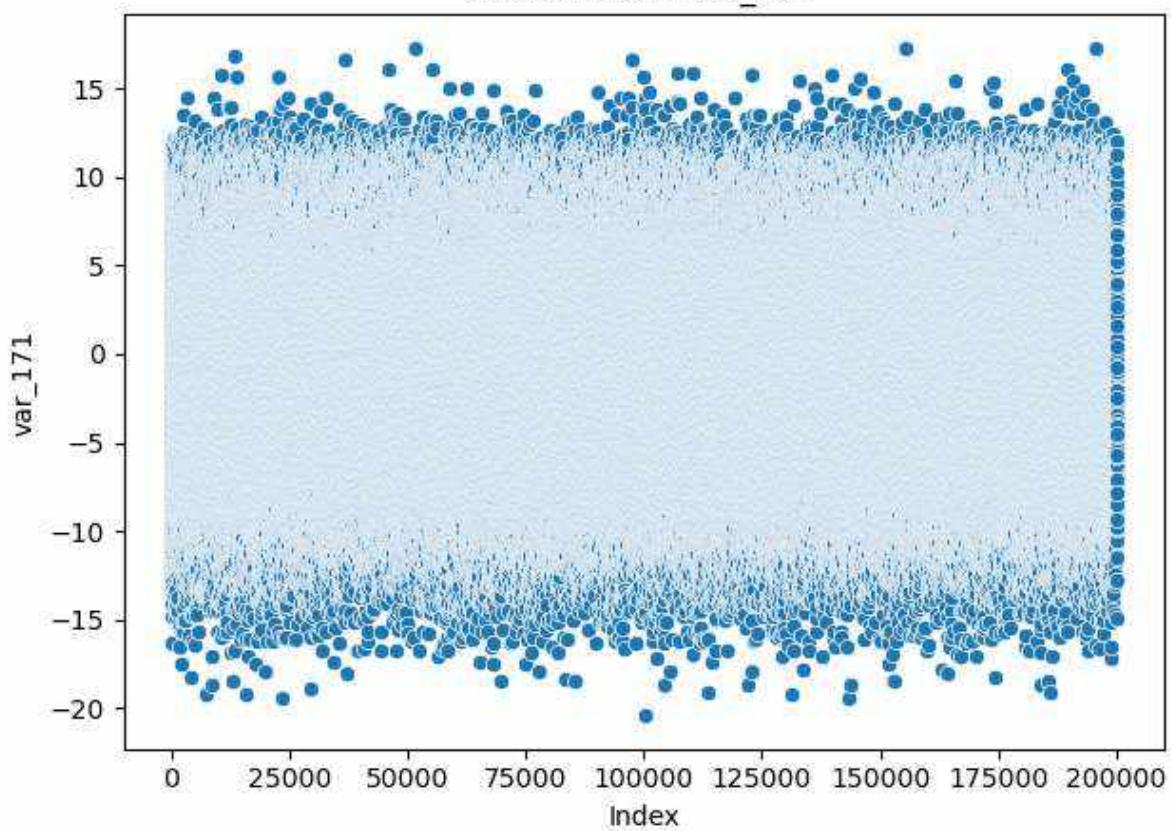




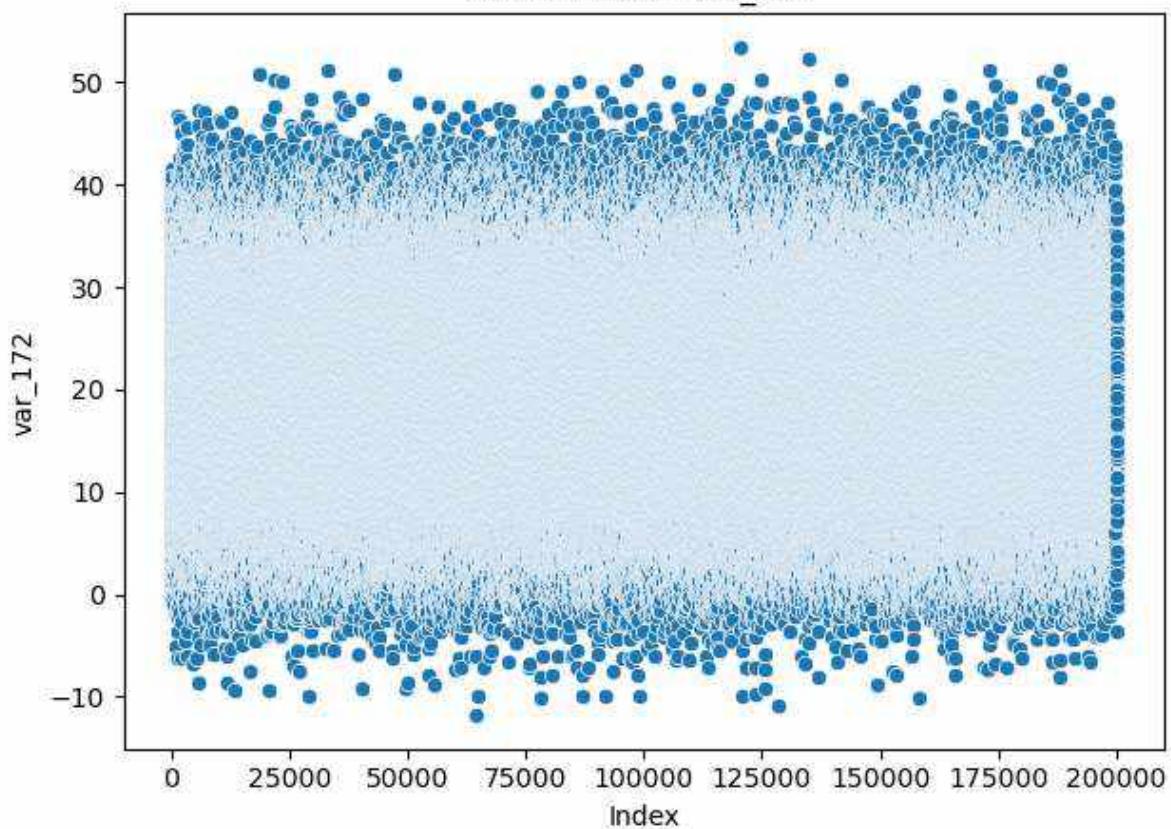
Scatter Plot - var\_170



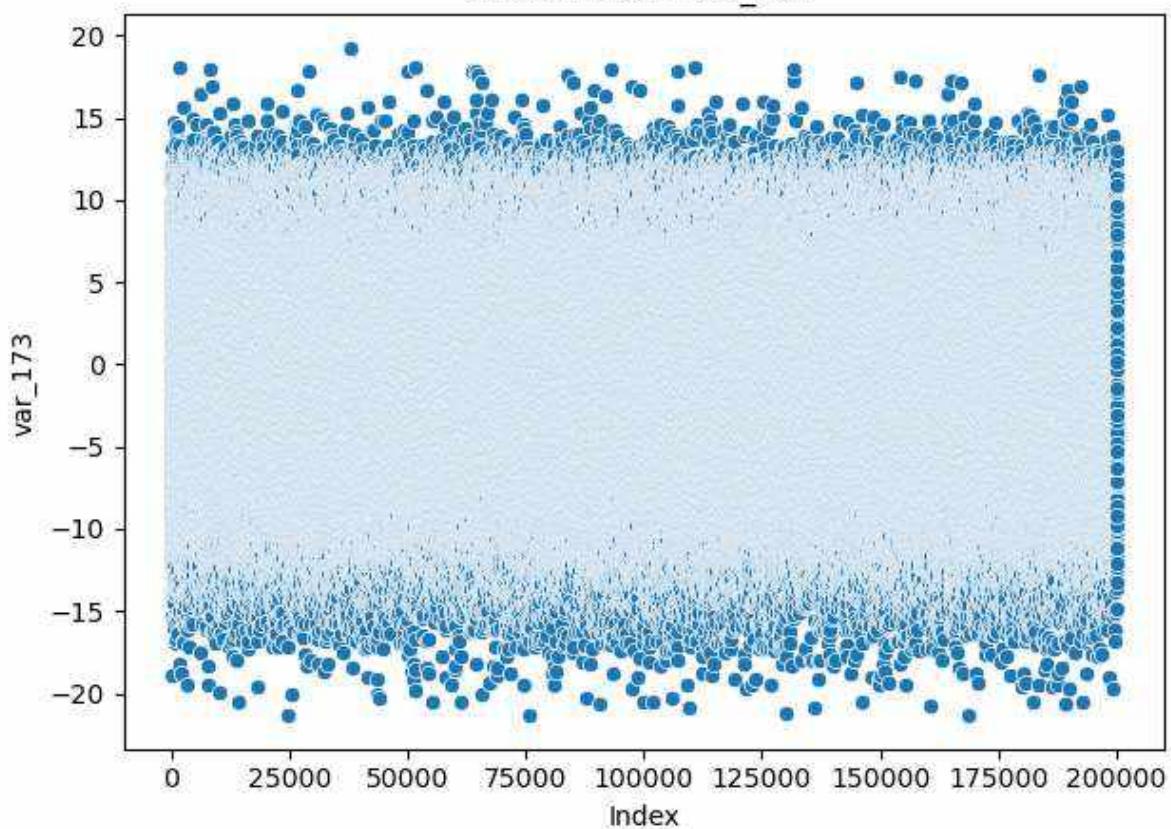
Scatter Plot - var\_171



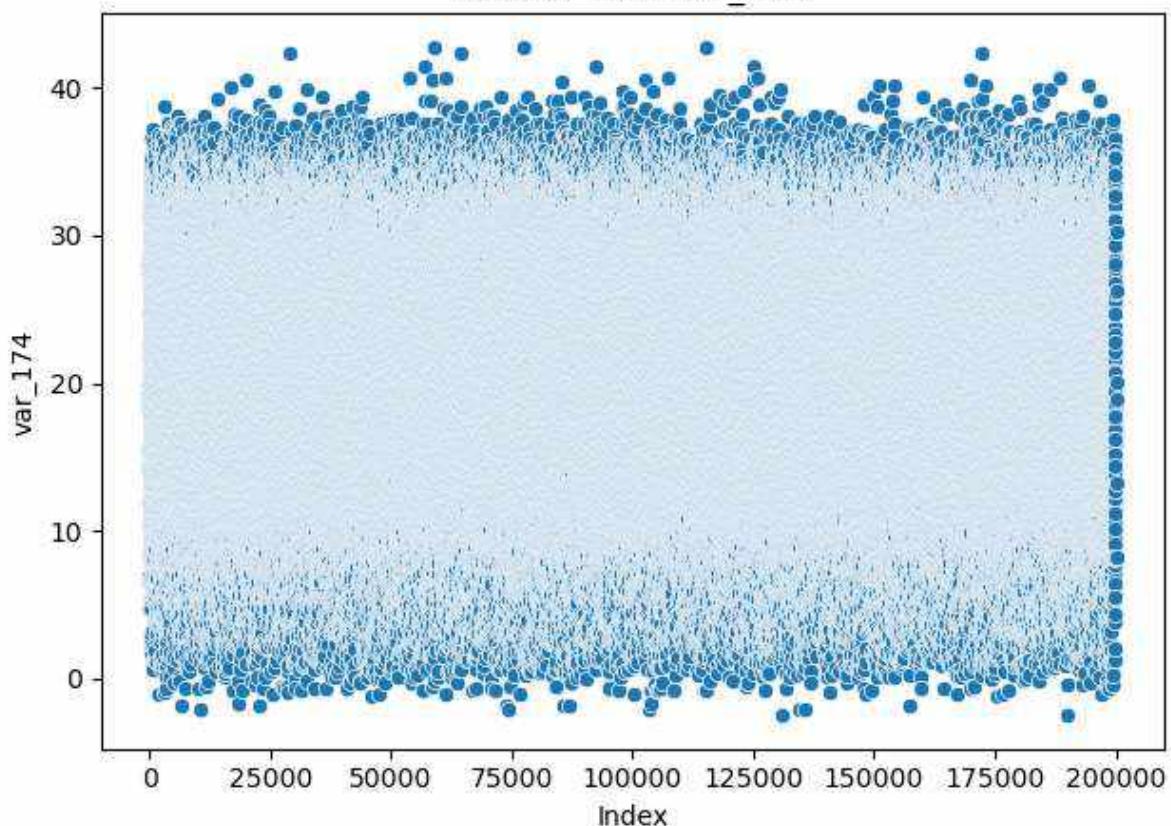
Scatter Plot - var\_172



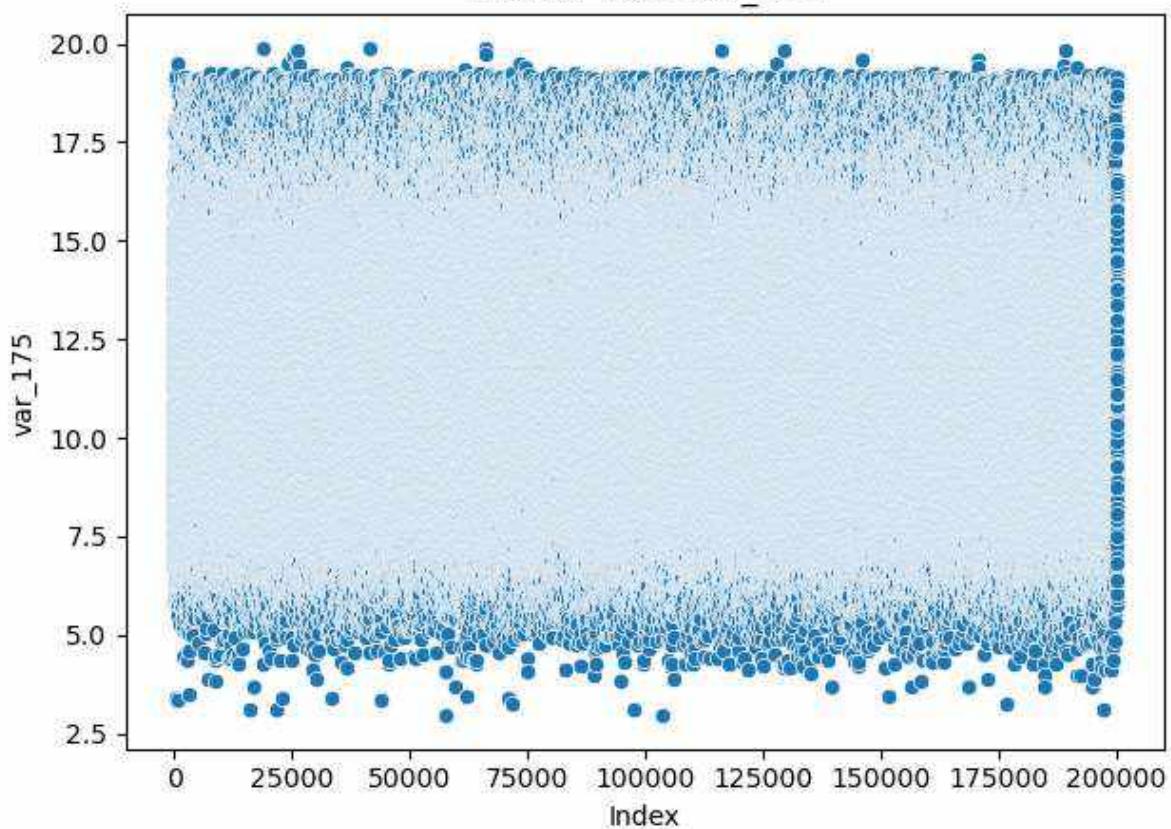
Scatter Plot - var\_173

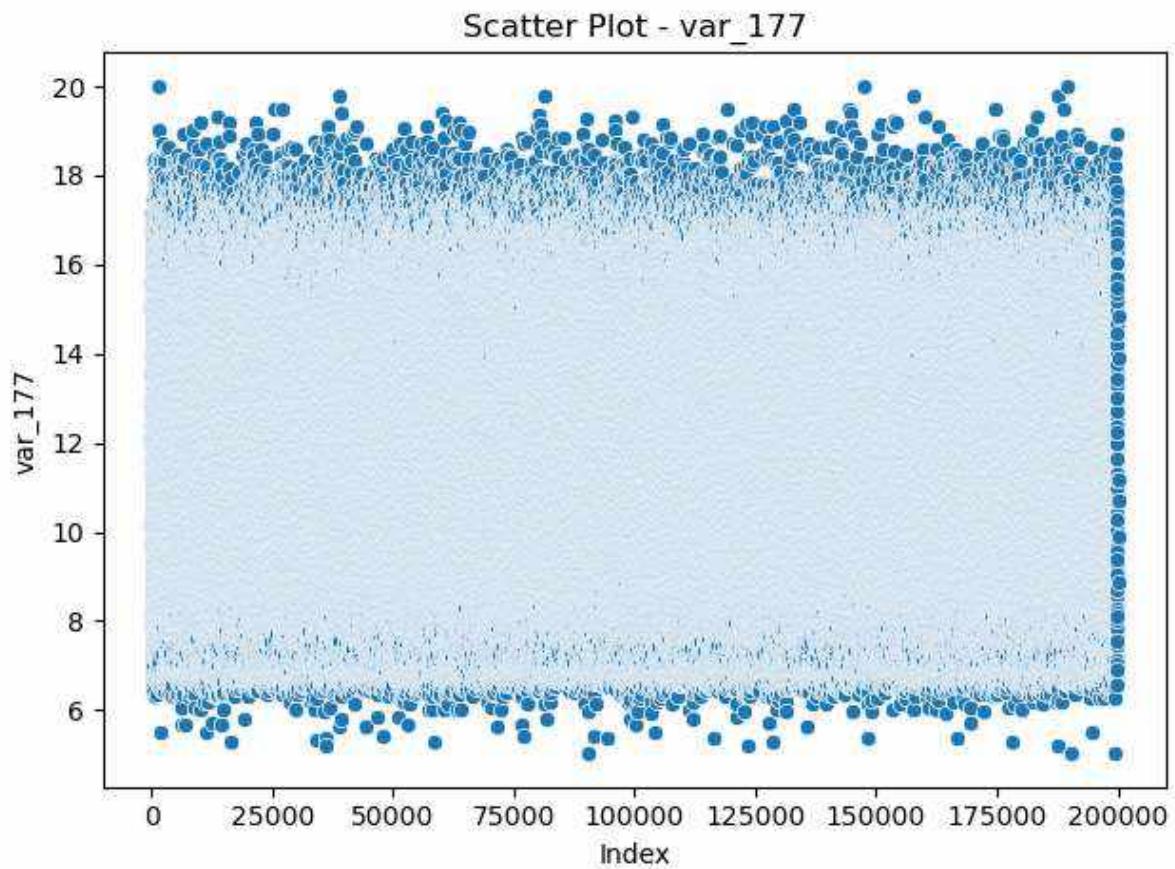
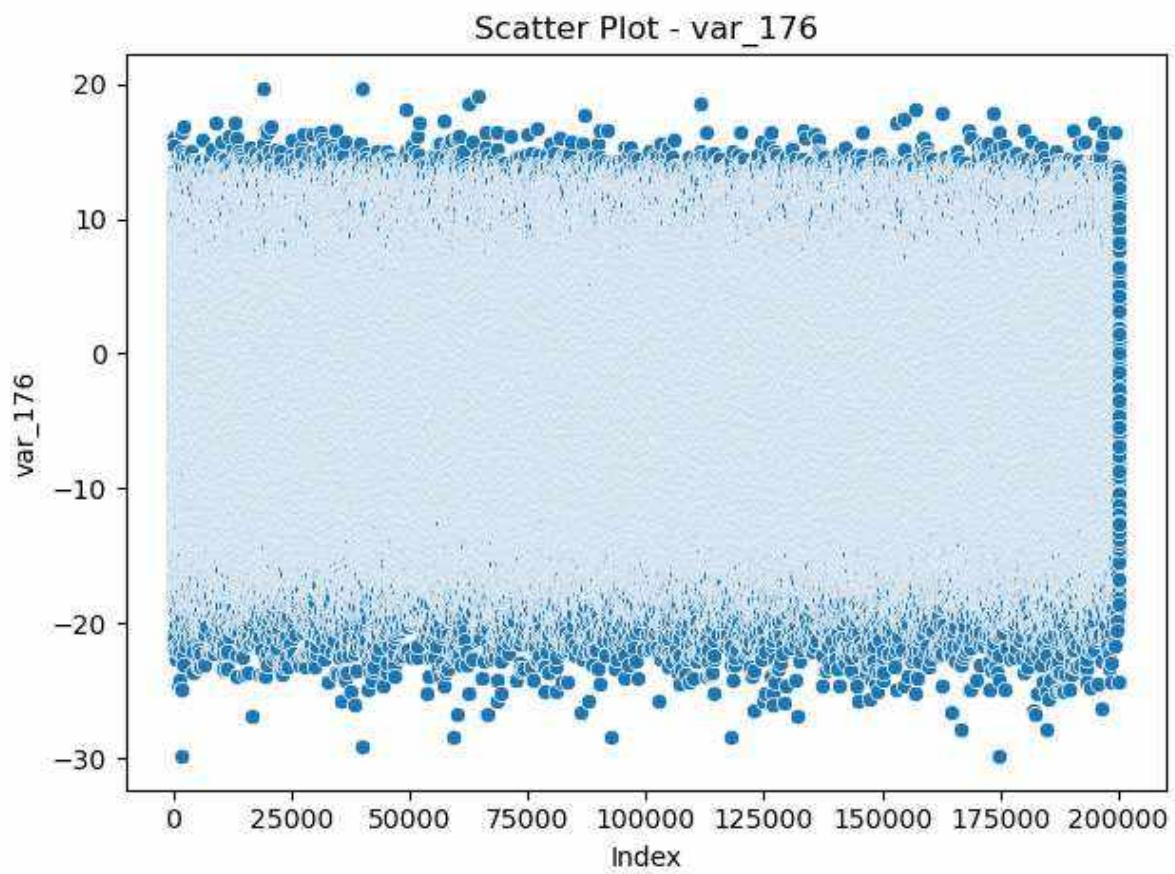


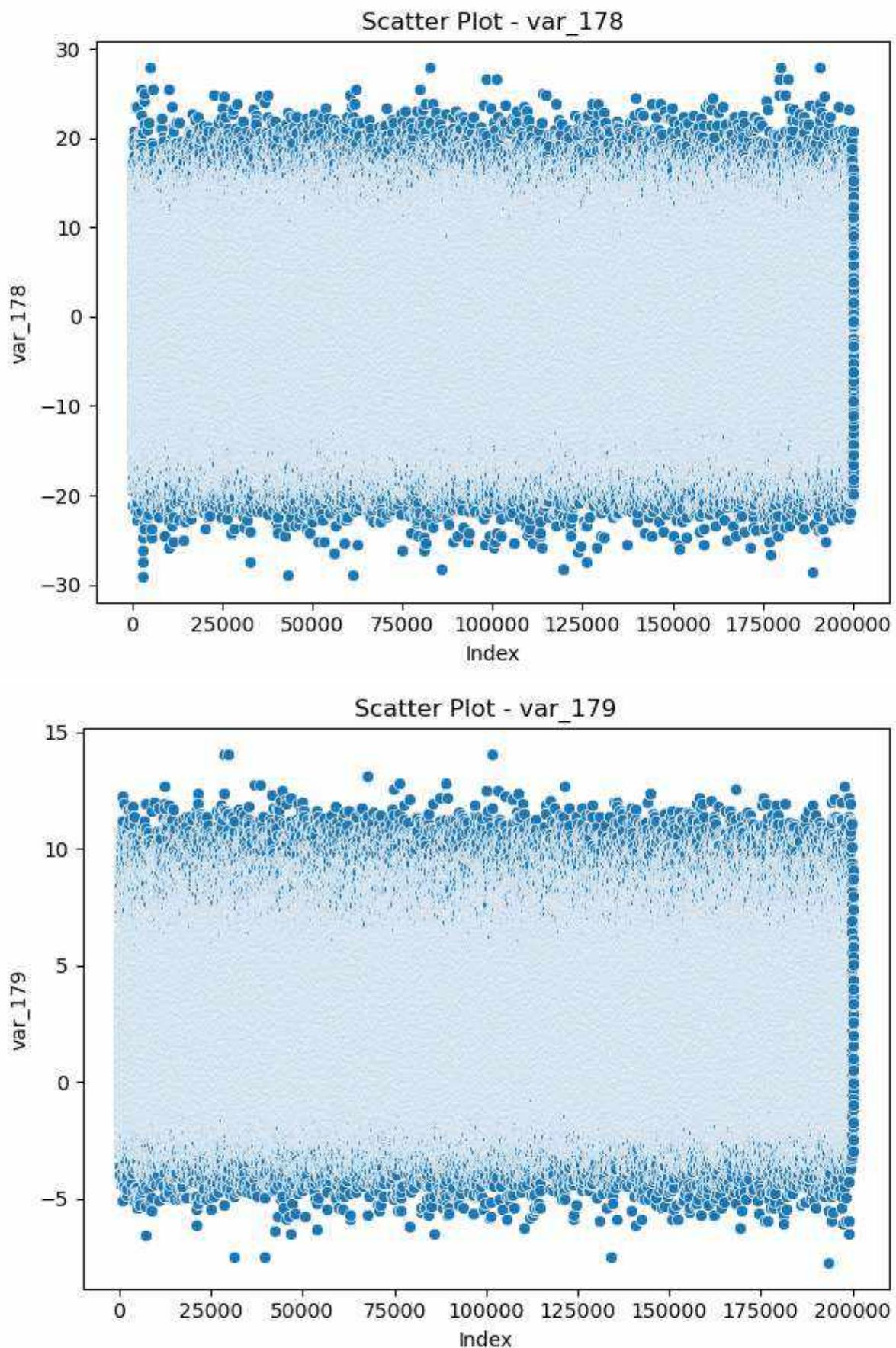
Scatter Plot - var\_174

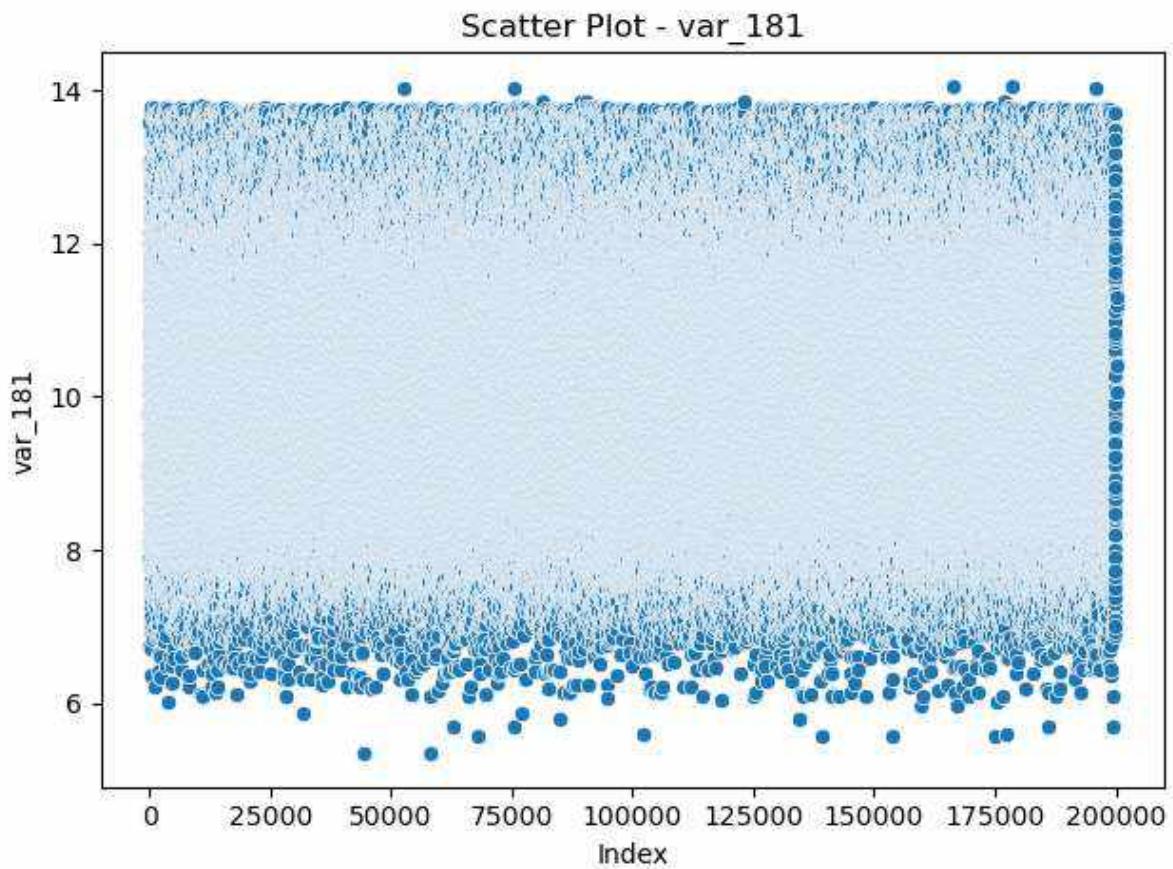
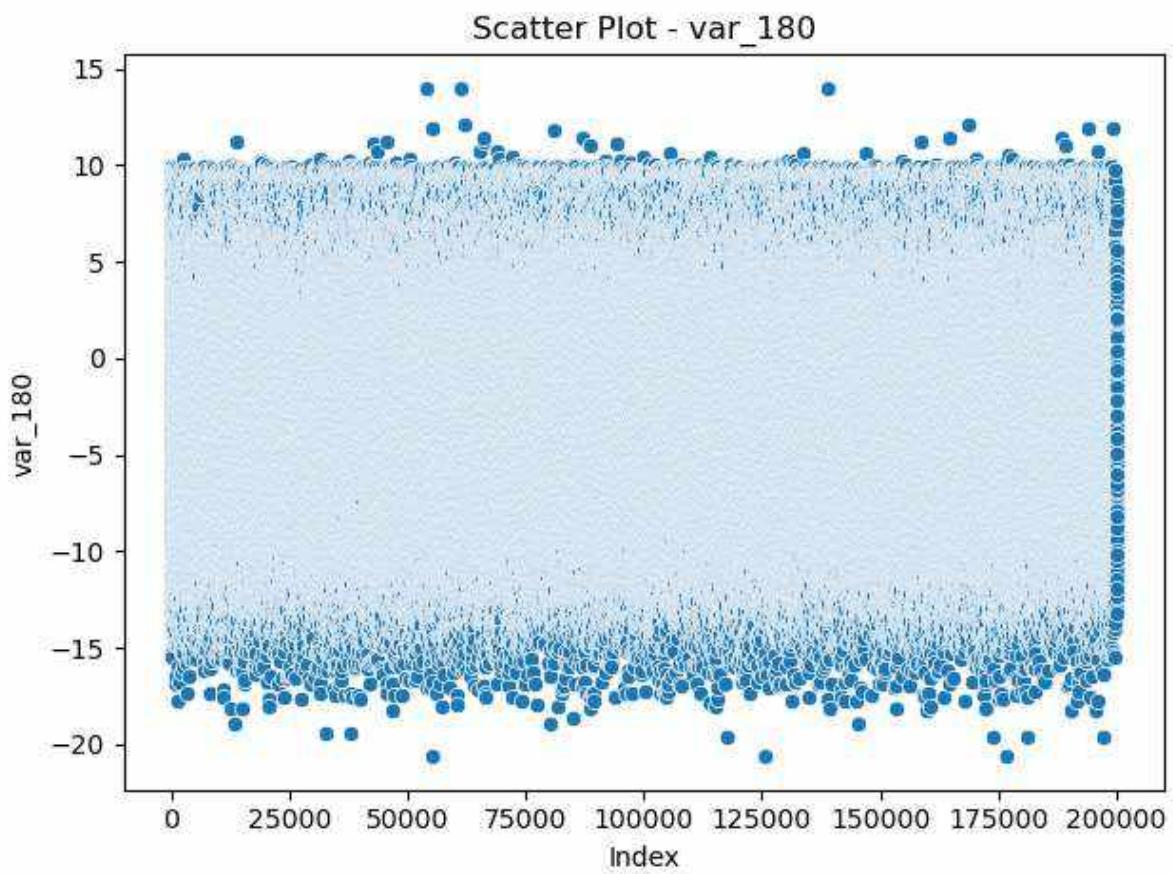


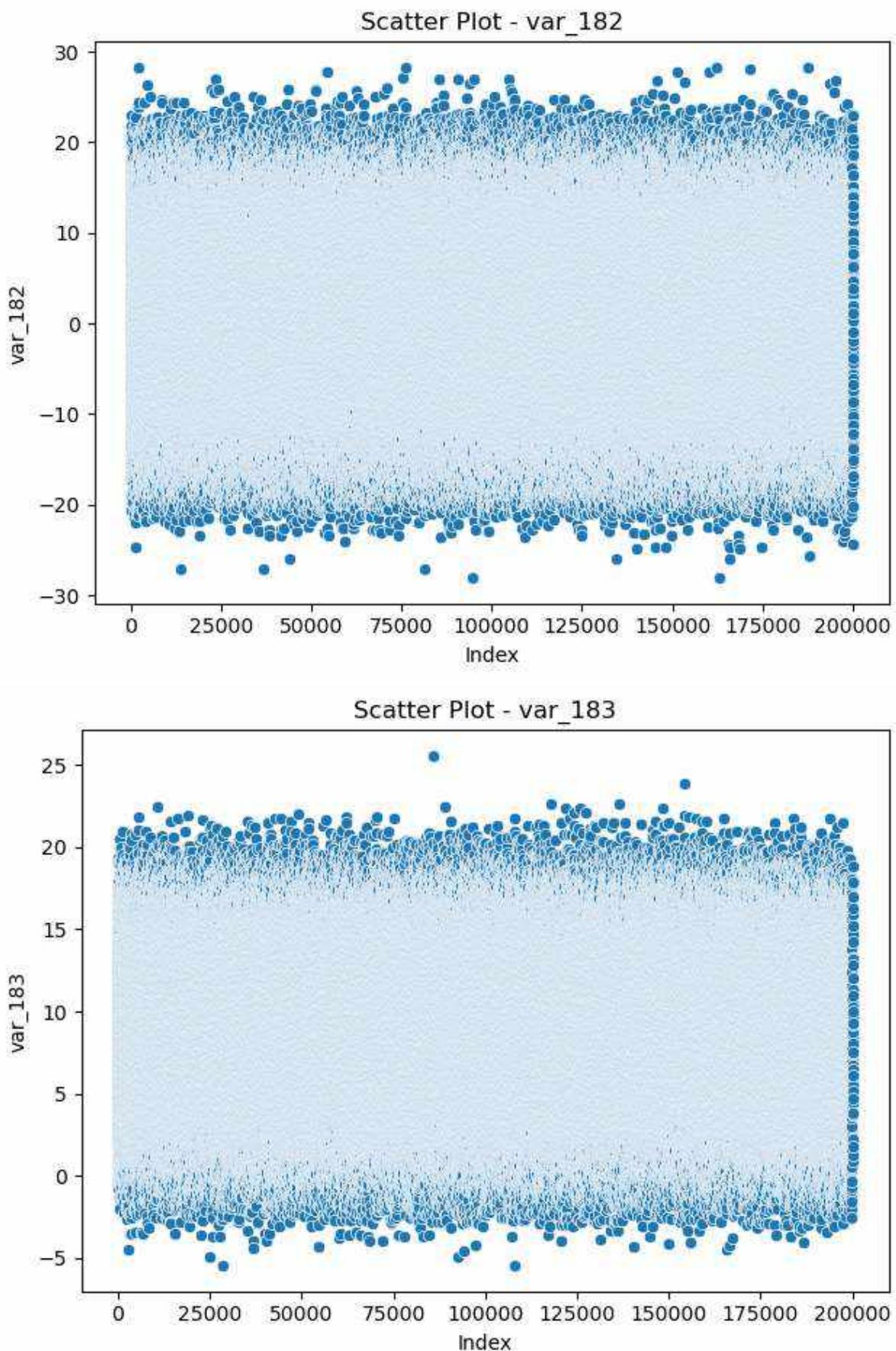
Scatter Plot - var\_175

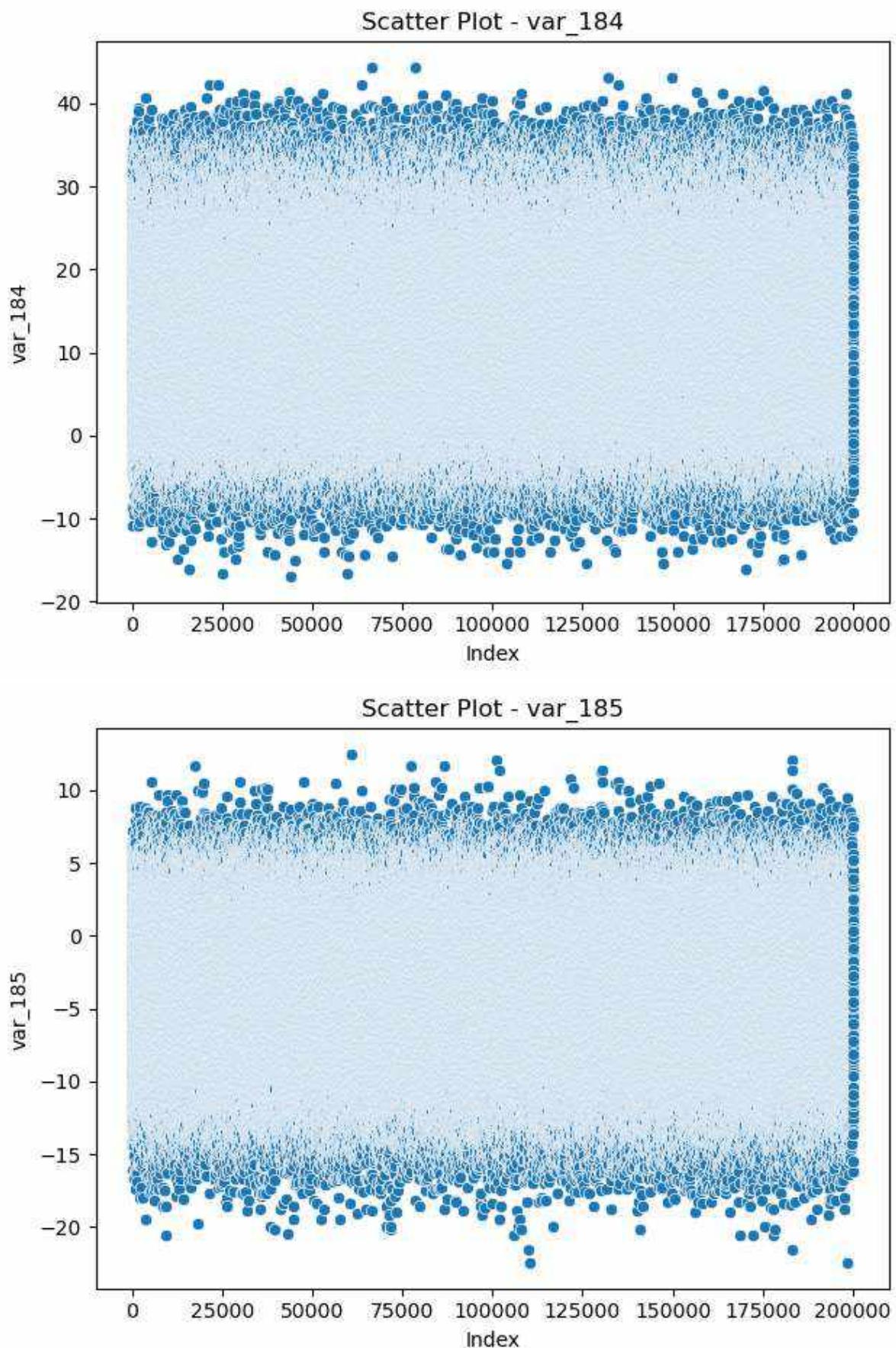




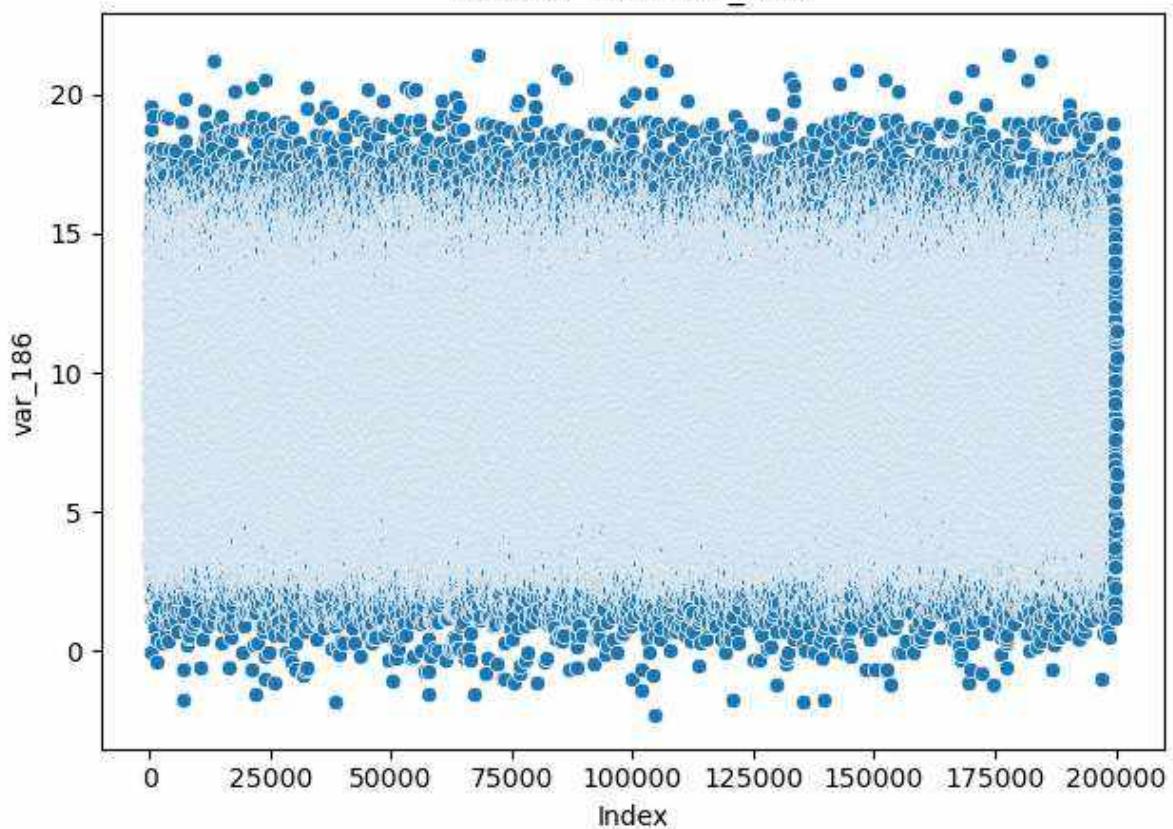




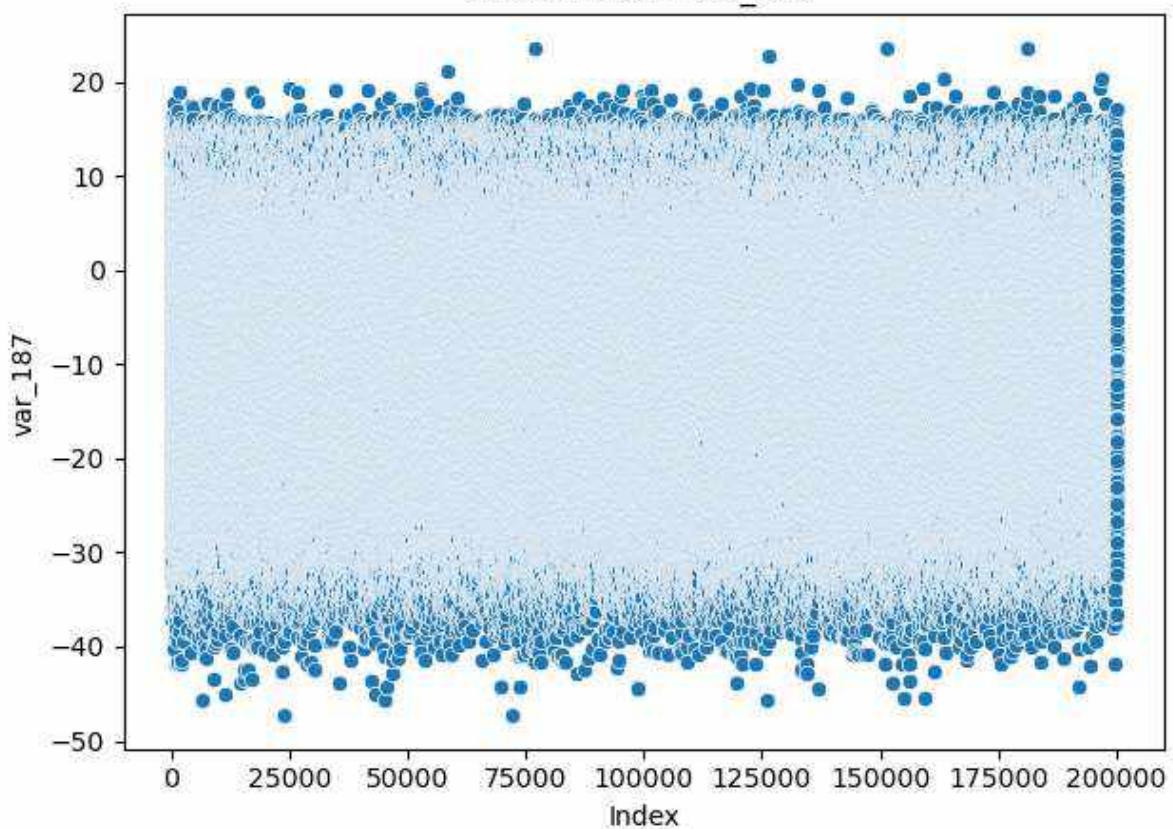


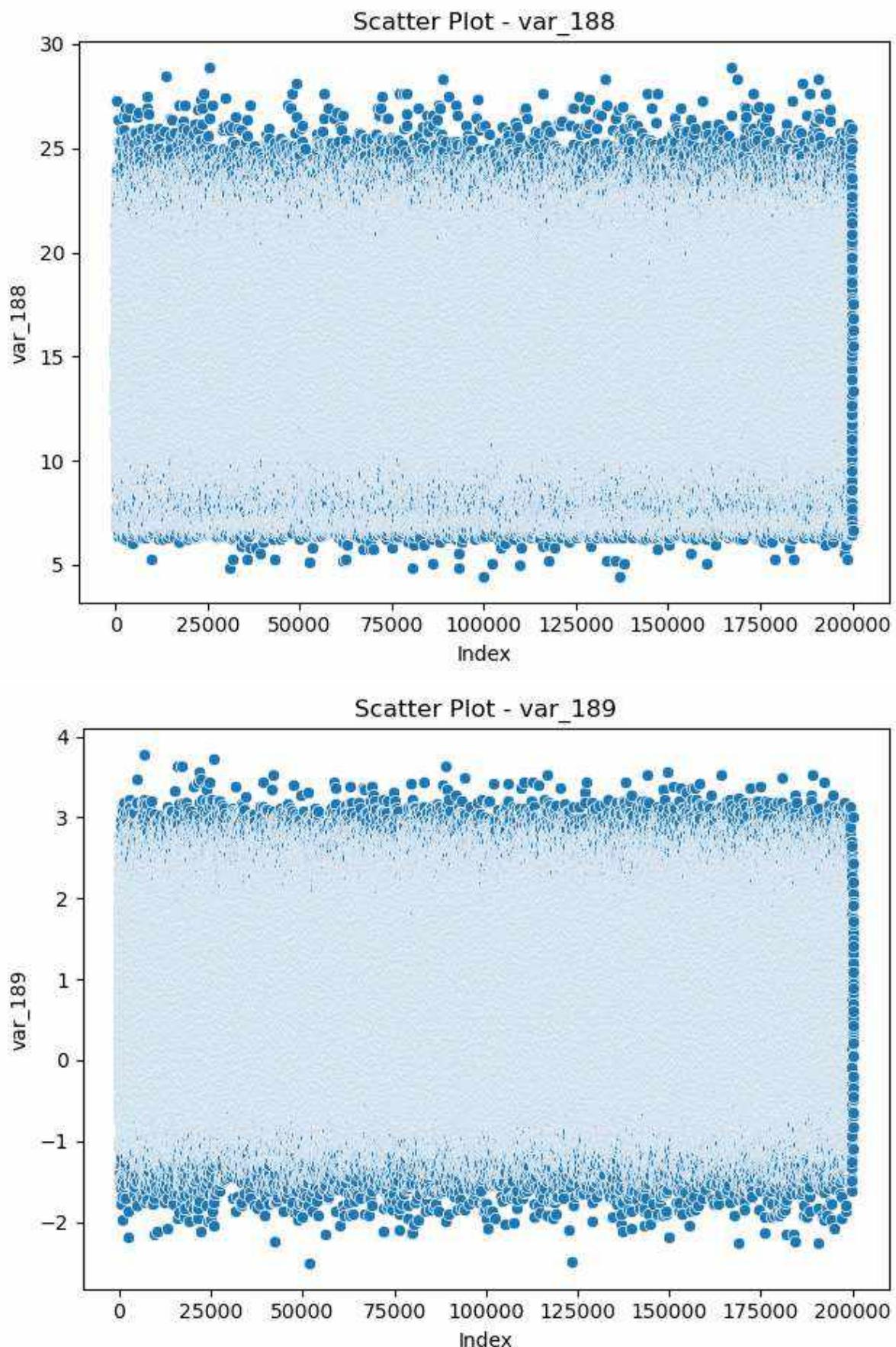


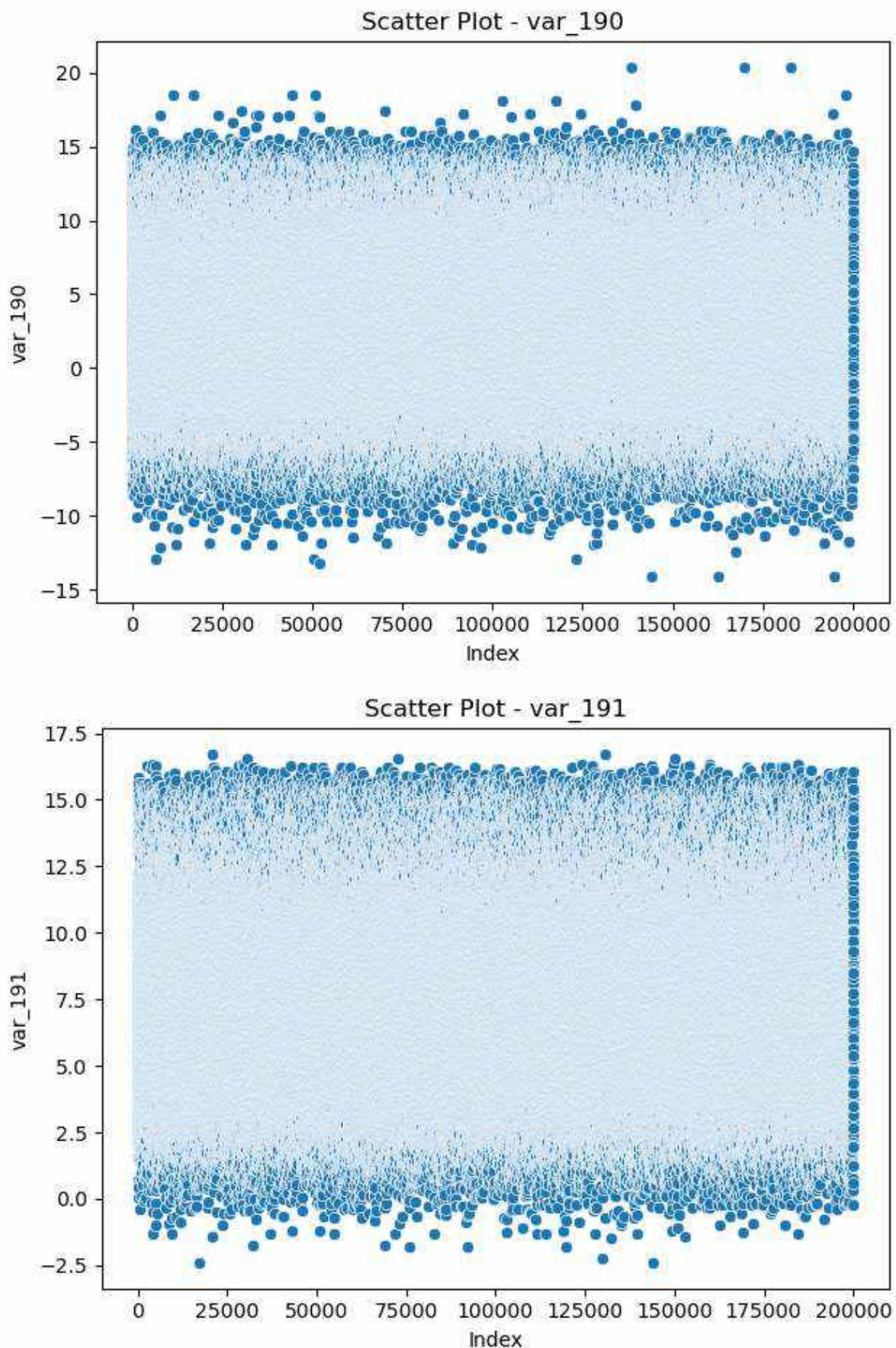
Scatter Plot - var\_186



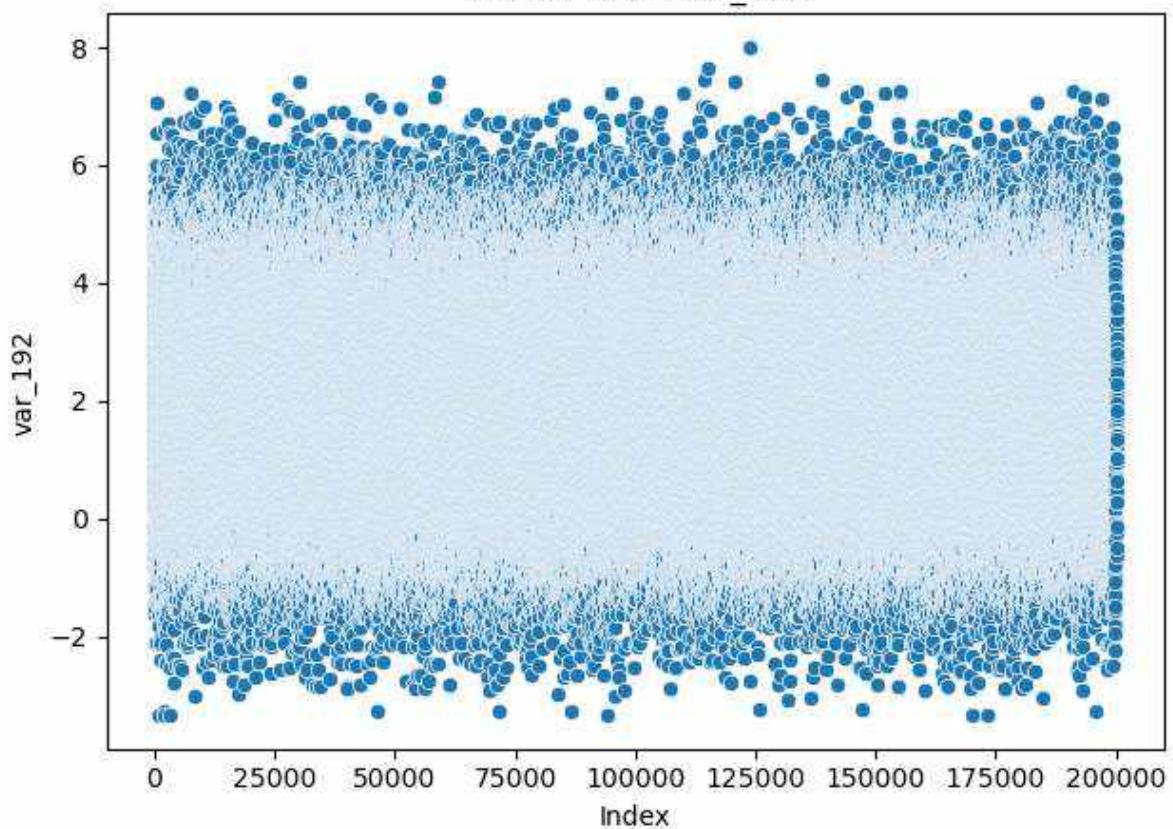
Scatter Plot - var\_187



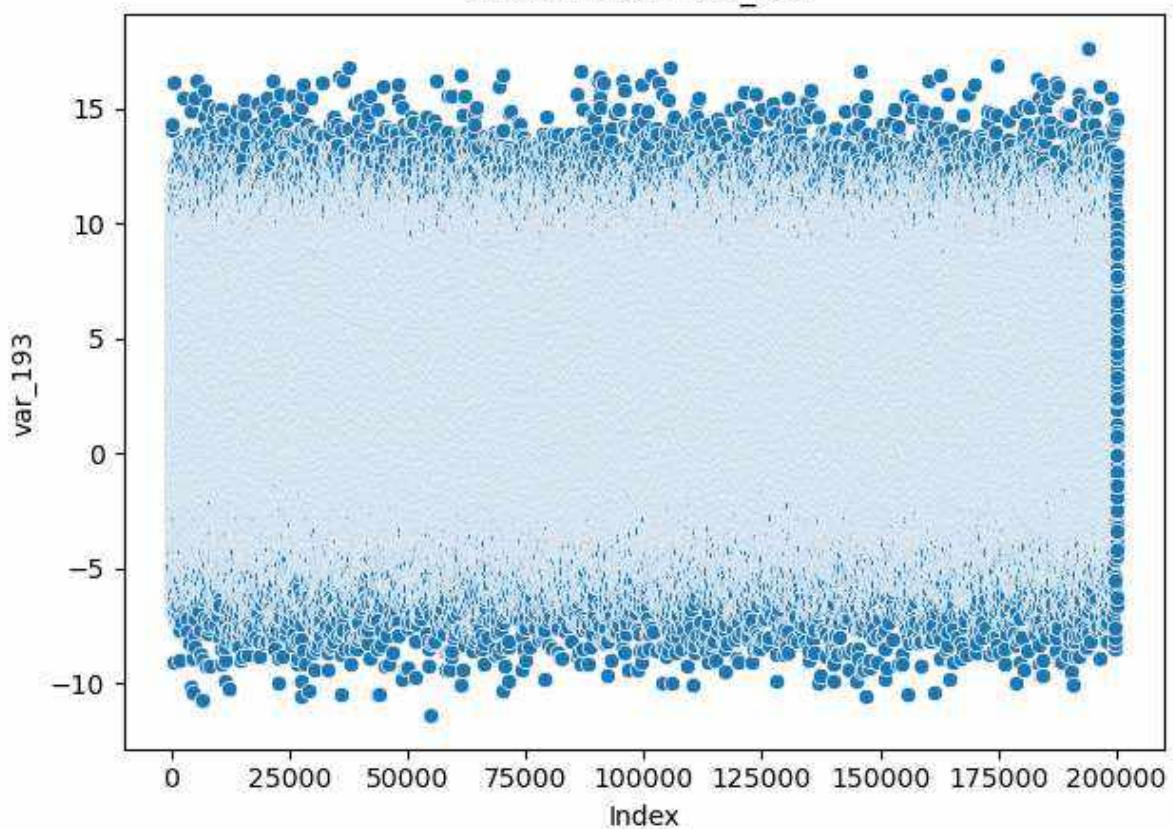




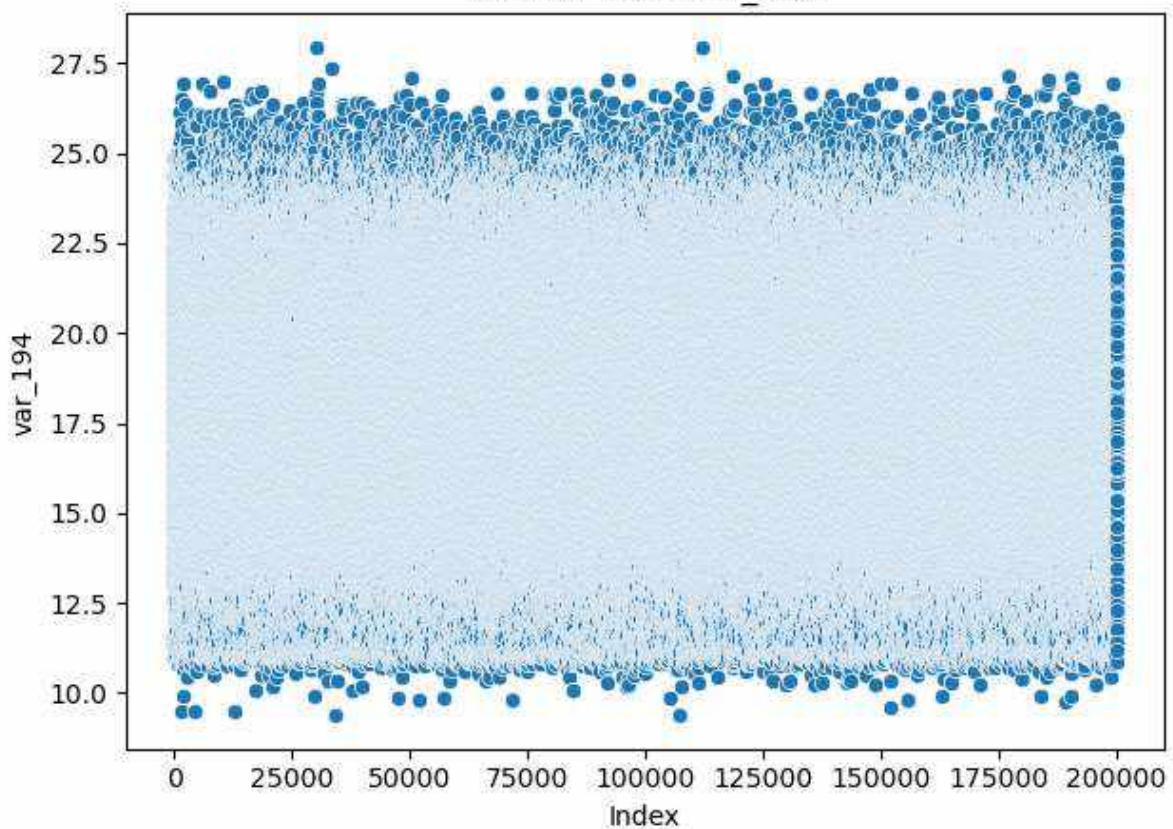
Scatter Plot - var\_192



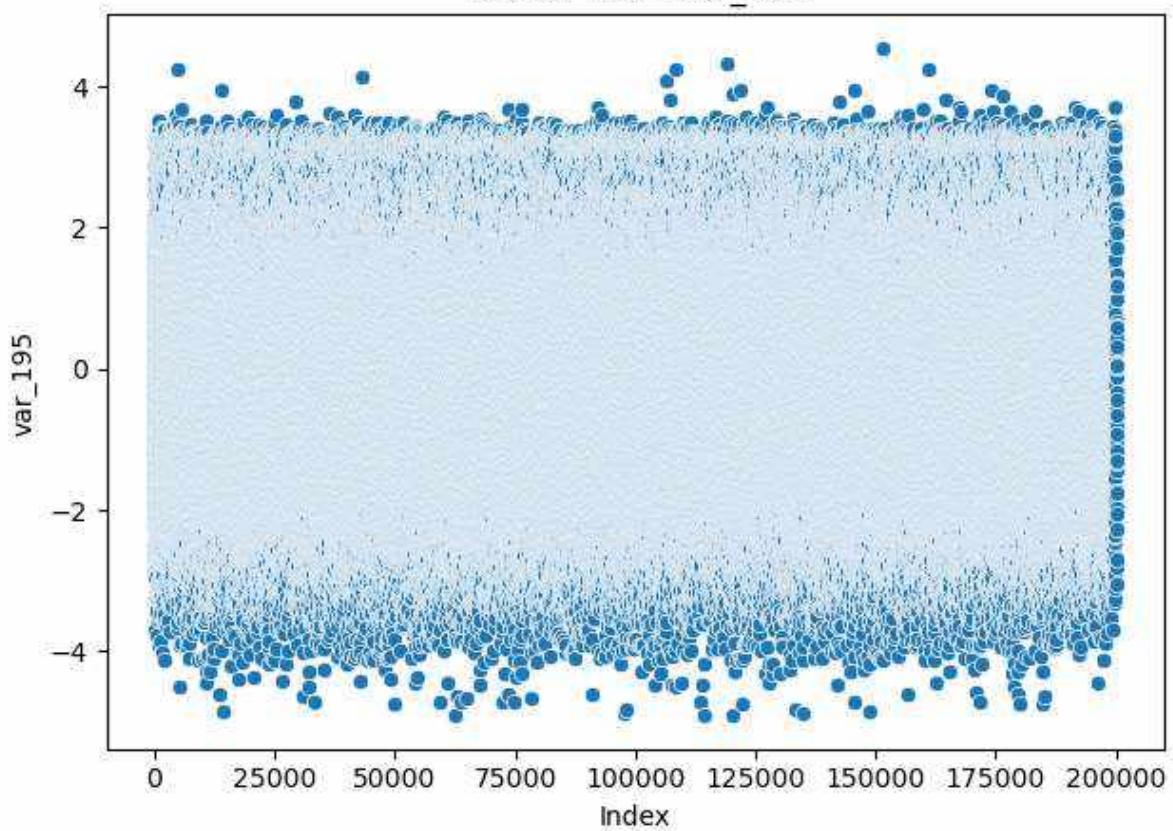
Scatter Plot - var\_193



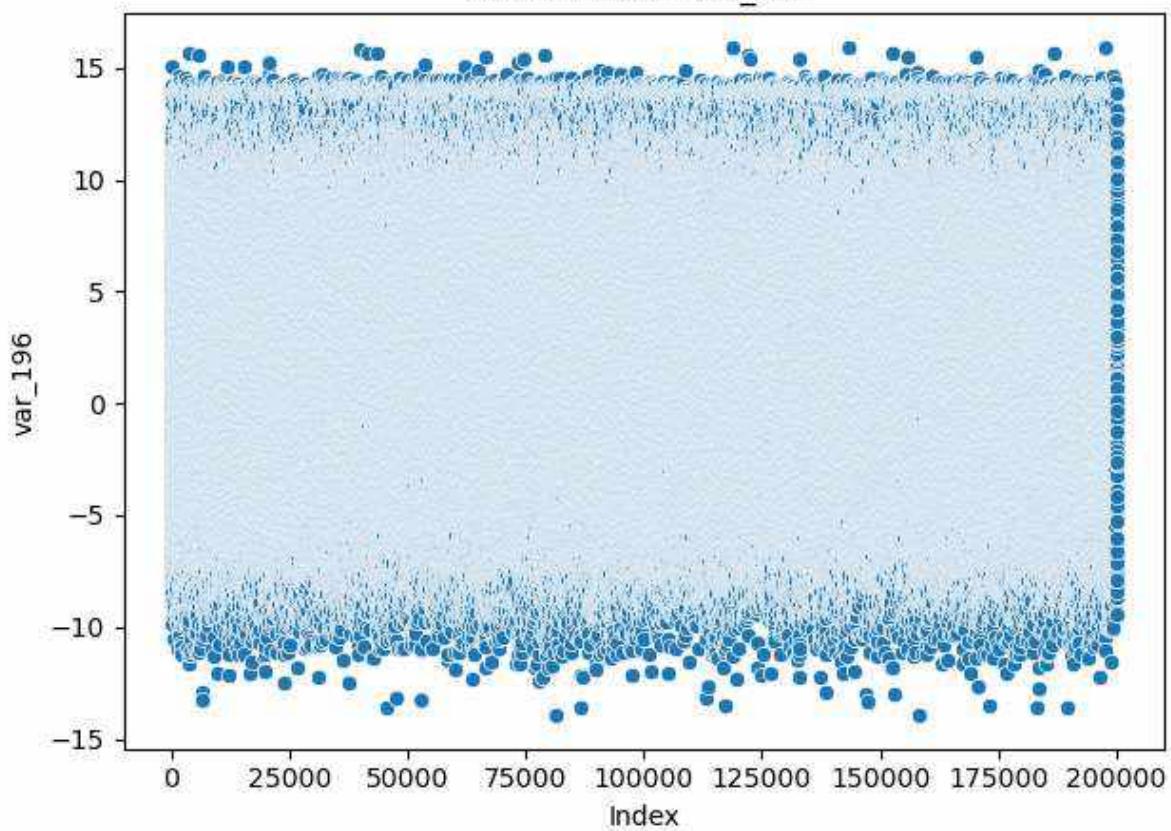
Scatter Plot - var\_194



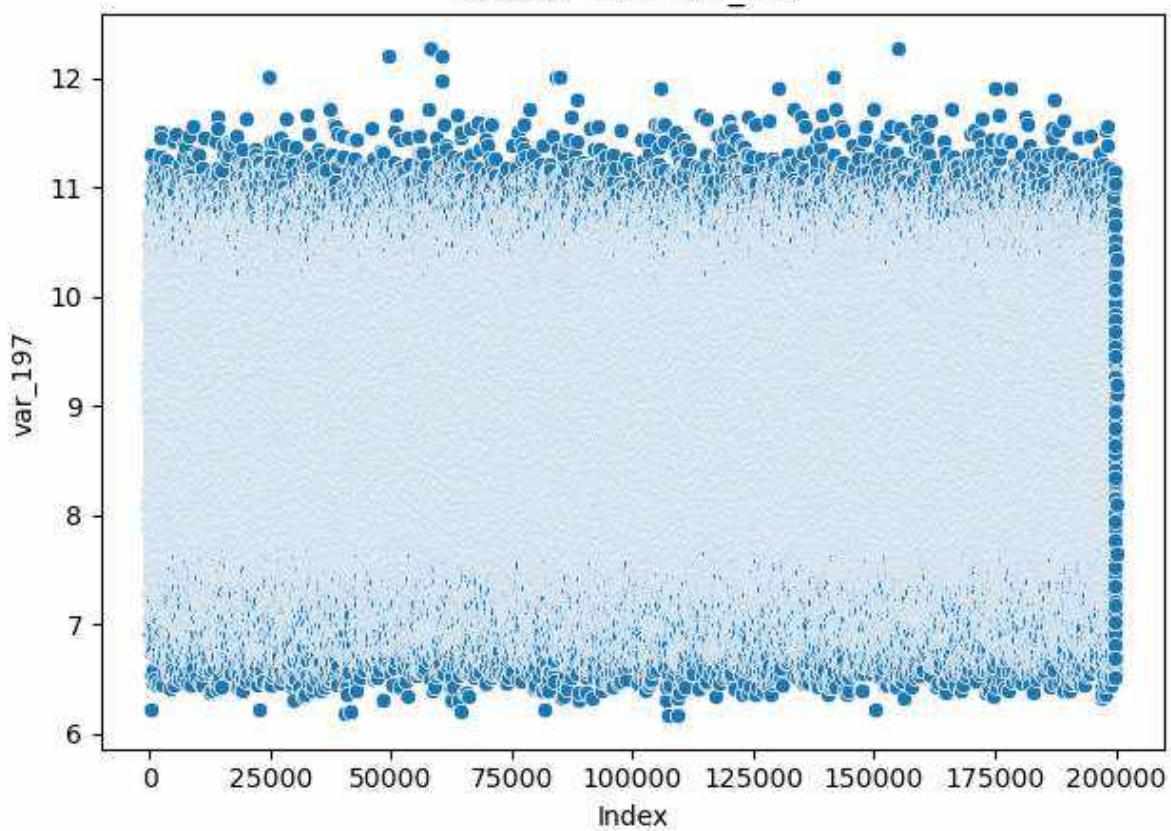
Scatter Plot - var\_195

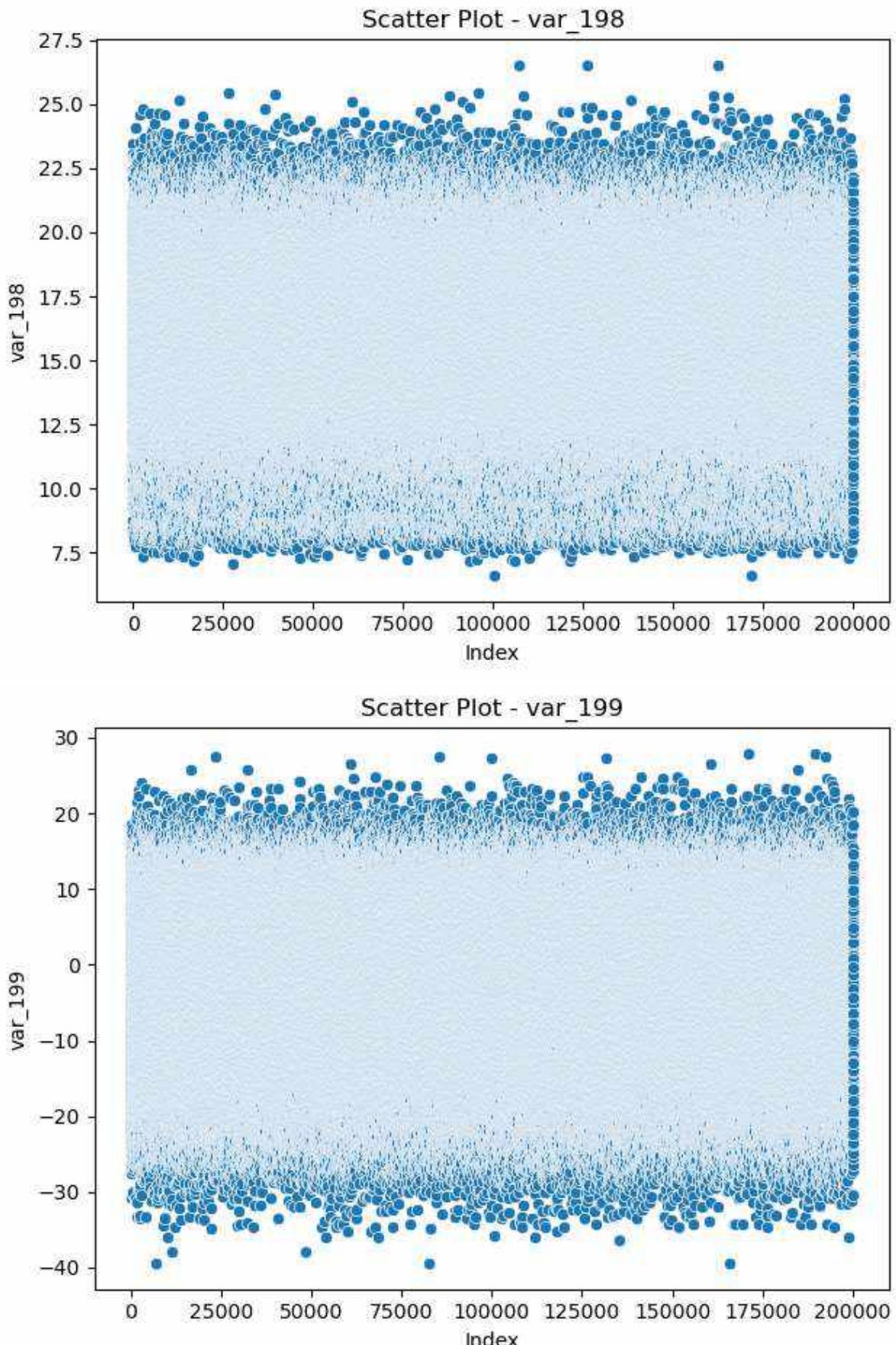


Scatter Plot - var\_196



Scatter Plot - var\_197





## Distribution of Target Variable

```
In [27]: # Counting the occurrences of each target value  
target_counts = data_training_set['target'].value_counts()  
  
# Calculating the percentage distribution of the target variable
```

```

total_samples = len(data_training_set)
target_percentage = target_counts / total_samples * 100

# Printing the target value counts and percentages
print("Target Value Counts:")
for target, count in target_counts.items():
    percentage = target_percentage[target]
    print(f"Target {target}: {count} ({percentage:.2f}%)")

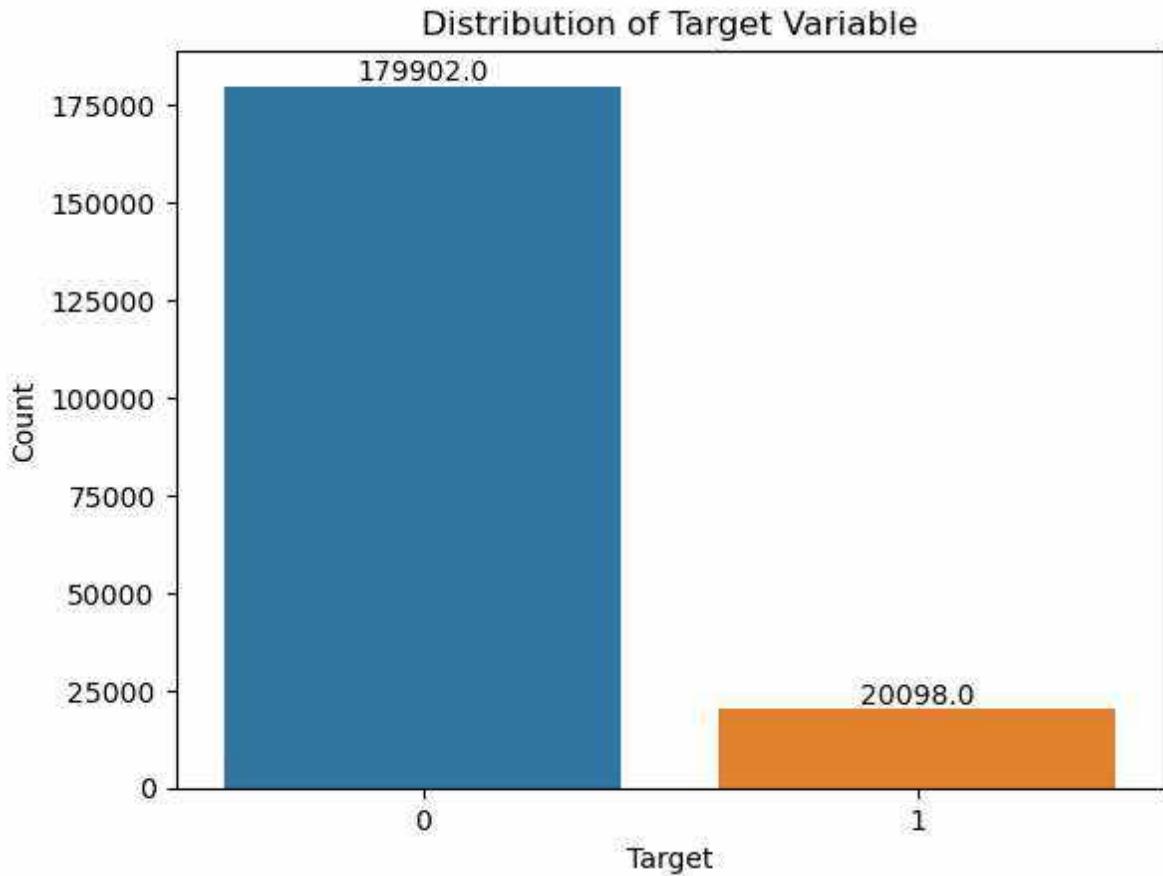
# Plotting the distribution of the target variable
sns.barplot(x=target_counts.index, y=target_counts.values)
plt.title('Distribution of Target Variable')
plt.xlabel('Target')
plt.ylabel('Count')

ax = plt.gca()
for p in ax.patches:
    ax.annotate(f"{p.get_height()}", (p.get_x() + p.get_width() / 2, p.get_height(),
                                    ha='center', va='bottom'))

plt.show()

```

Target Value Counts:  
 Target 0: 179902 (89.95%)  
 Target 1: 20098 (10.05%)



## Analyze relationships between features

- Correlation (Heatmap) and Pair plot

### Correlation (Heatmap) for training\_set

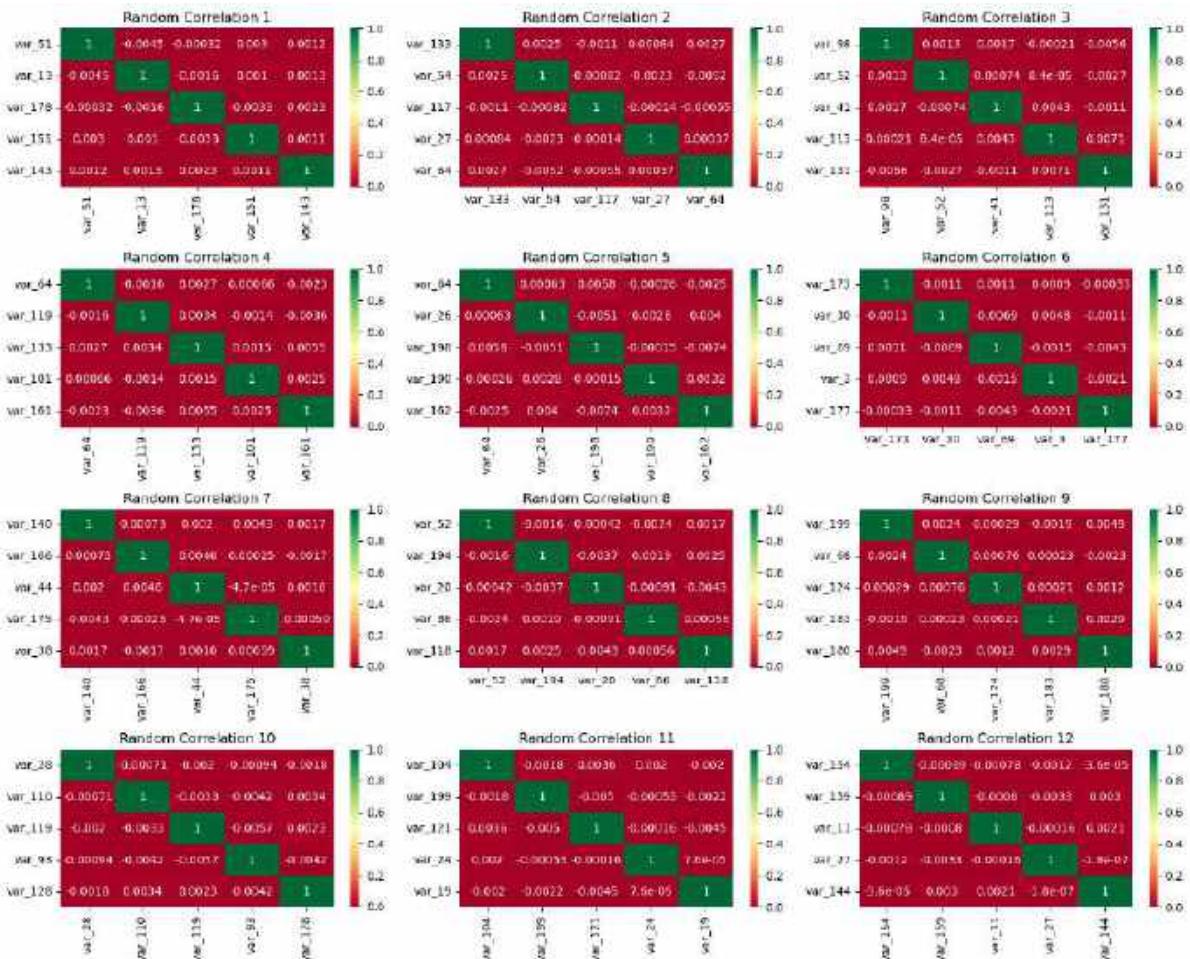
In [28]: # correlation matrix: randomly selected a features of the training\_set

```
# Exclude ID_code and target columns if present
columns_to_analyze = [column for column in data_training_set.columns if column not
num_heatmaps = 12

# subplots for correlation matrices
fig, axes = plt.subplots((num_heatmaps + 2) // 3, 3, figsize=(15, (num_heatmaps + 2) * 1.5))

# plotting correlation matrices
for i in range(num_heatmaps):
    random_columns = random.sample(columns_to_analyze, 5)
    row = i // 3
    col = i % 3
    ax = axes[row, col]
    sns.heatmap(data_training_set[random_columns].corr(), annot=True, cmap='RdYlGn')
    ax.set_title(f'Random Correlation {i + 1}')

# Adjusting spacing between subplots
plt.tight_layout()
plt.show()
```



## Correlation (Heatmap) for test\_set

In [29]: # correlation matrix: randomly selected a features of the test\_set

```
num_heatmaps = 12

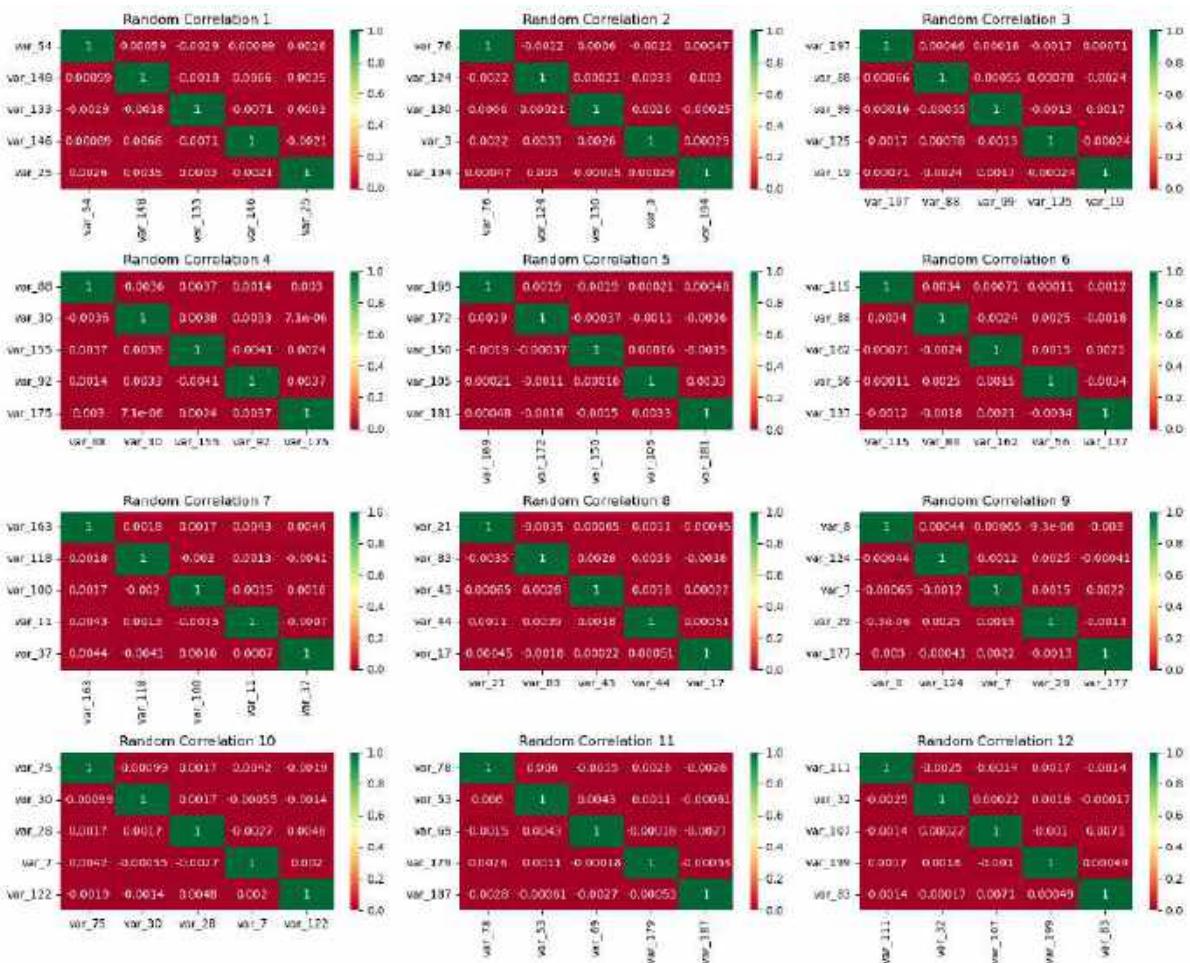
fig, axes = plt.subplots((num_heatmaps + 2) // 3, 3, figsize=(15, (num_heatmaps + 2) * 1.5))
```

```

for i in range(num_heatmaps):
    random_columns = random.sample(columns_to_analyze, 5)
    row = i // 3
    col = i % 3
    ax = axes[row, col]
    sns.heatmap(data_test_set[random_columns].corr(), annot=True, cmap='RdYlGn', ax=ax)
    ax.set_title(f'Random Correlation {i + 1}')

plt.tight_layout()
plt.show()

```



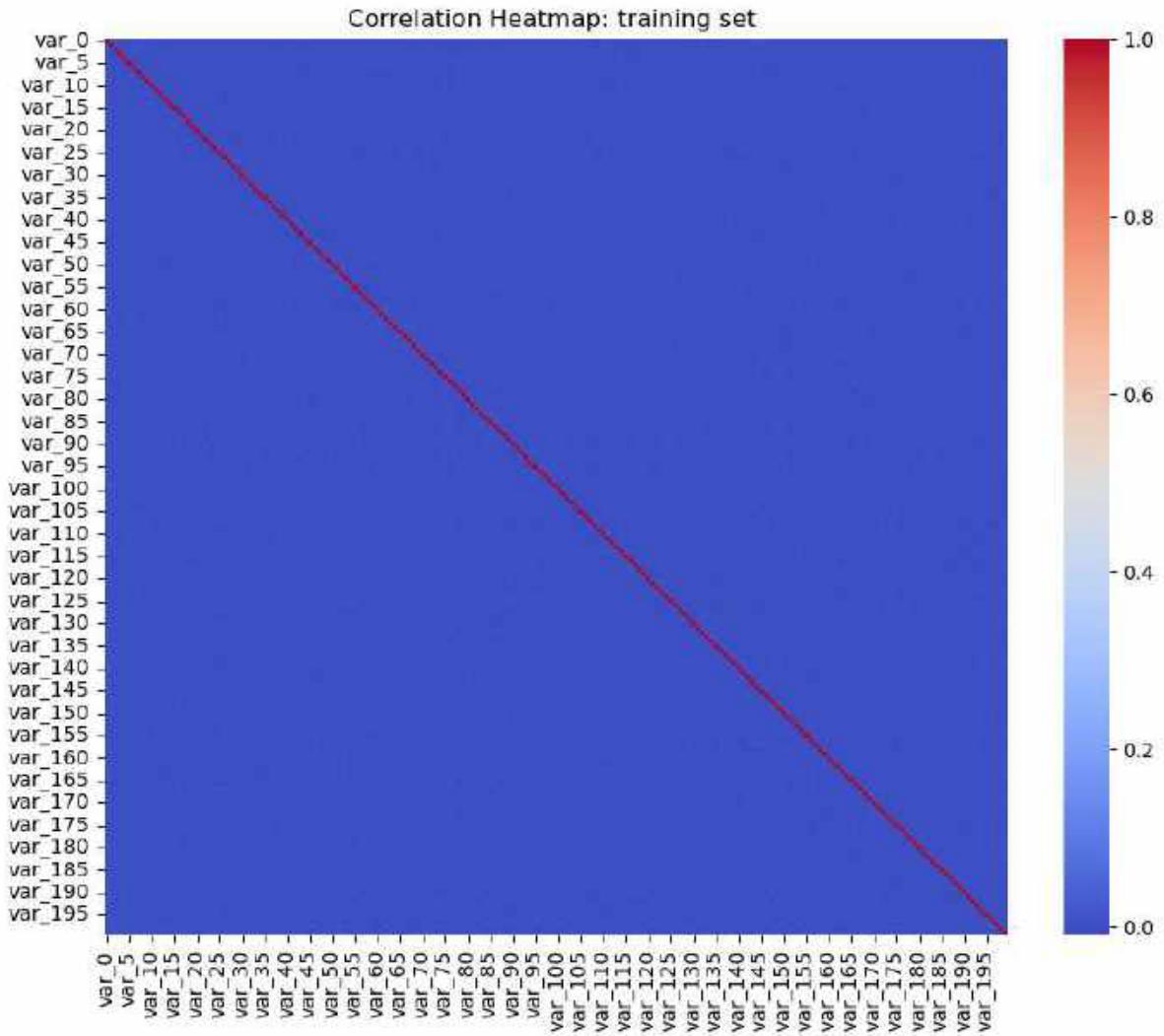
```

In [30]: # columns for correlation matrix
selected_columns = data_training_set.columns[2:]
corr_matrix_test = data_training_set[selected_columns].corr()

plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix_test, annot=False, cmap="coolwarm")
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)

plt.title("Correlation Heatmap: training set ", fontsize=12)
plt.show()

```

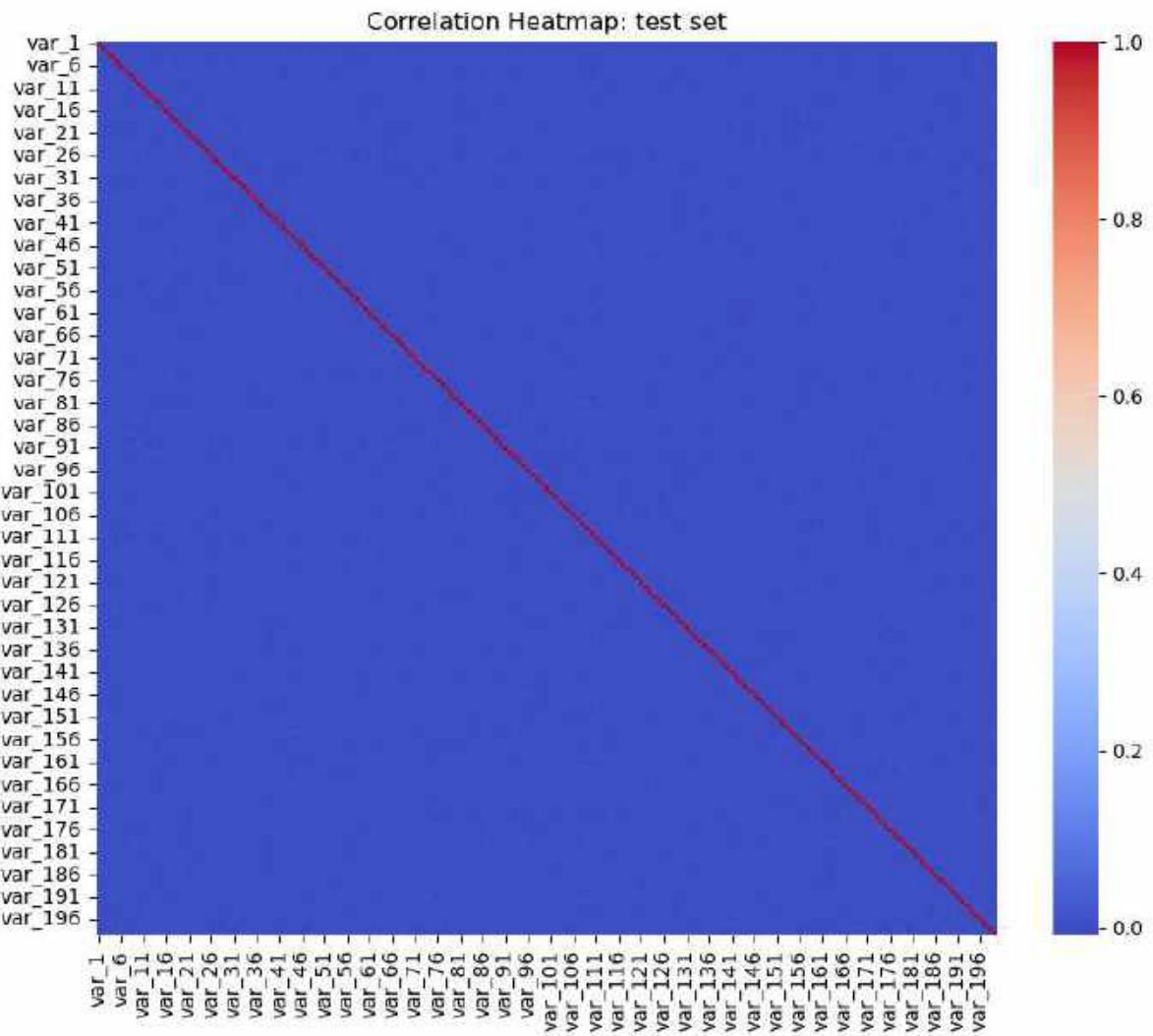


## Correlation (Heatmap) for test\_set

```
In [31]: # Selecting columns for correlation matrix
selected_columns = data_test_set.iloc[:, 2:]
corr_matrix_test = selected_columns.corr()

plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix_test, annot=False, cmap="coolwarm")
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.title("Correlation Heatmap: test set", fontsize=12)

plt.show()
```



The correlation between the features is very small.

## Pair plot

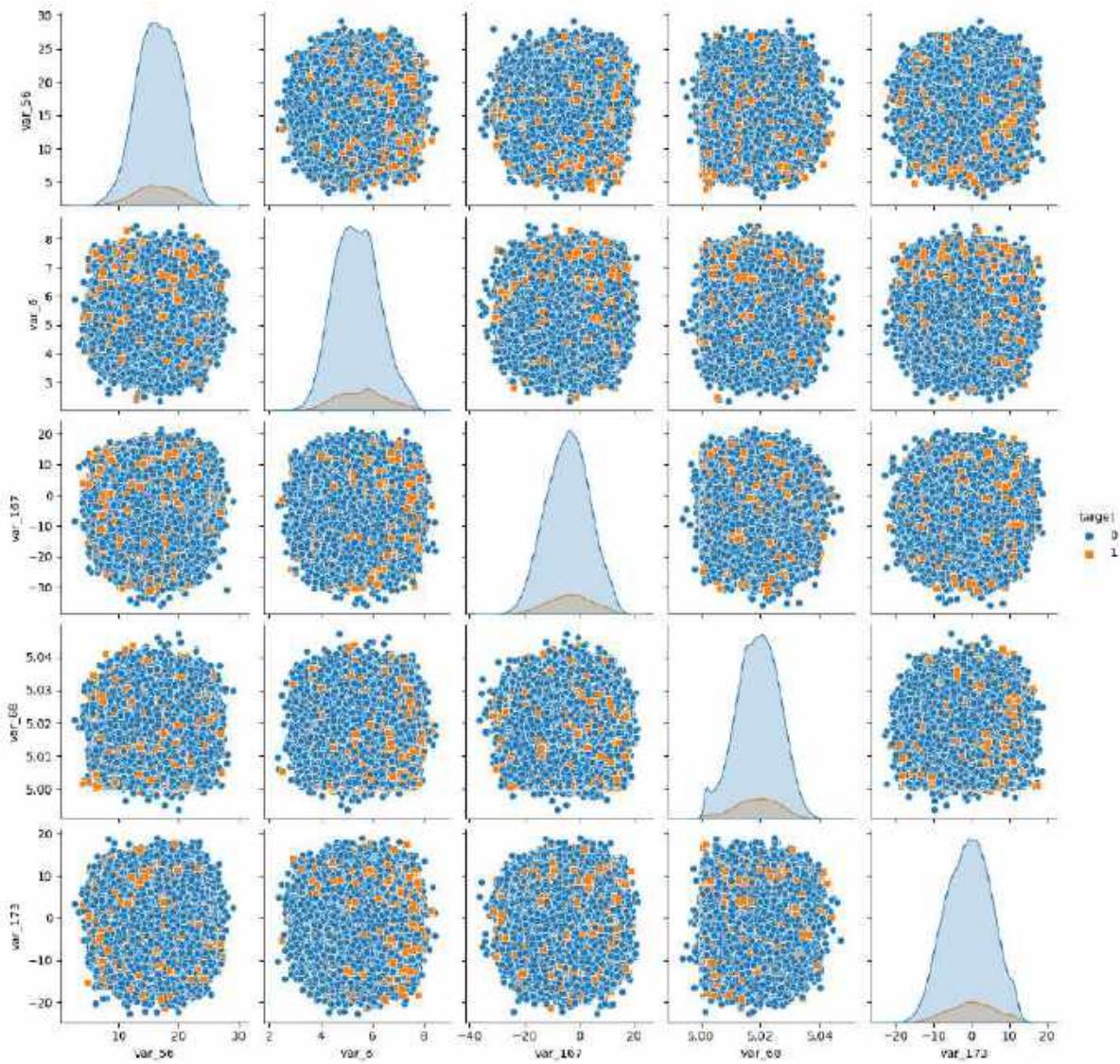
```
In [32]: # Pair plot: training_set(Randomly select columns )

# Exclude ID_code and target columns if present
columns_to_analyze = [column for column in data_training_set.columns if column not

# Selecting a subset of columns to analyze (e.g., var_0 to var_9)
num_columns = 5

# Randomly select columns to include in the pair plot
random_columns = random.sample(columns_to_analyze, num_columns)

# data points will be colored based on their corresponding target values.
sns.pairplot(data_training_set[random_columns + ['target']], hue="target", markers:
plt.show()
```

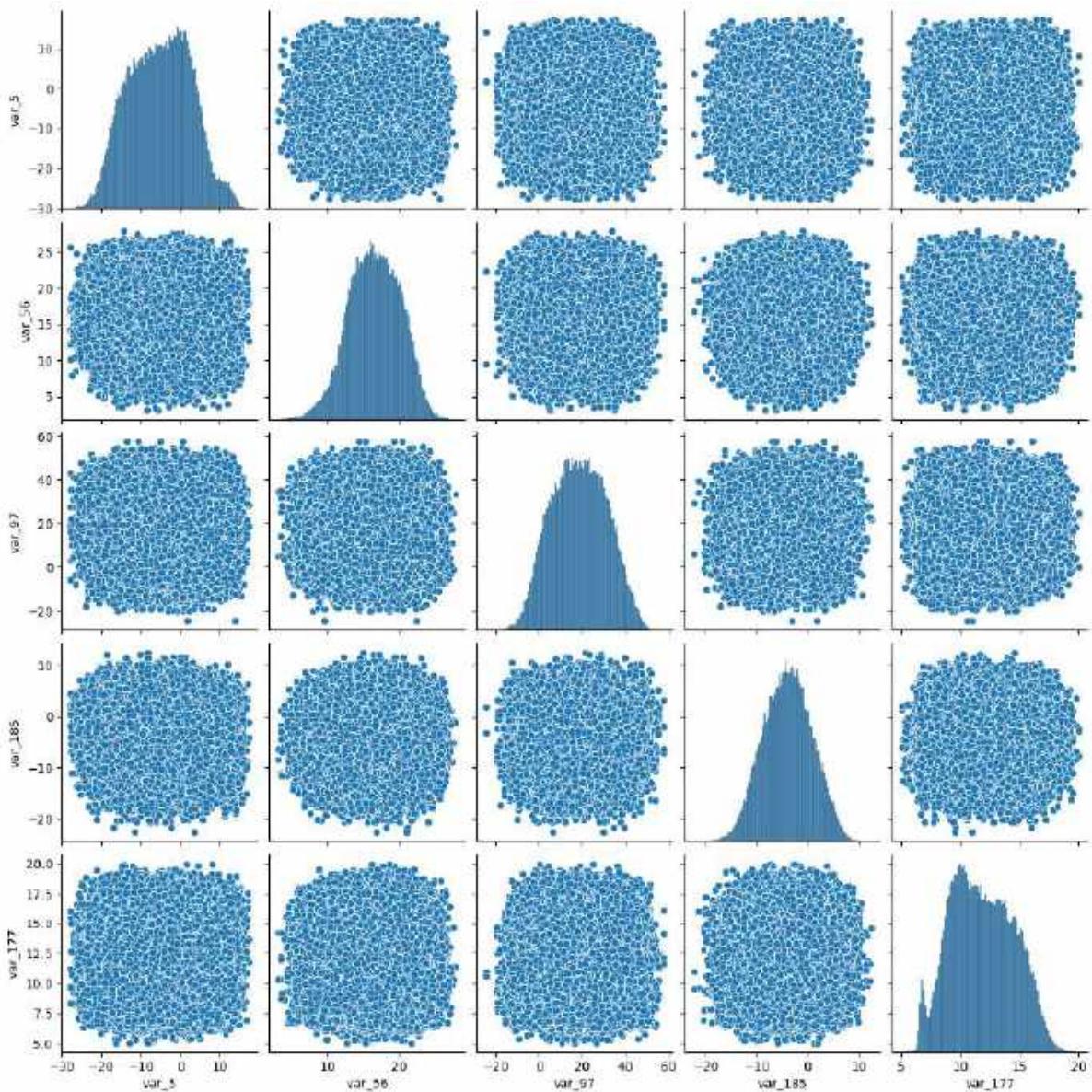


```
In [33]: # Pair plot: test_set(Randomly select columns)

columns_to_analyze = [column for column in data_test_set.columns if column != 'ID_'
num_columns = 5

# Randomly select columns to include in the pair plot
random_columns = random.sample(columns_to_analyze, num_columns)

sns.pairplot(data_test_set[random_columns], markers="o")
plt.show()
```



- Exploratory Data Analysis (EDA) for every possible question:
  - Identify the variable and their types
  - Clean your data (error, remove duplicates, missing values, Outliers)
  - Transformation (Standardization, Normalization, encoding categorical to numerical)
  - Data Visualization (use the suitable visualization that you need)

```
In [34]: # Display information about variables and their types

print("data_test_set:")
data_test_set.info()

print("\n data_training_set:")
data_training_set.info()
```

```

data_test_set:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200000 entries, 0 to 199999
Columns: 201 entries, ID_code to var_199
dtypes: float64(200), object(1)
memory usage: 306.7+ MB

data_training_set:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200000 entries, 0 to 199999
Columns: 202 entries, ID_code to var_199
dtypes: float64(200), int64(1), object(1)
memory usage: 308.2+ MB

```

## Normality test for Outlier detection

In [35]:

```

# Function to detect outliers
def detect_outliers(data, column_name, threshold=1.5):
    outliers = None

    if pd.api.types.is_numeric_dtype(data):
        alpha = 0.05
        stat, p = stats.shapiro(data.dropna())

        if p > alpha:
            # Normal distribution, using Z-score method (Shapiro-Wilk test)
            z_scores = np.abs(stats.zscore(data))
            outliers = data[z_scores > threshold]
        else:
            # Non-normal distribution, use Tukey's method
            q1 = np.percentile(data, 25)
            q3 = np.percentile(data, 75)
            iqr = q3 - q1
            lower_bound = q1 - threshold * iqr
            upper_bound = q3 + threshold * iqr
            outliers = data[(data < lower_bound) | (data > upper_bound)]

    return outliers

```

In [36]:

```

# Function to visualize outliers
def visualize_outliers(column_name, outliers):
    if outliers is not None:
        plt.figure(figsize=(10, 6))
        plt.boxplot(outliers.values, showfliers=False)
        plt.scatter(range(1, len(outliers)+1), outliers.values, color='red', marker='x')
        plt.xlabel('Outliers of ' + column_name)
        plt.ylabel('Values')
        plt.title('Outliers - ' + column_name)
        plt.legend()
        plt.show()
        print("Total outliers in column", column_name + ":", len(outliers))
    else:
        print('No outliers detected for column:', column_name)

    # Exclude ID_code and target columns if present
    numeric_columns_training = [column for column in data_training_set.columns if column != 'ID_code' and column != 'target']
    numeric_columns_test = [column for column in data_test_set.columns if column not in numeric_columns_training]

```

In [37]:

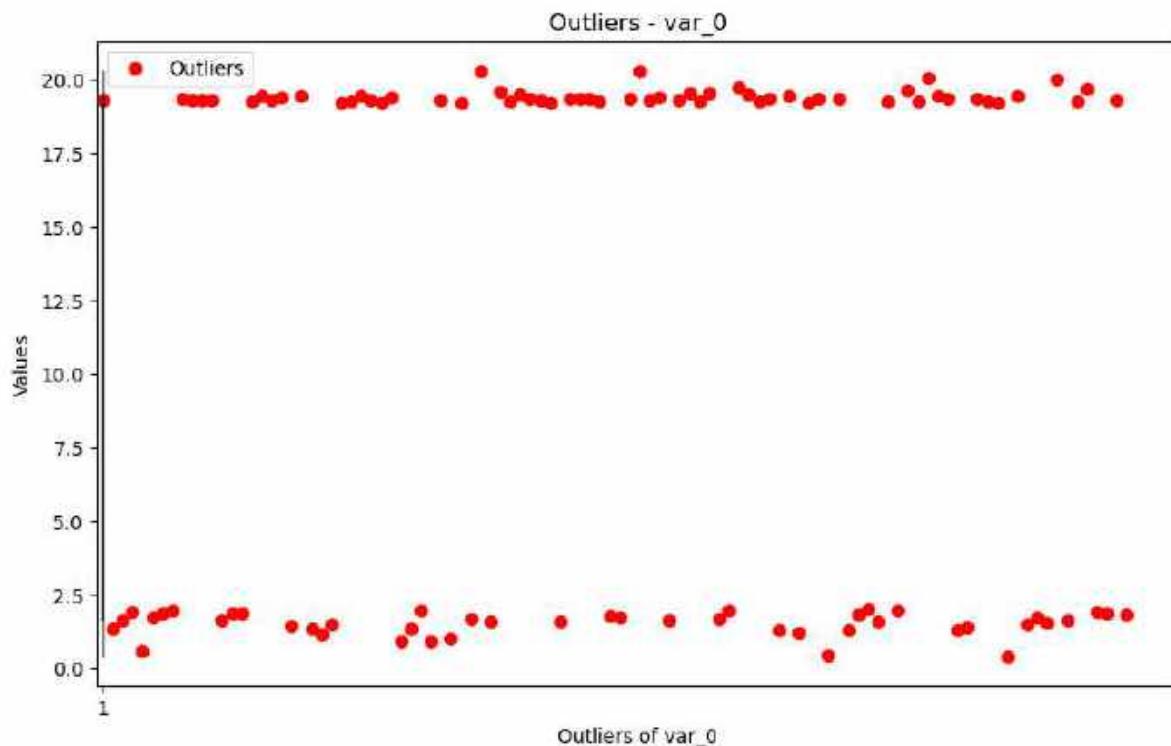
```

# Detect outliers for each column in the training set
for column in numeric_columns_training:
    outliers = detect_outliers(data_training_set[column], column)
    visualize_outliers(column, outliers)

```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.
```

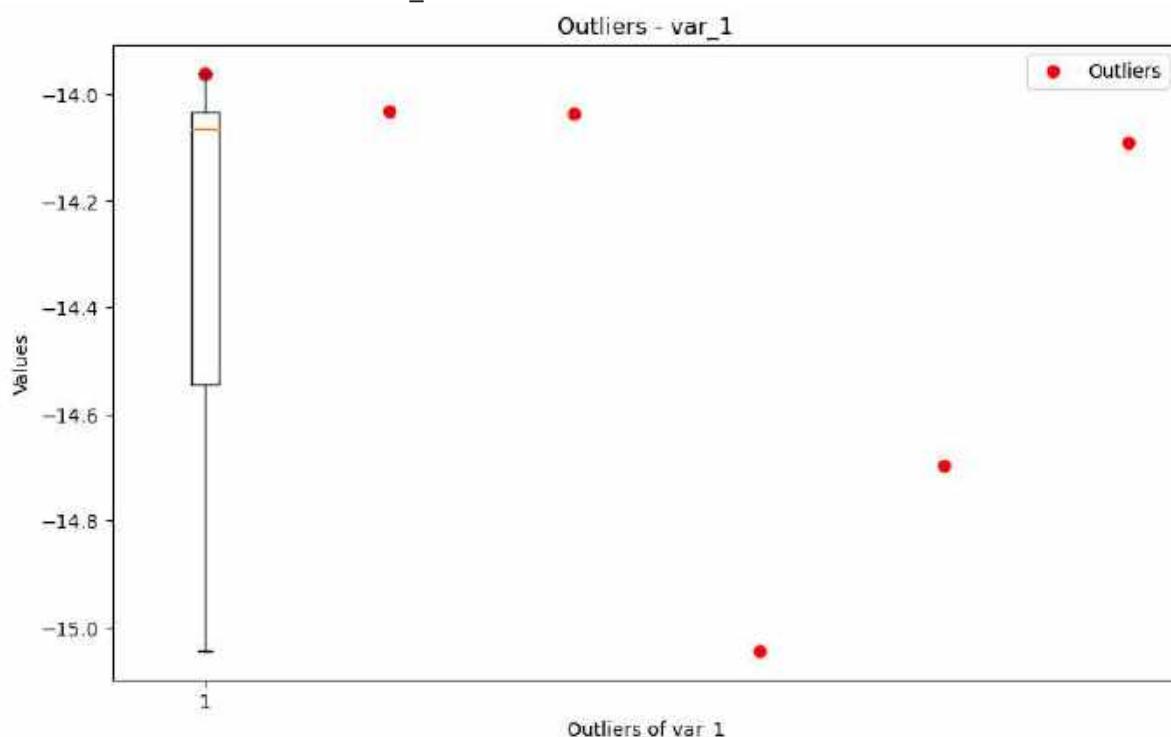
```
    warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.
```

```
    warnings.warn("p-value may not be accurate for N > 5000.")
```

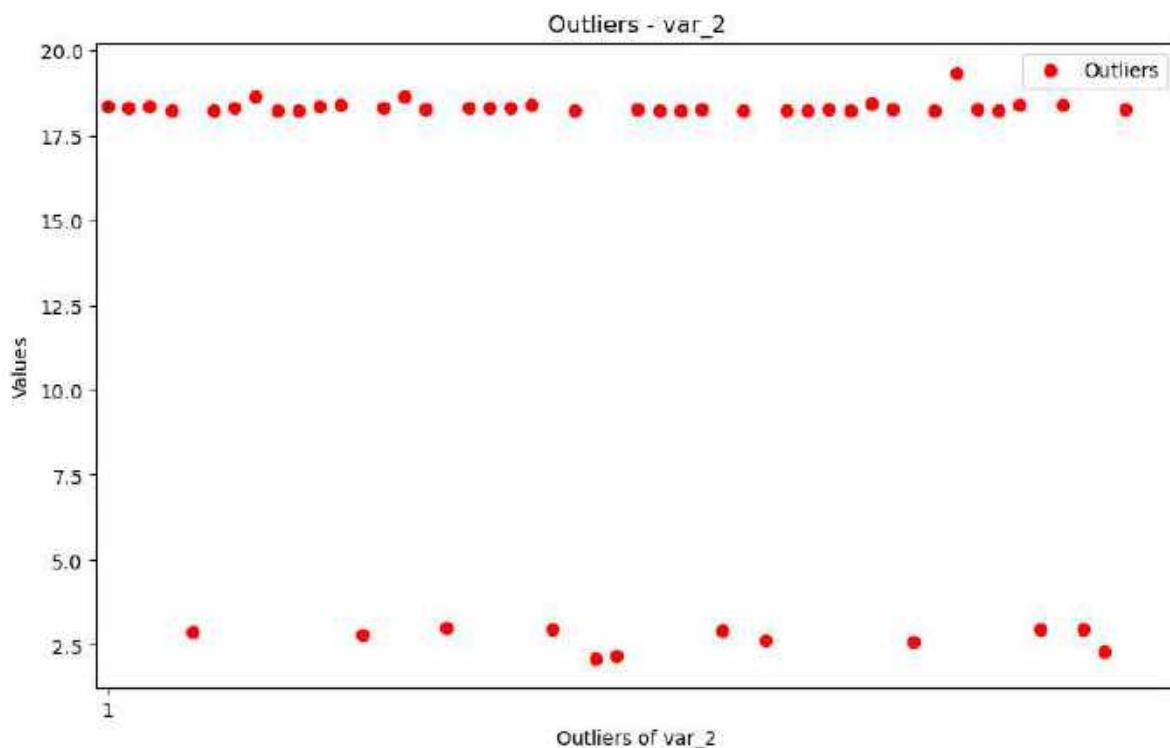
Total outliers in column var\_0: 104



Total outliers in column var\_1: 6

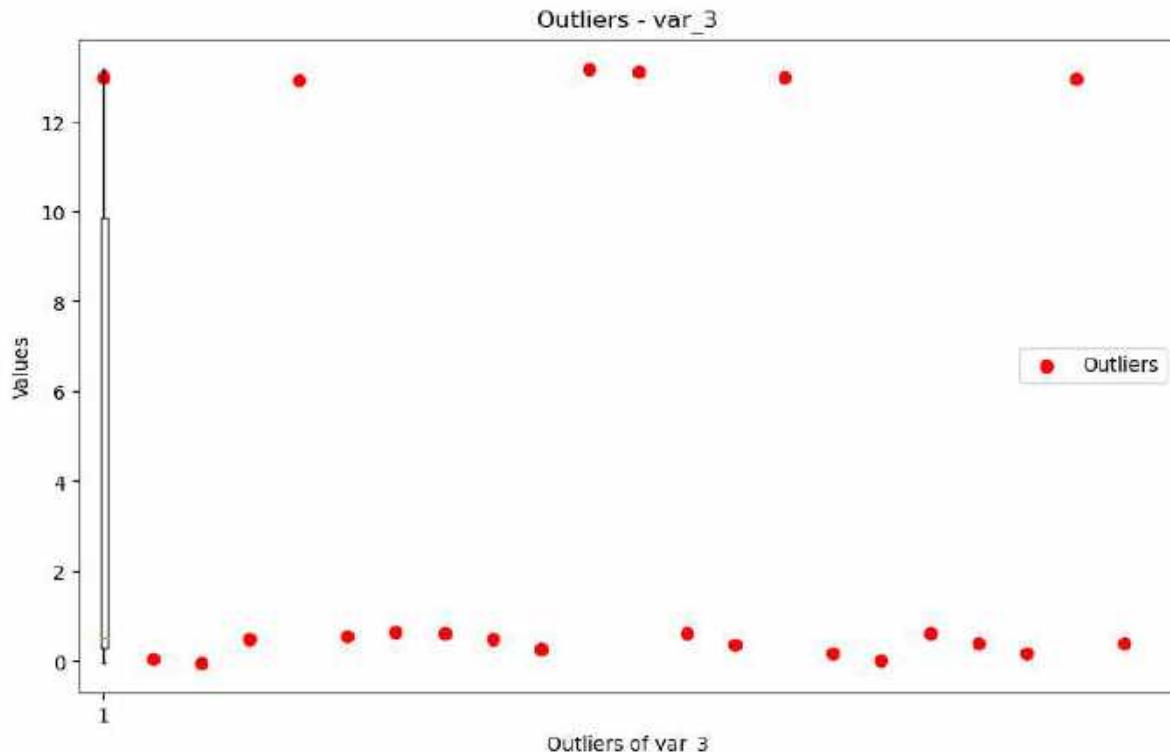
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.
```

```
    warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_2: 49

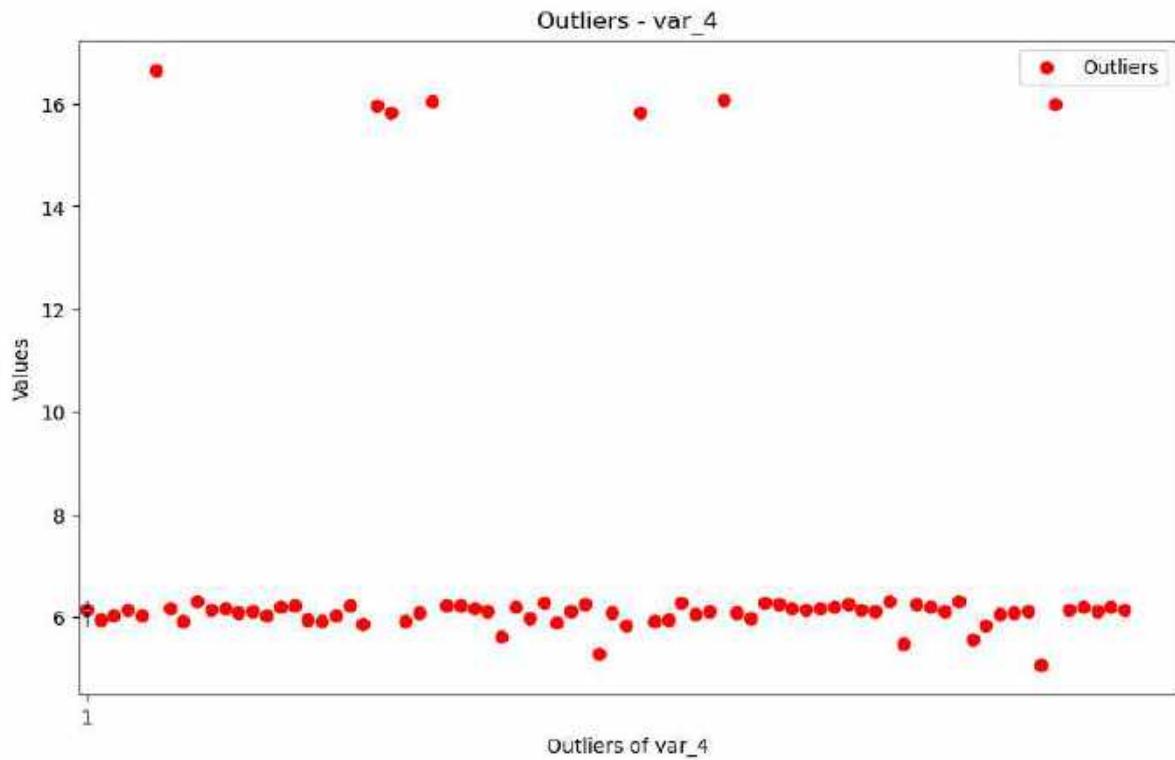
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.
```

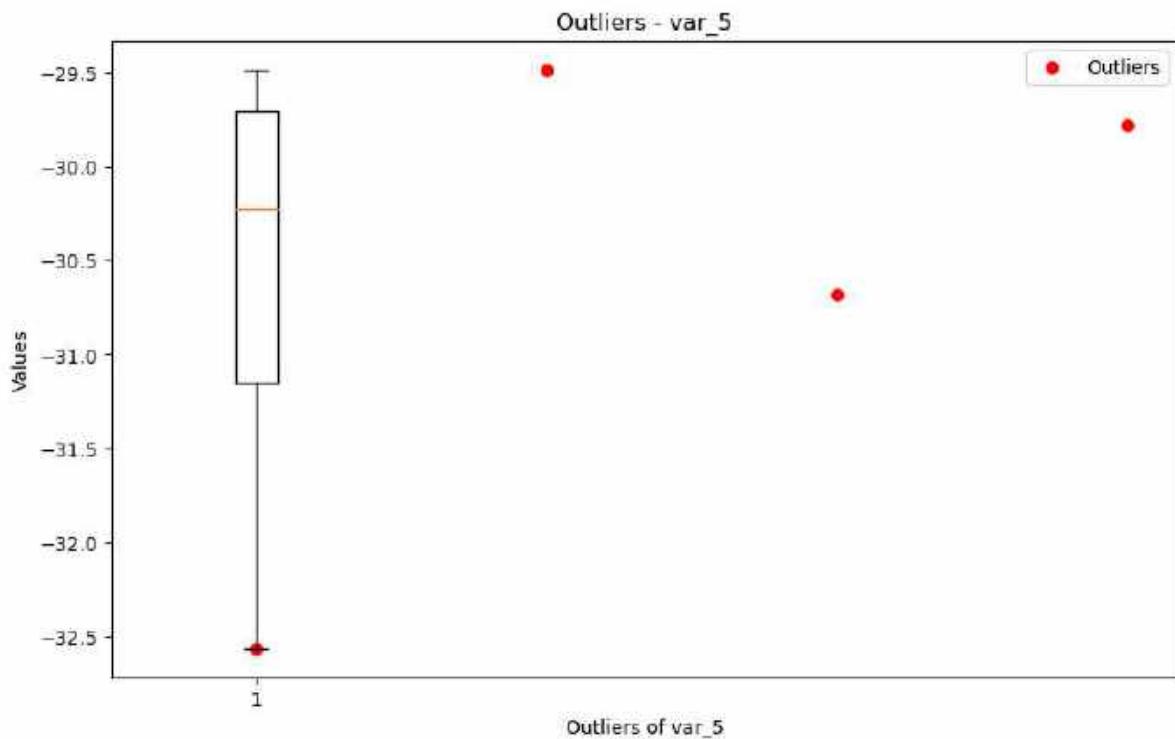
```
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_3: 22



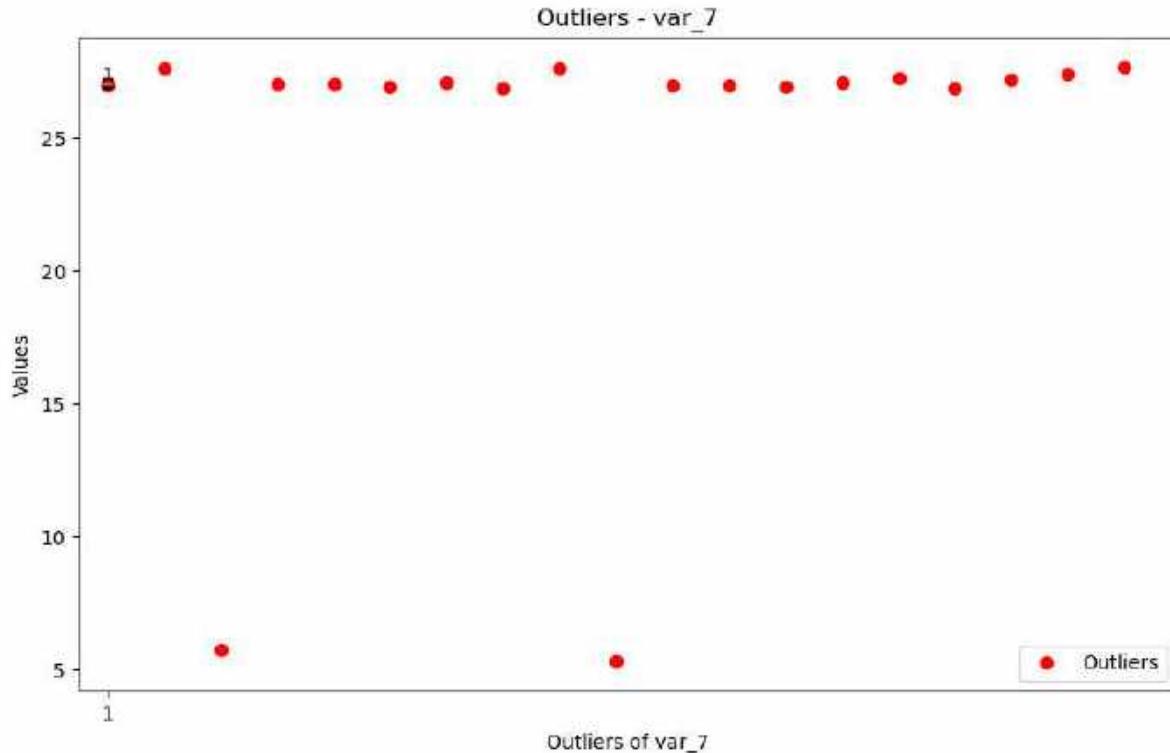
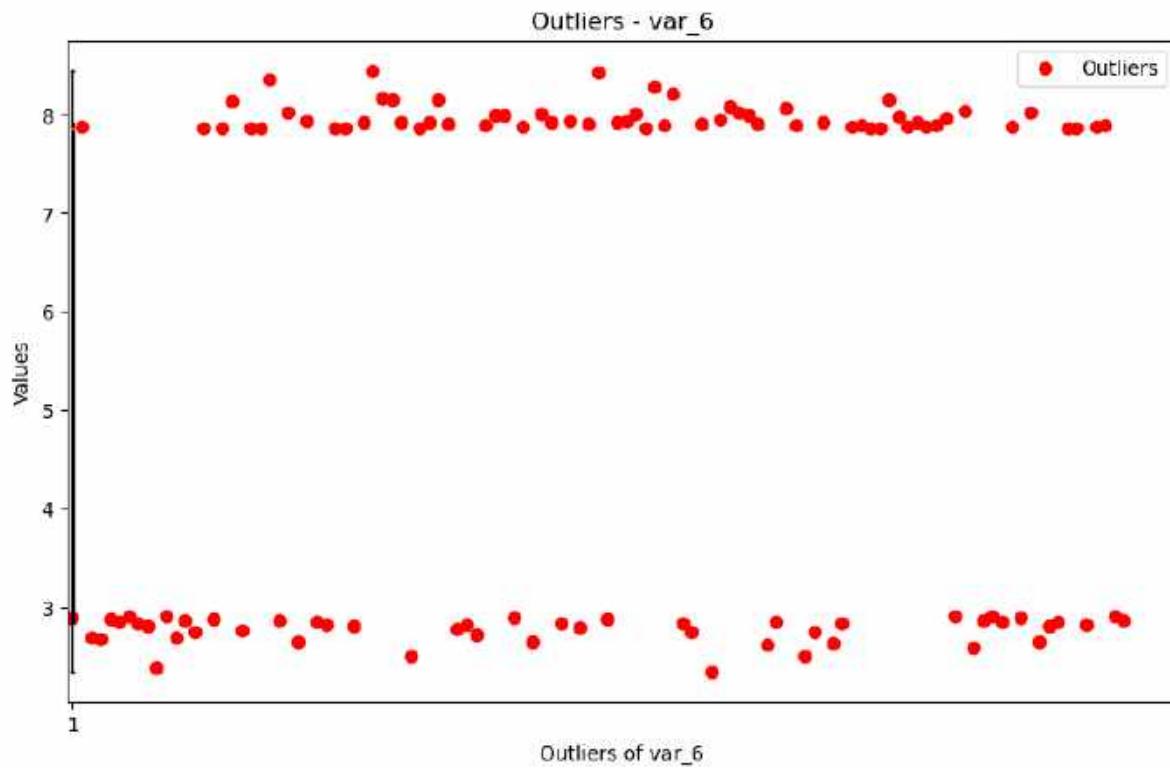
Total outliers in column var\_4: 76

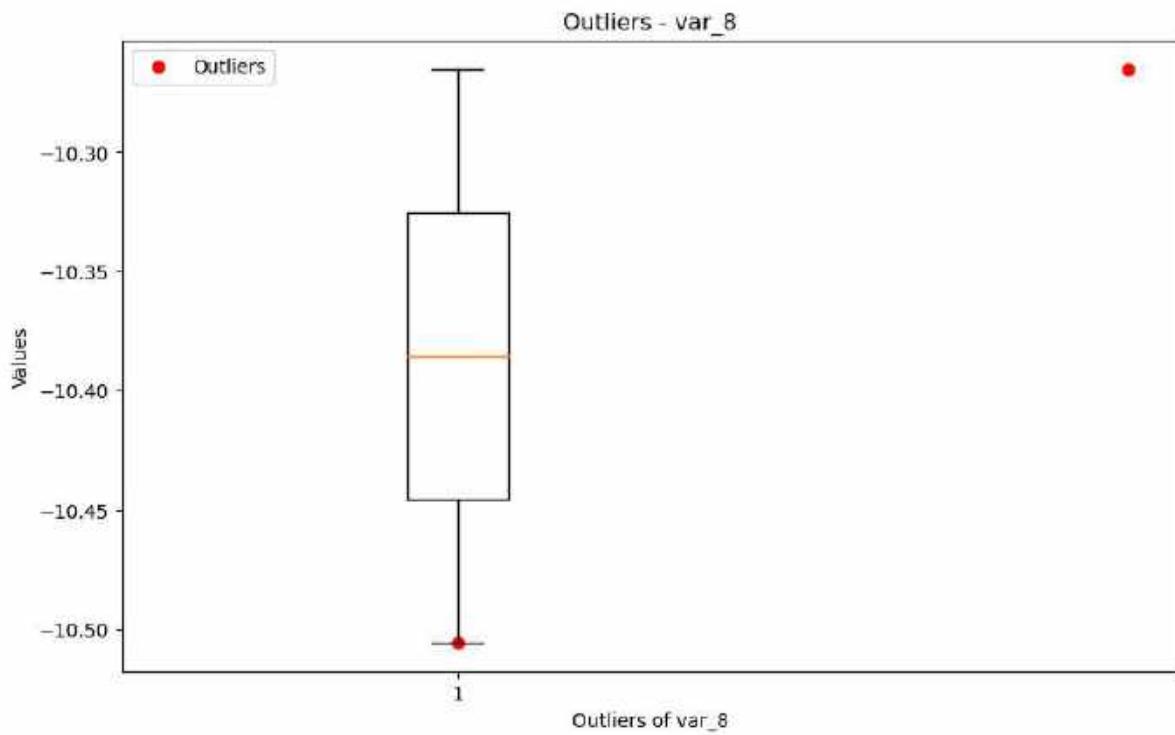
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_5: 4

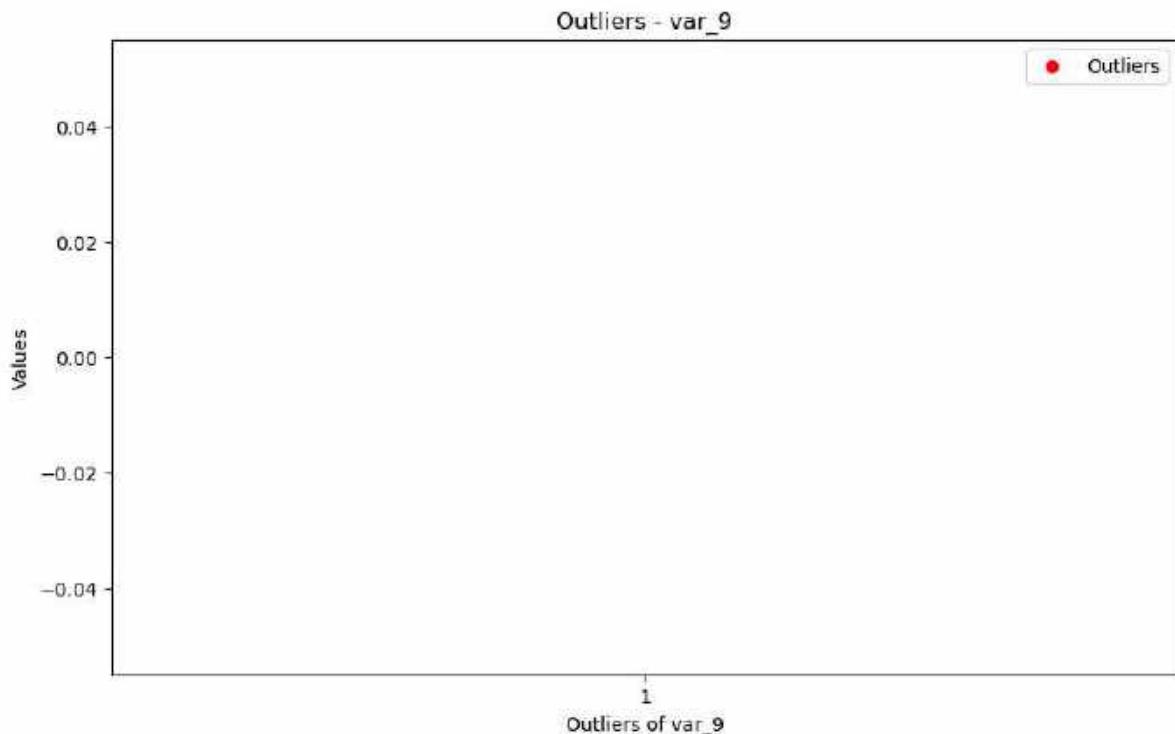
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





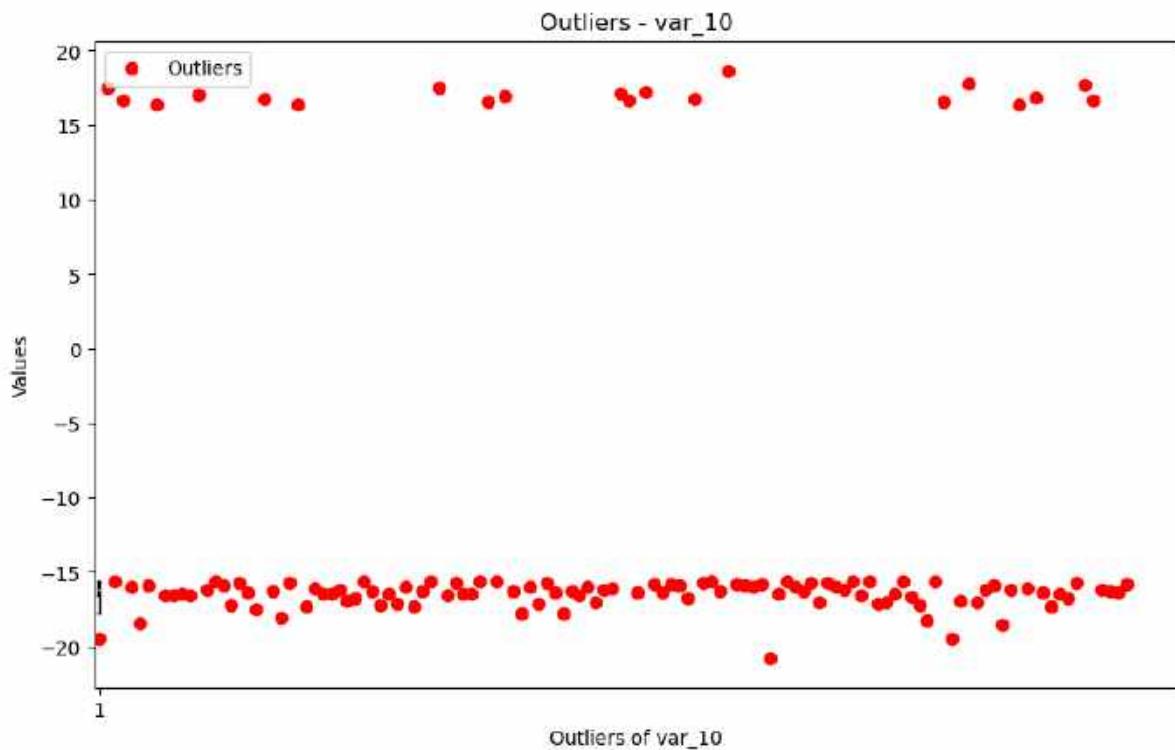
Total outliers in column var\_8: 2

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

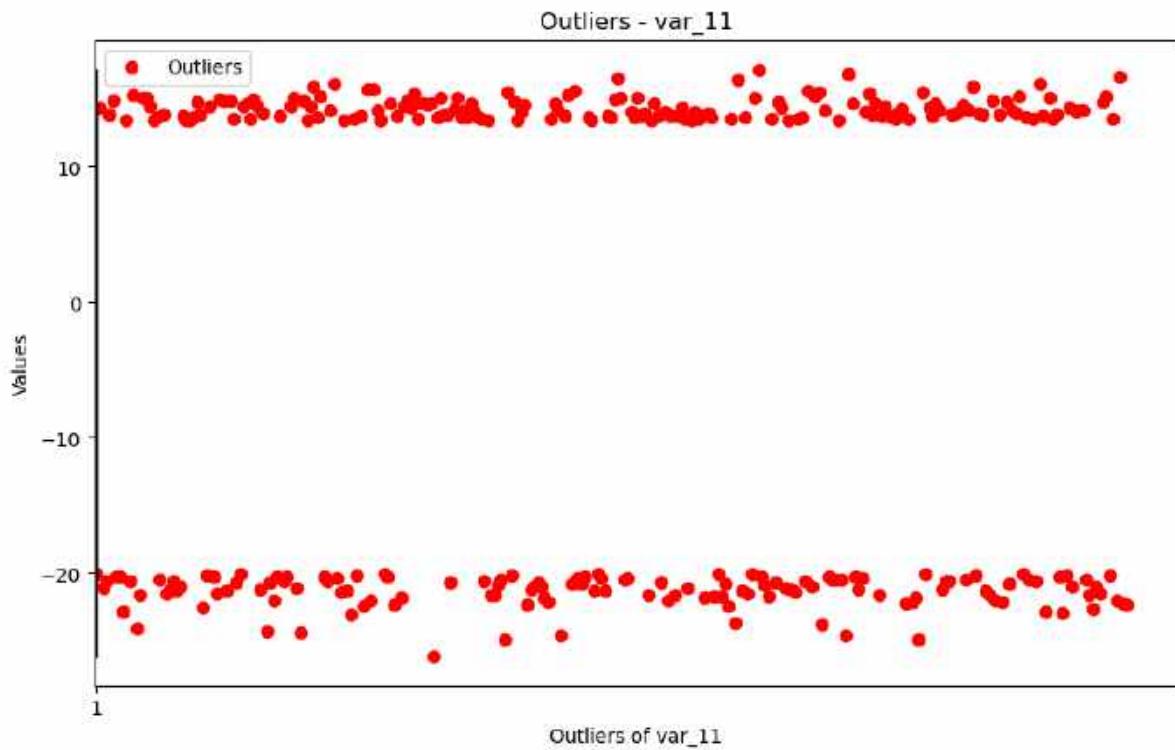


Total outliers in column var\_9: 0

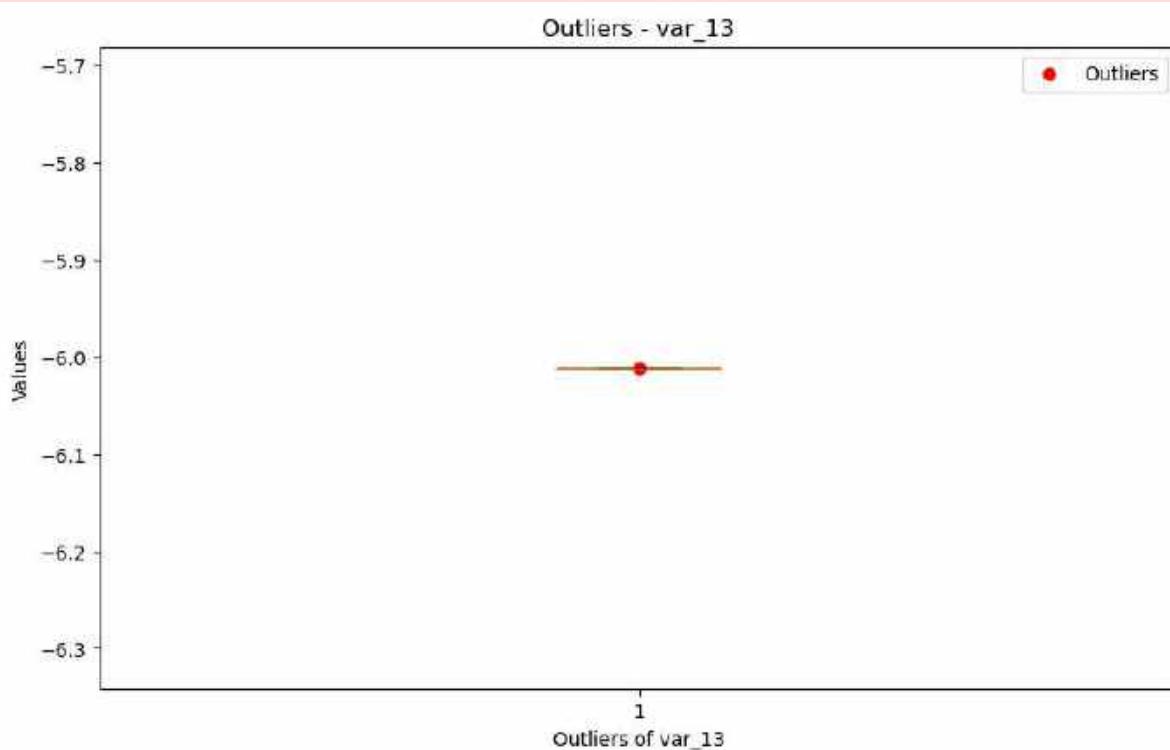
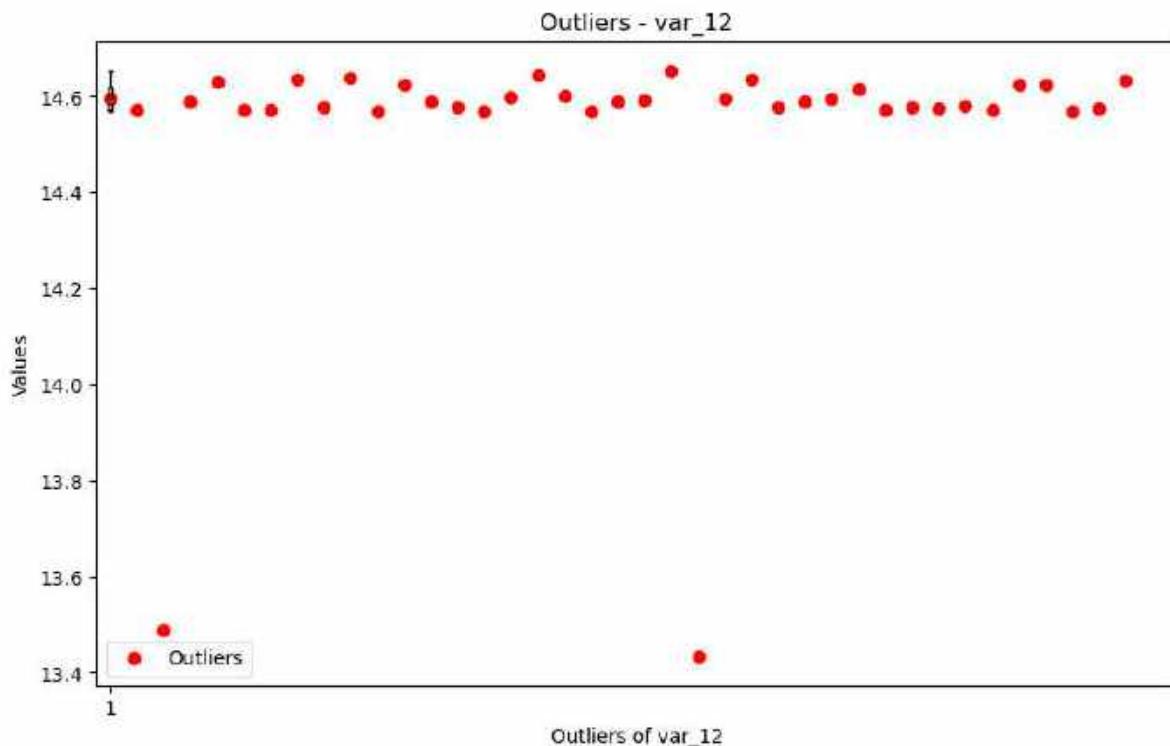
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

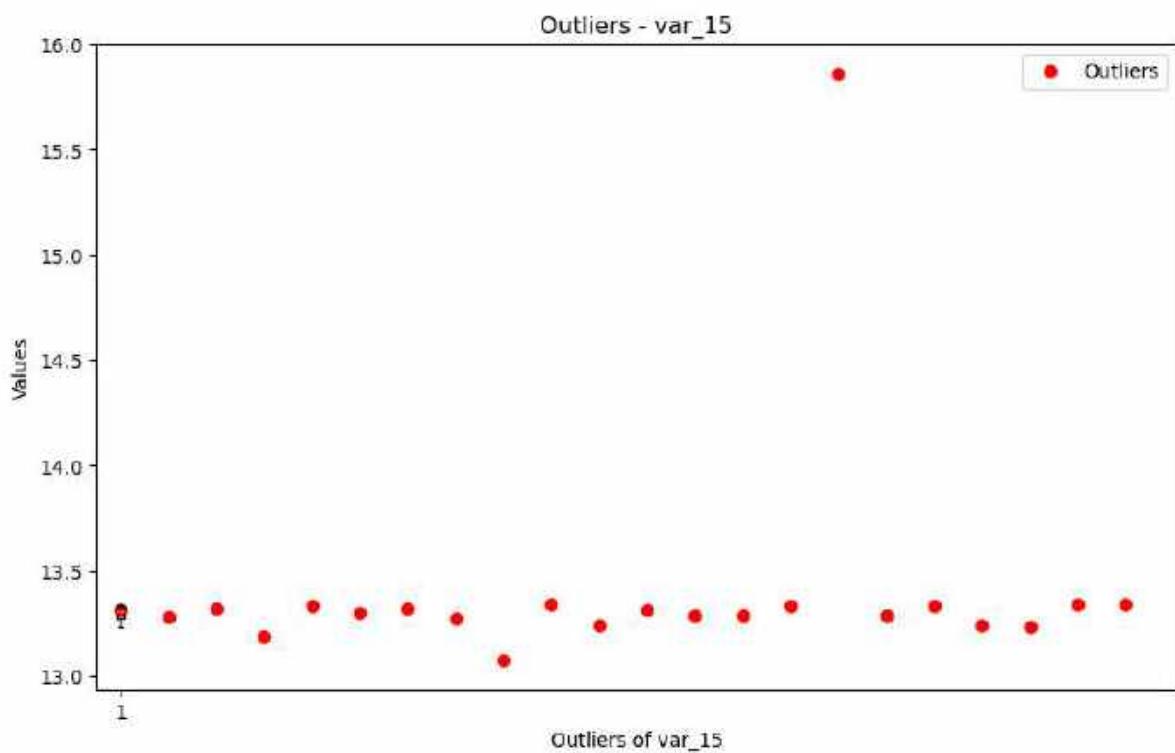
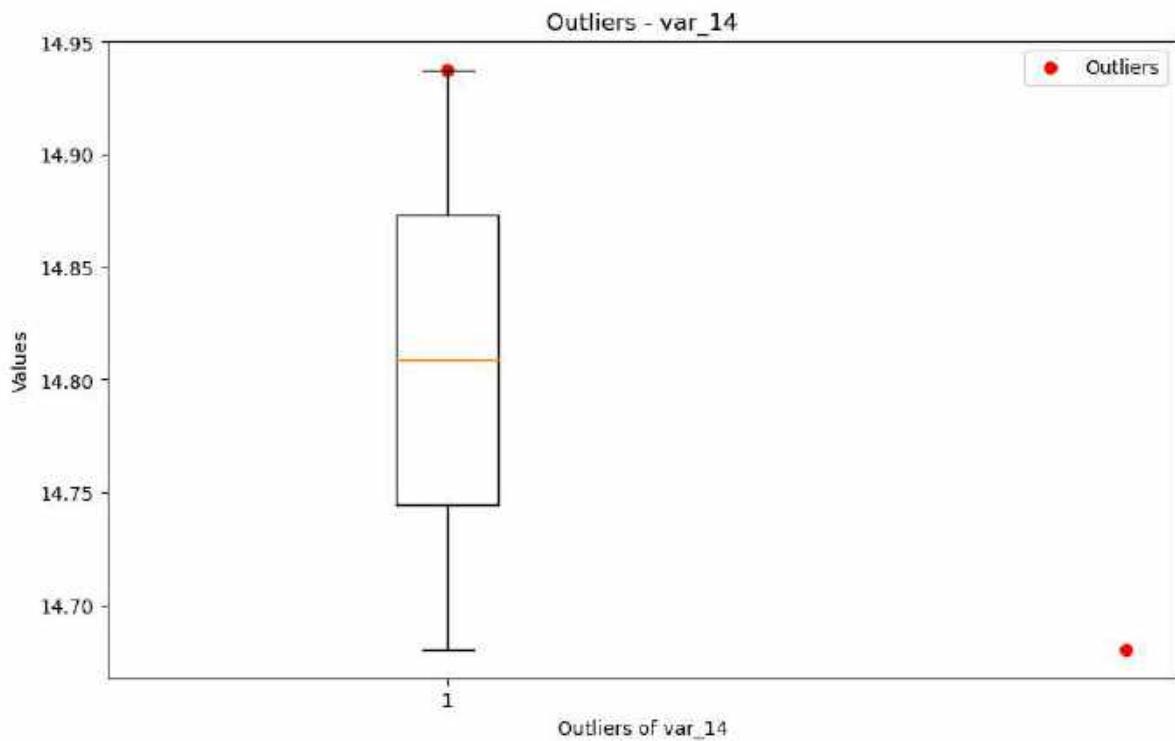


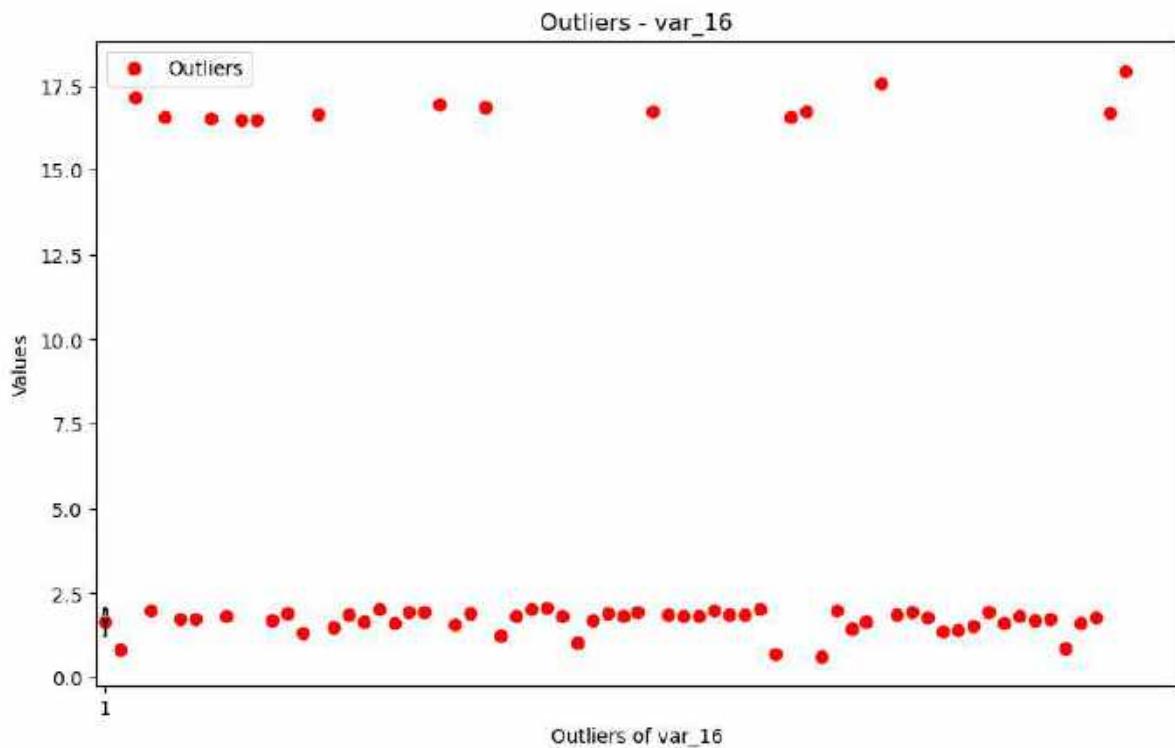
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_10: 125
```



```
Total outliers in column var_11: 309  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

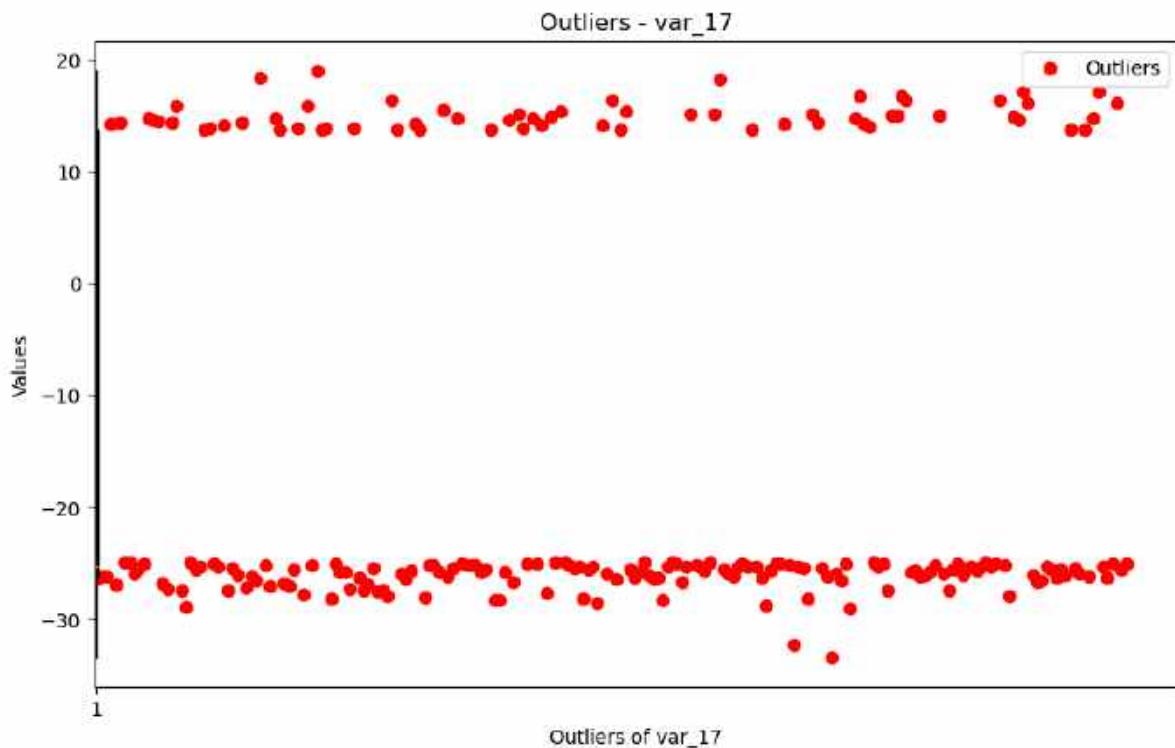






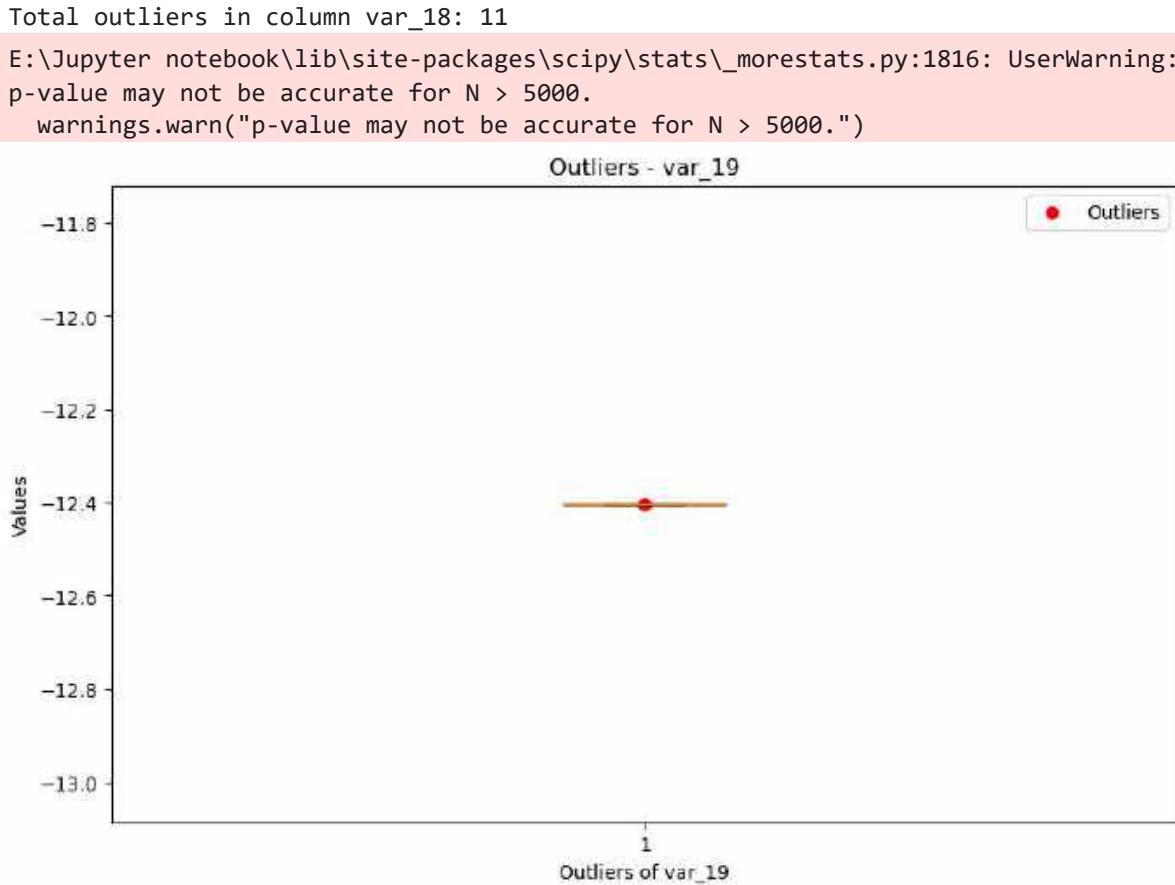
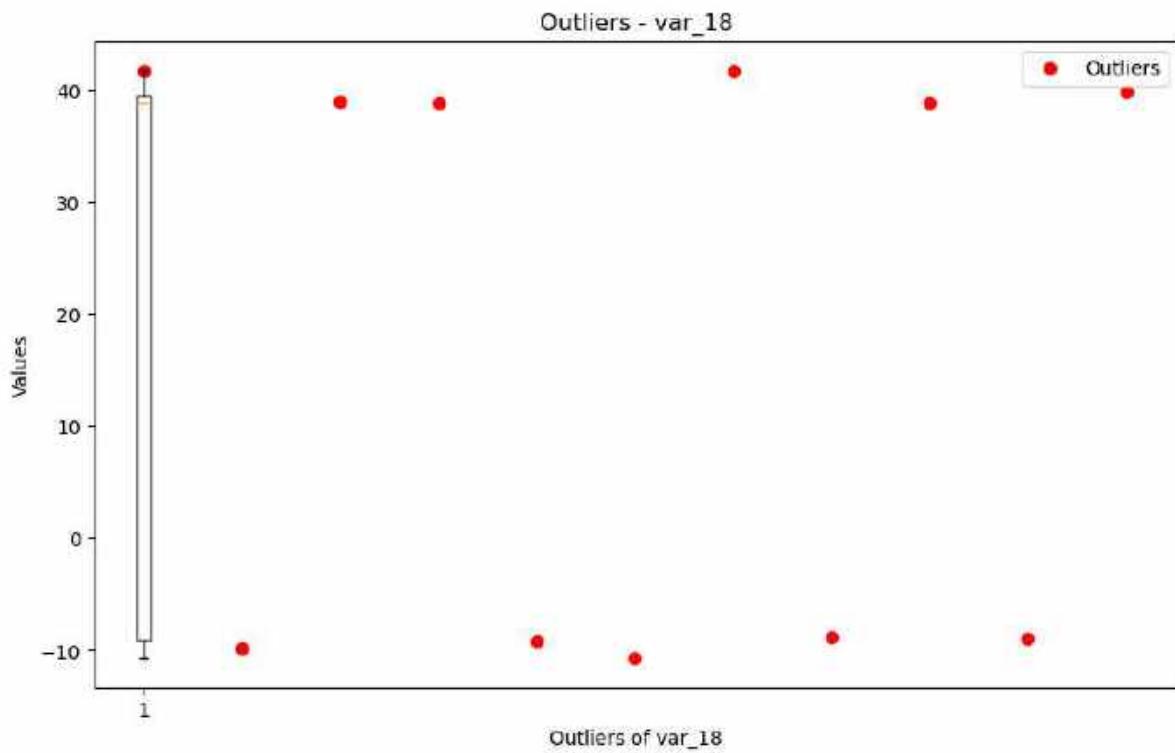
Total outliers in column var\_16: 68

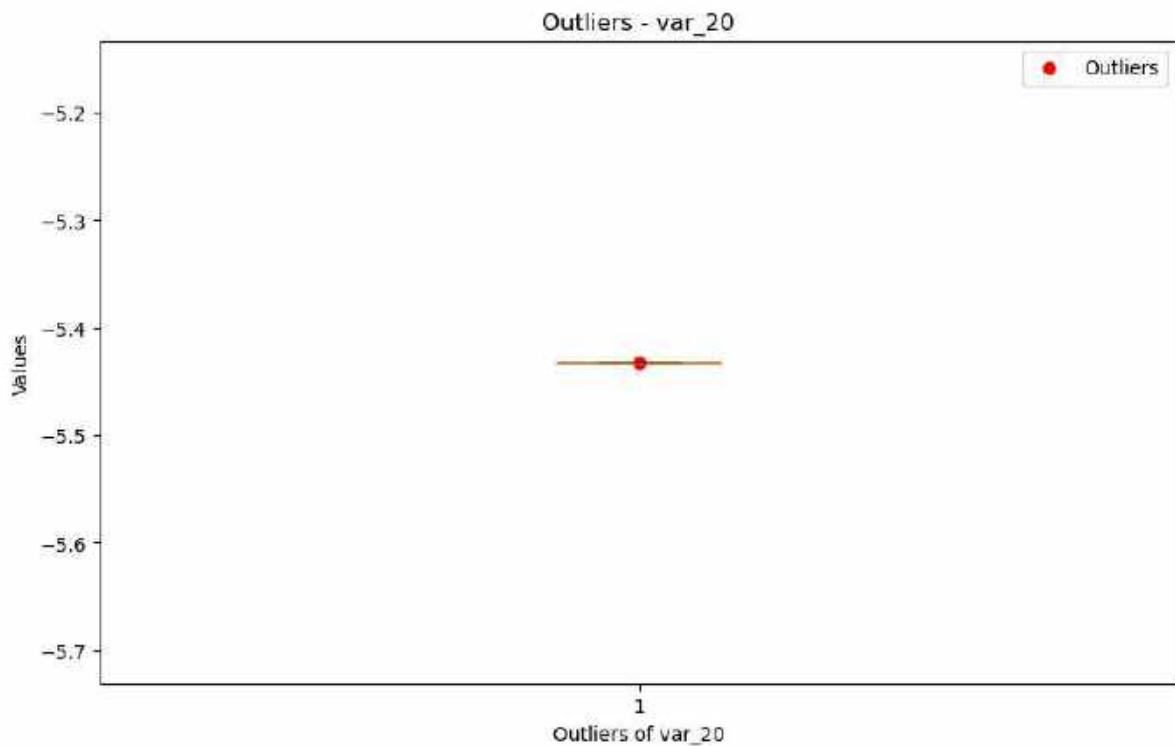
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_17: 221

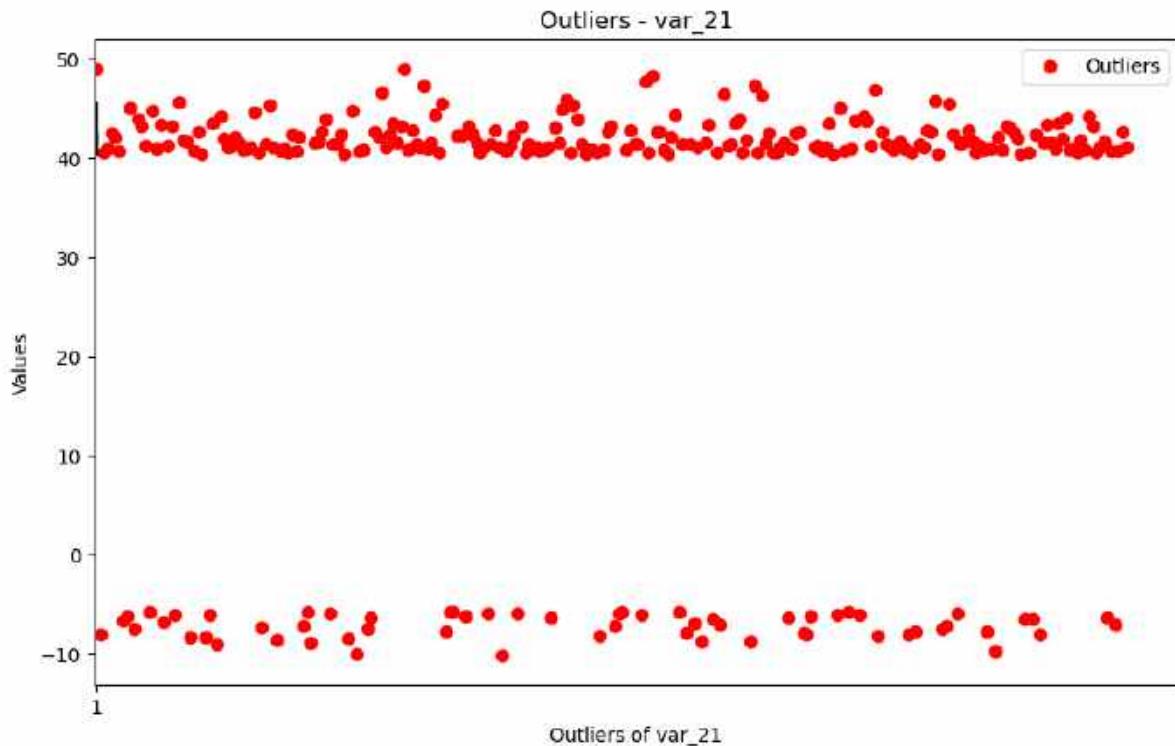
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





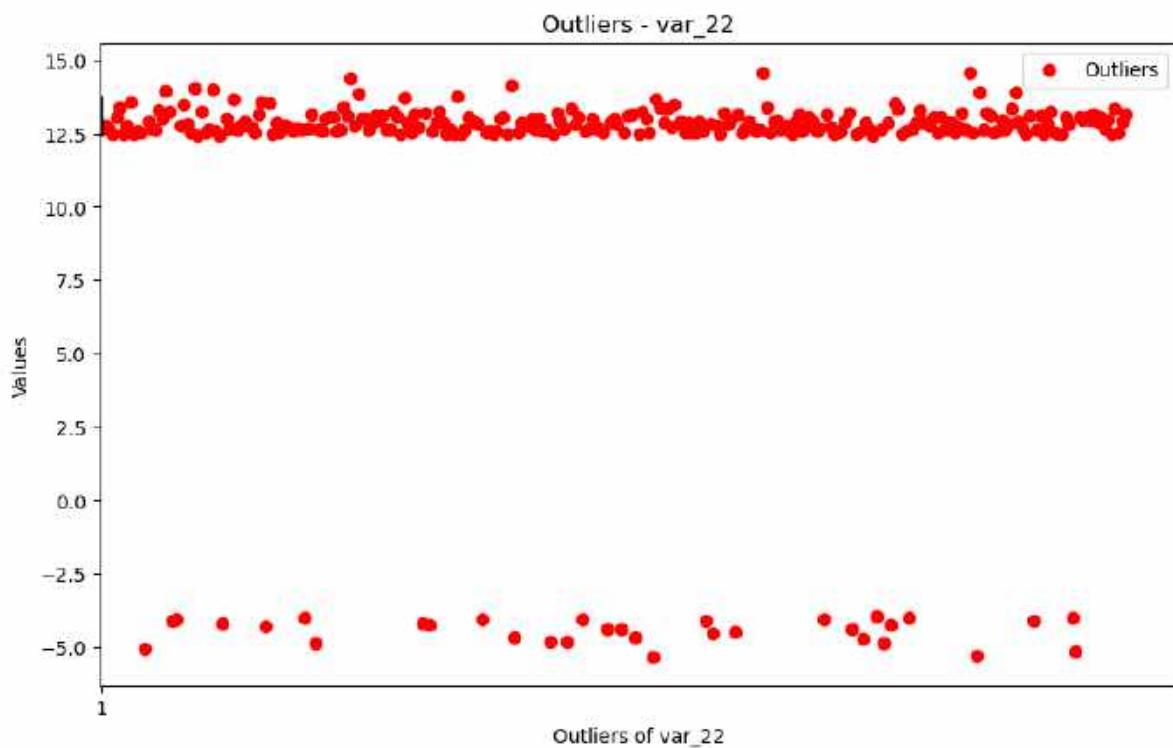
Total outliers in column var\_20: 1

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_21: 275

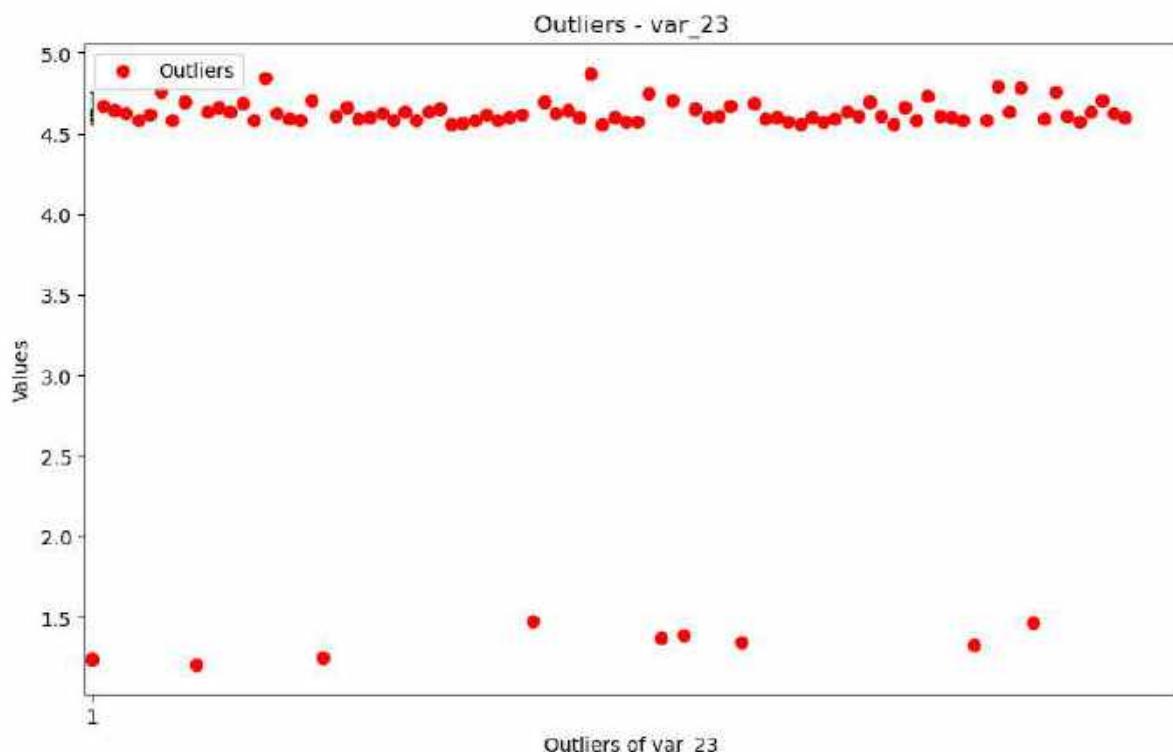
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.
```

```
    warnings.warn("p-value may not be accurate for N > 5000.")
```

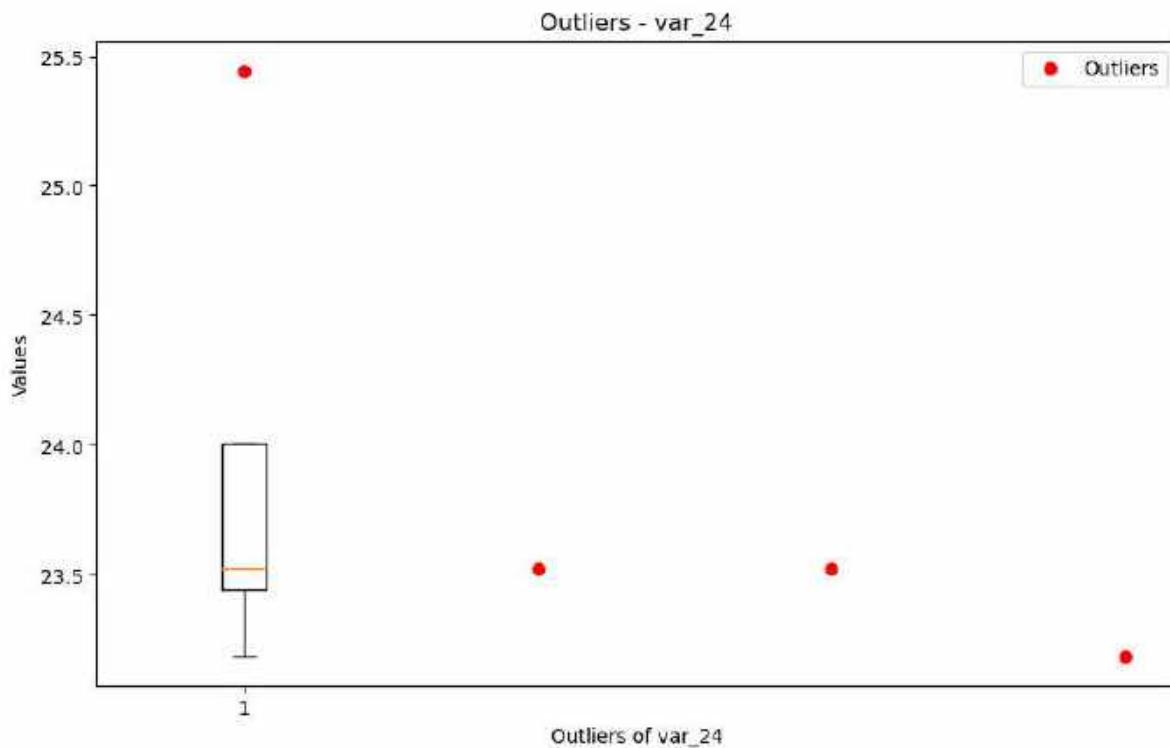
Total outliers in column var\_22: 289



Total outliers in column var\_23: 90

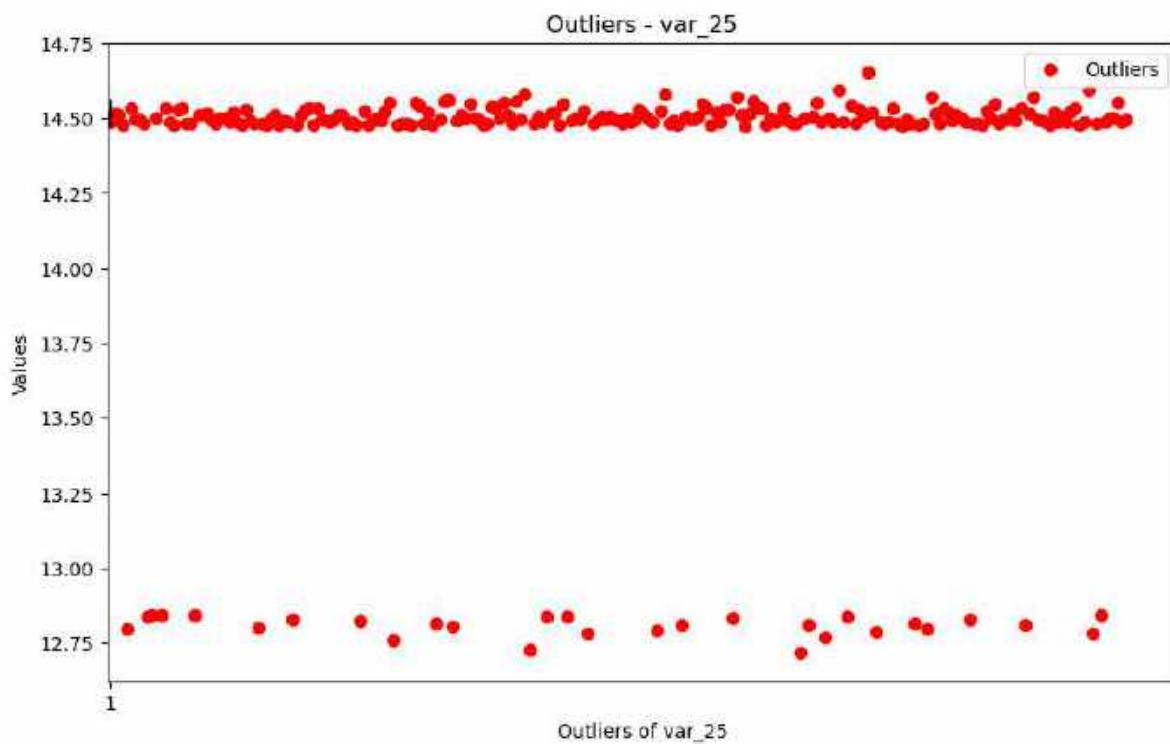
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.
```

```
    warnings.warn("p-value may not be accurate for N > 5000.")
```



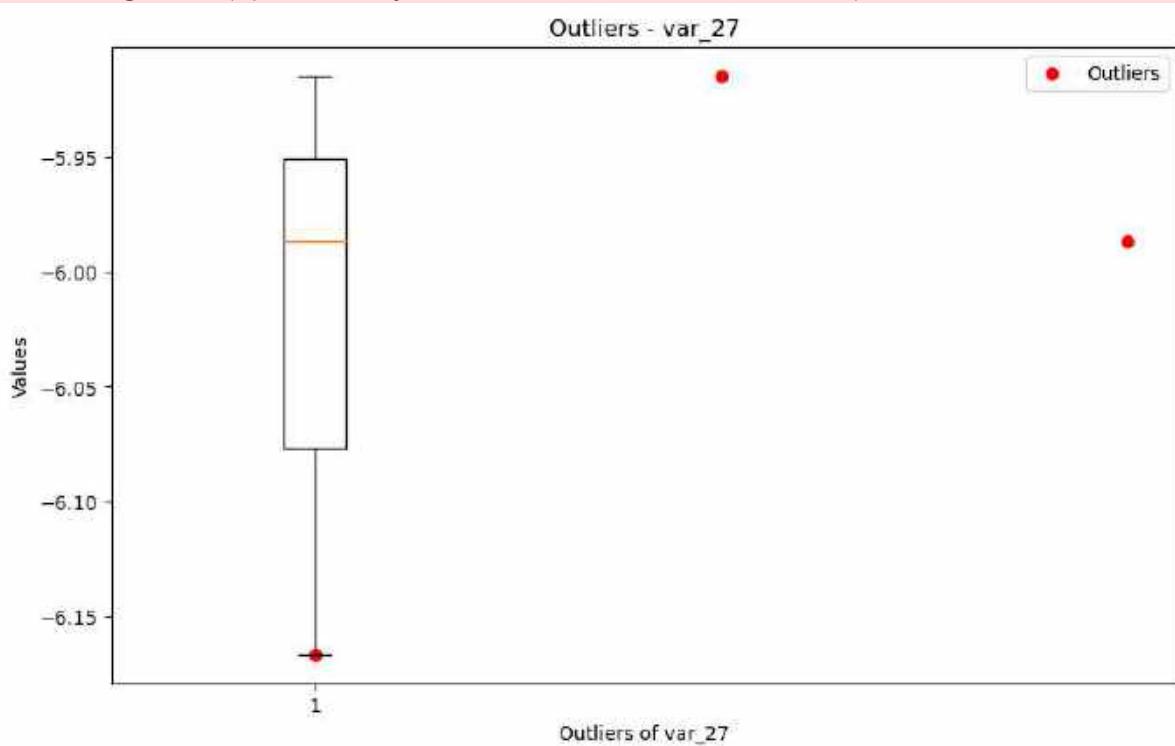
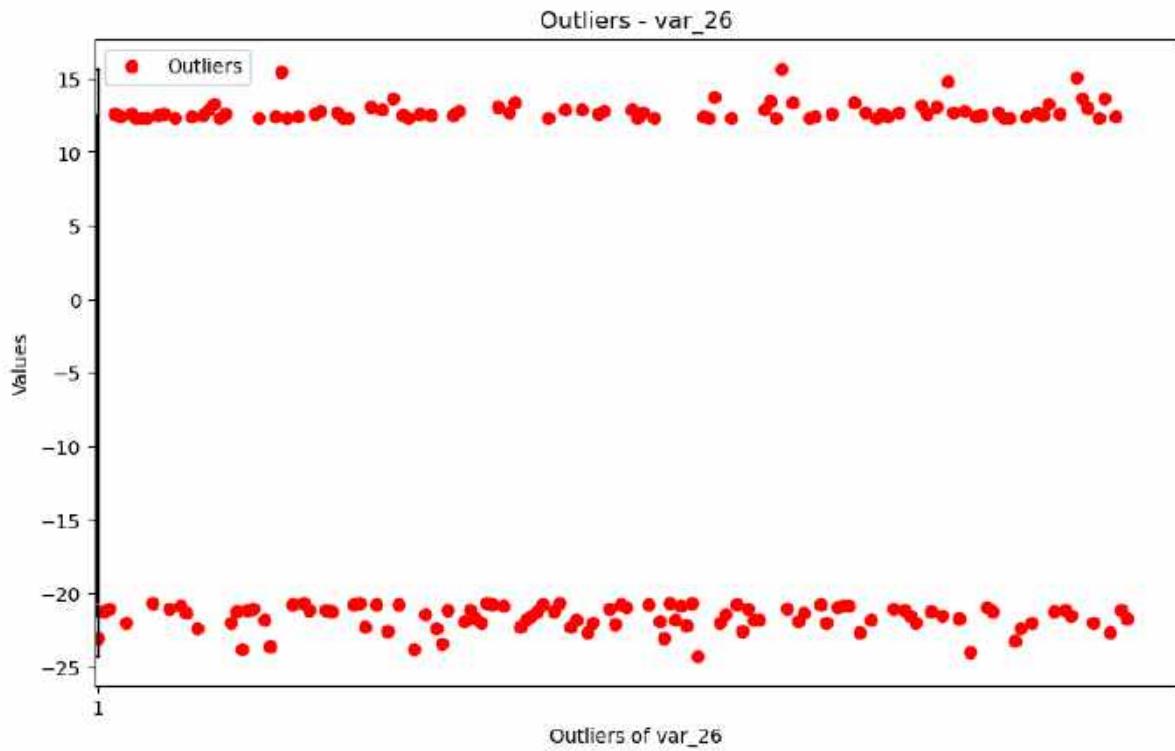
Total outliers in column var\_24: 4

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



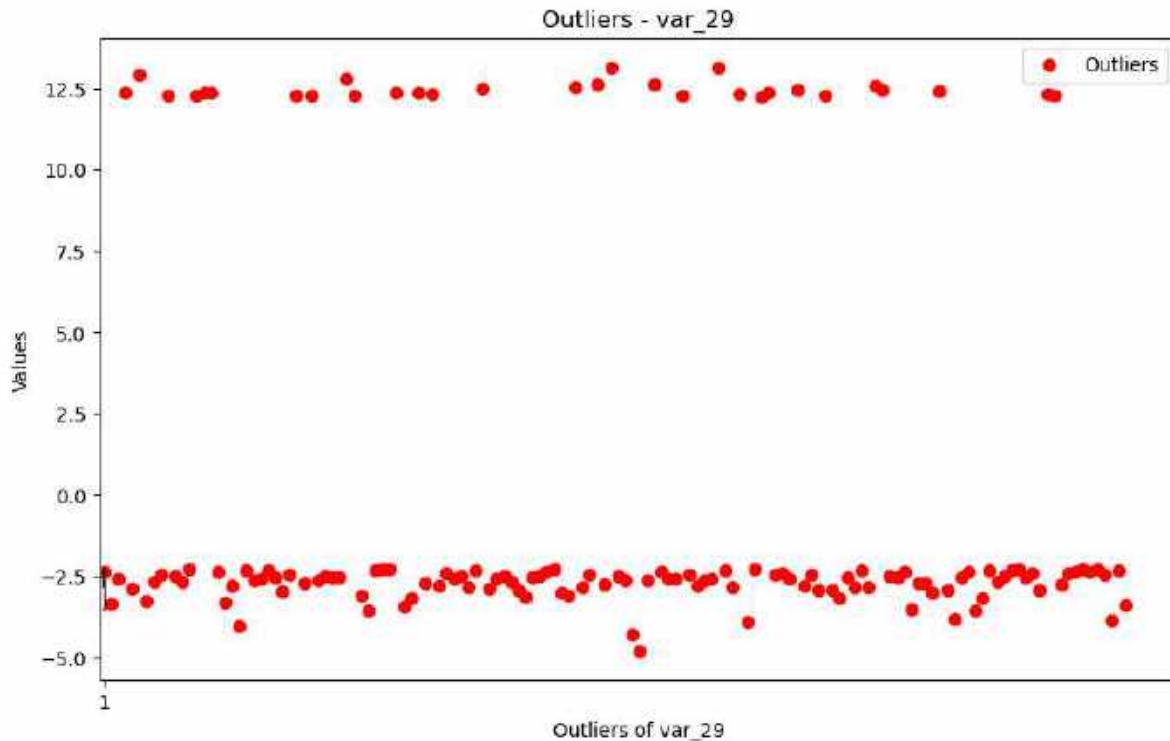
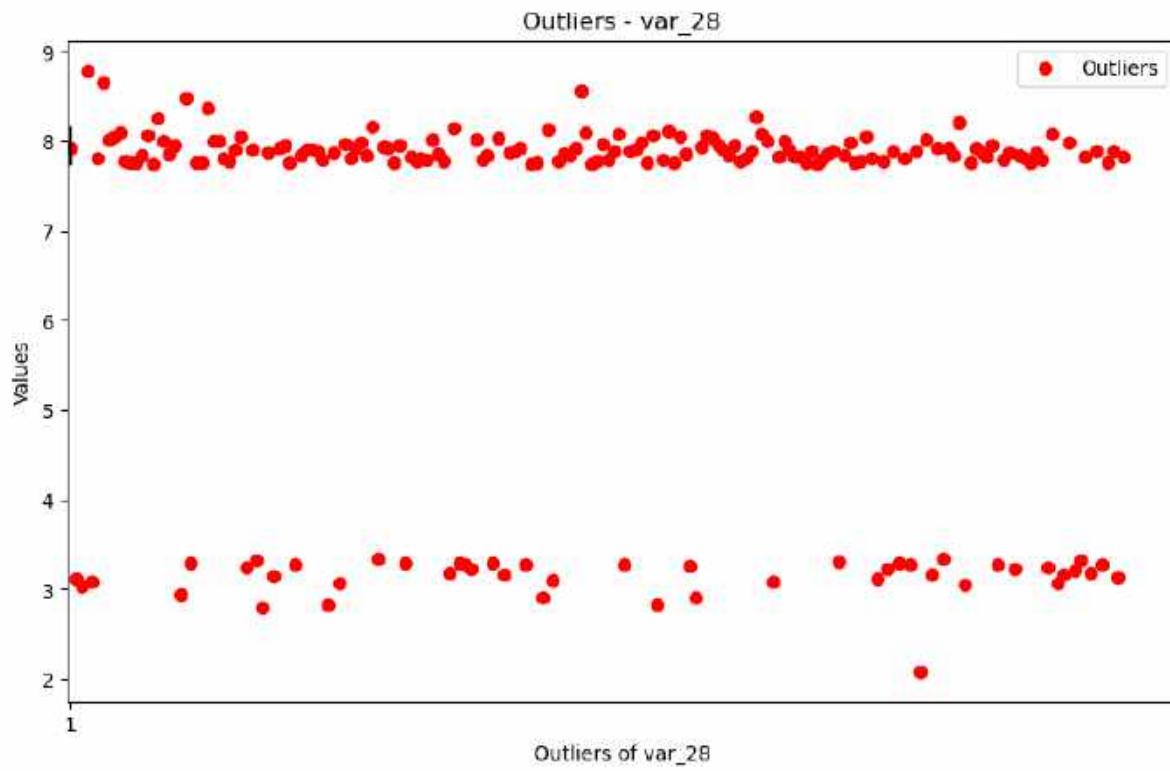
Total outliers in column var\_25: 241

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

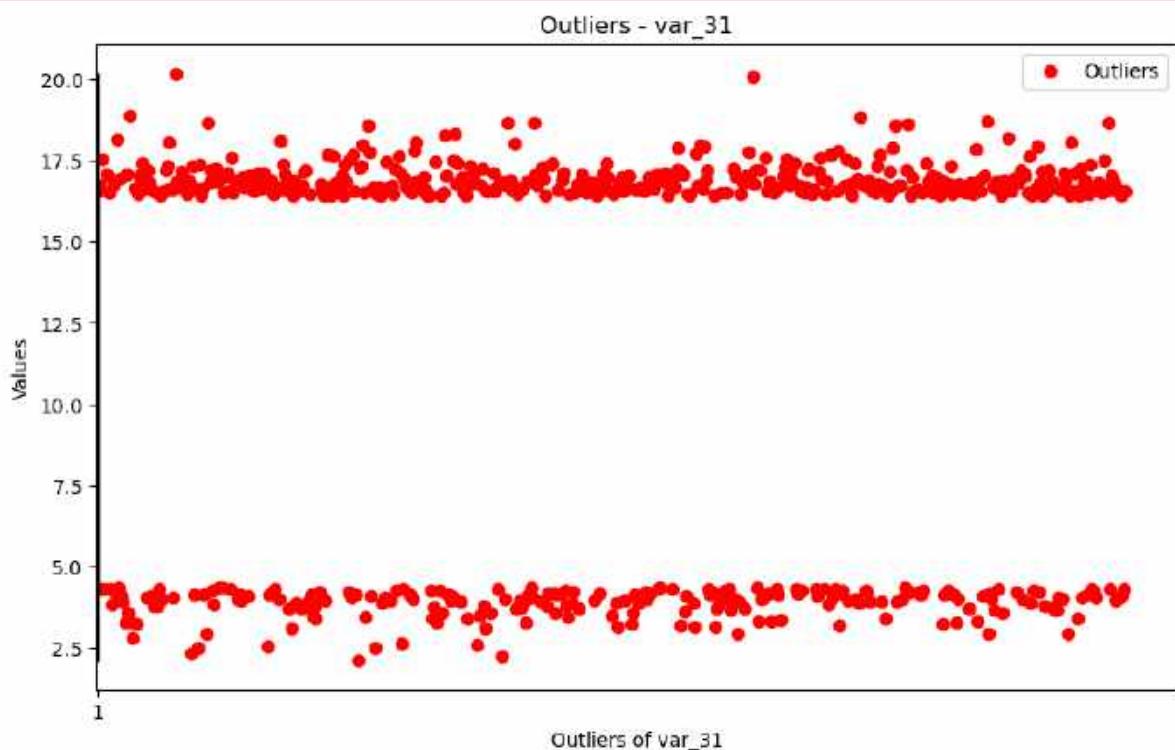
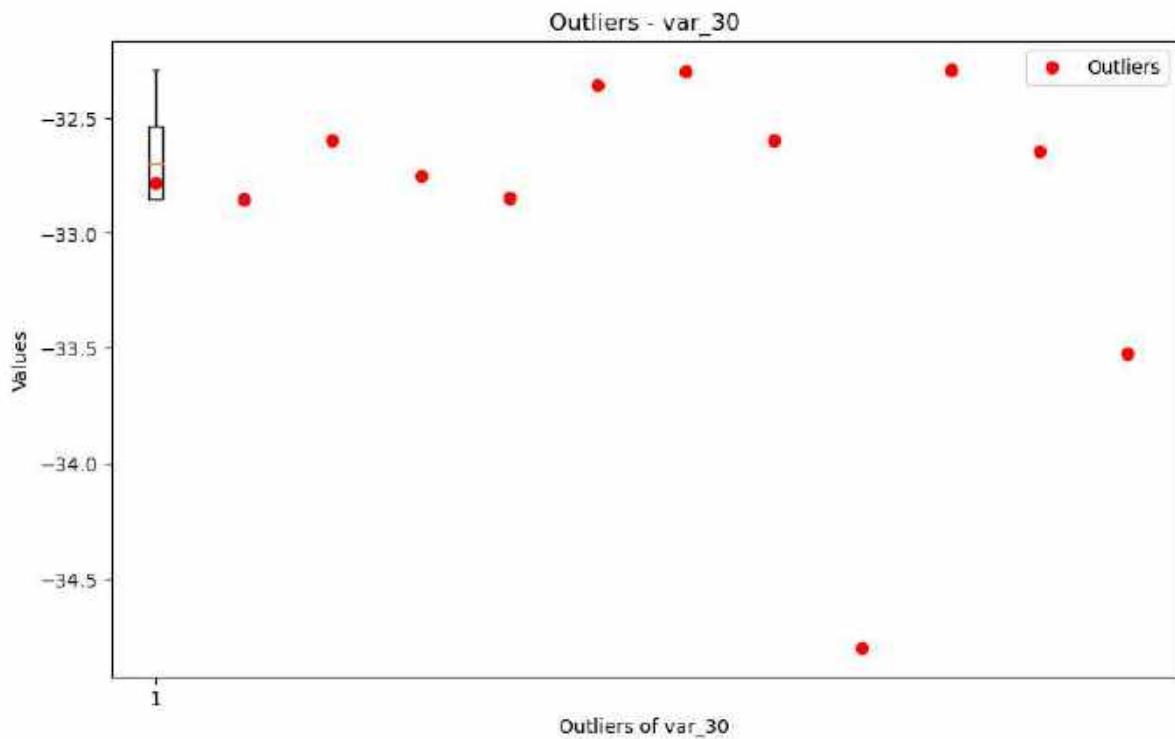


Total outliers in column var\_27: 3

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

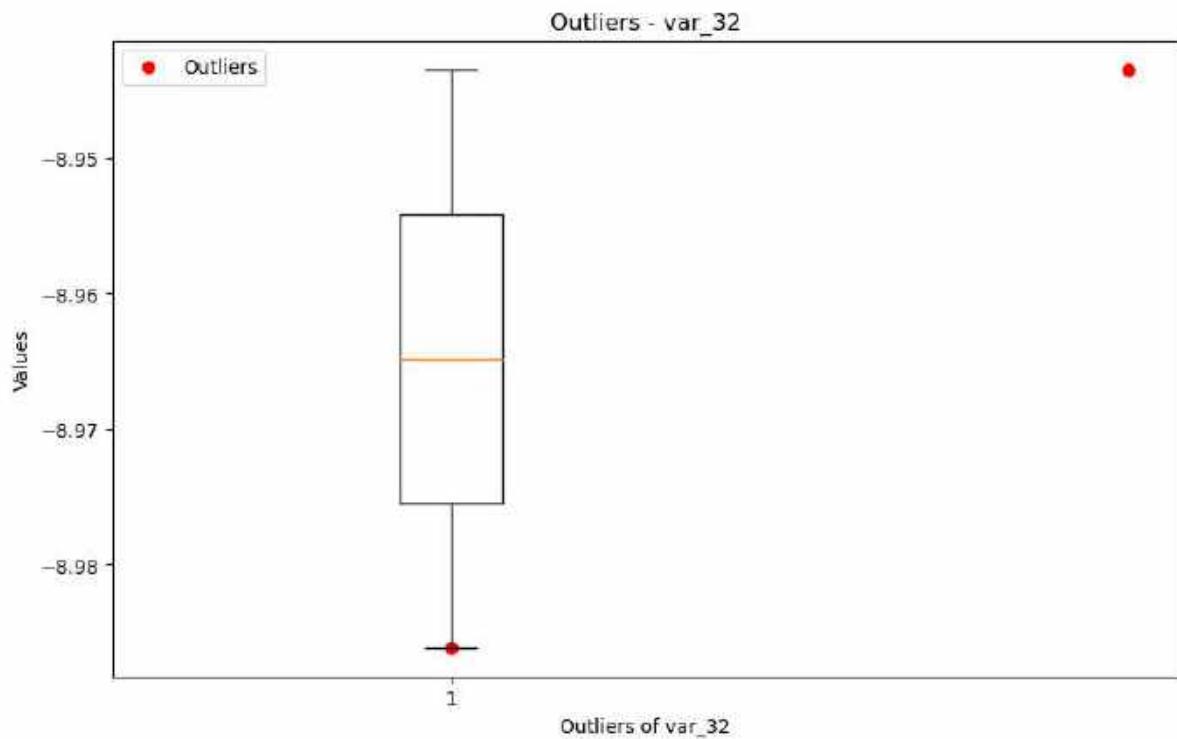


Total outliers in column var\_29: 144



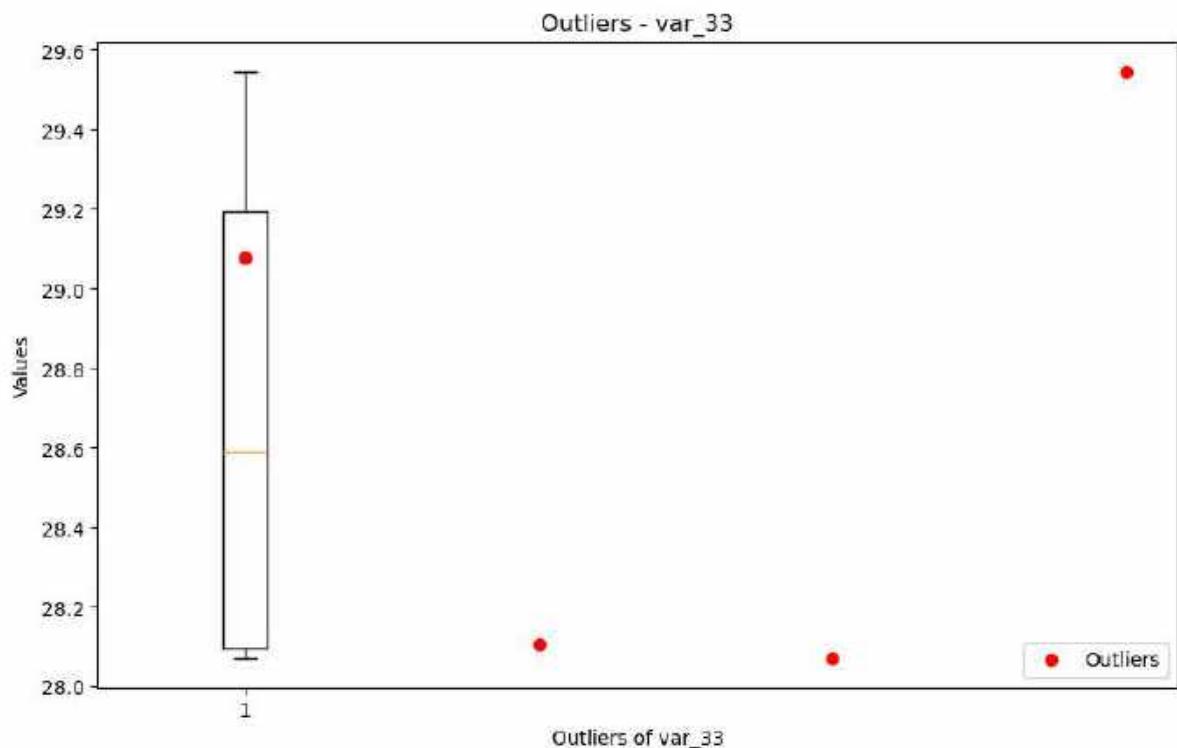
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_31: 63



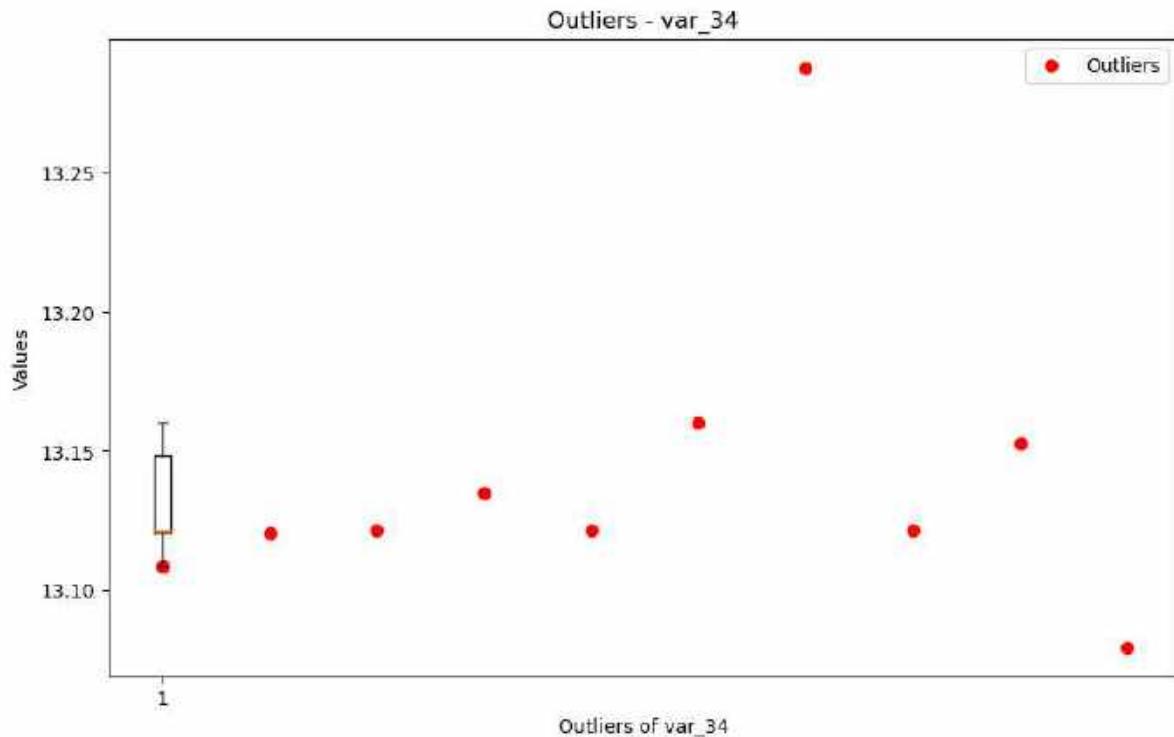
Total outliers in column var\_32: 2

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



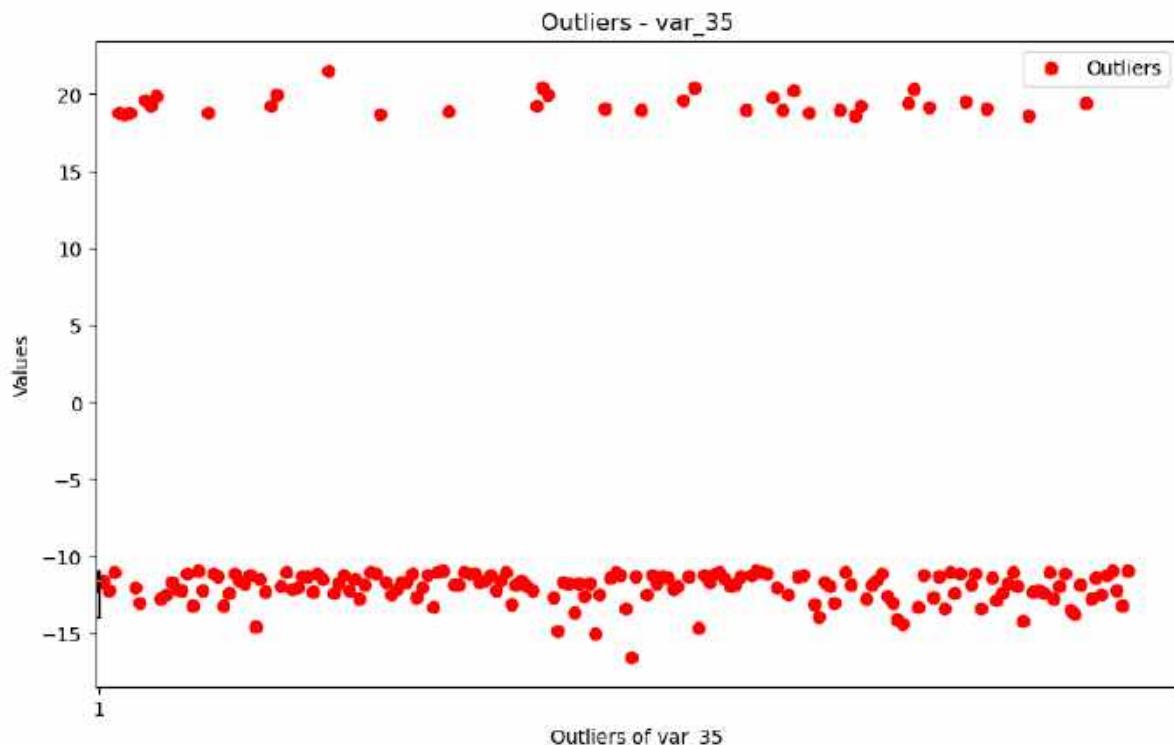
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_33: 4



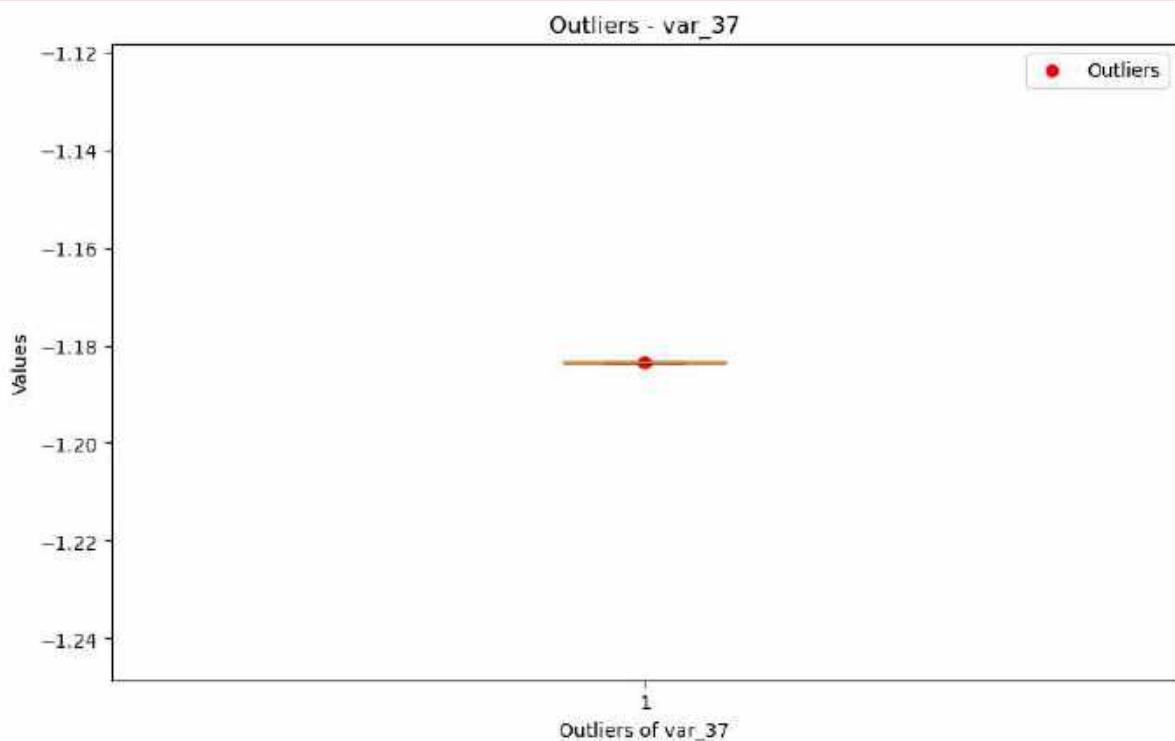
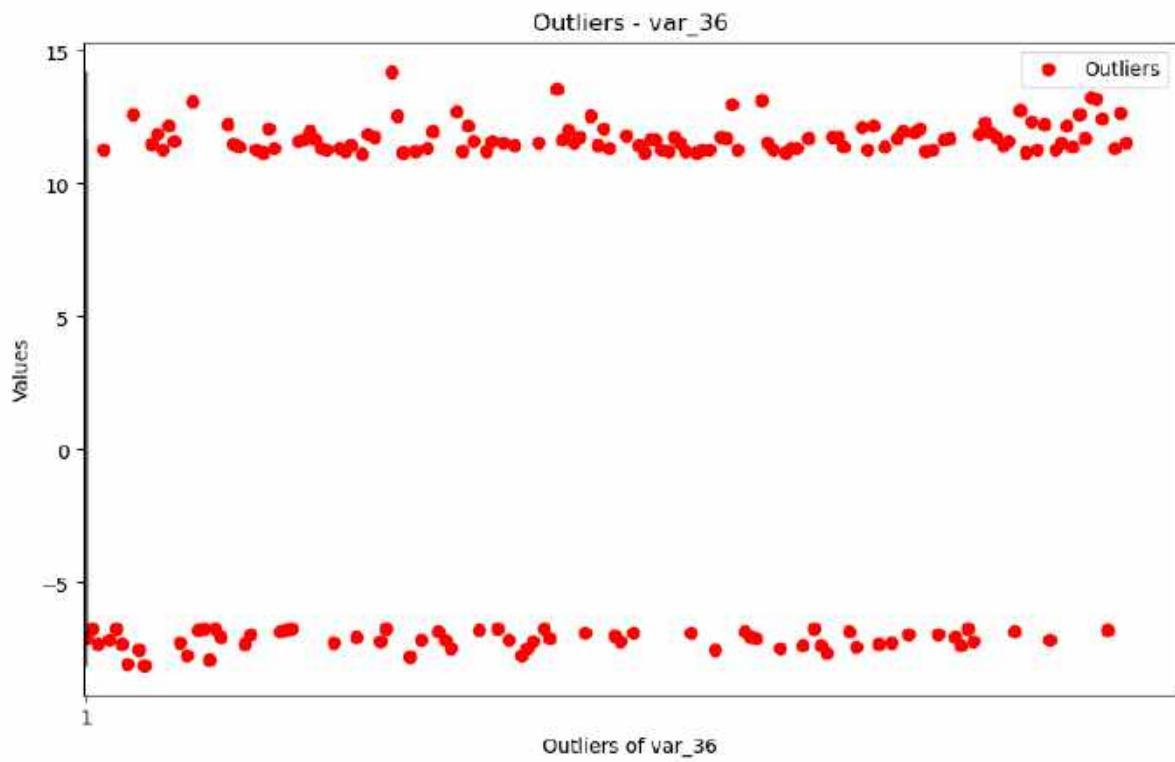
Total outliers in column var\_34: 10

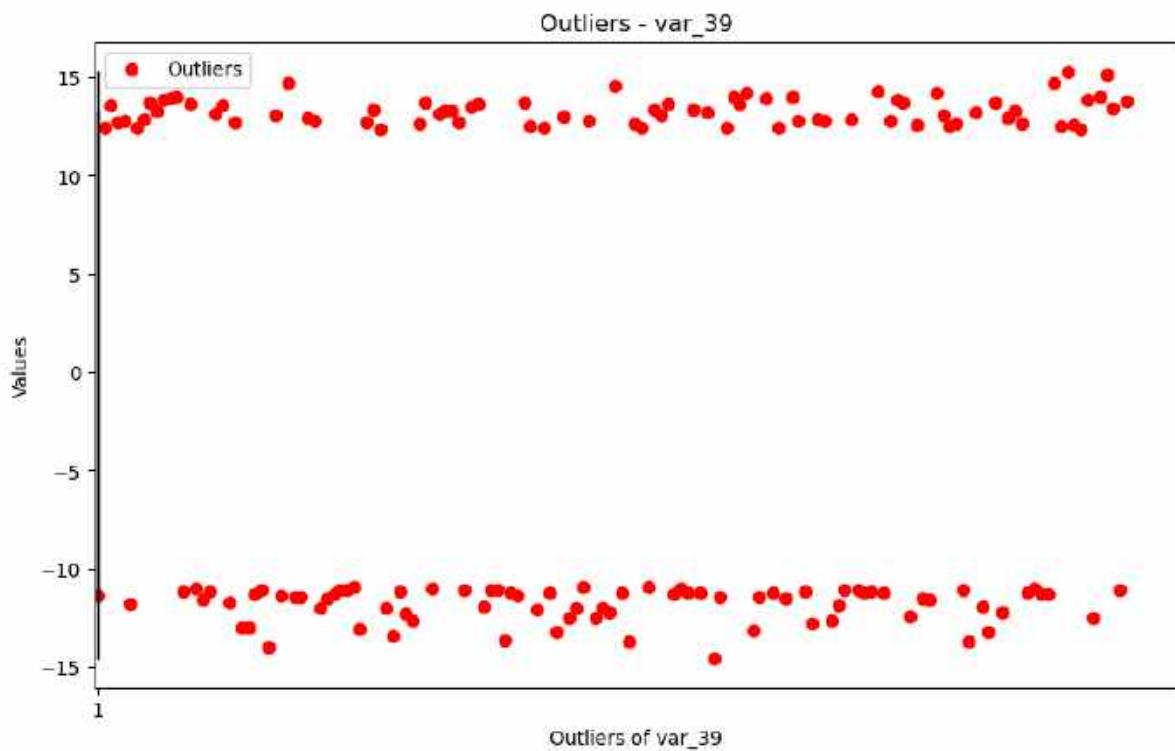
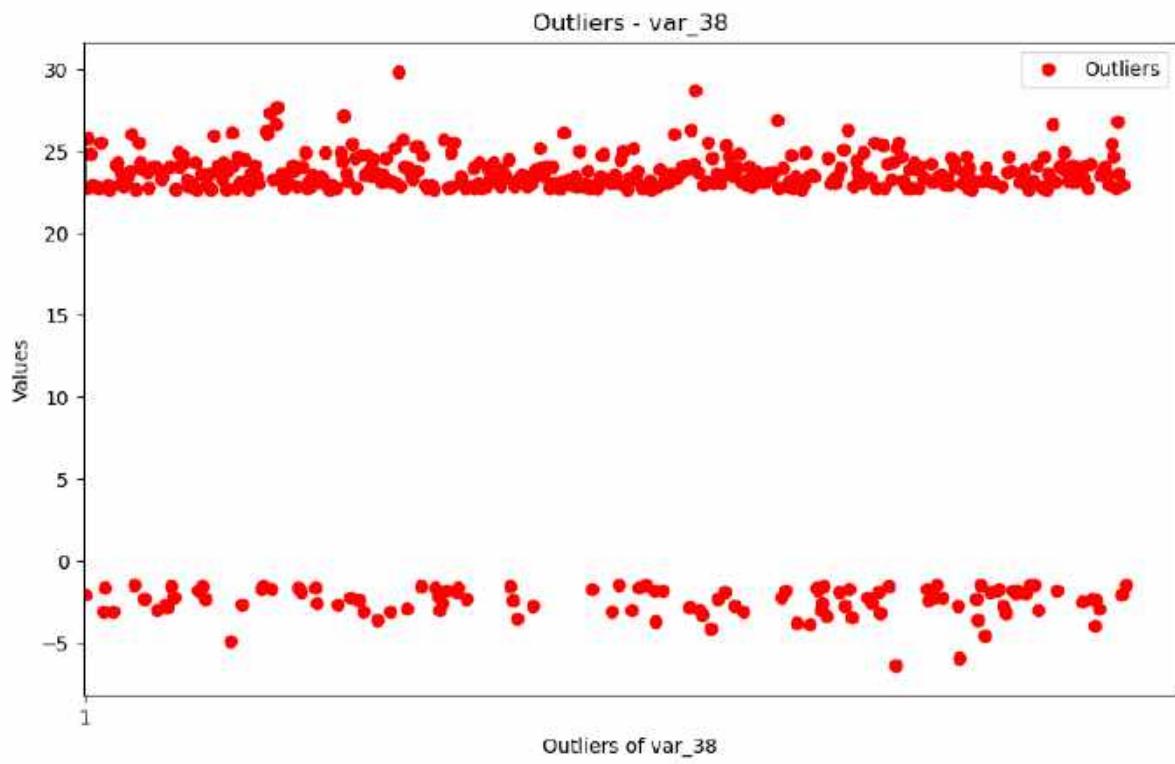
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

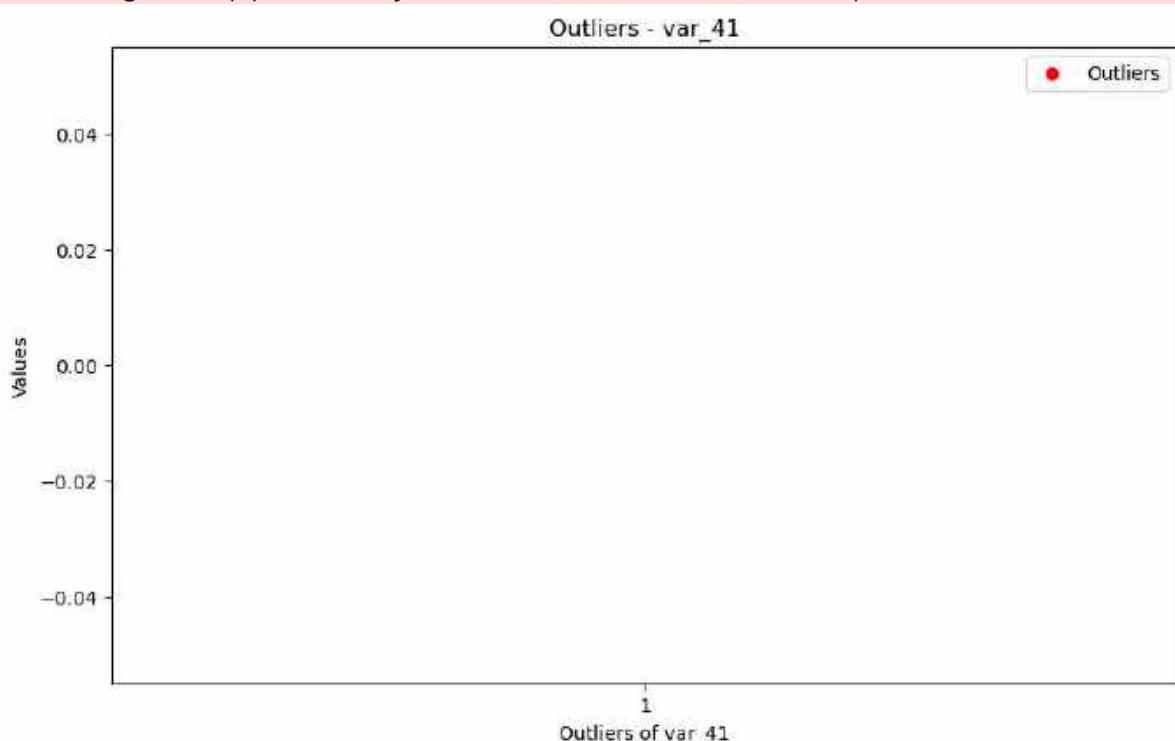
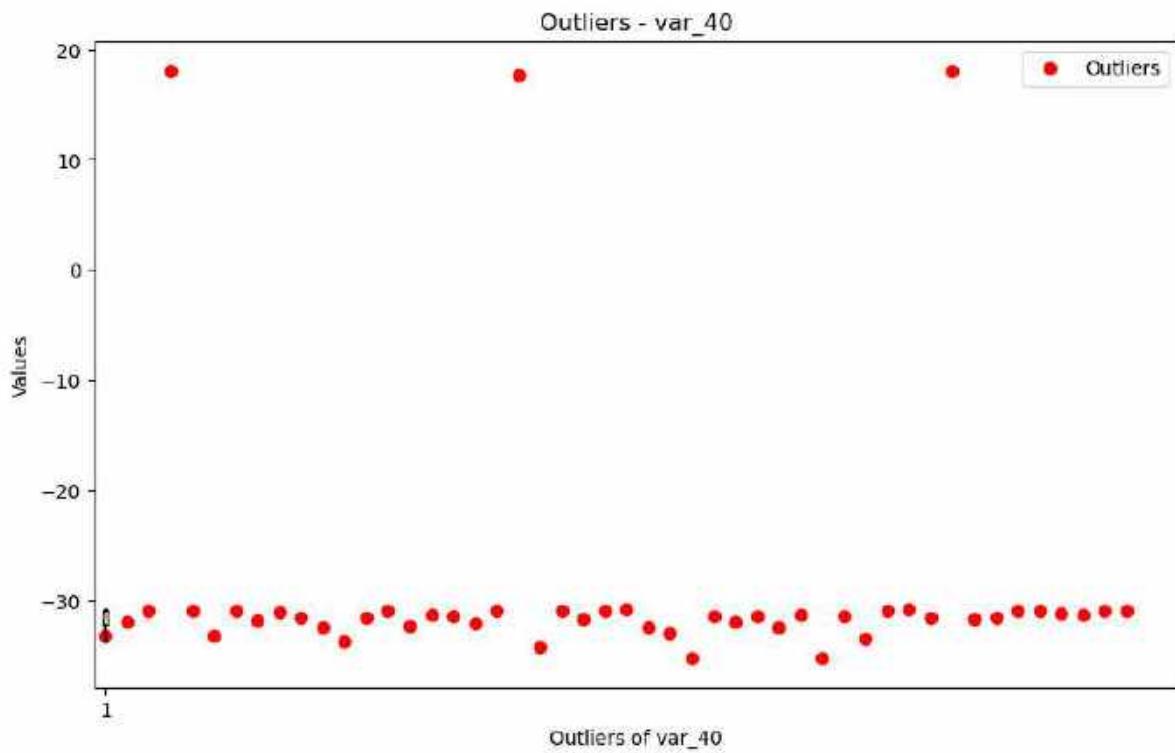


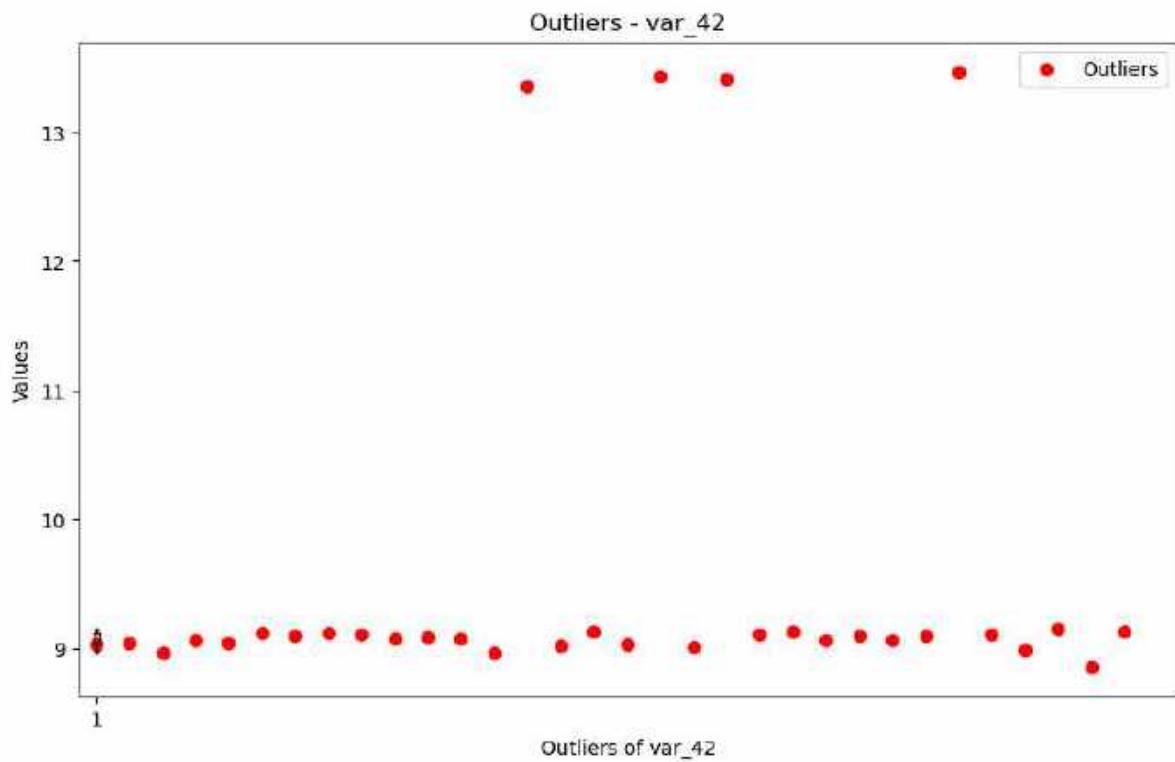
Total outliers in column var\_35: 198

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



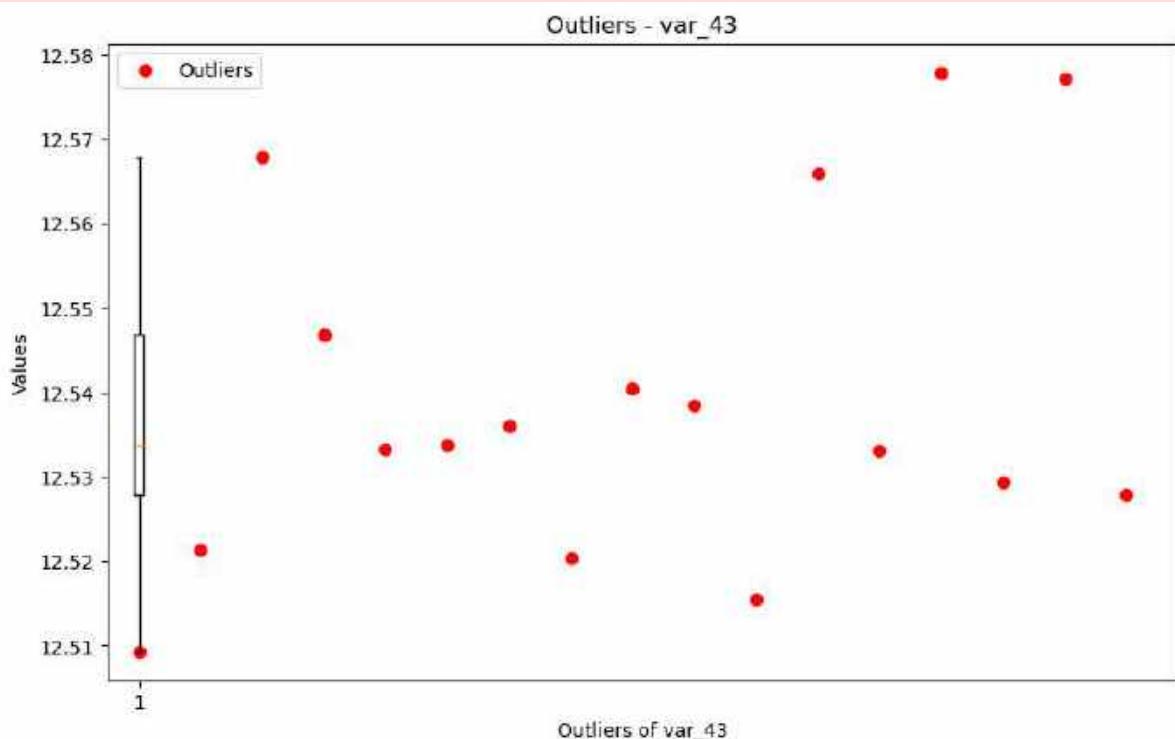






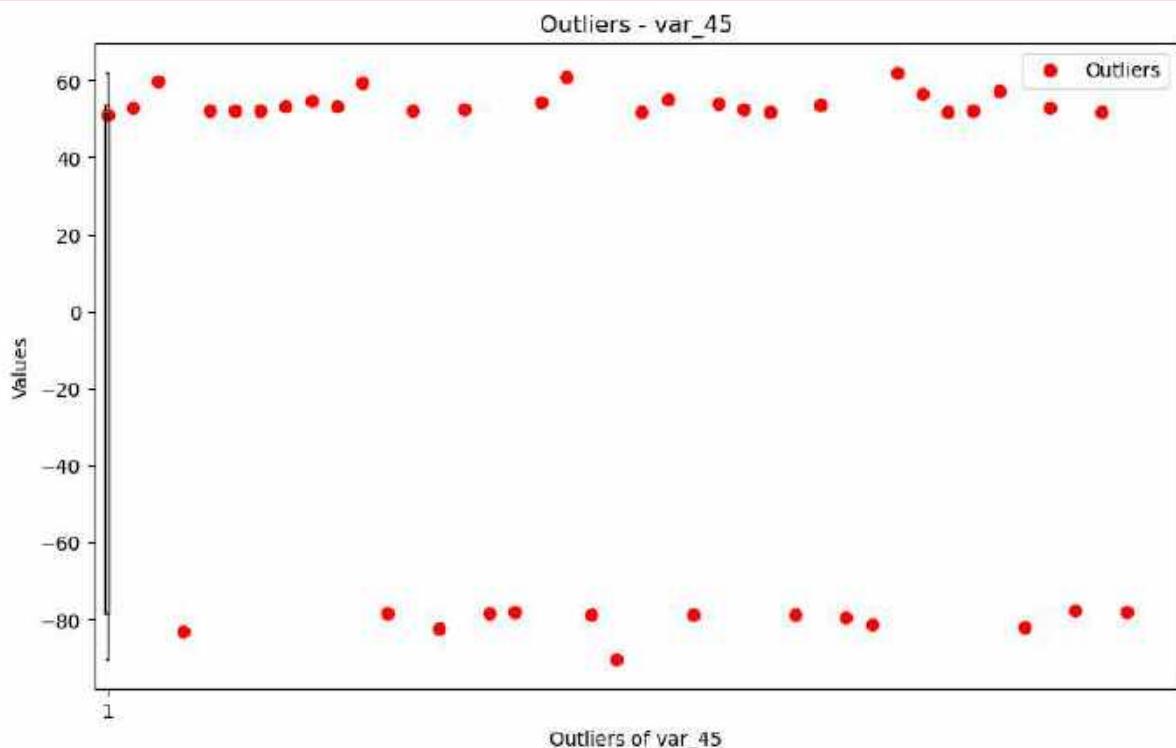
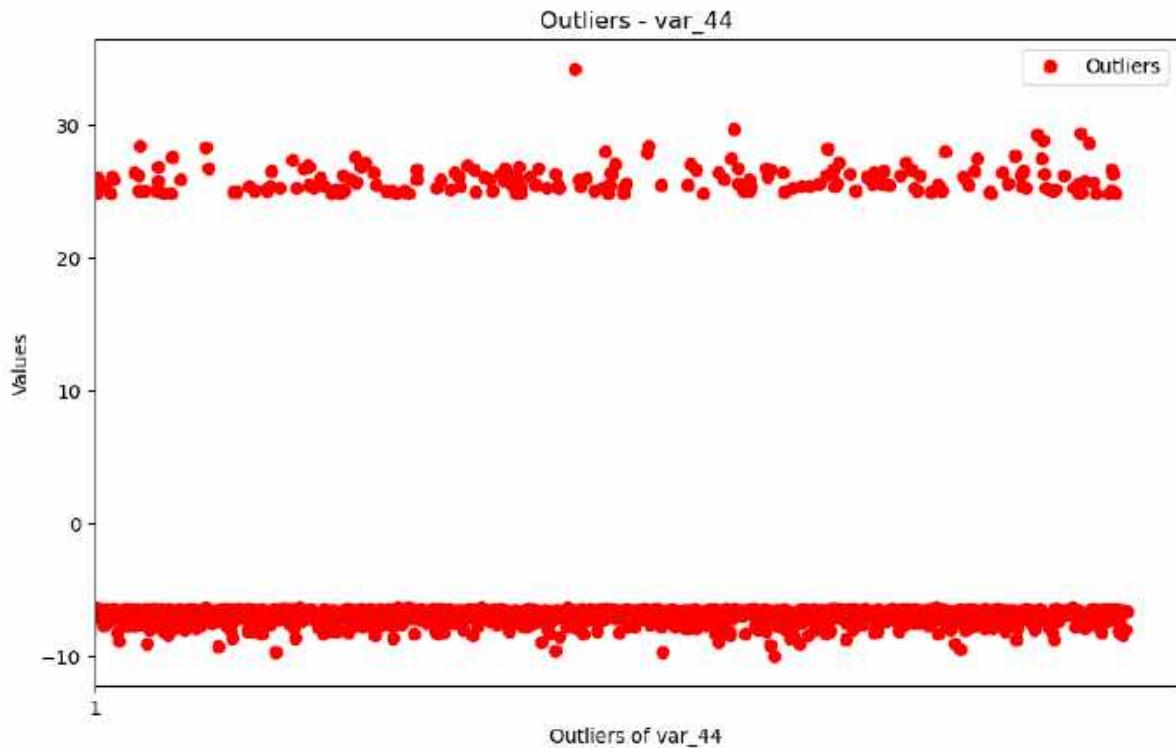
Total outliers in column var\_42: 32

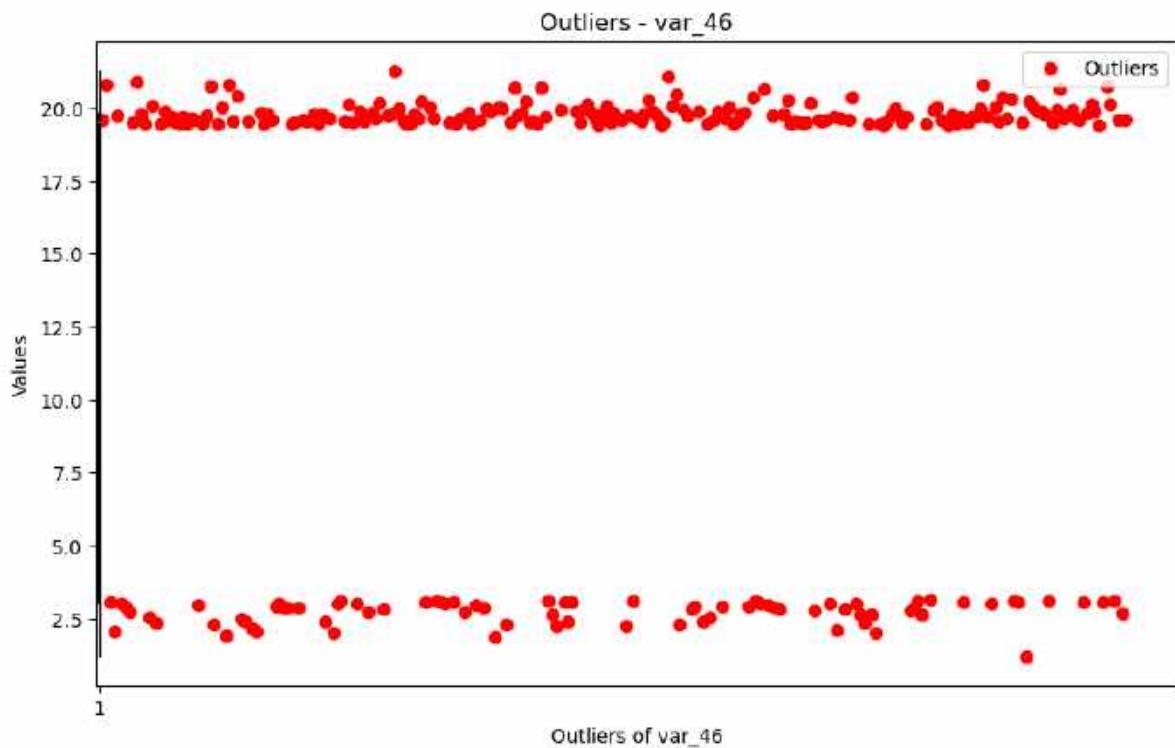
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

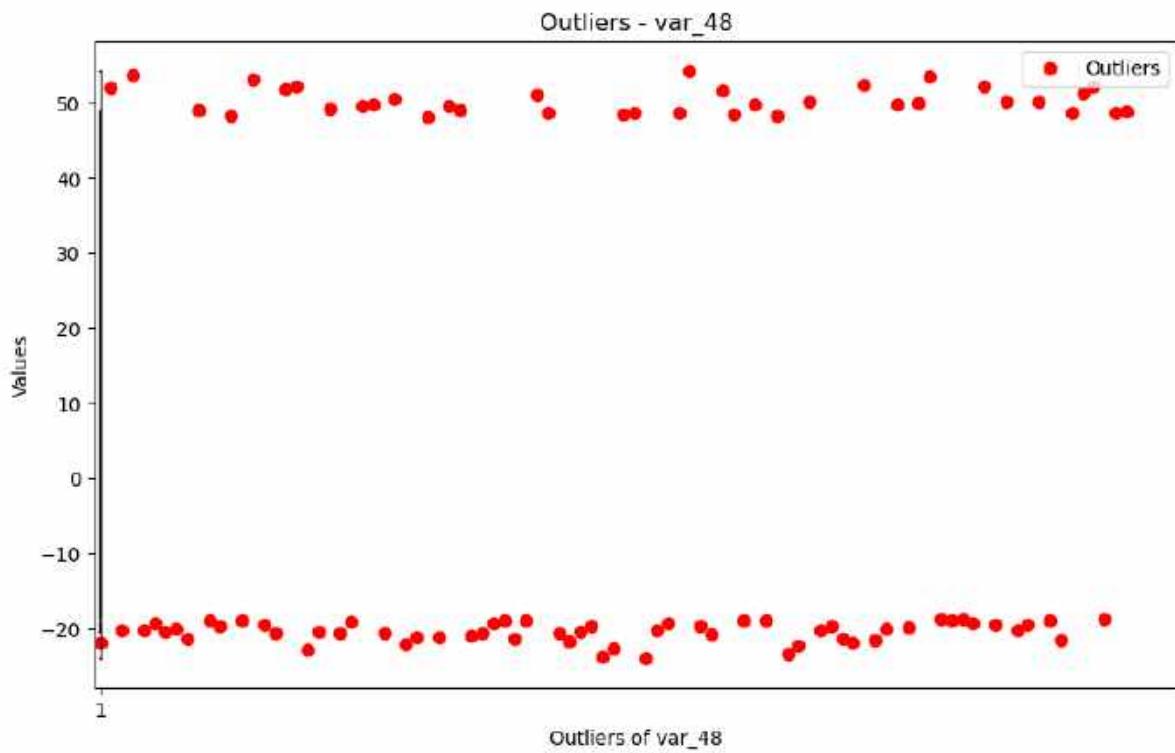
Total outliers in column var\_43: 17





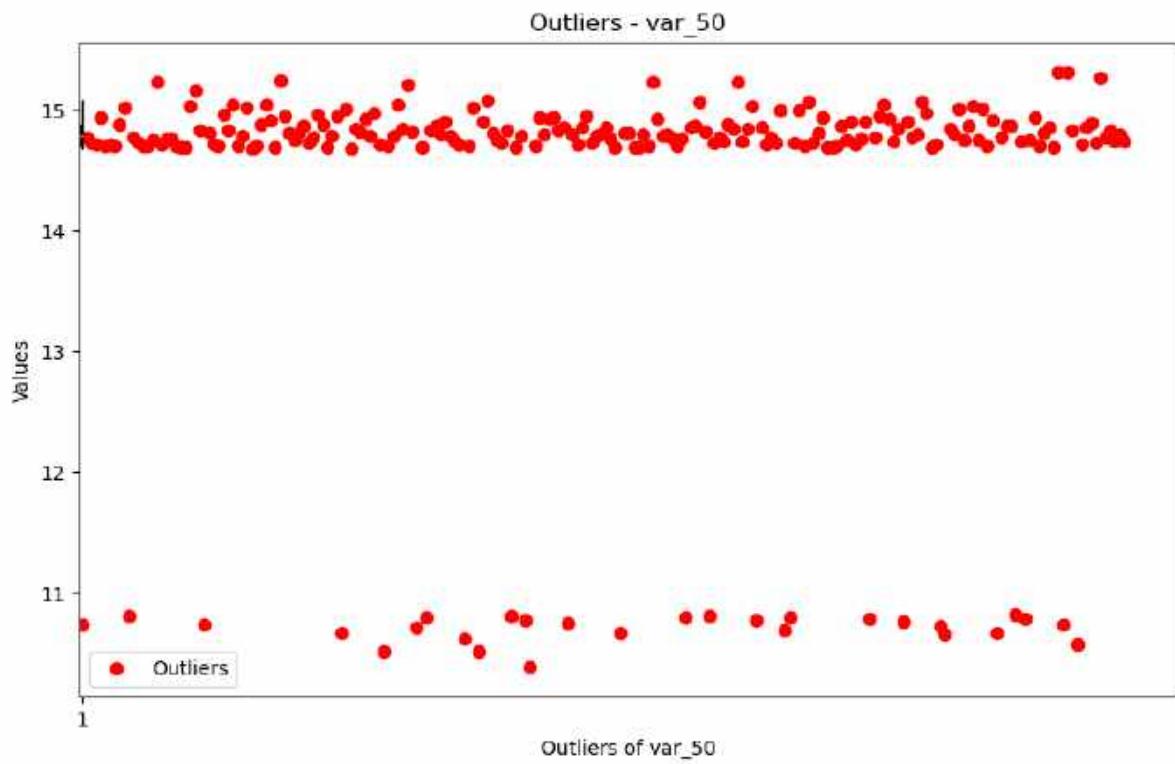
Total outliers in column var\_47: 2

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



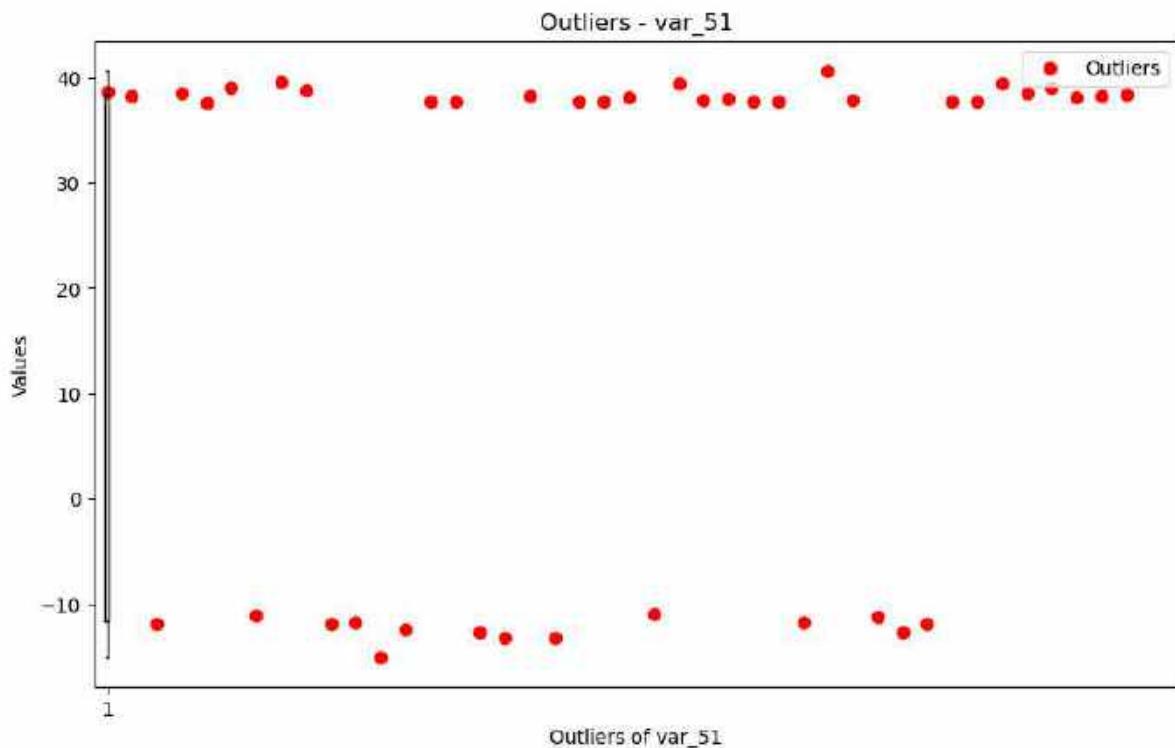
Total outliers in column var\_49: 7

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



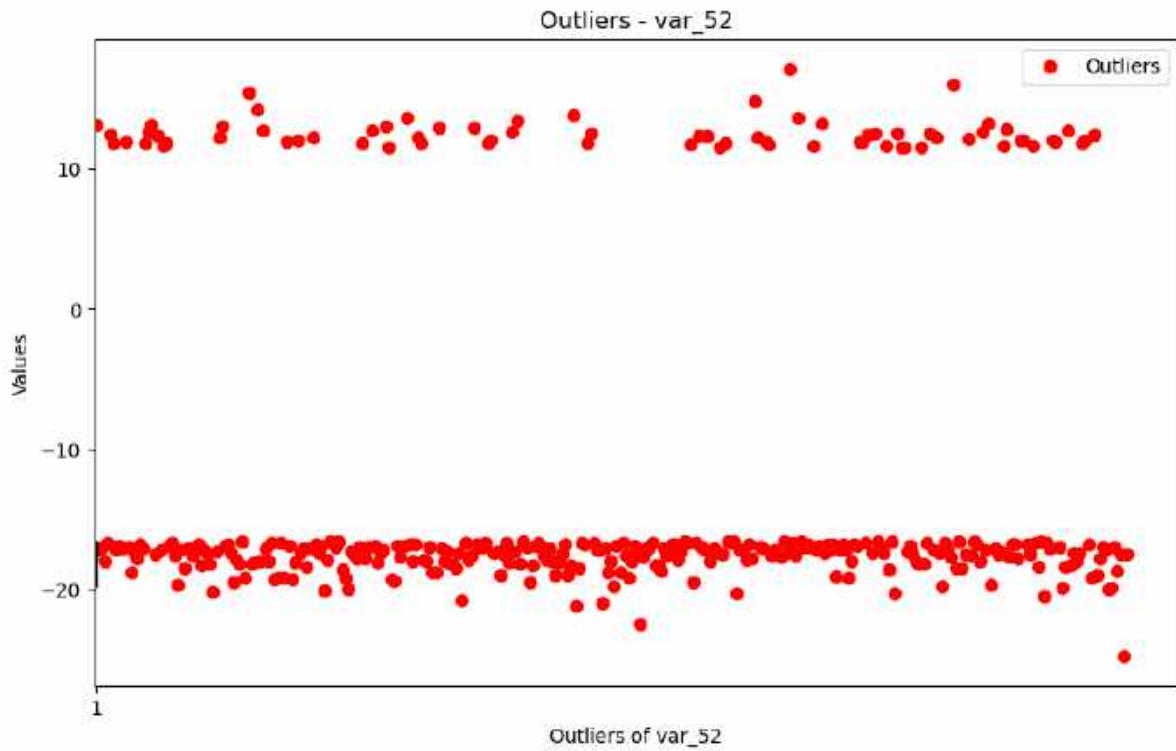
Total outliers in column var\_50: 222

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



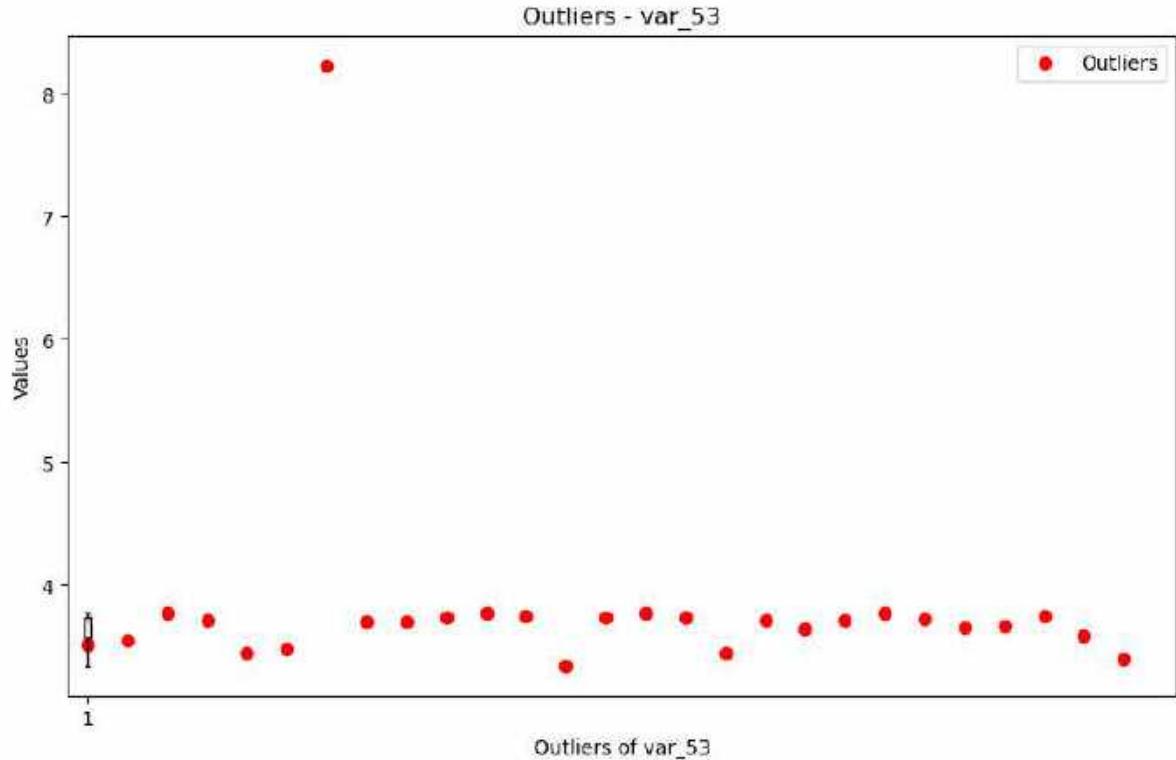
Total outliers in column var\_51: 42

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



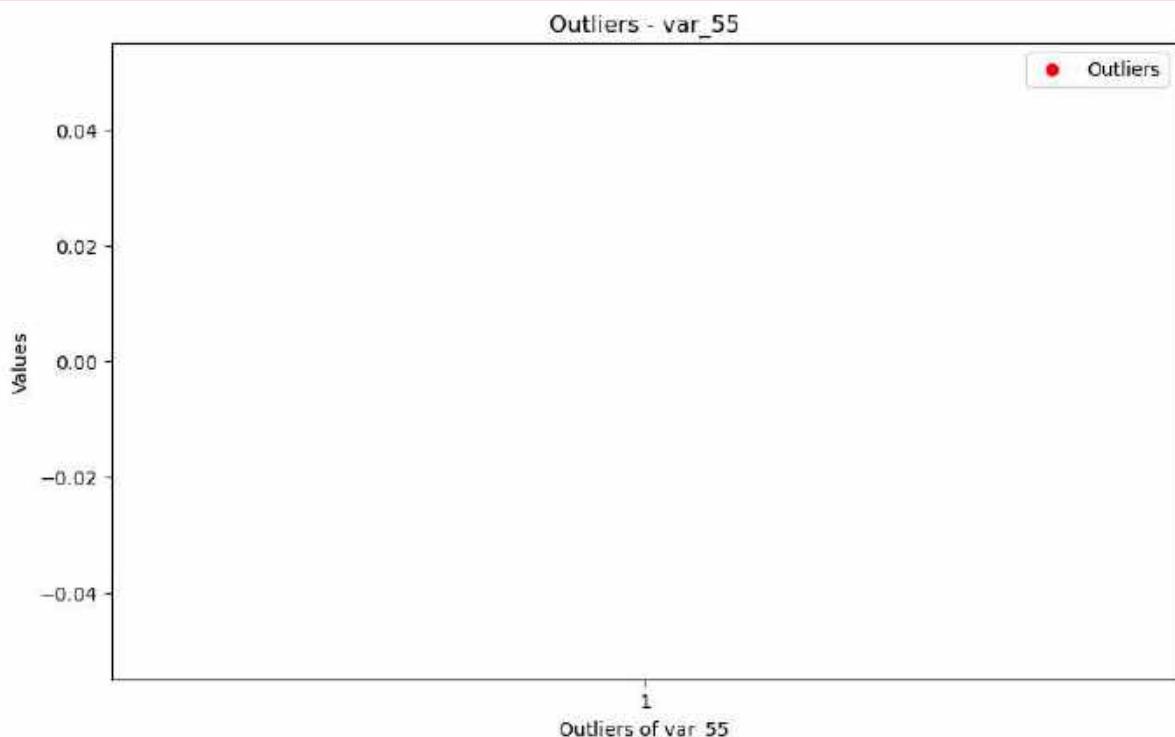
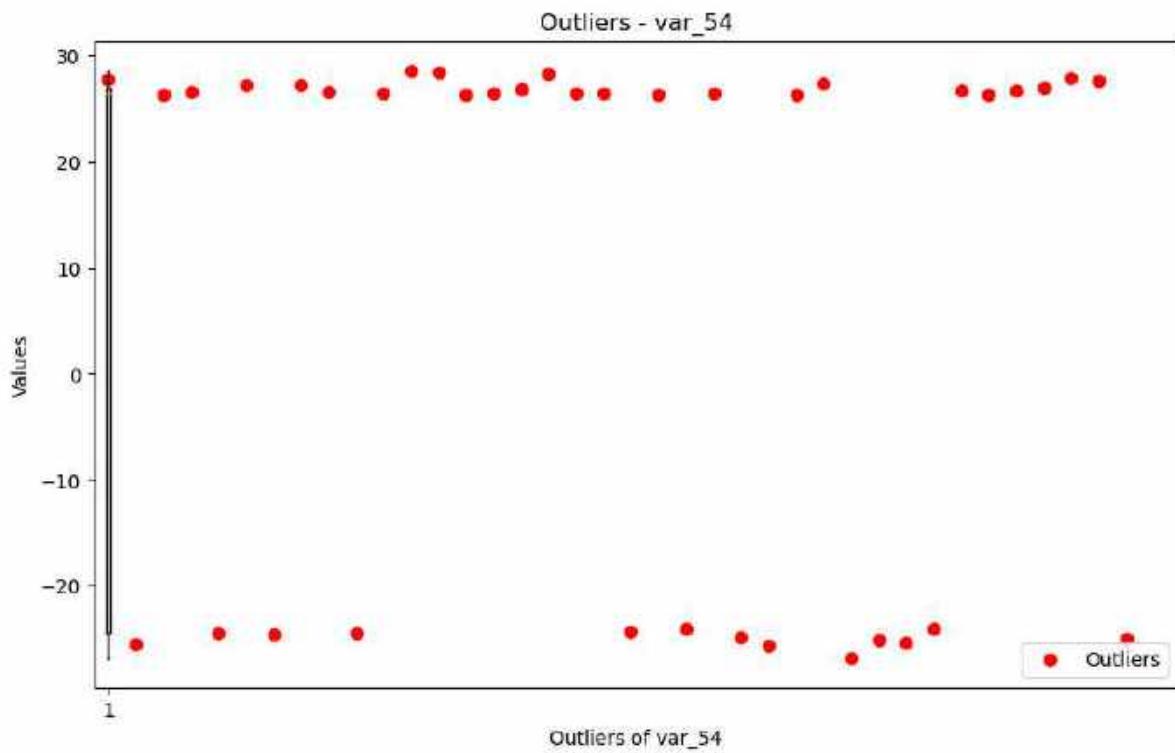
Total outliers in column var\_52: 353

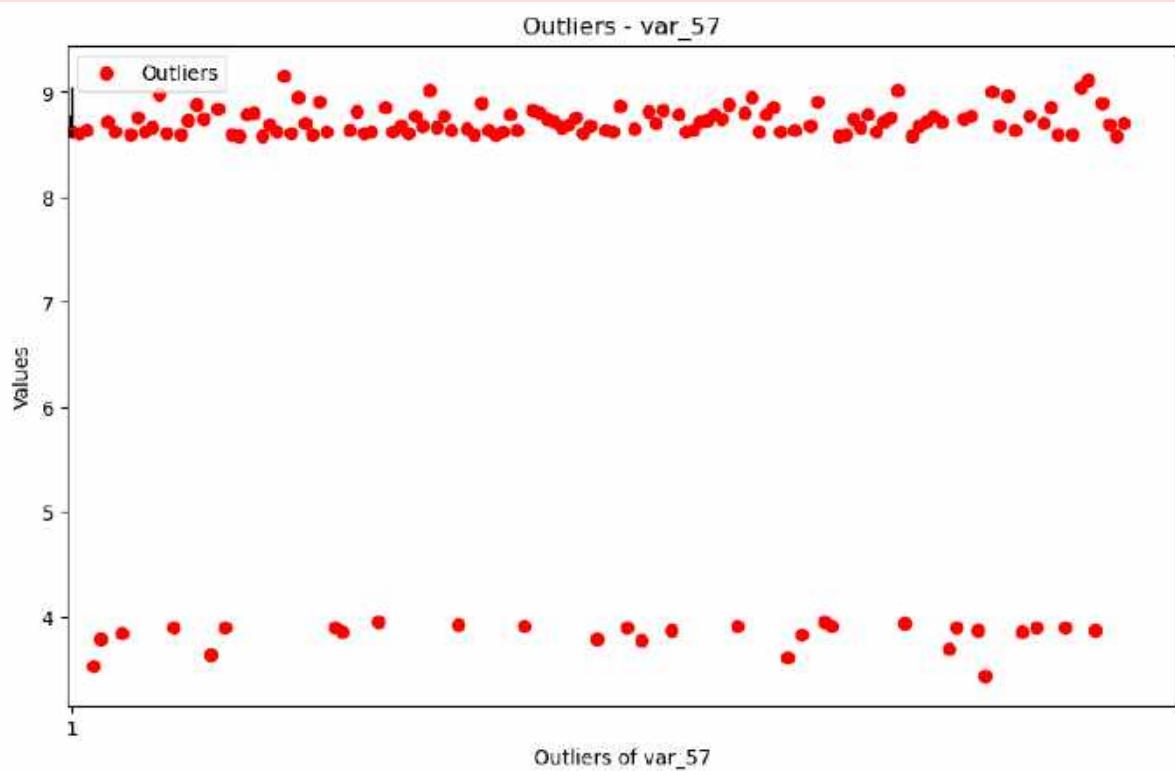
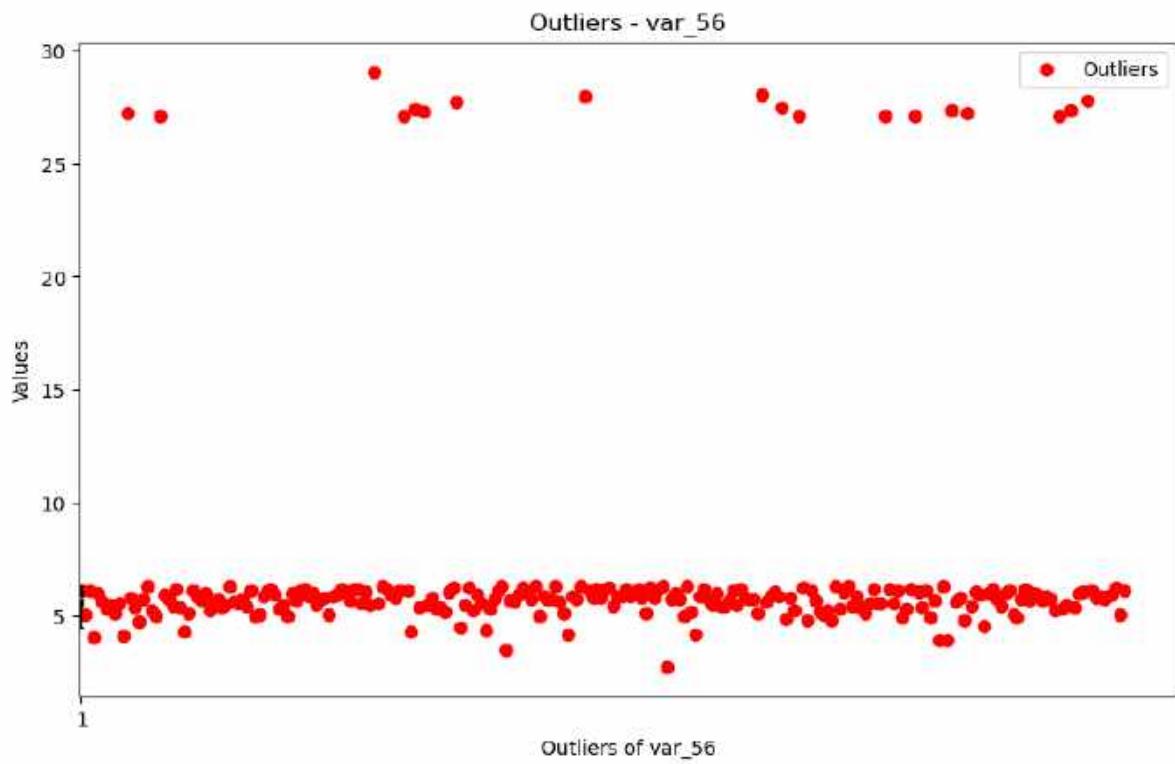
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

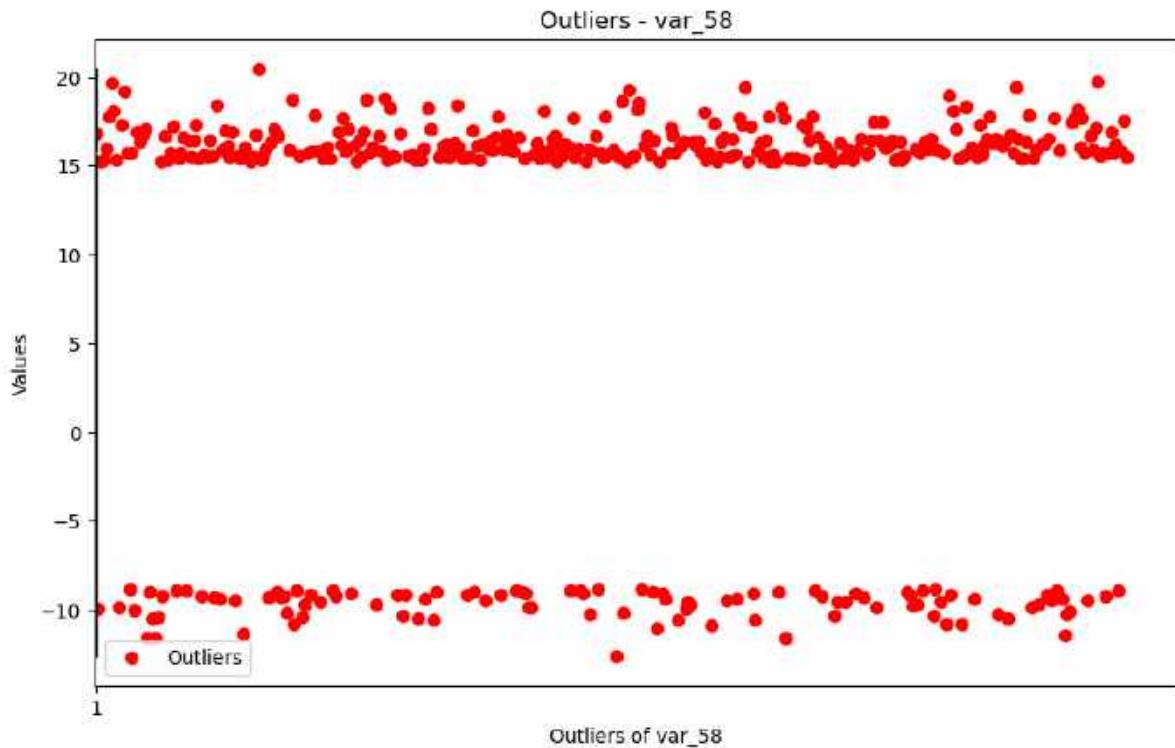


Total outliers in column var\_53: 27

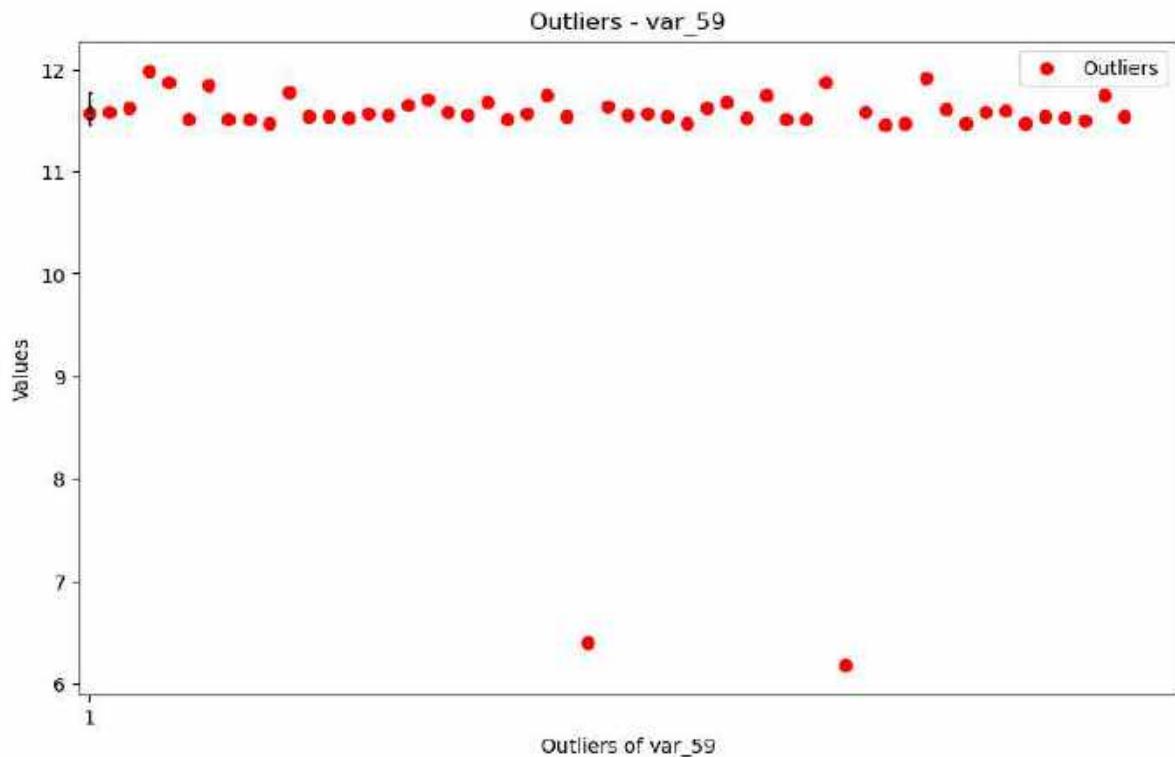
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



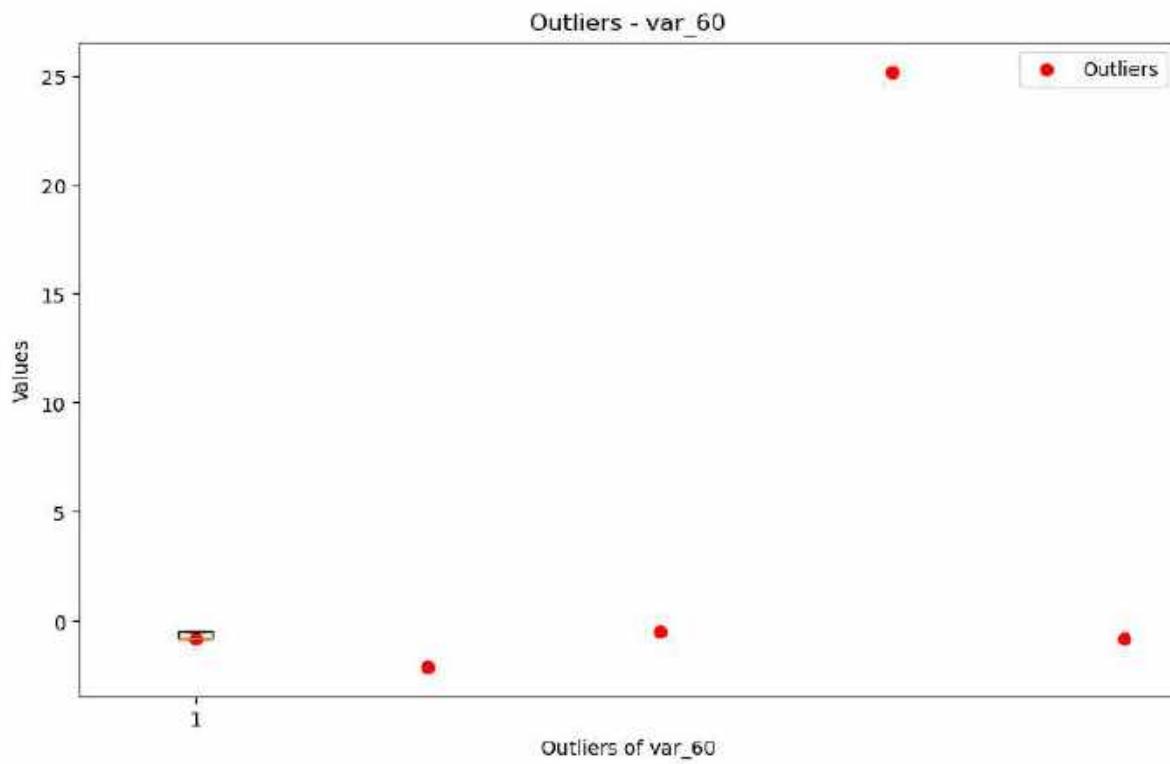




```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_58: 401
```

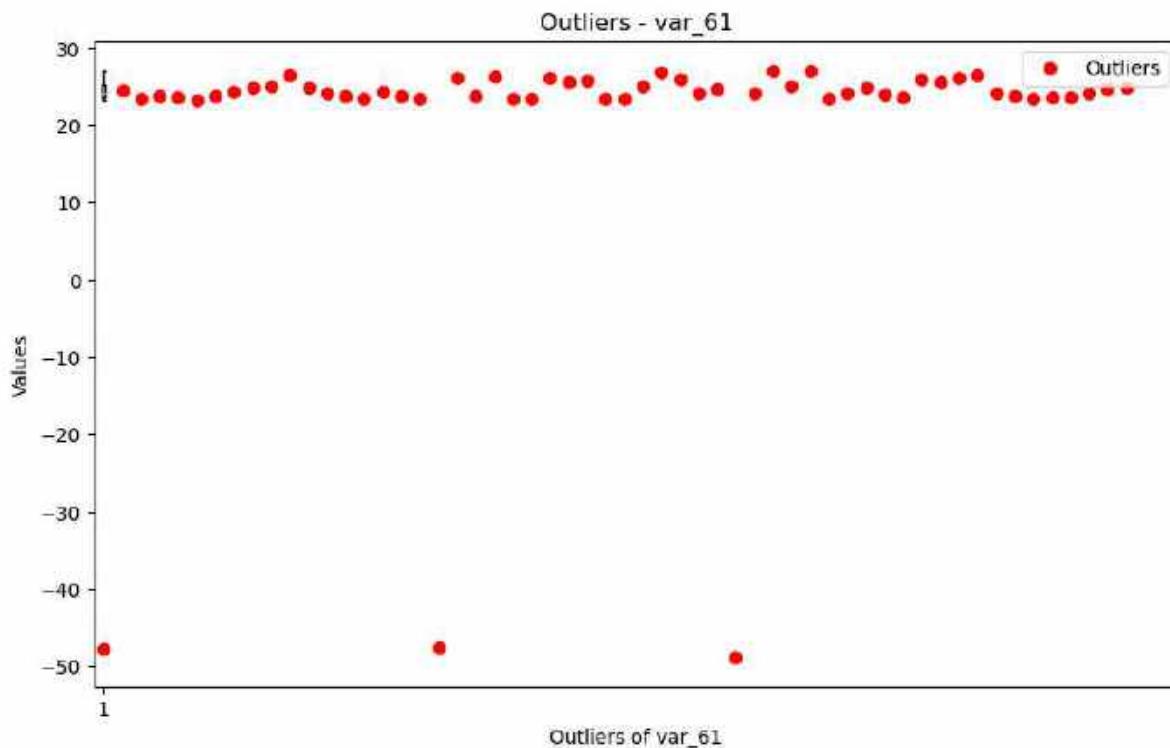


```
Total outliers in column var_59: 53  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



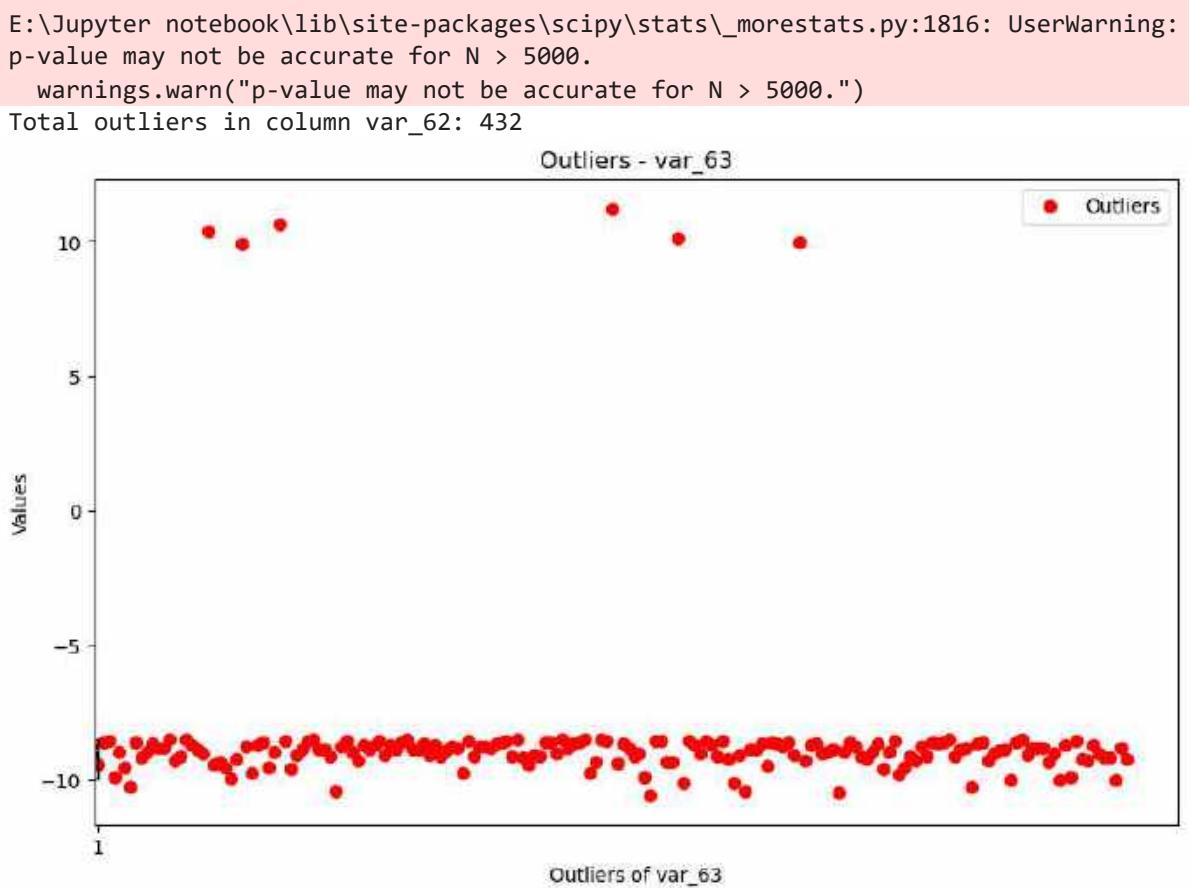
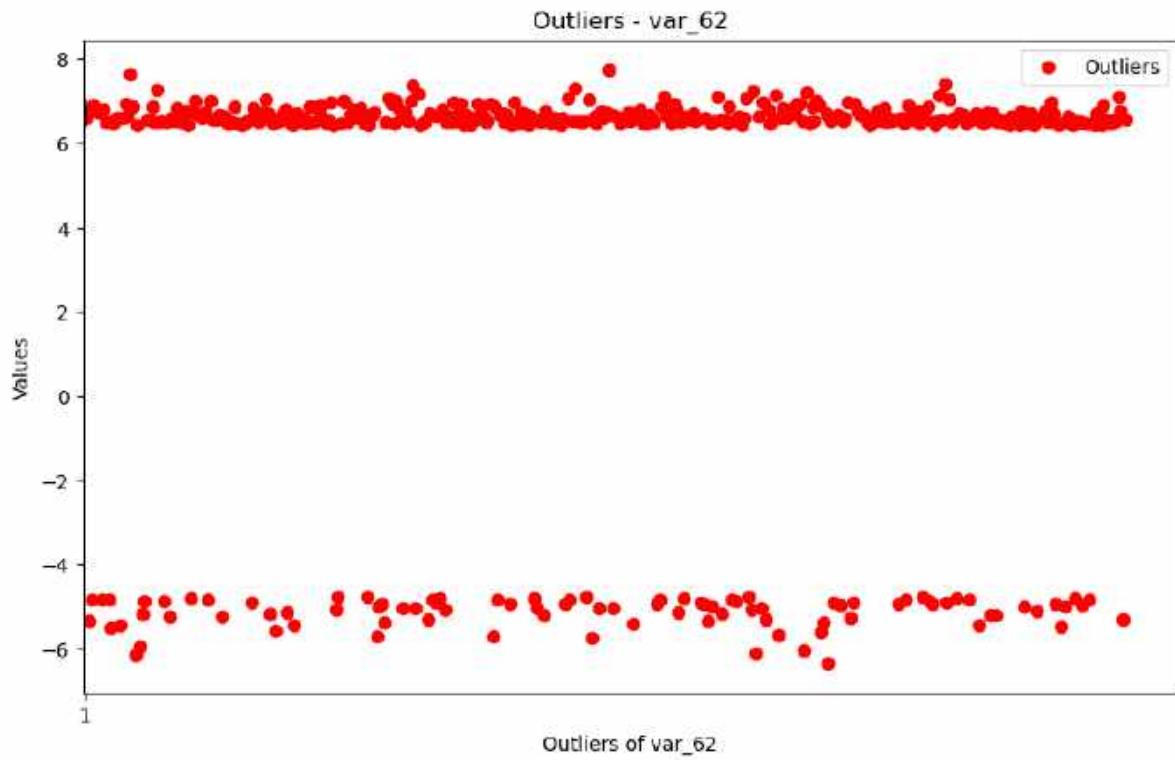
Total outliers in column var\_60: 5

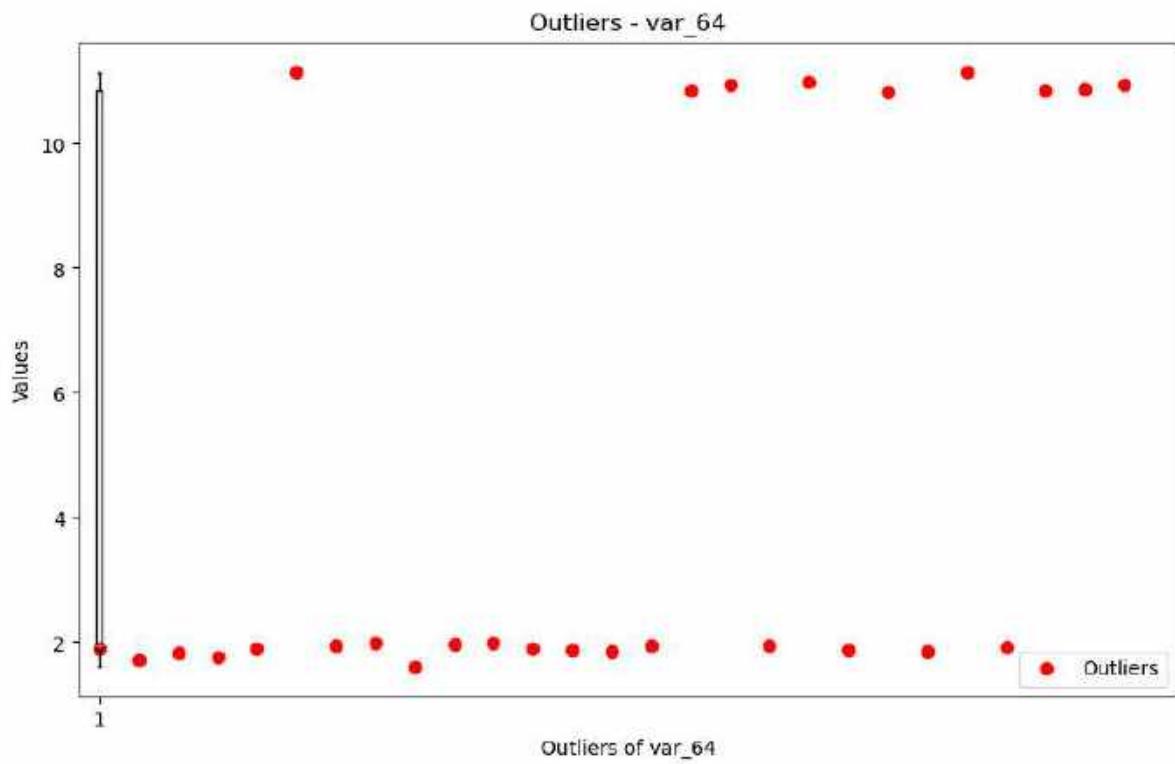
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_61: 56

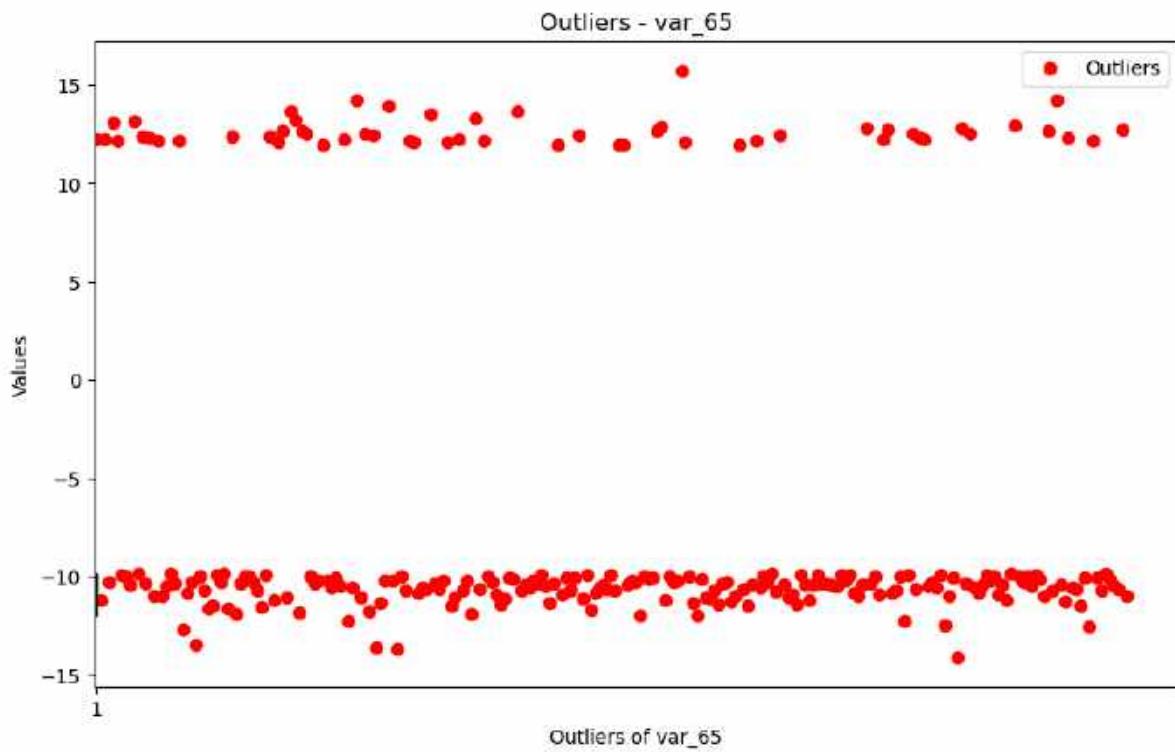
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





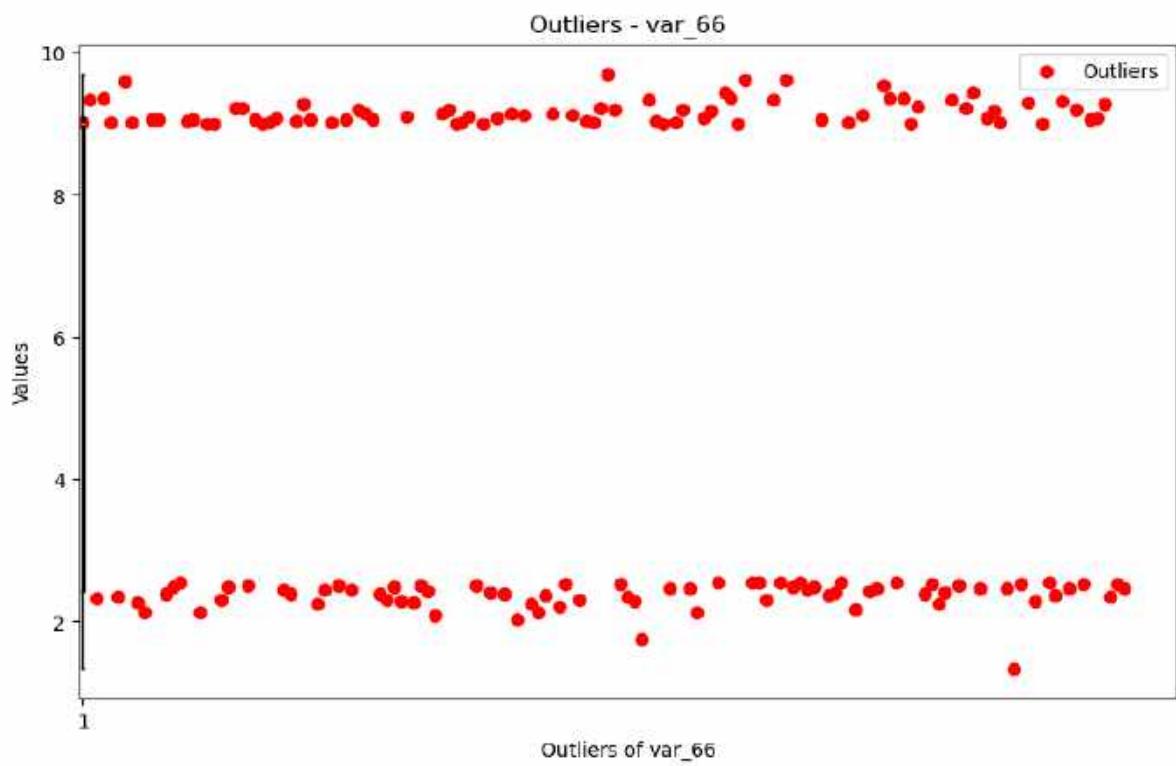
Total outliers in column var\_64: 27

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



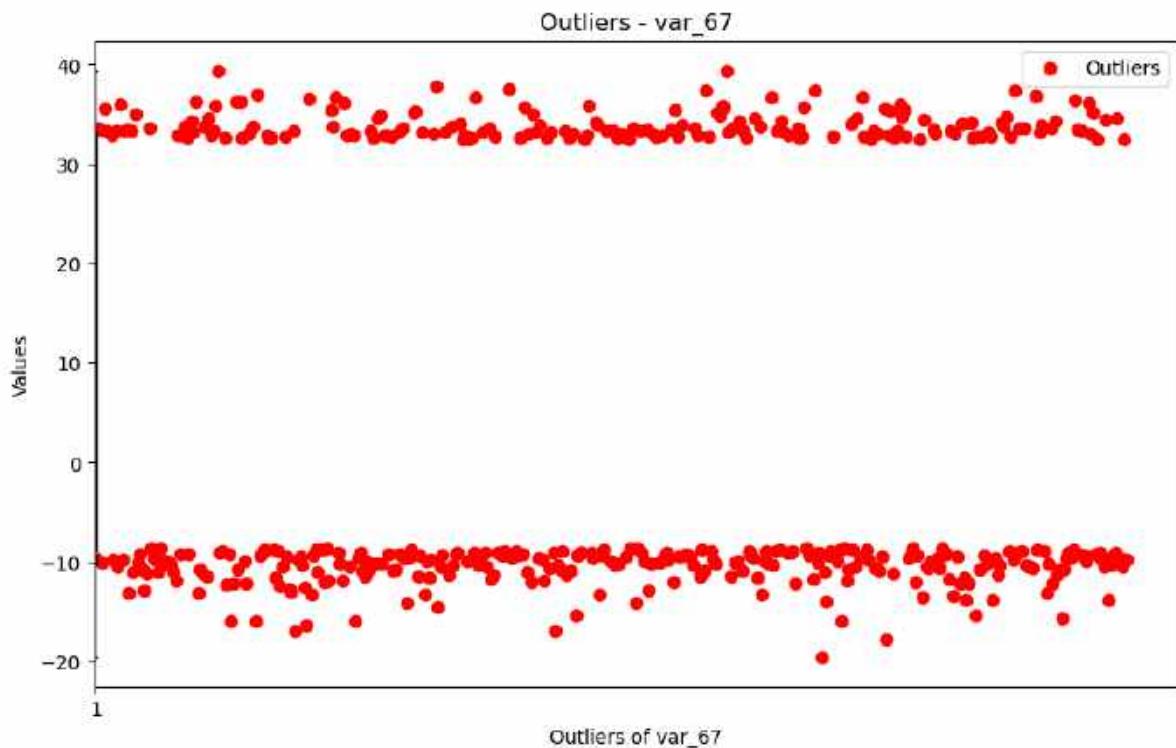
Total outliers in column var\_65: 251

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



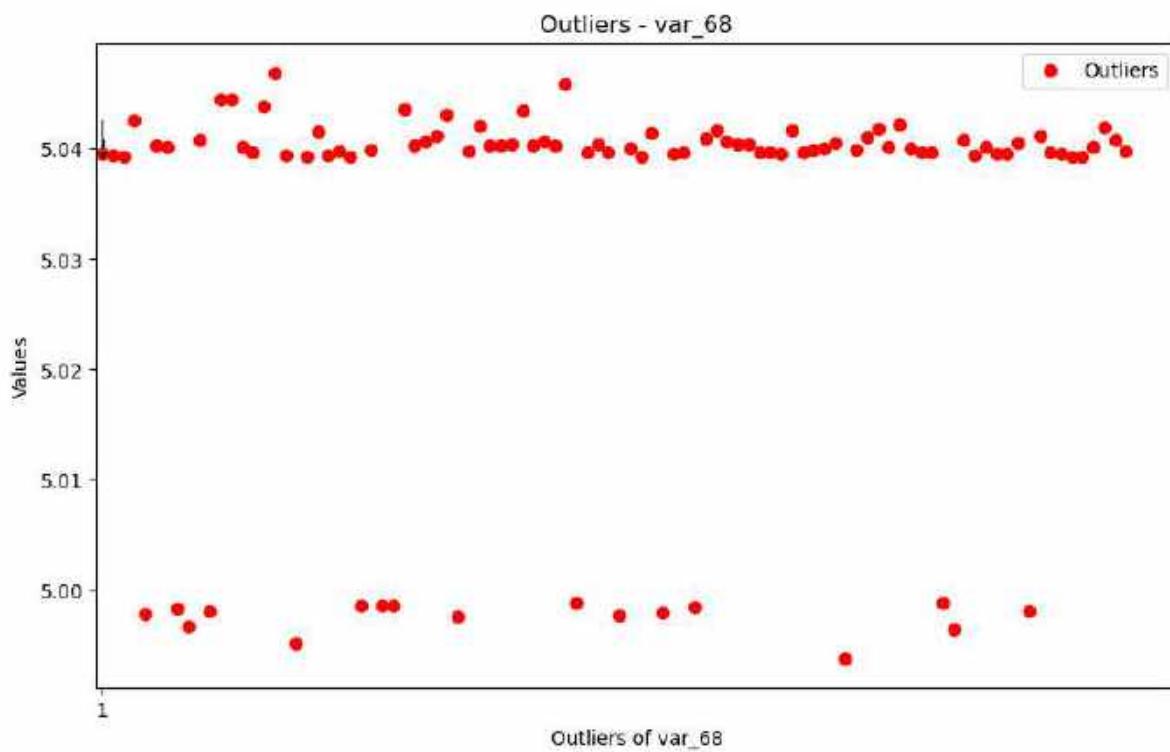
Total outliers in column var\_66: 152

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



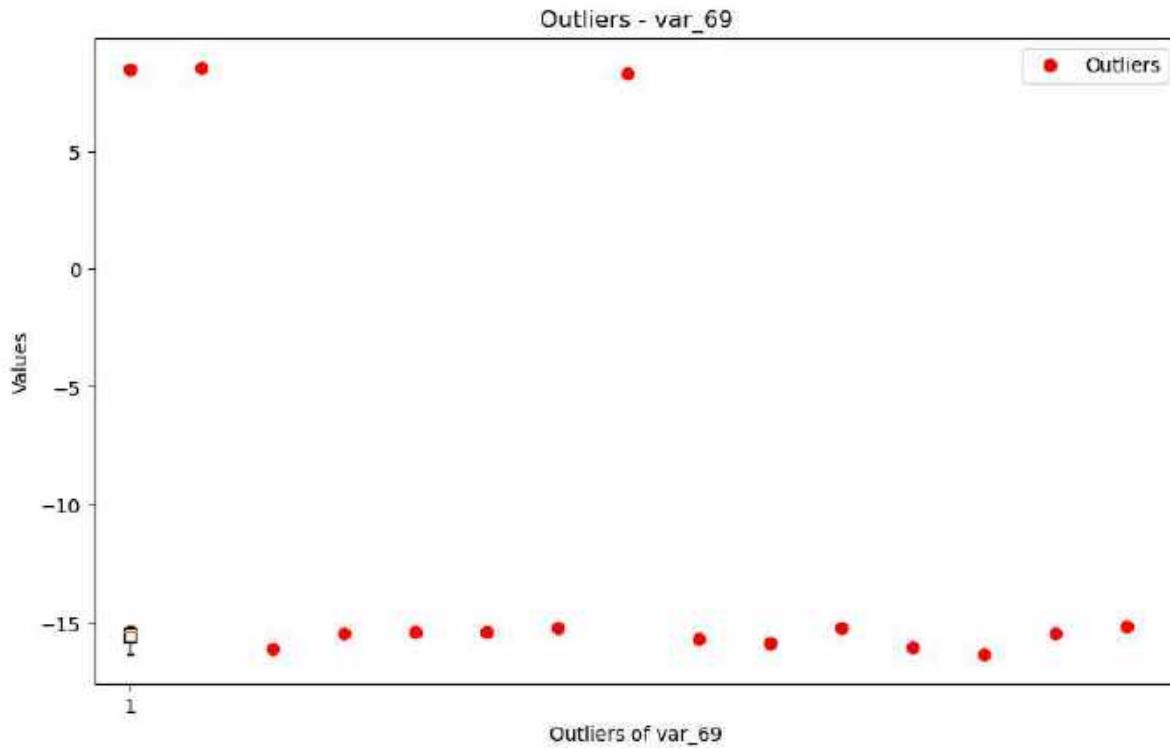
Total outliers in column var\_67: 465

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



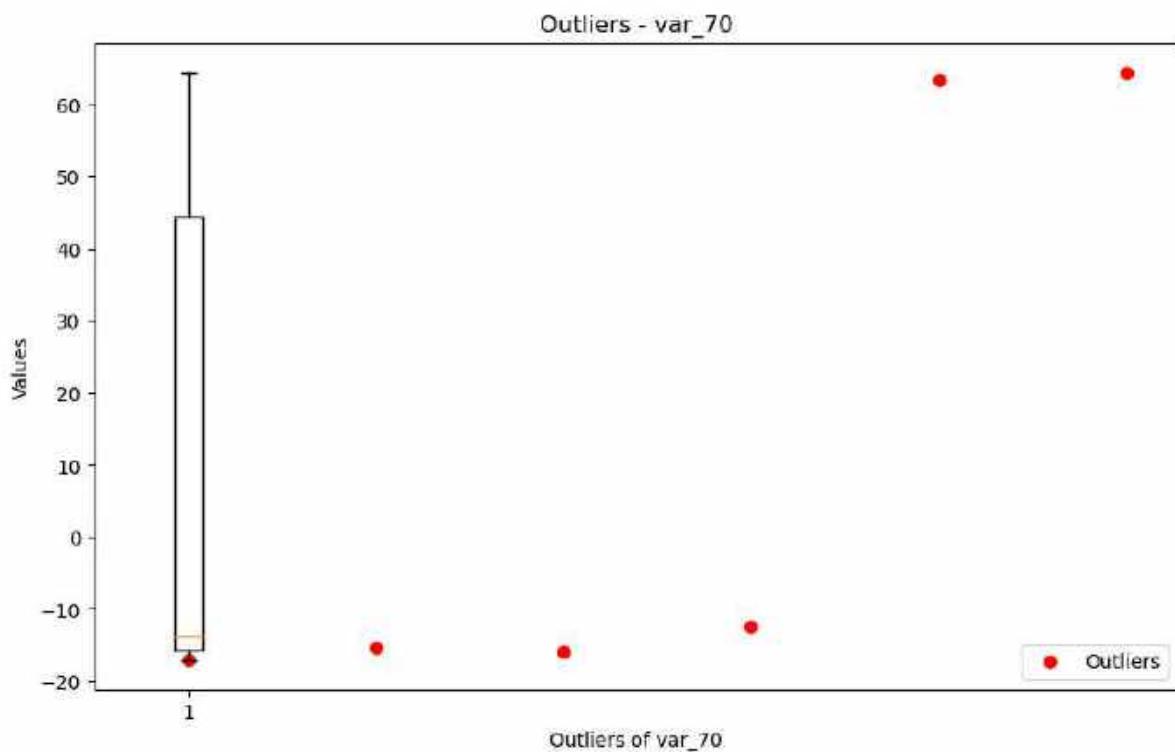
Total outliers in column var\_68: 96

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



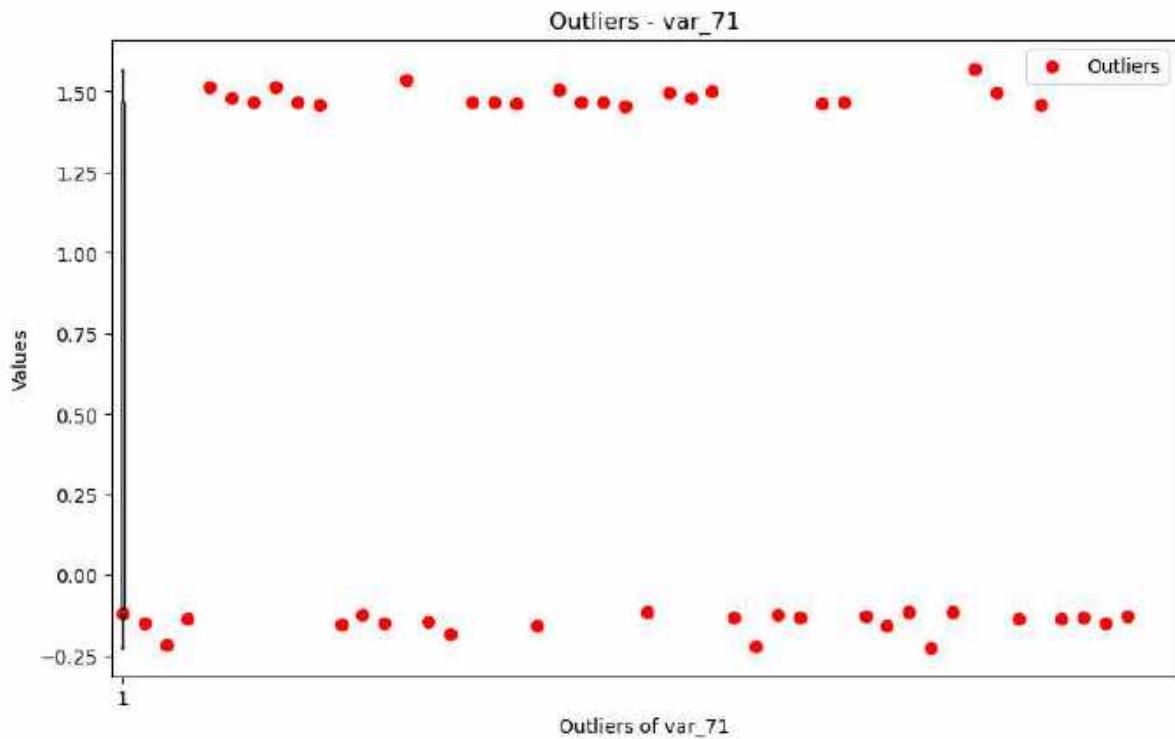
Total outliers in column var\_69: 15

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



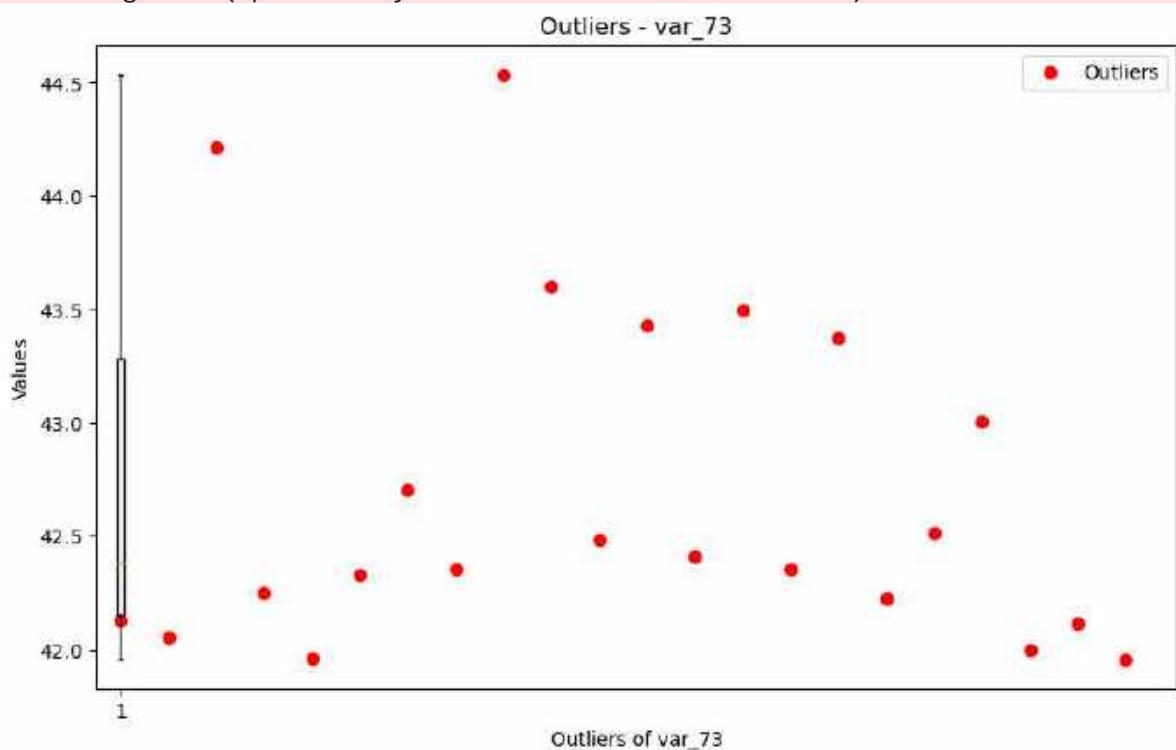
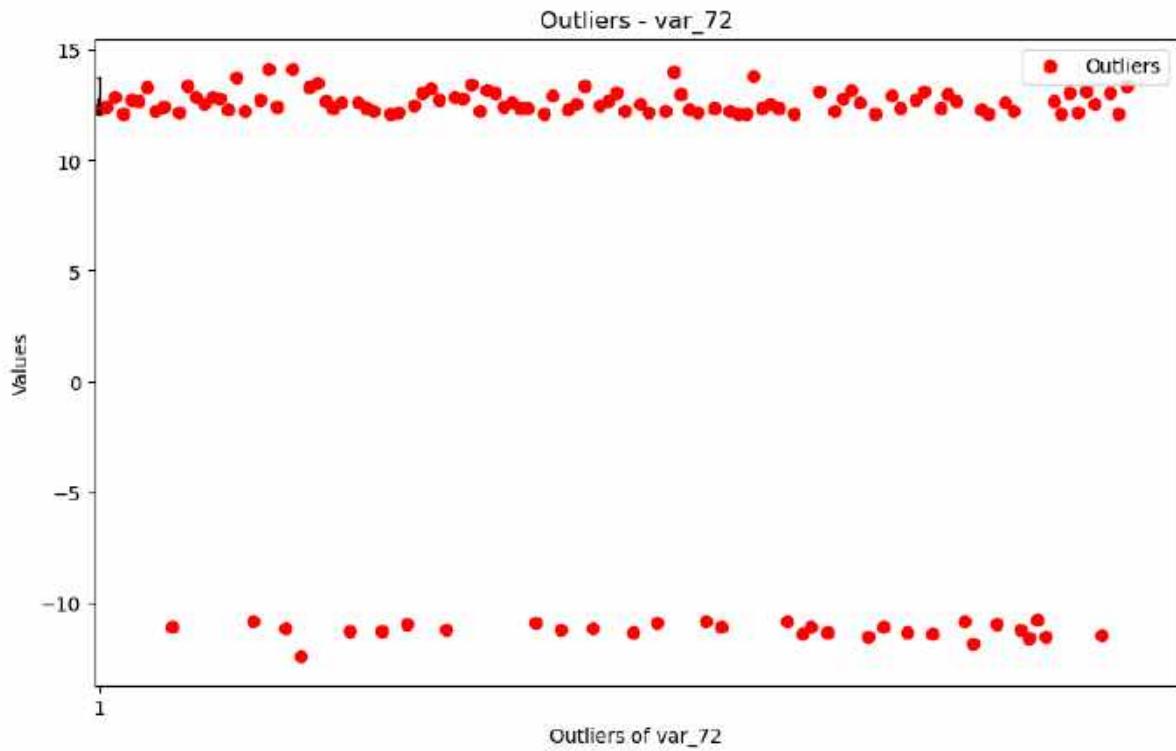
Total outliers in column var\_70: 6

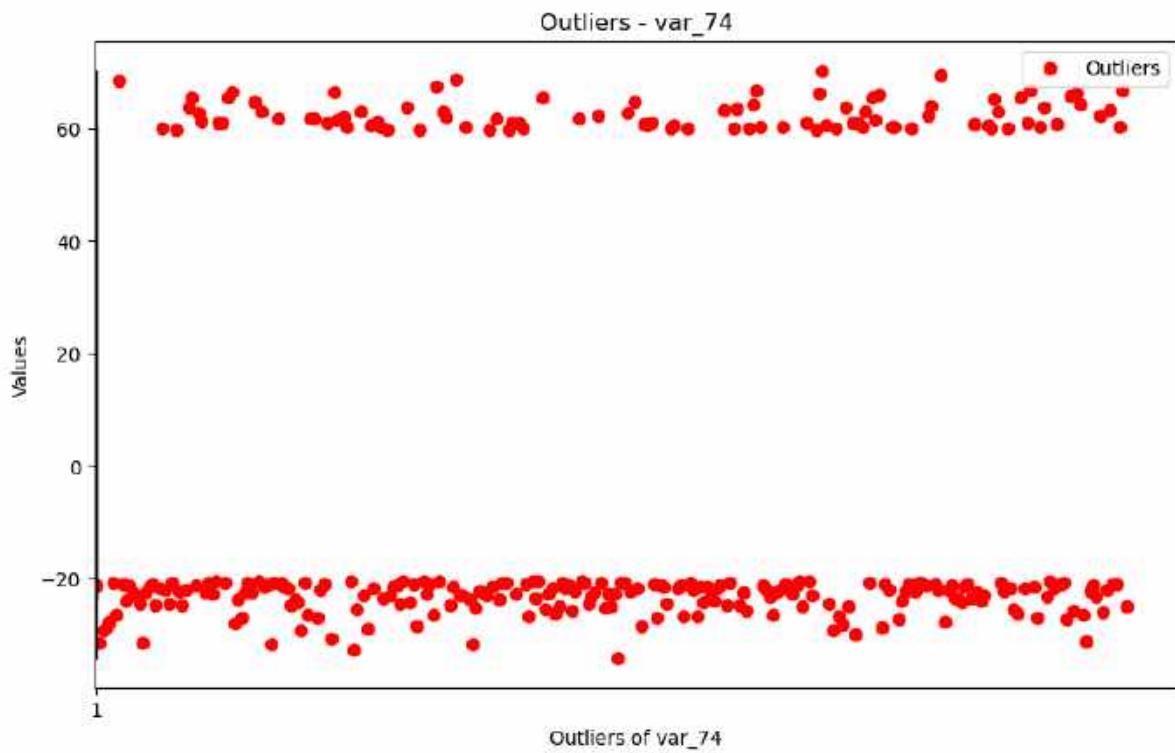
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_71: 47

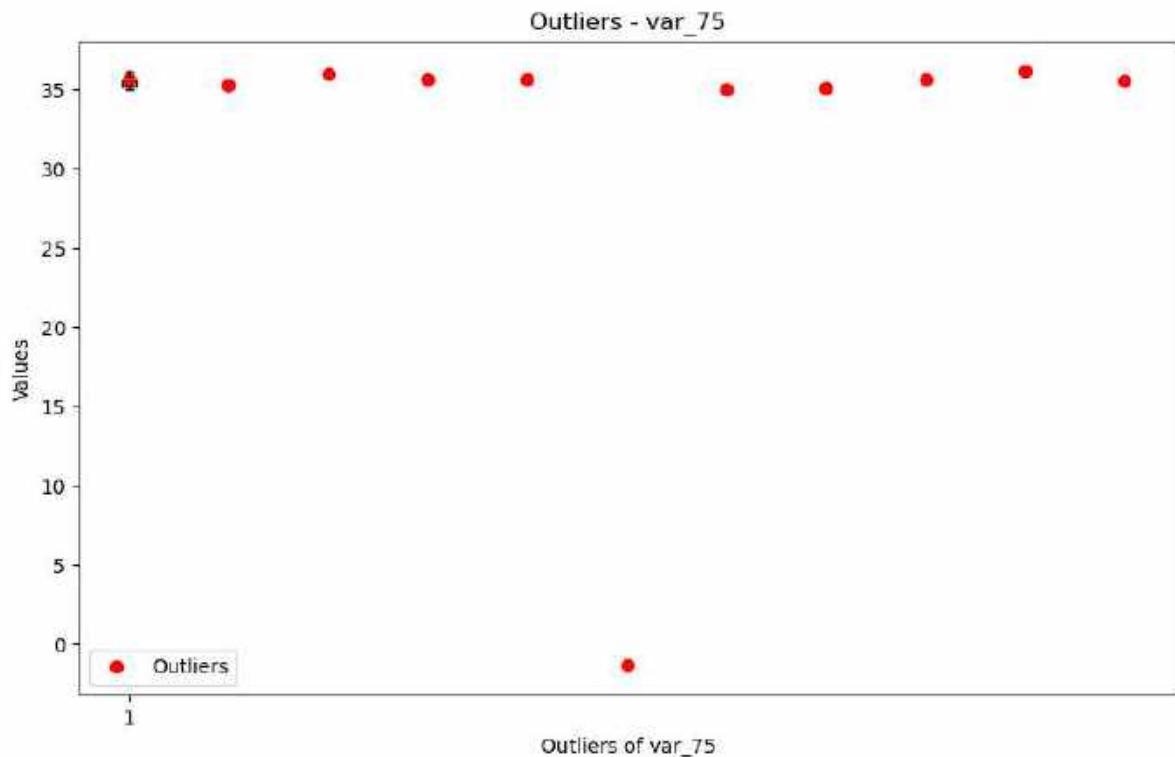
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





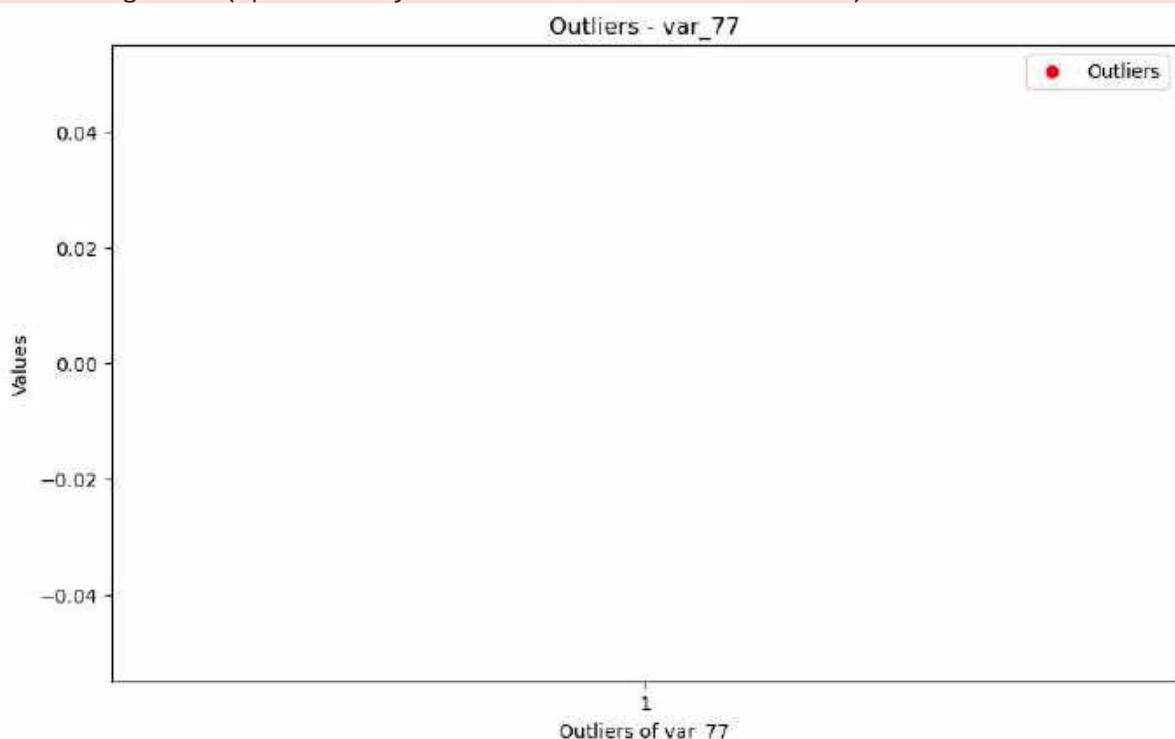
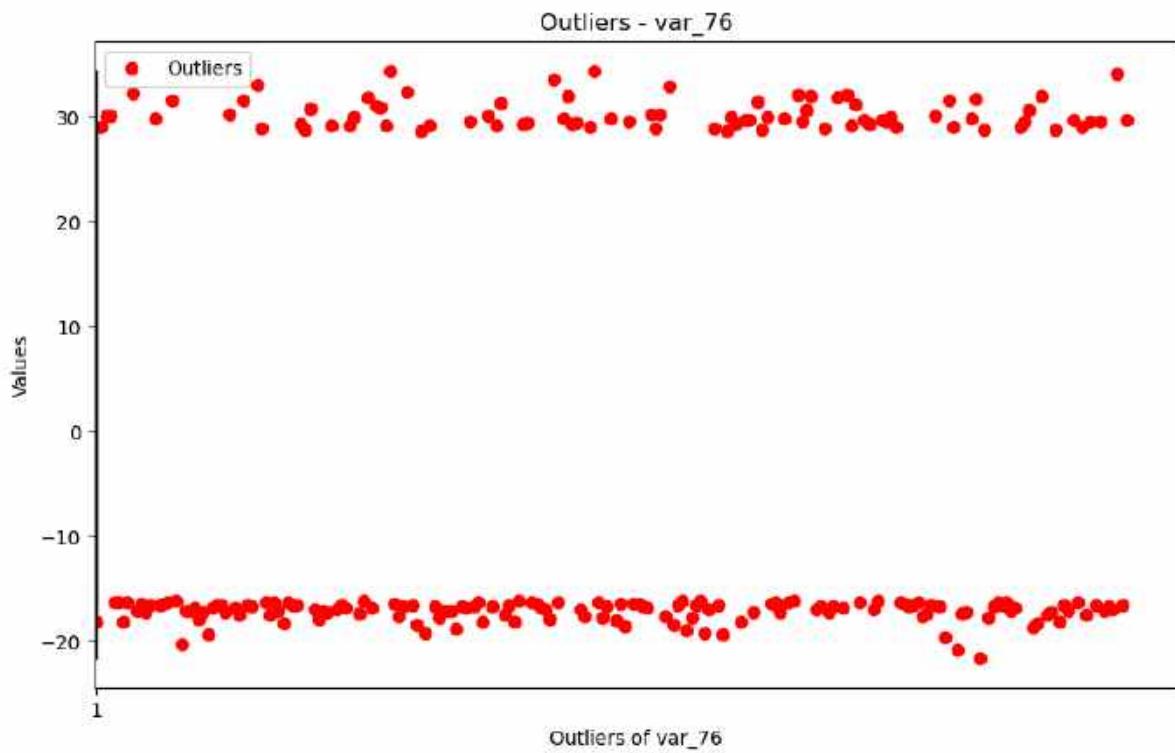
Total outliers in column var\_74: 313

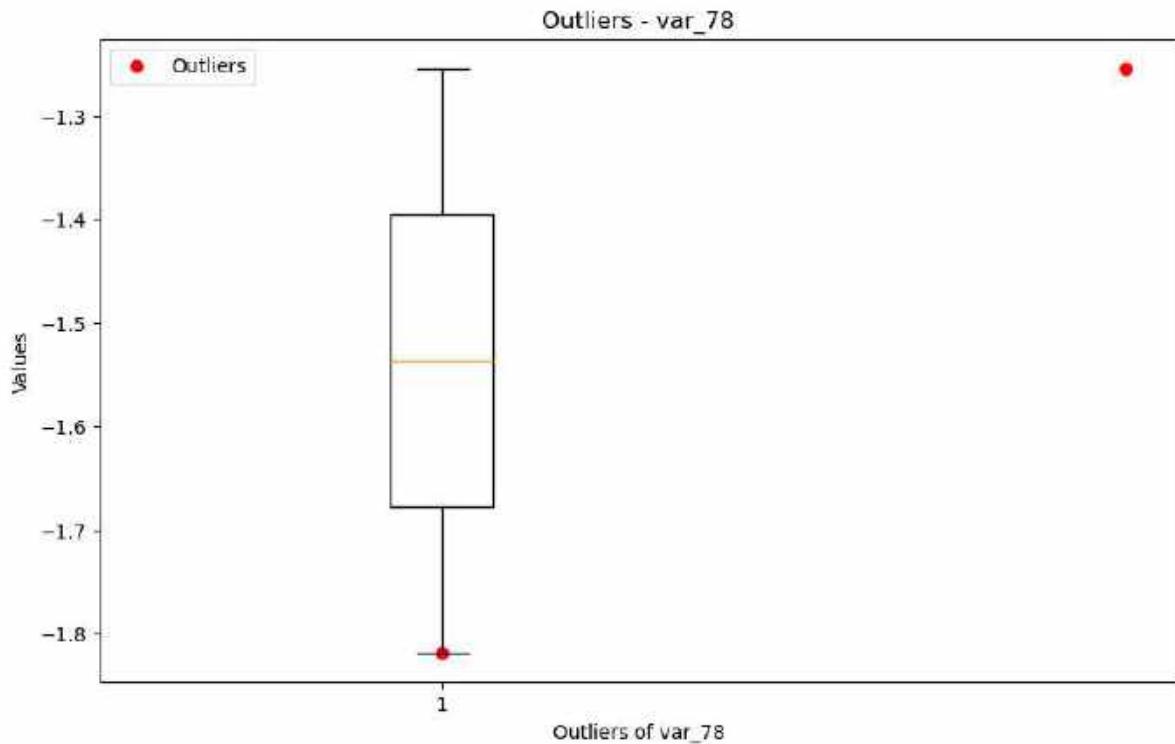
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_75: 11

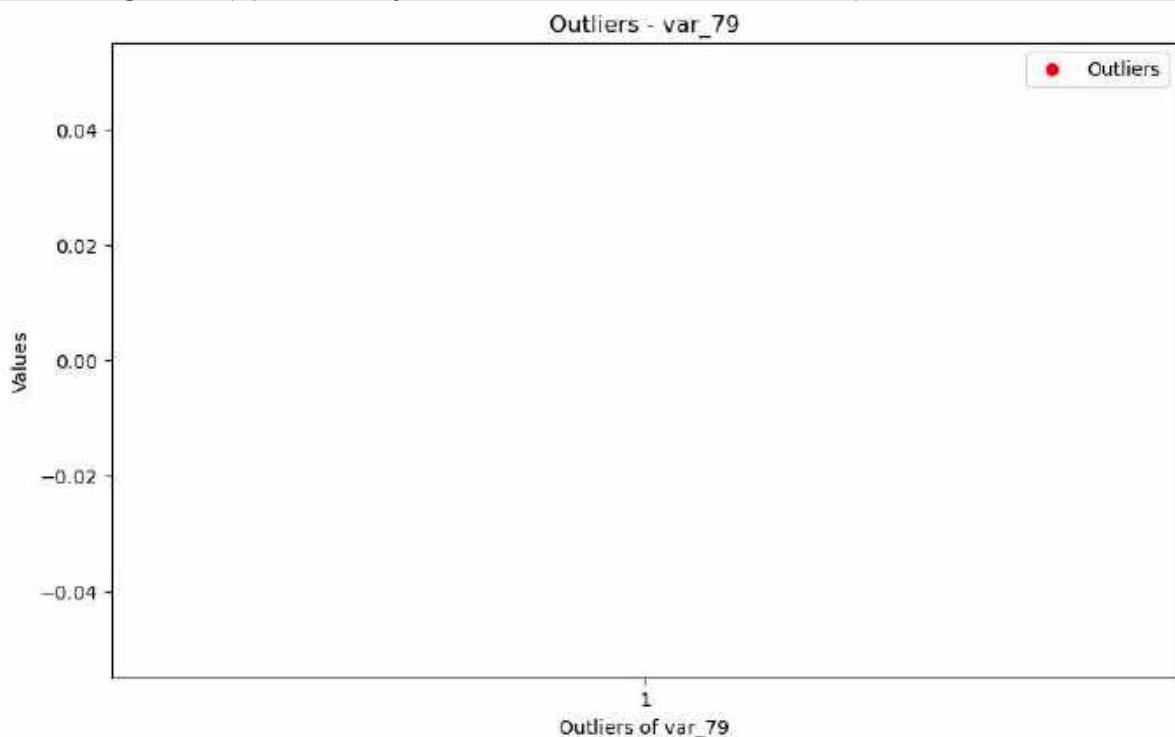
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





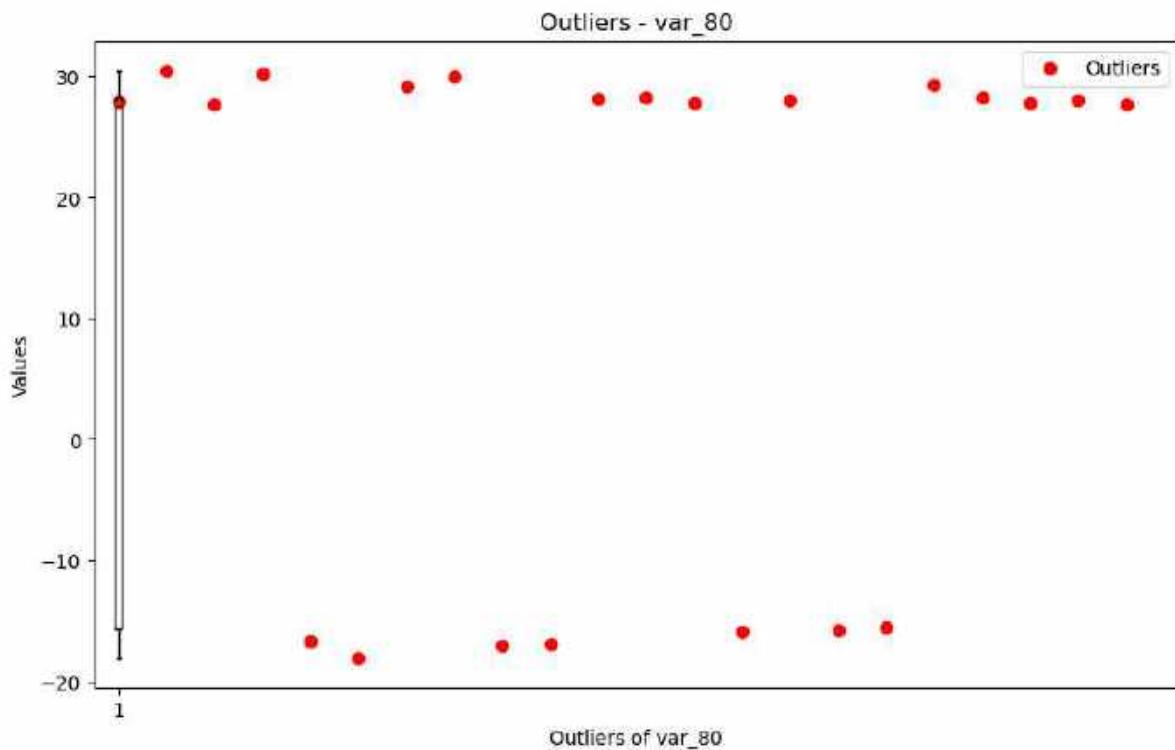
```
Total outliers in column var_78: 2
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



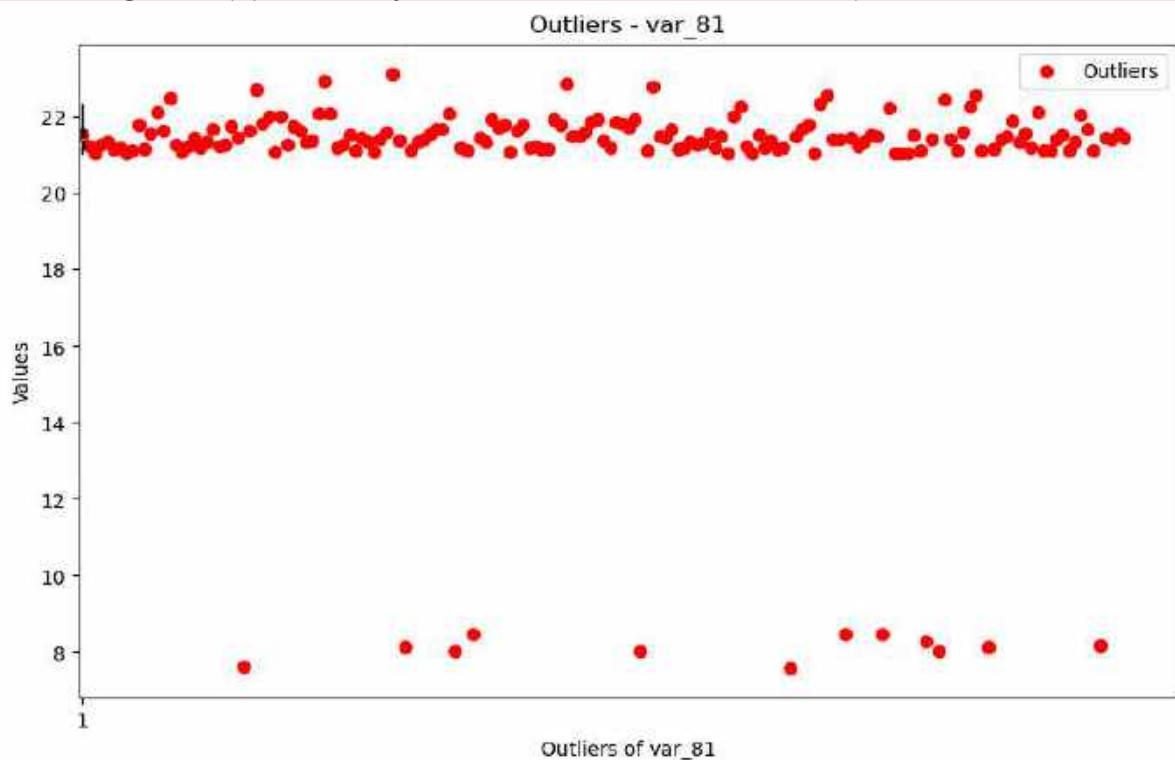
```
Total outliers in column var_79: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



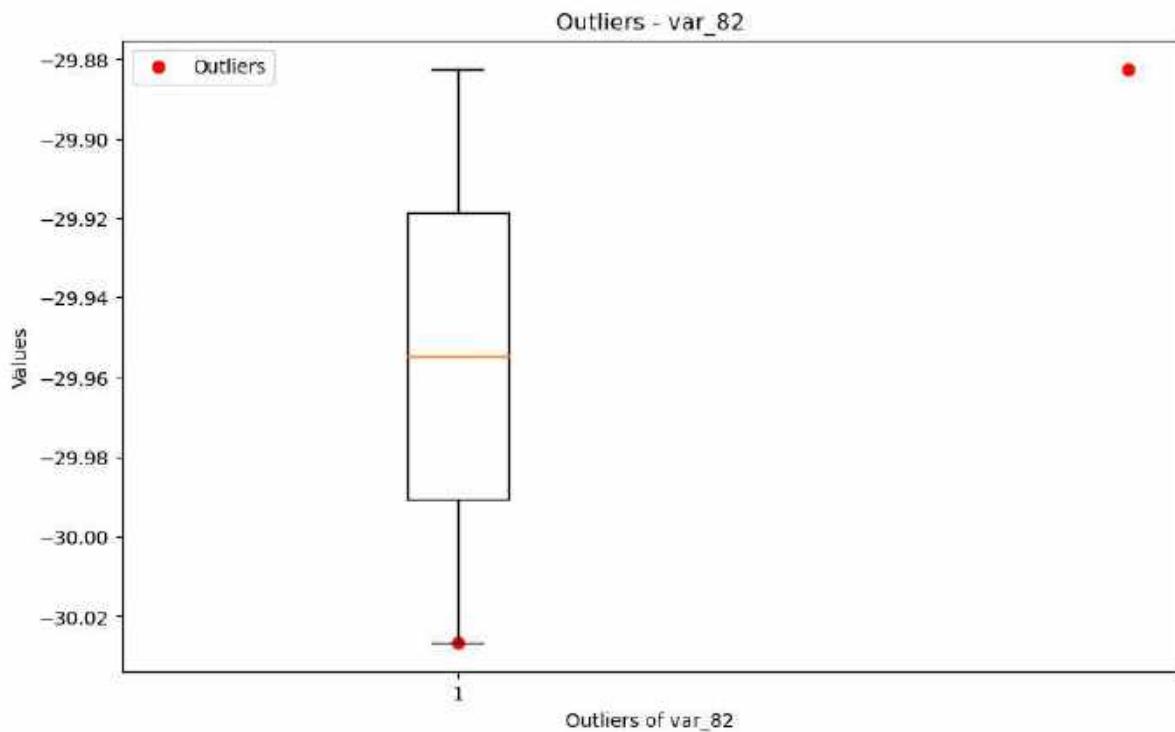
Total outliers in column var\_80: 22

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

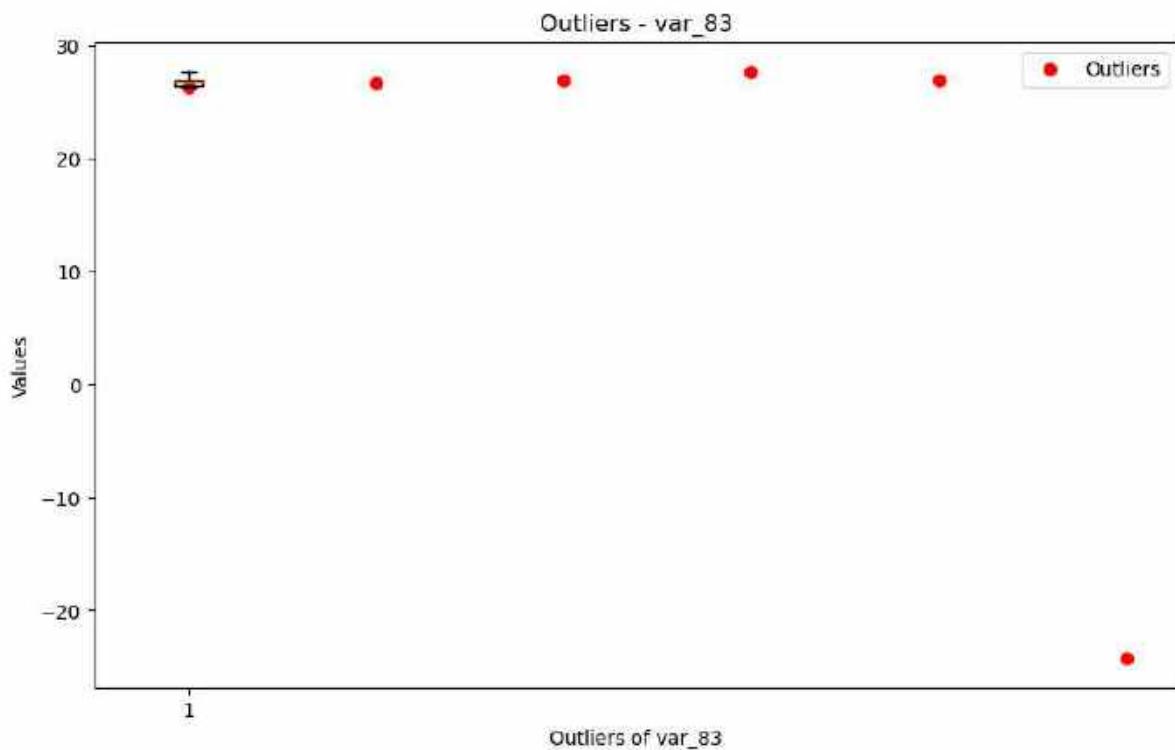


Total outliers in column var\_81: 169

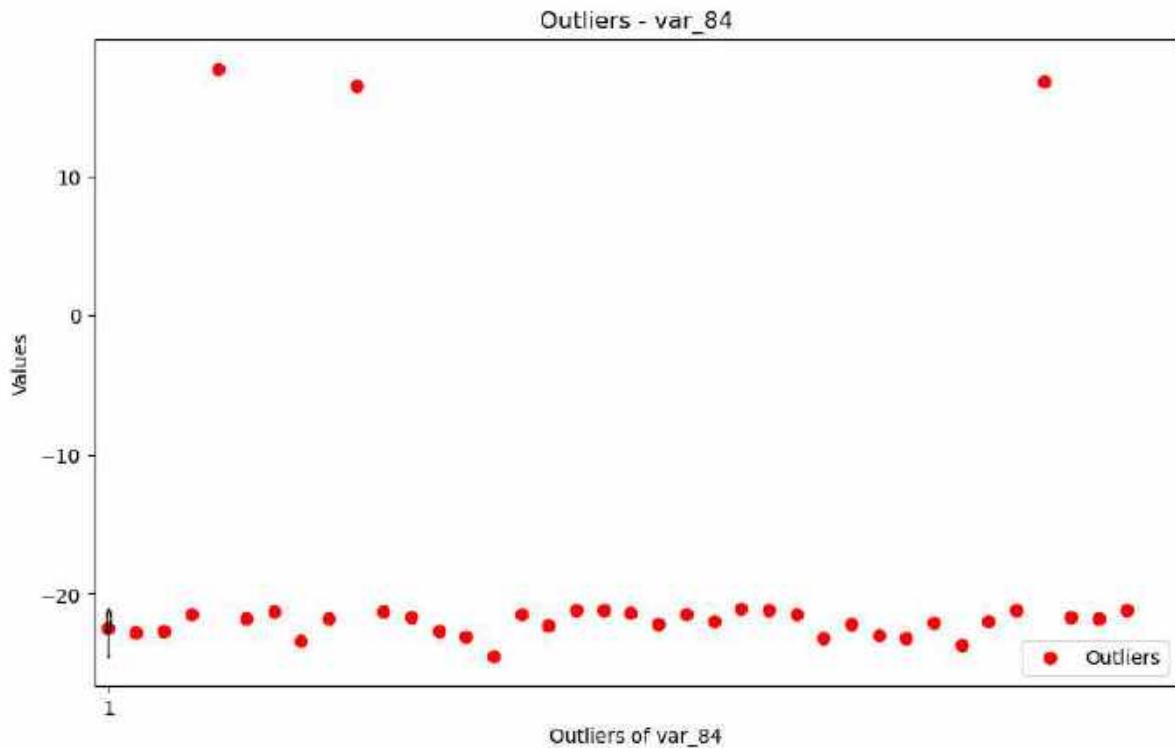
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_82: 2
```

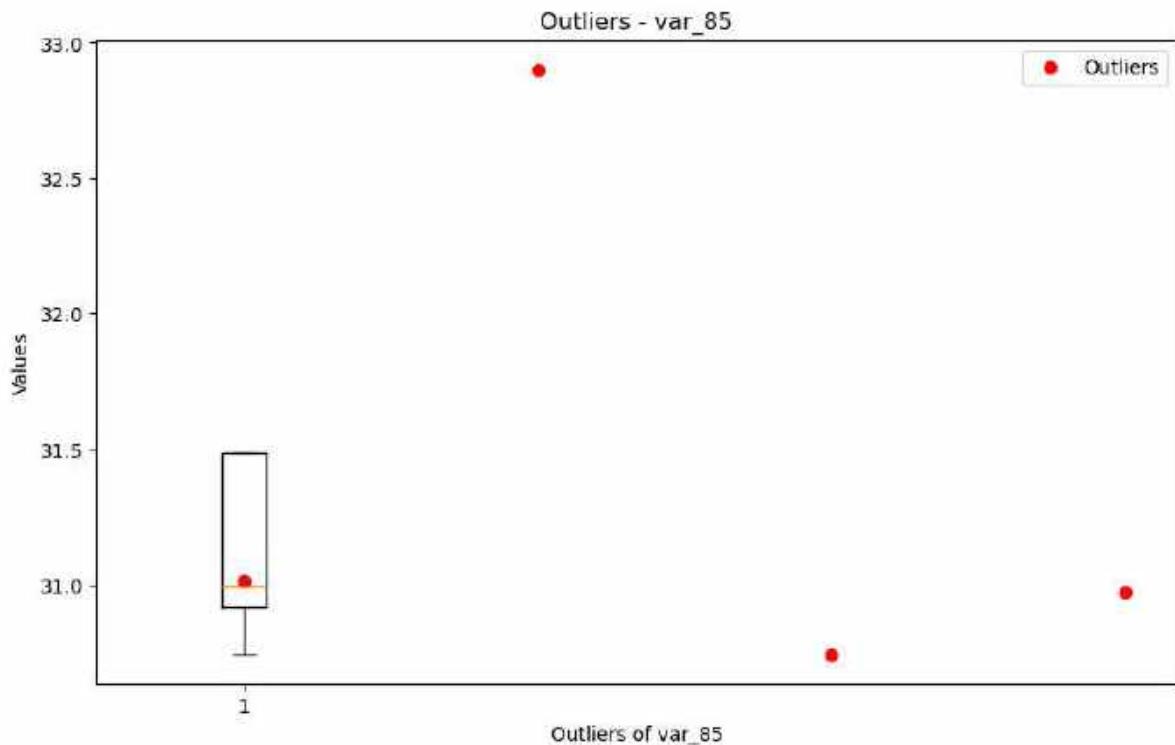


```
Total outliers in column var_83: 6  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



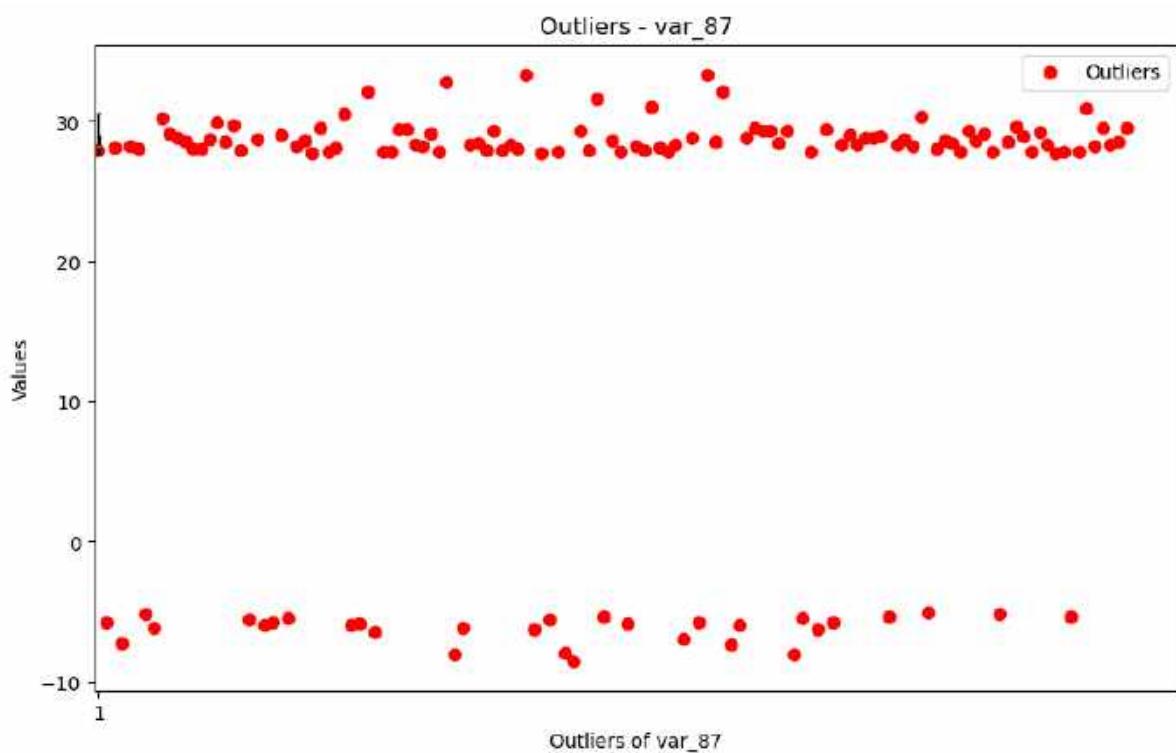
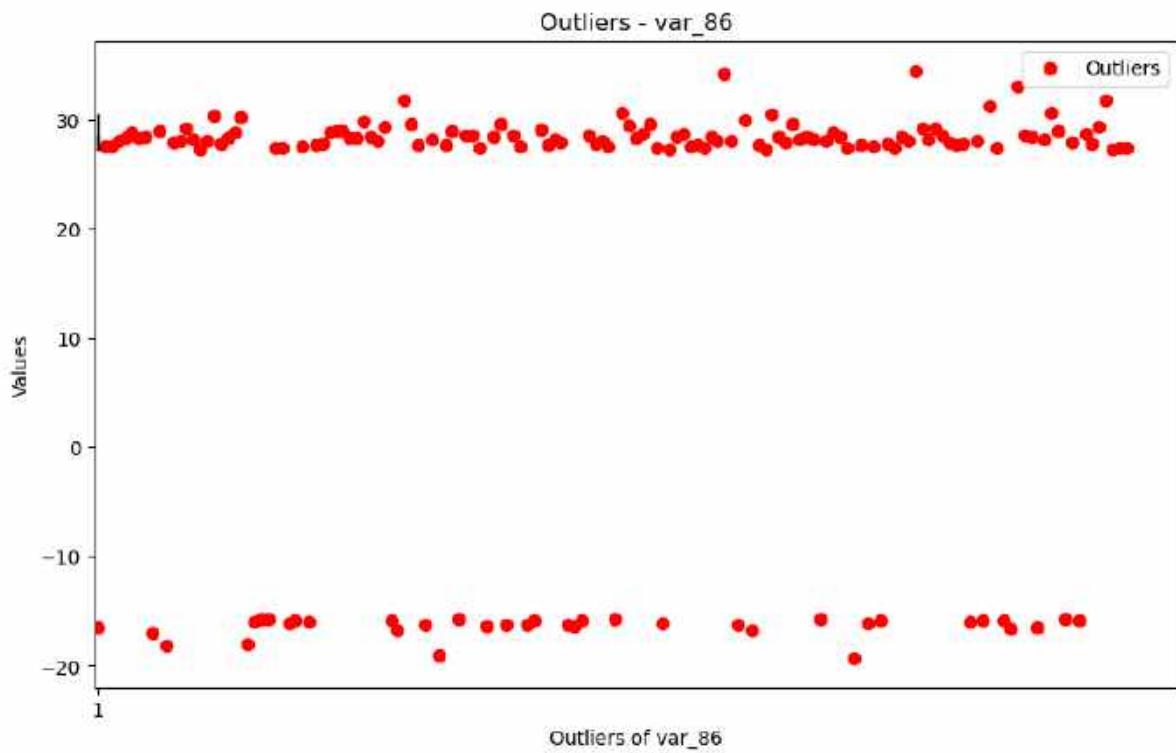
Total outliers in column var\_84: 38

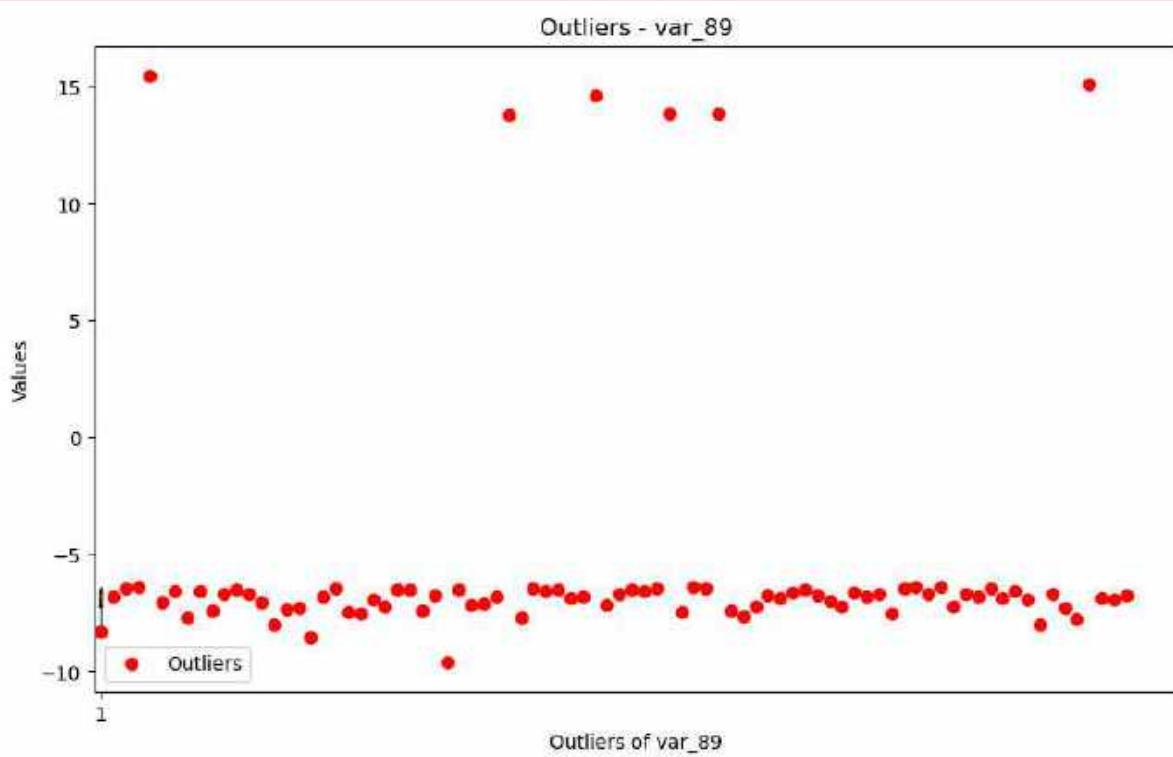
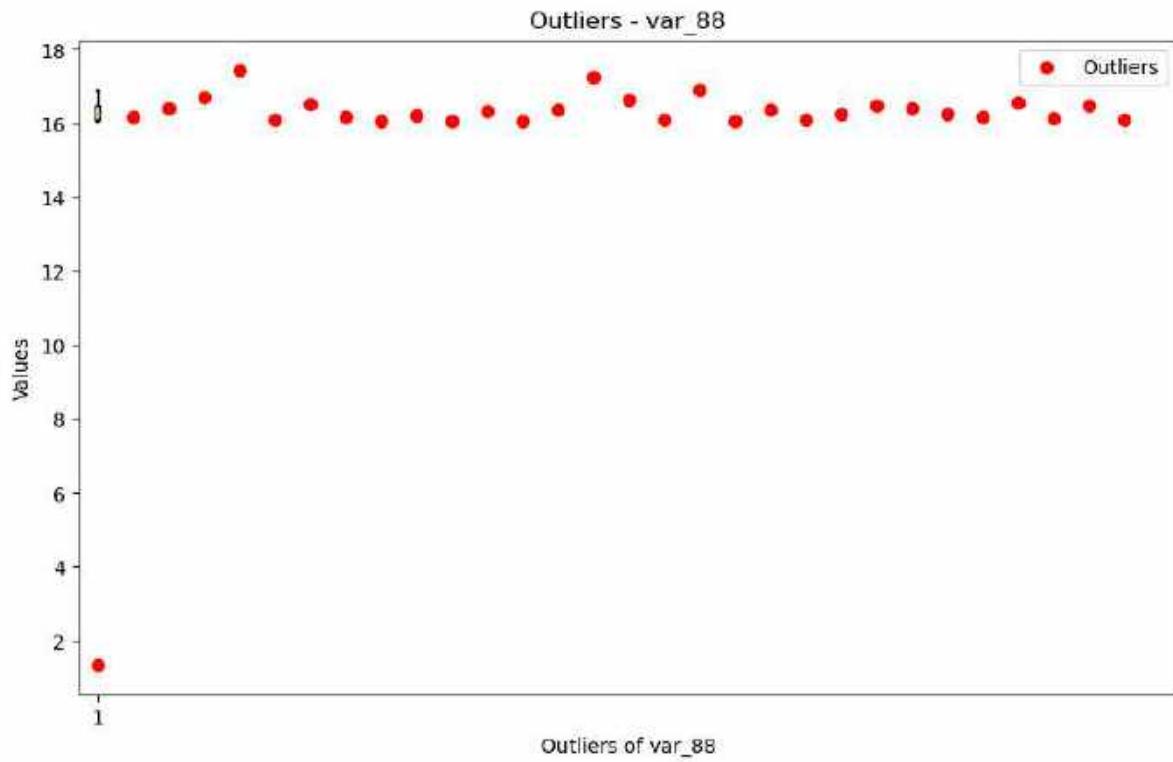
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

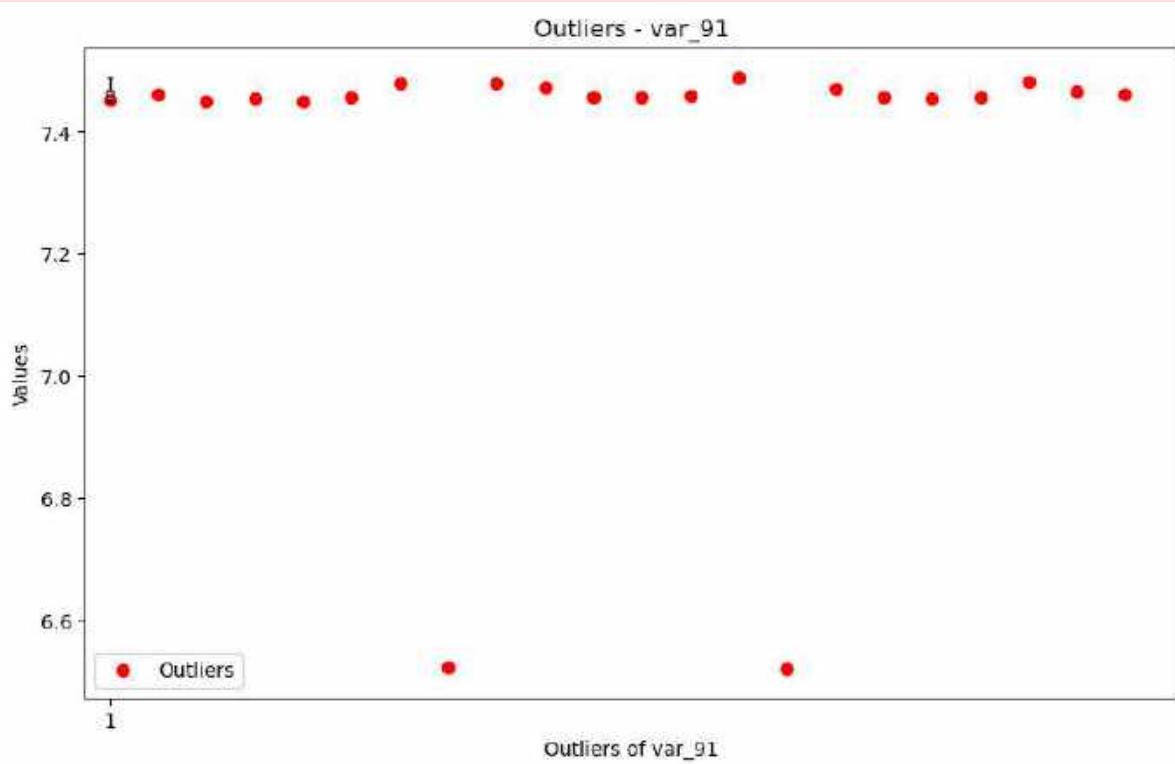
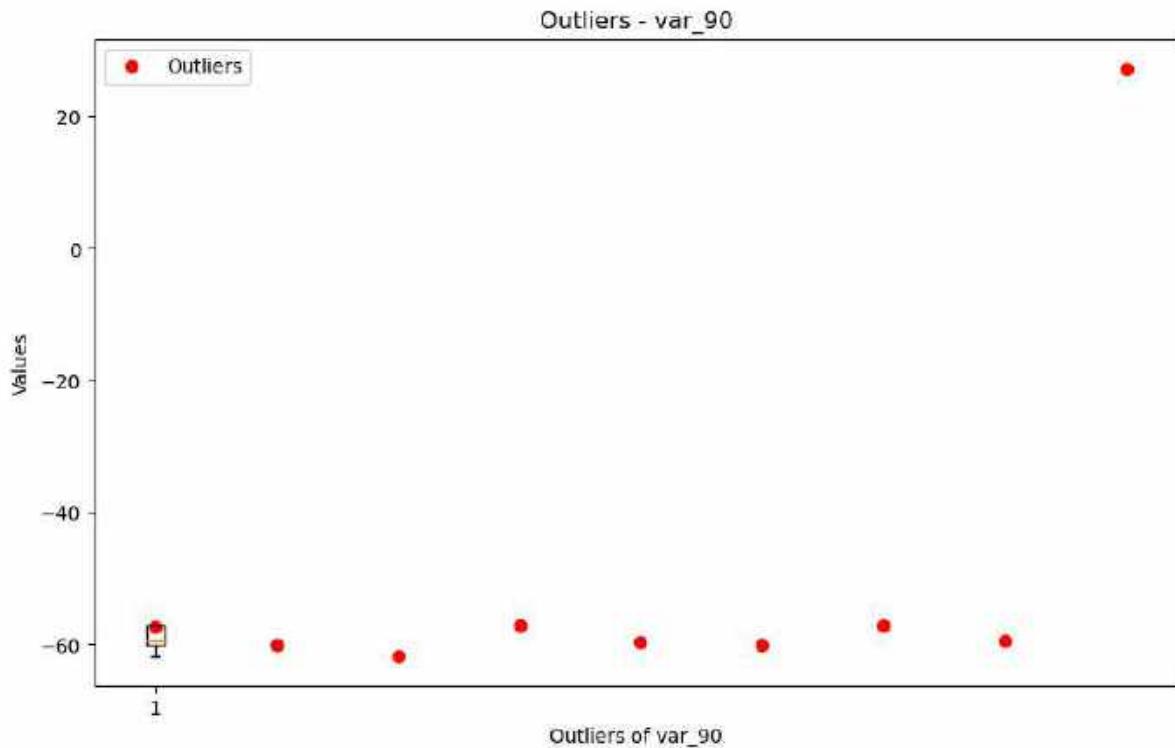


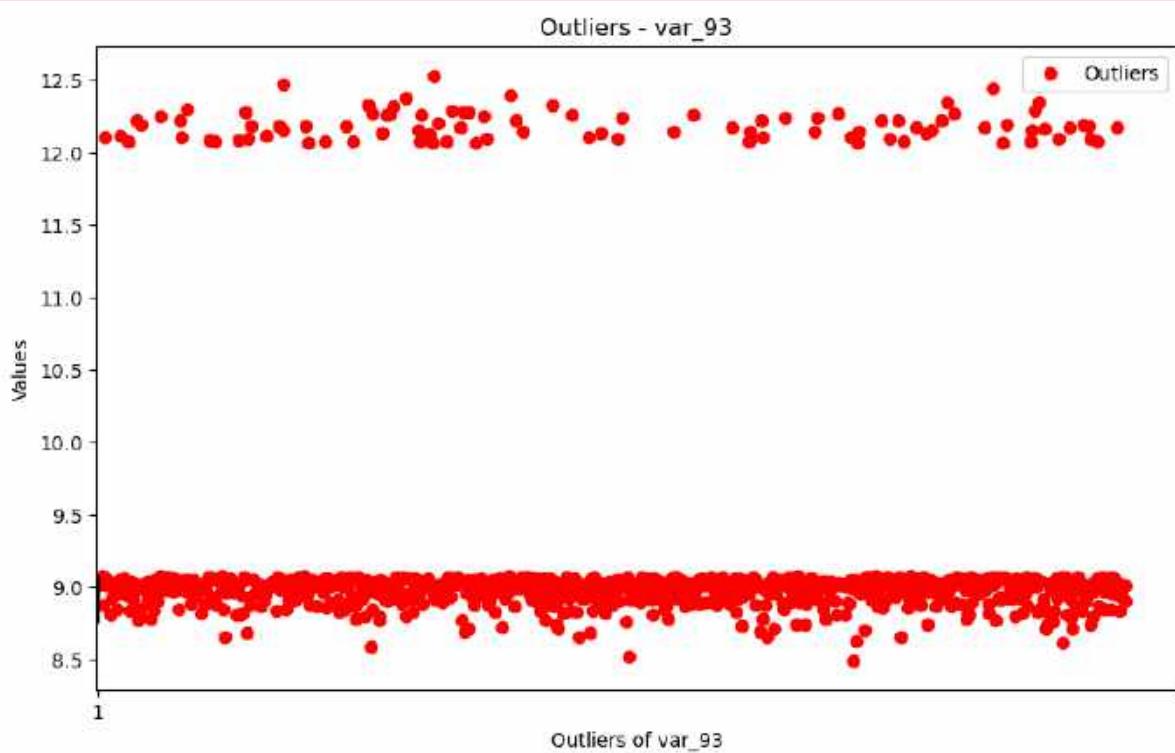
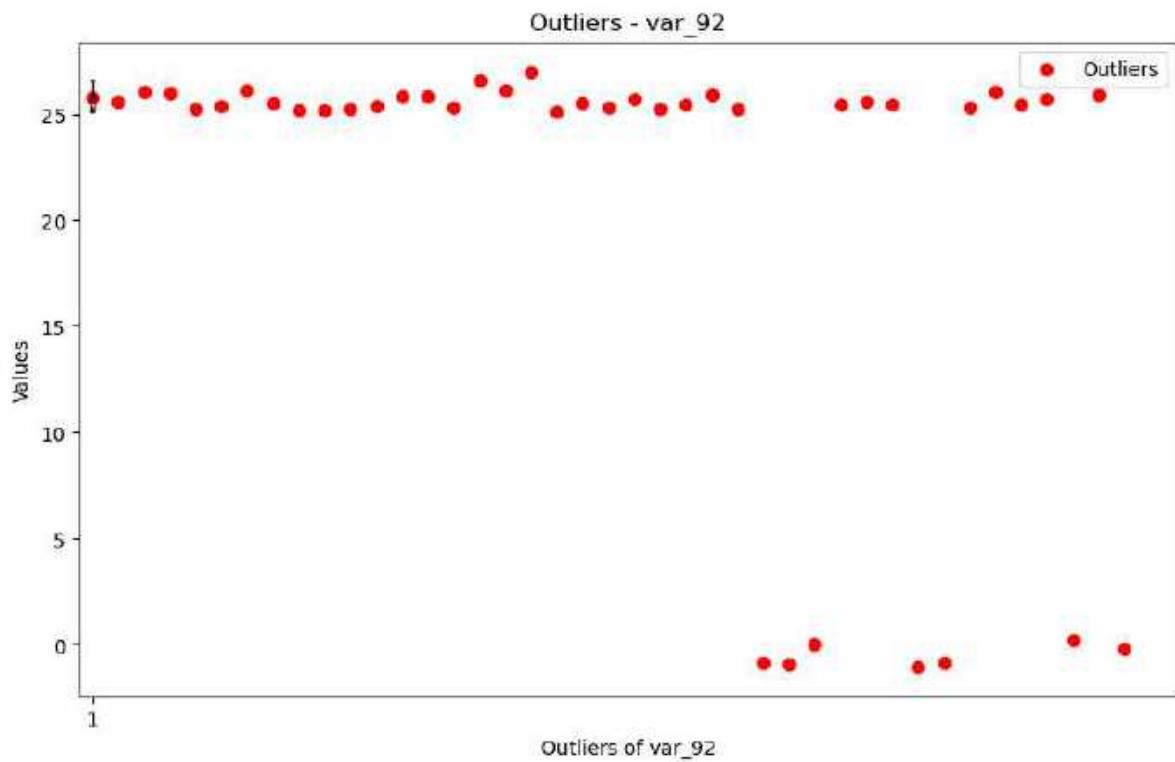
Total outliers in column var\_85: 4

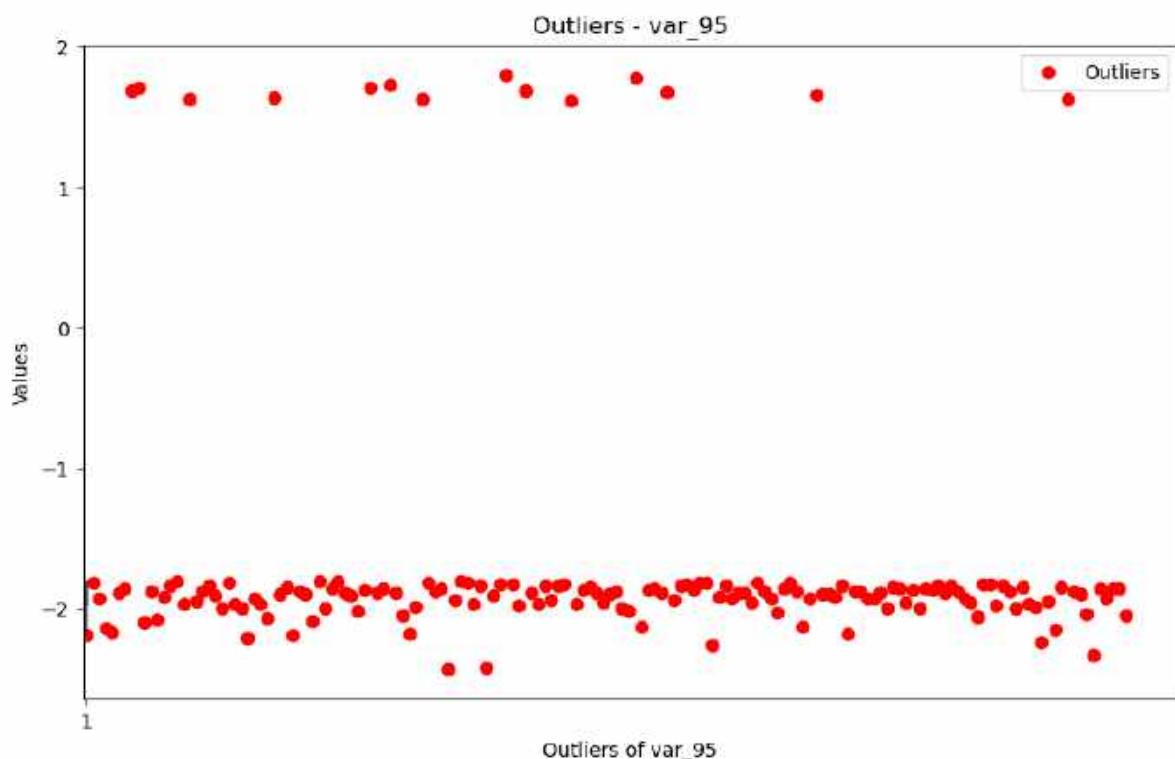
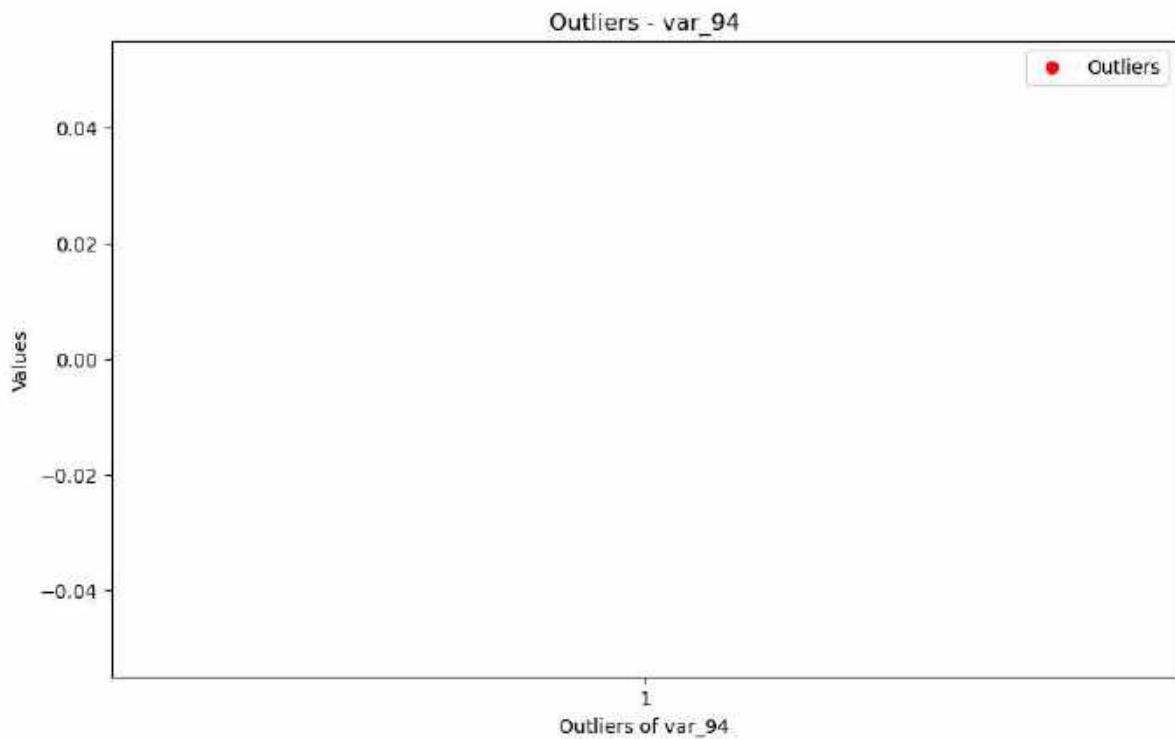
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

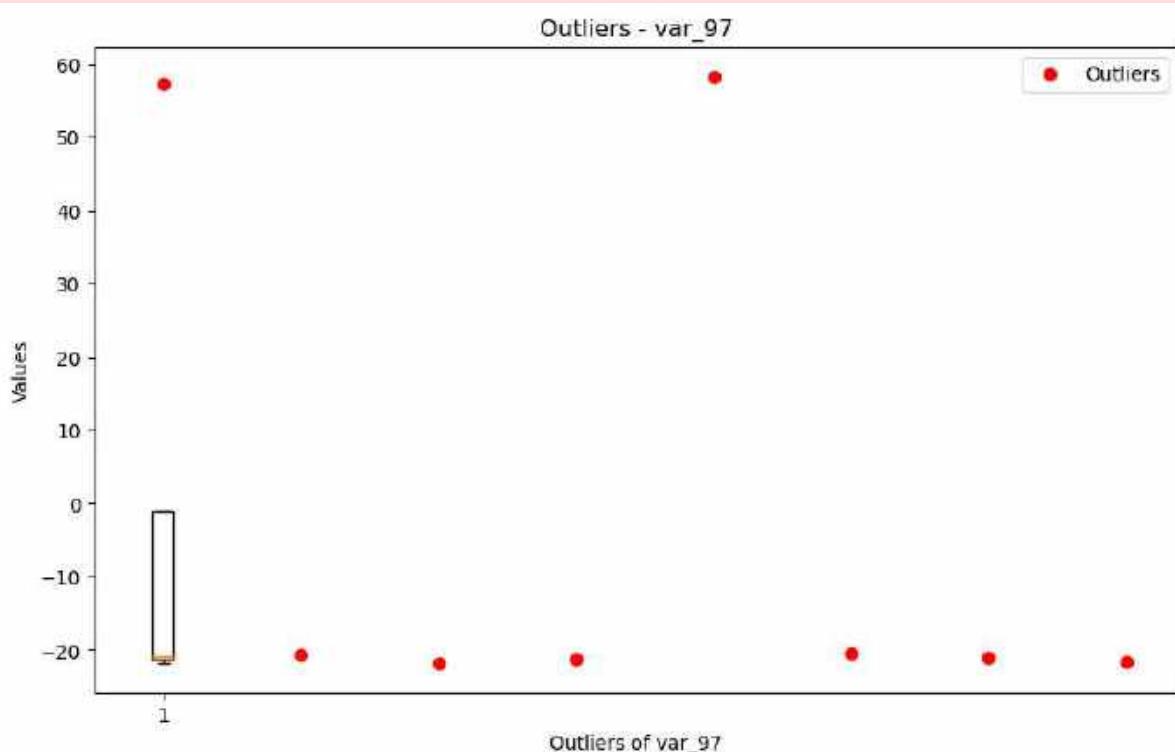
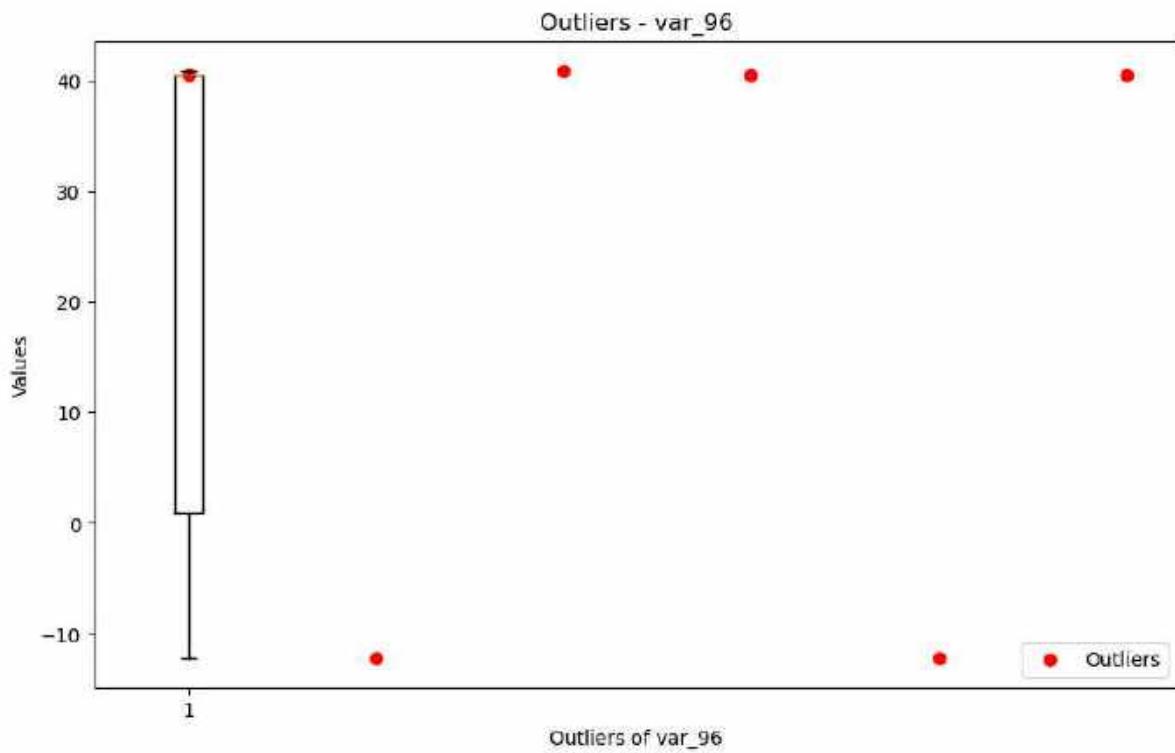


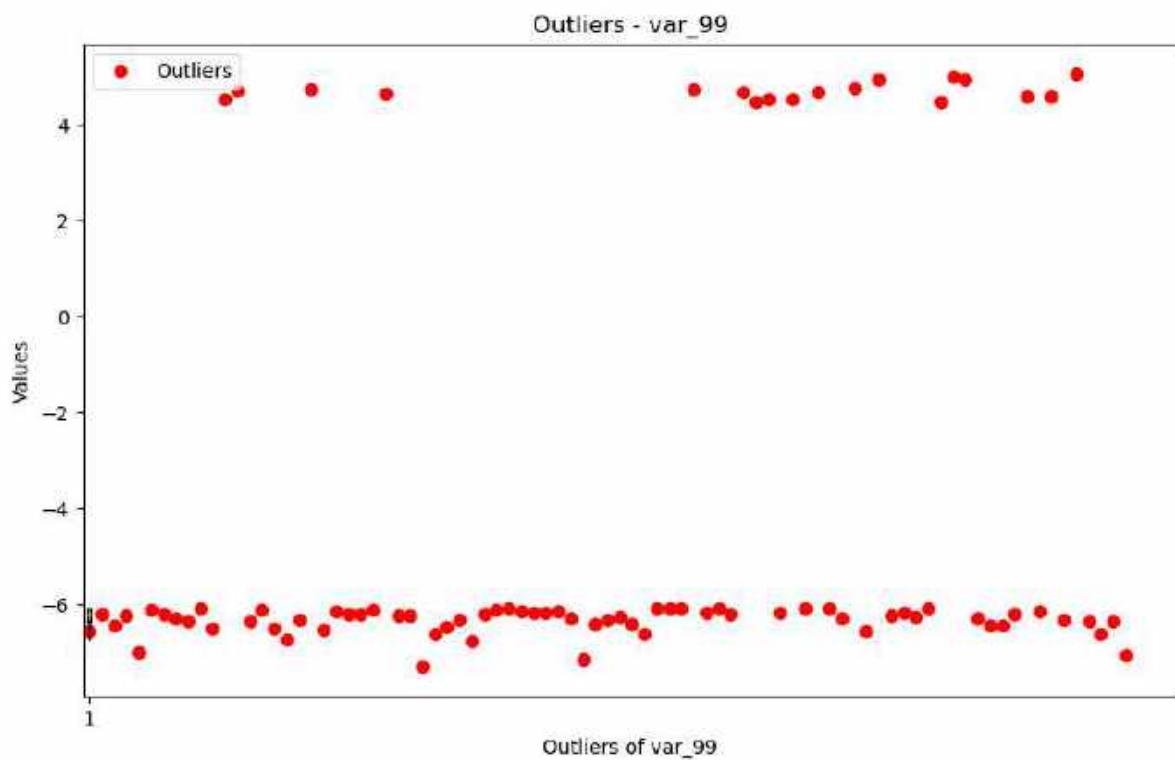
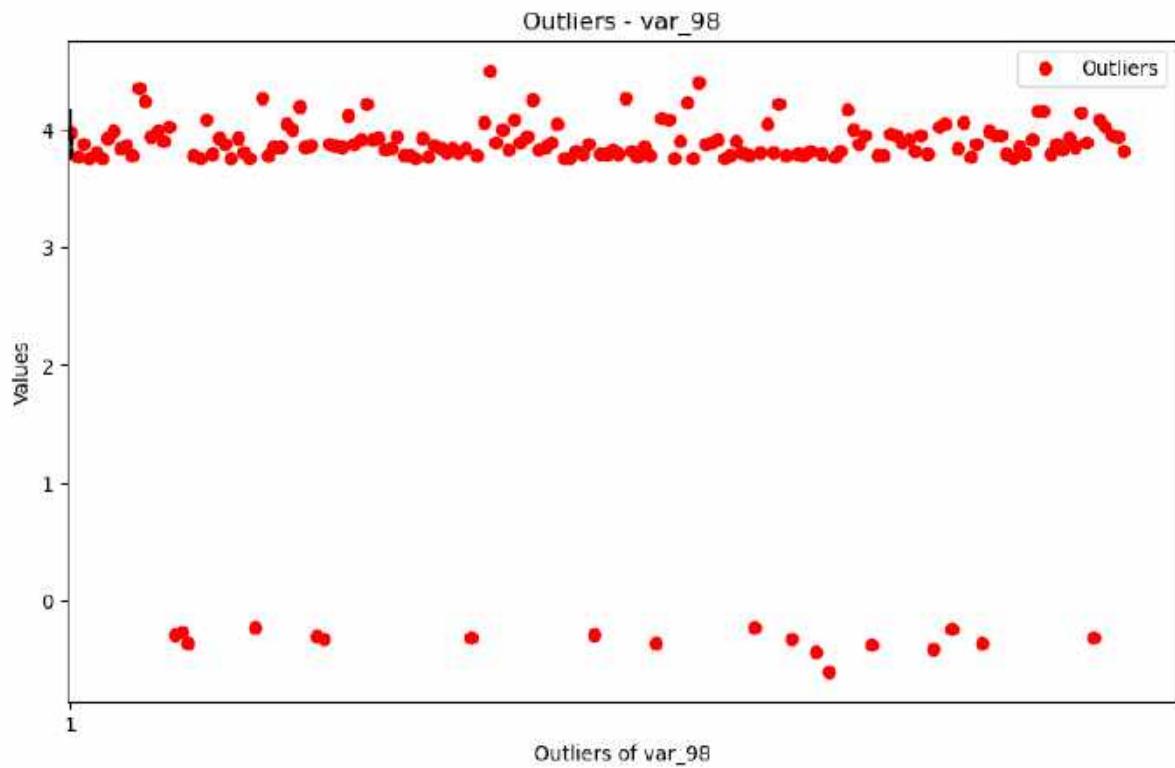


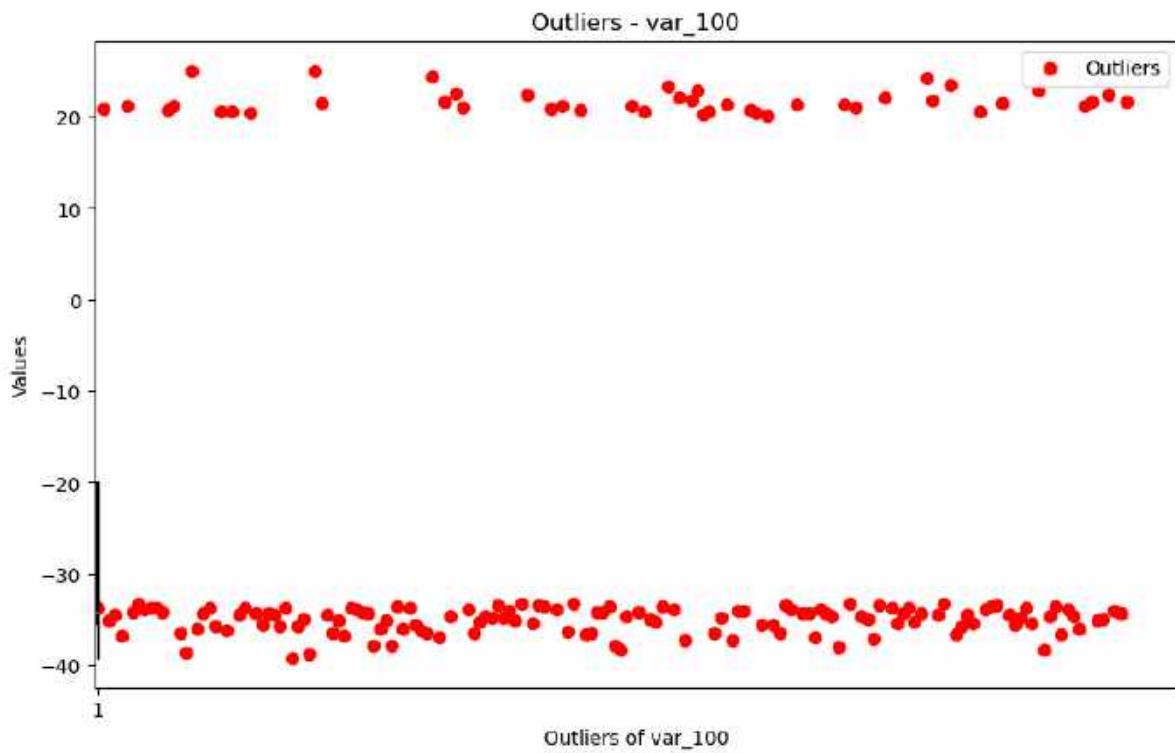












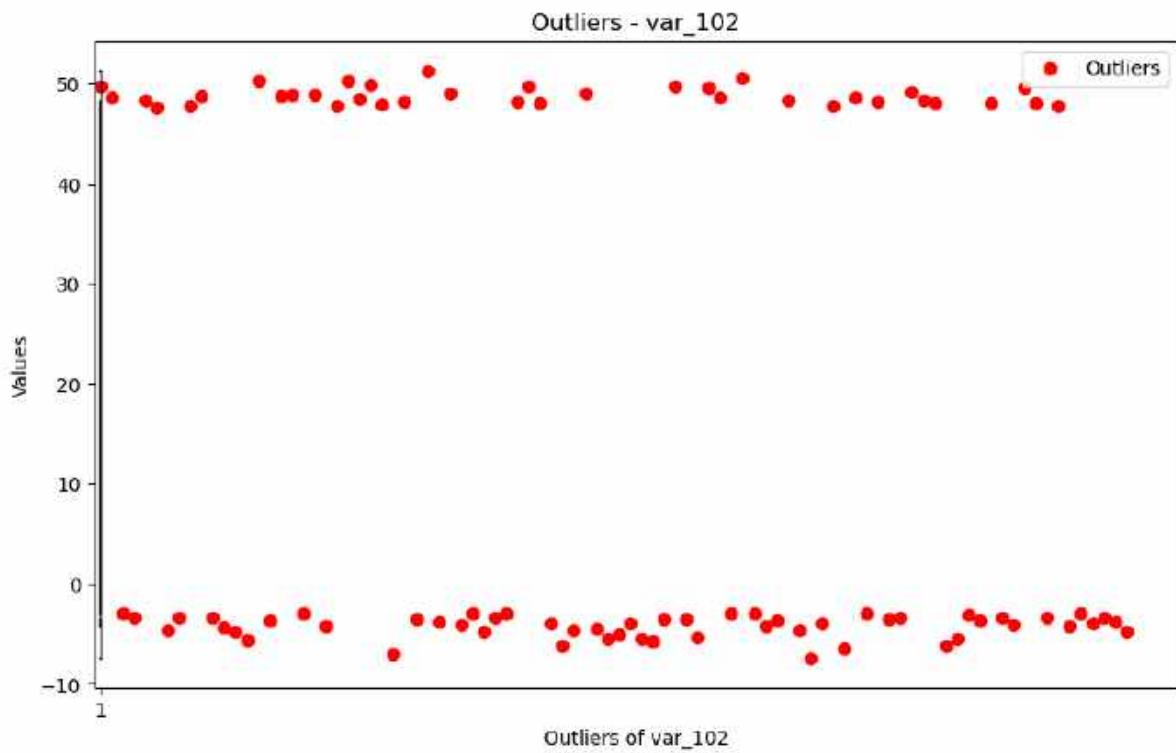
```
Total outliers in column var_100: 176
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



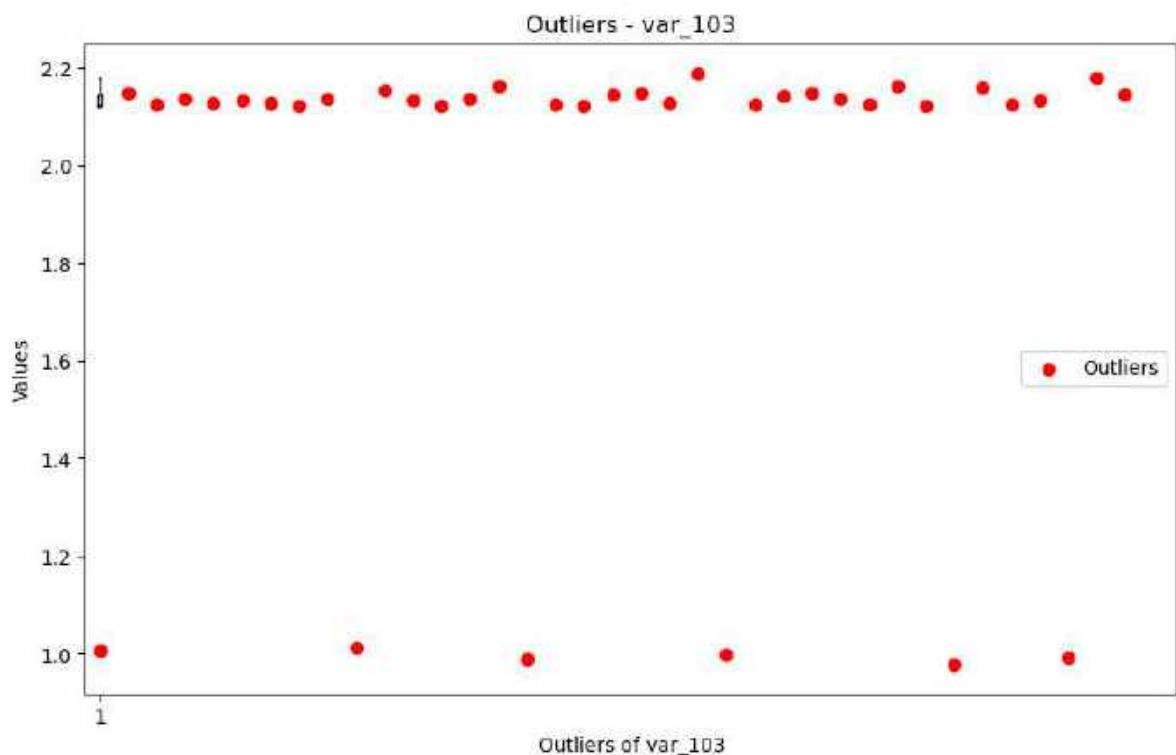
```
Total outliers in column var_101: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



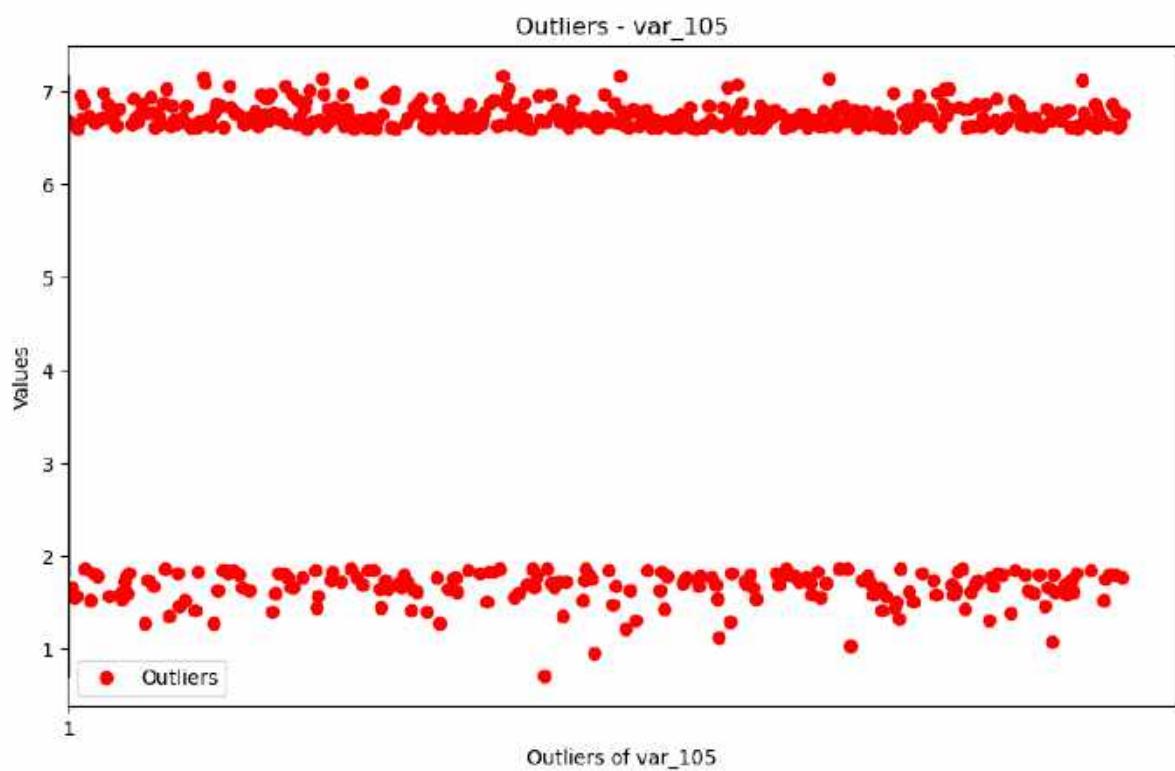
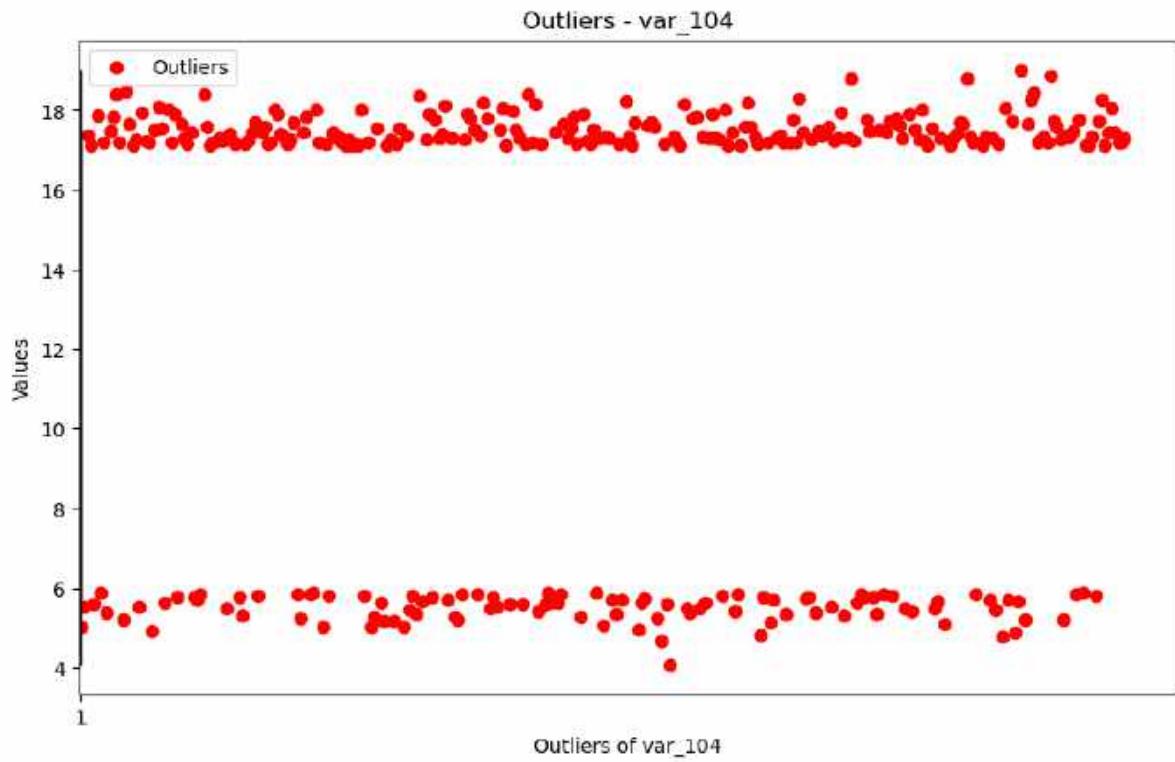
Total outliers in column var\_102: 92

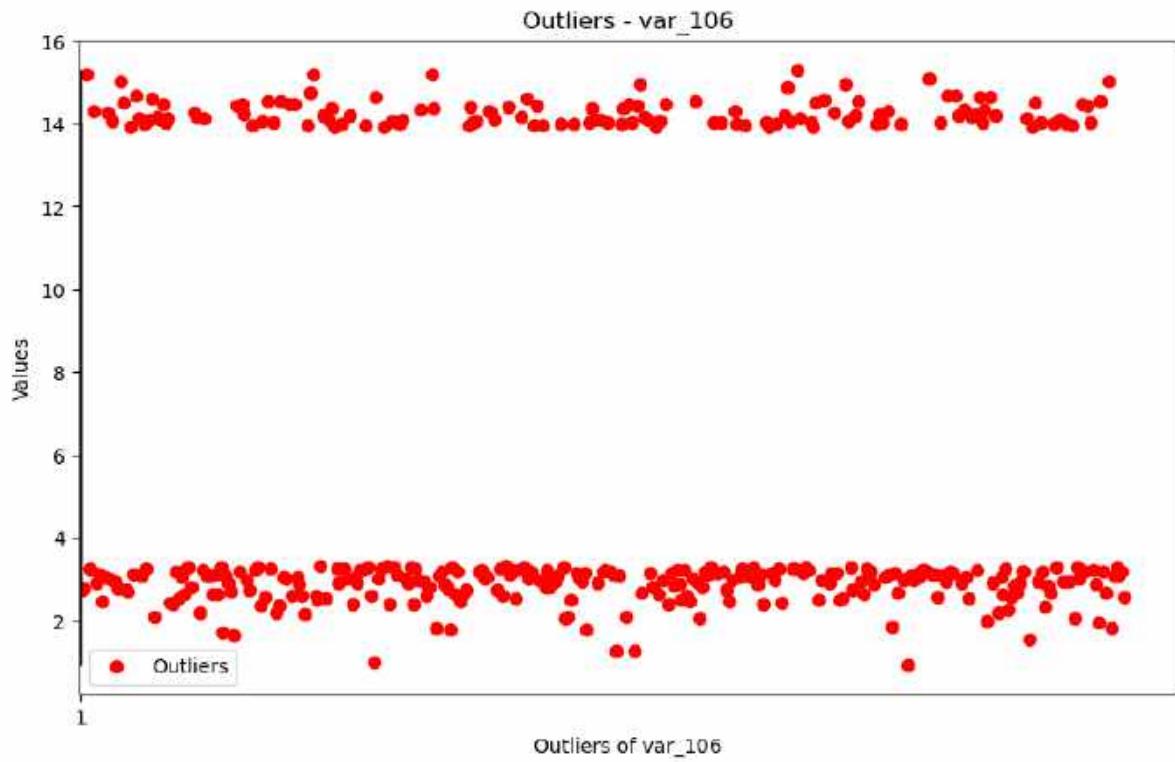
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



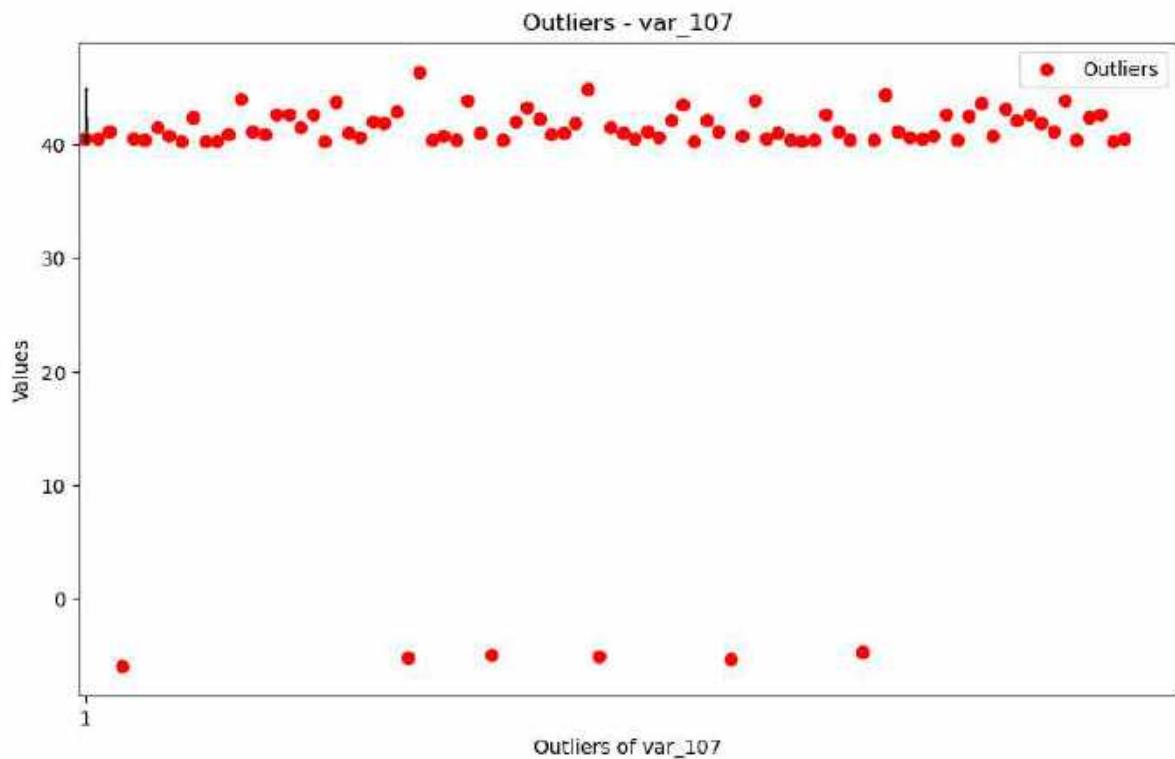
Total outliers in column var\_103: 37

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



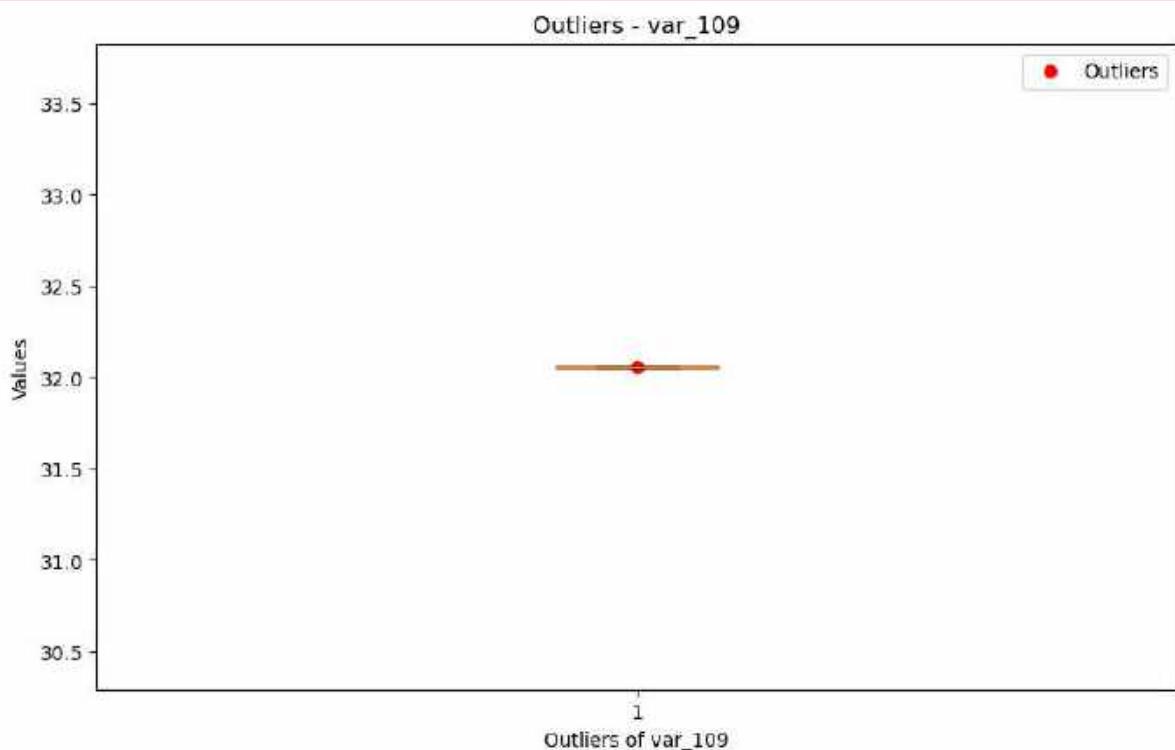
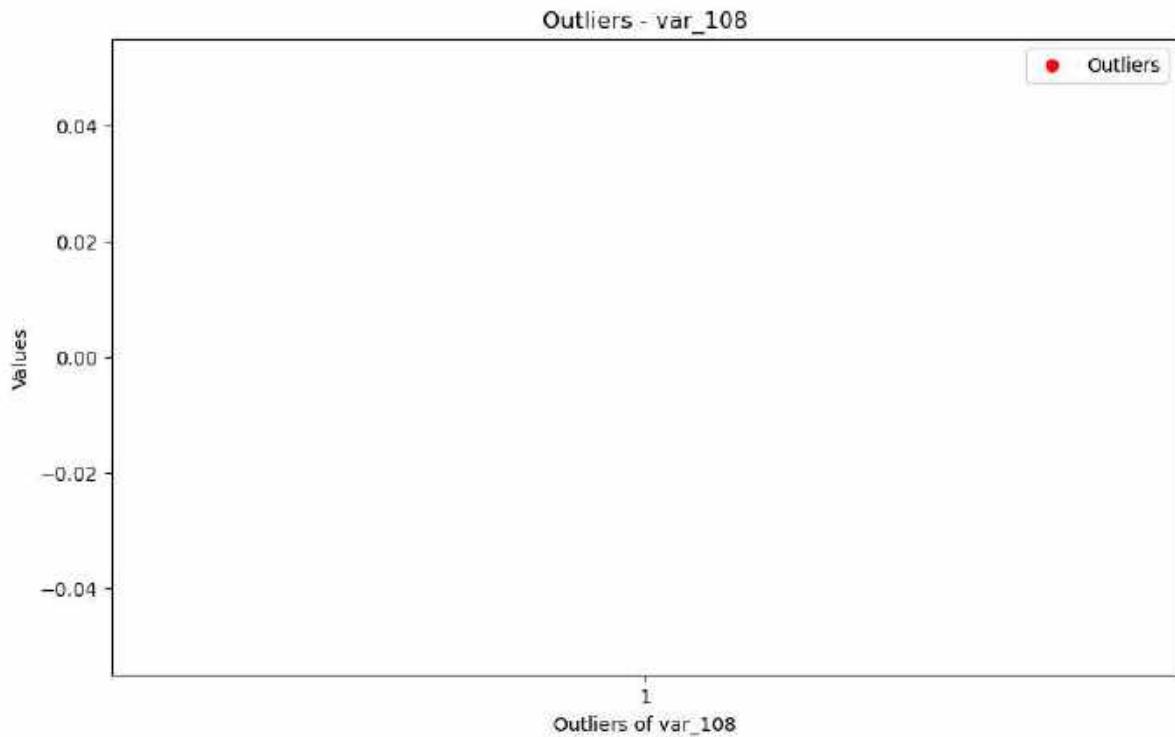


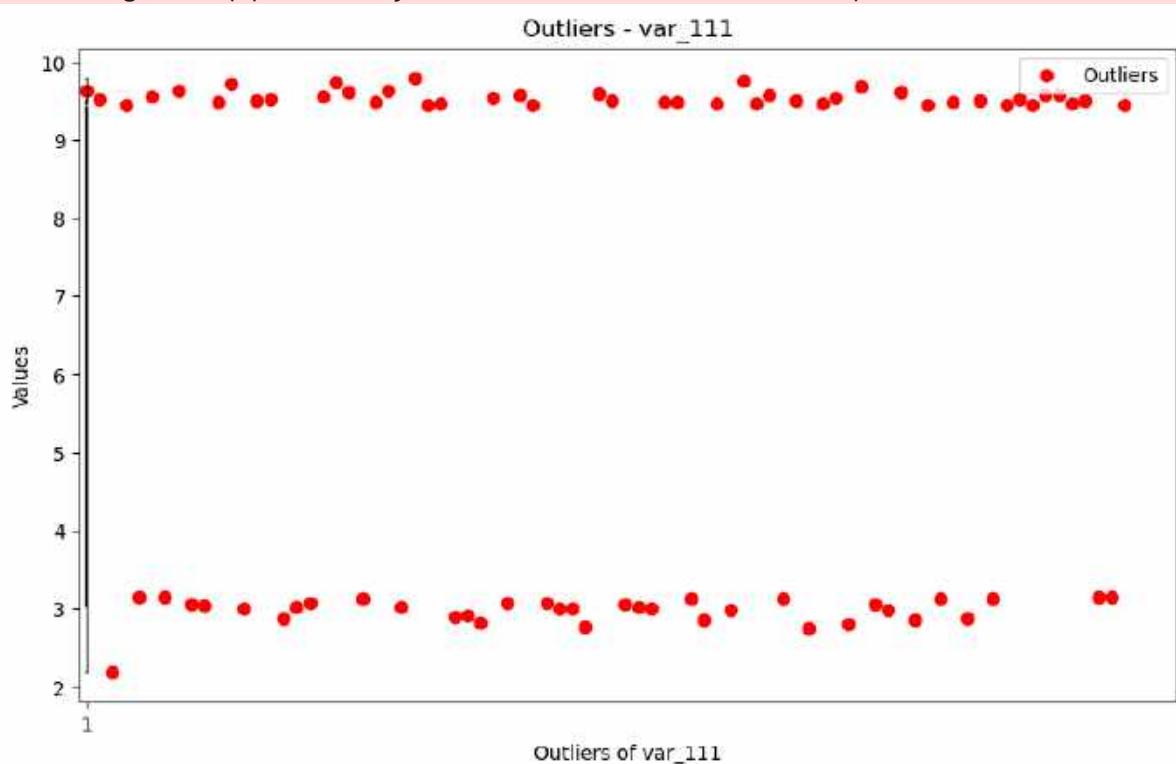
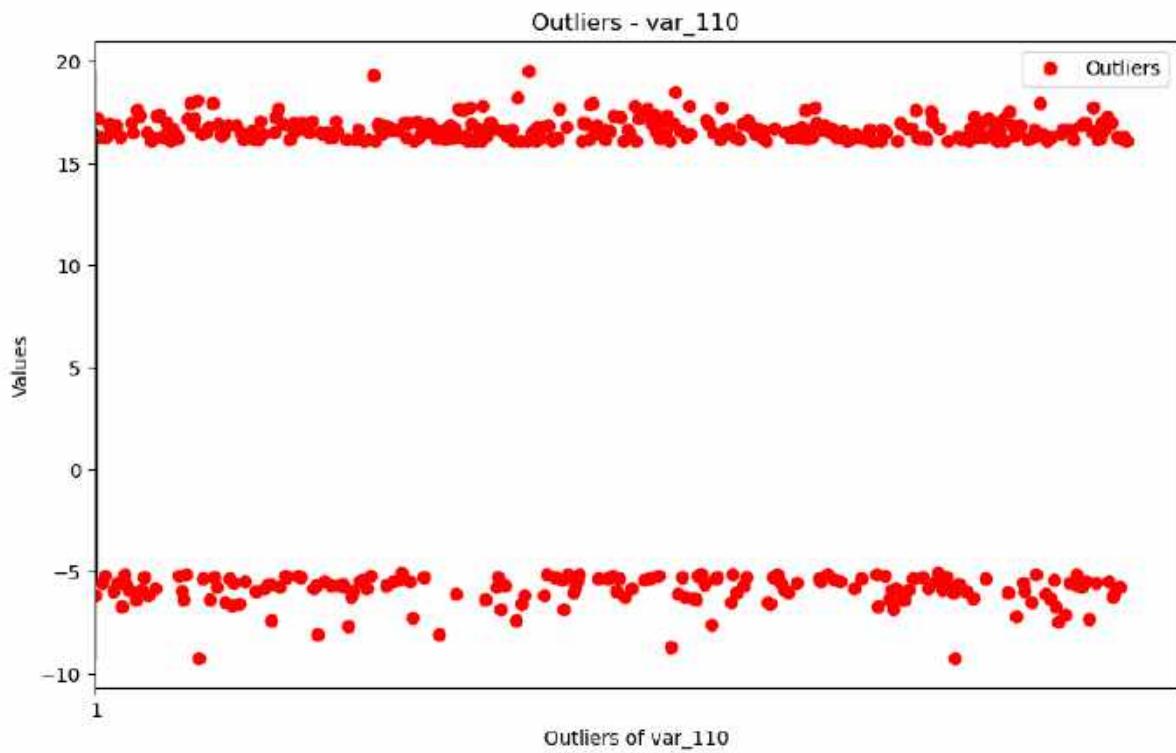
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_106: 397
```



```
Total outliers in column var_107: 88
```

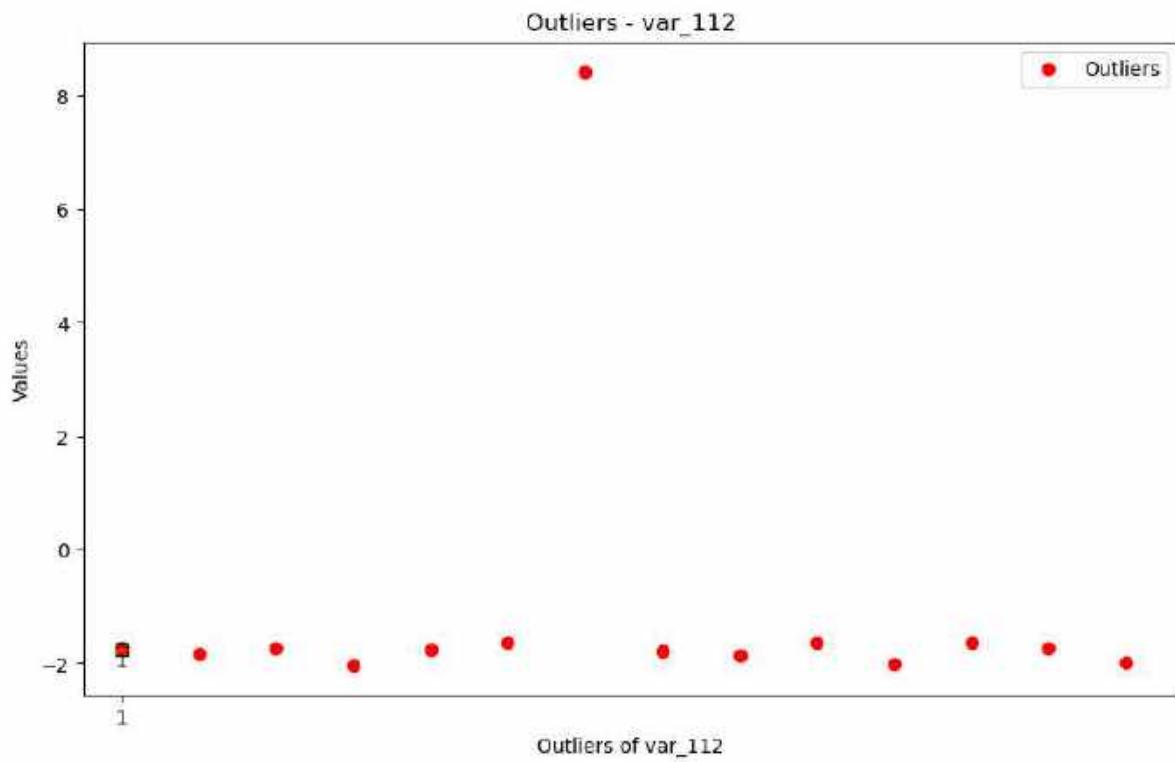
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





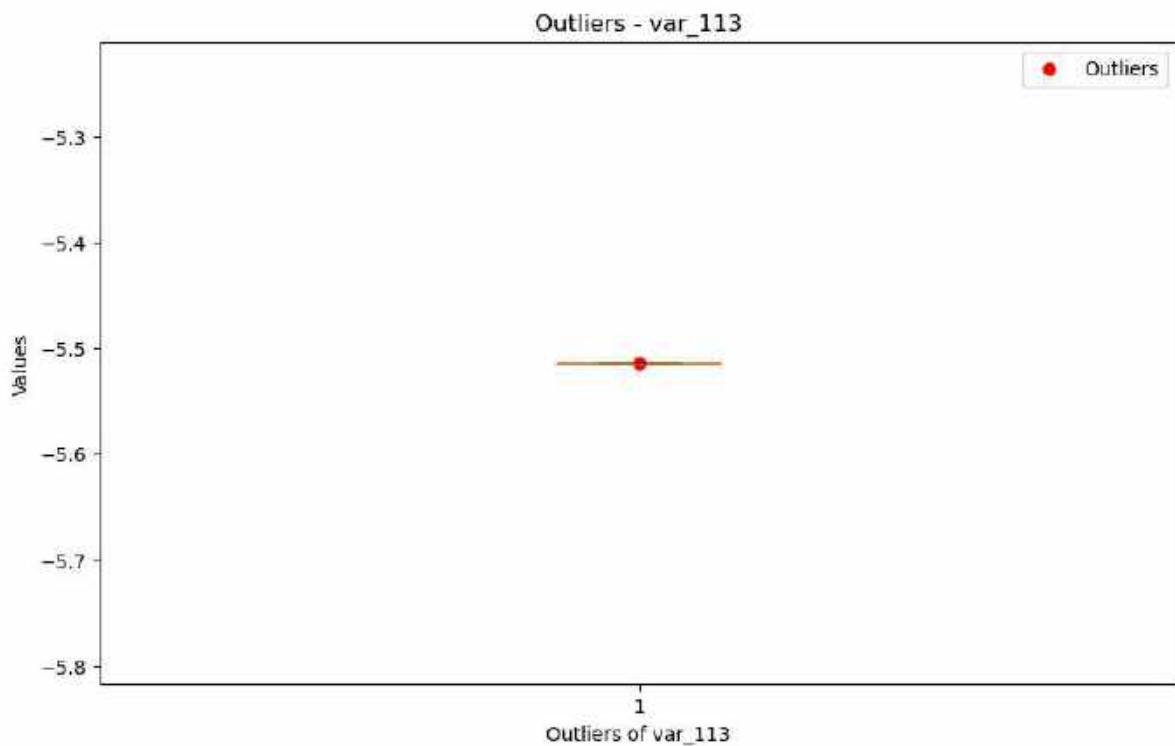
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_111: 80



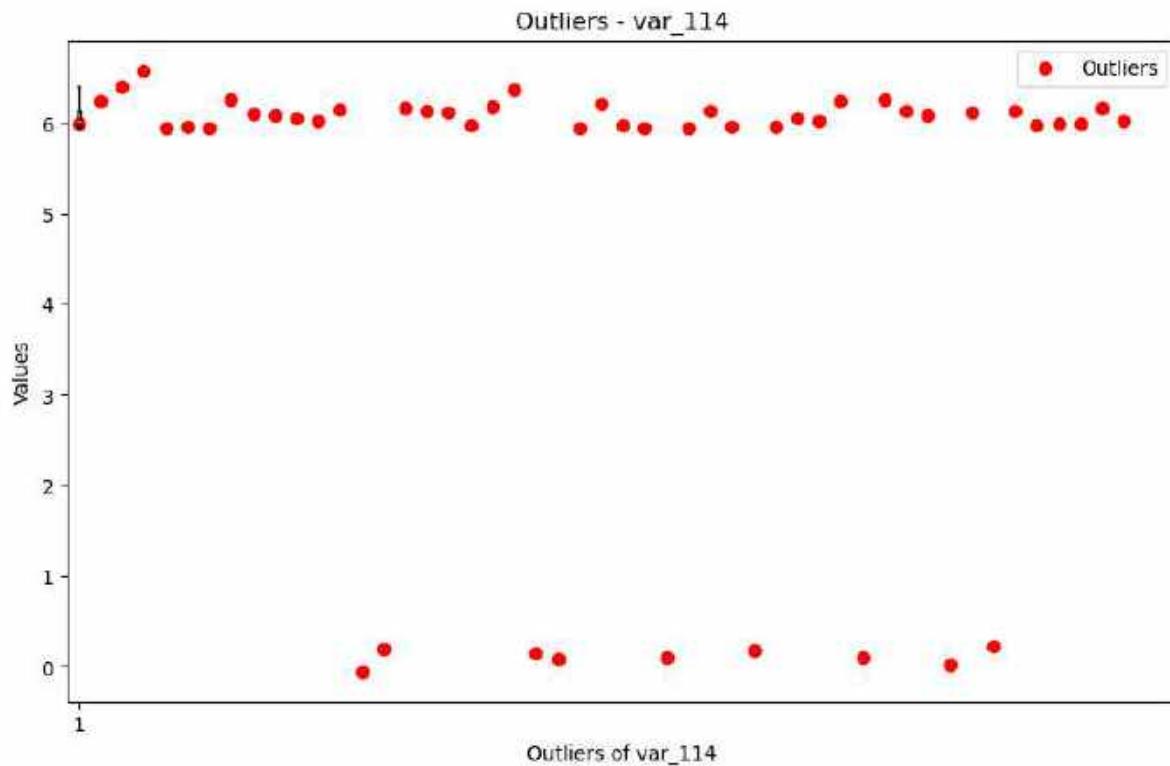
Total outliers in column var\_112: 14

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



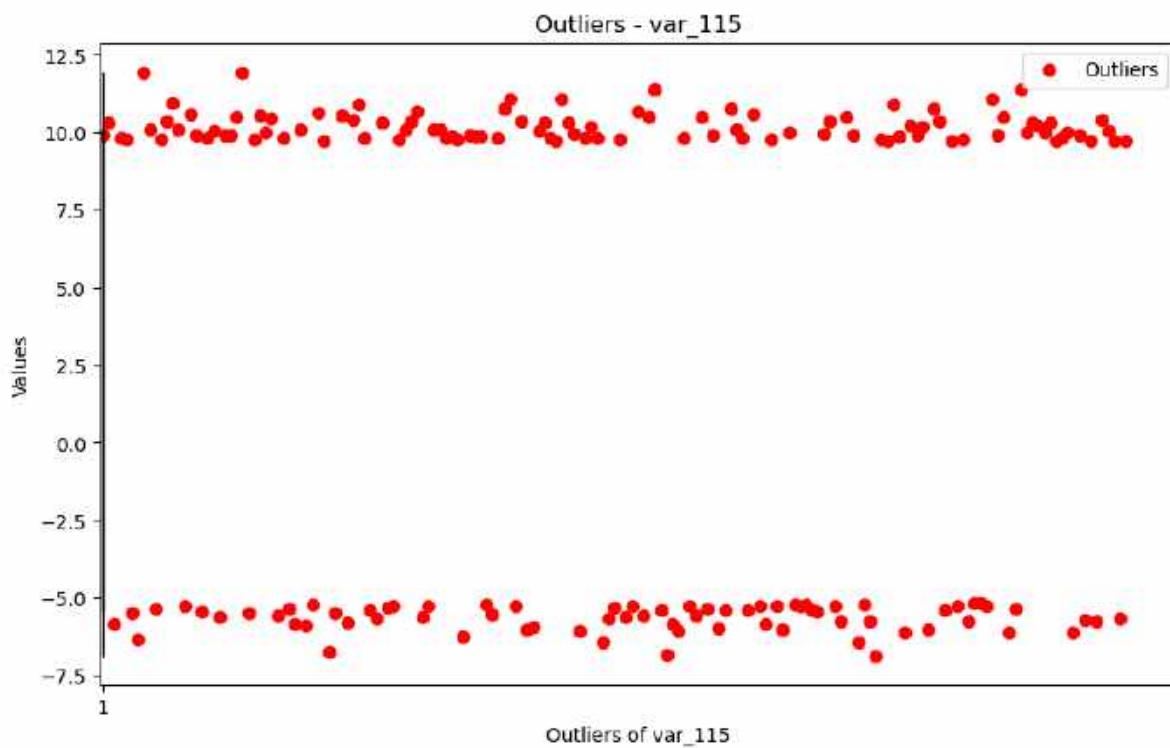
Total outliers in column var\_113: 1

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



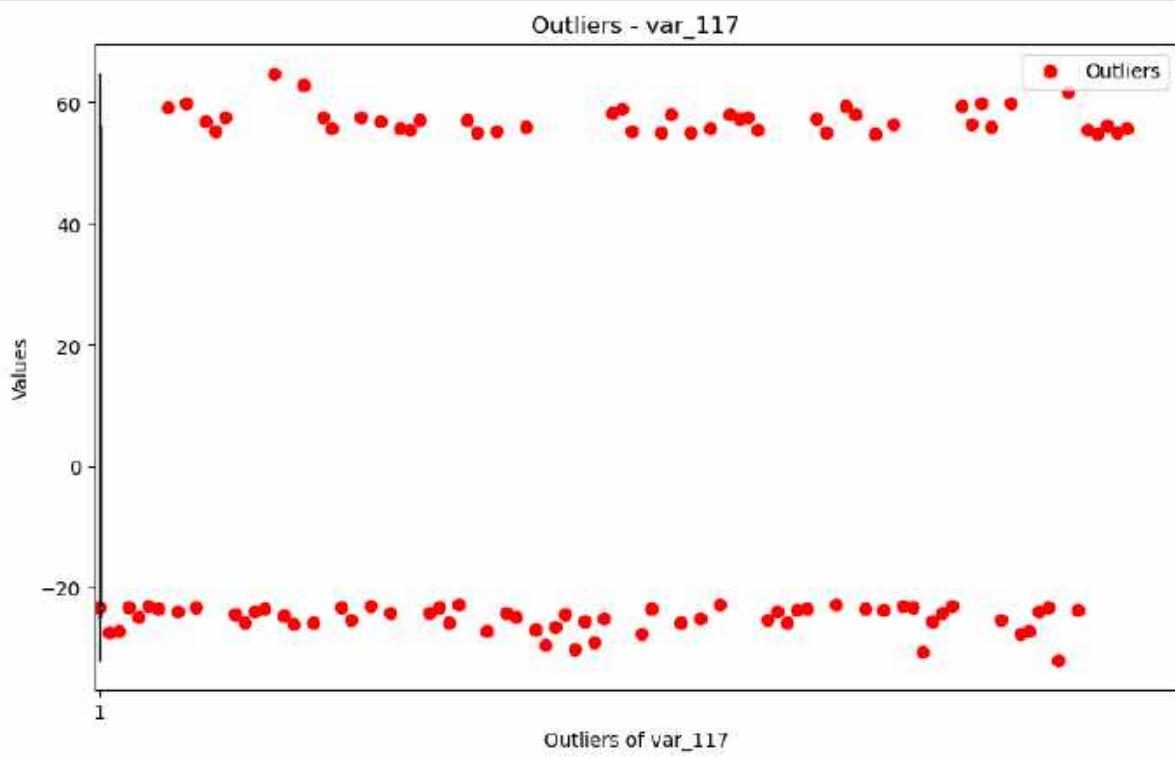
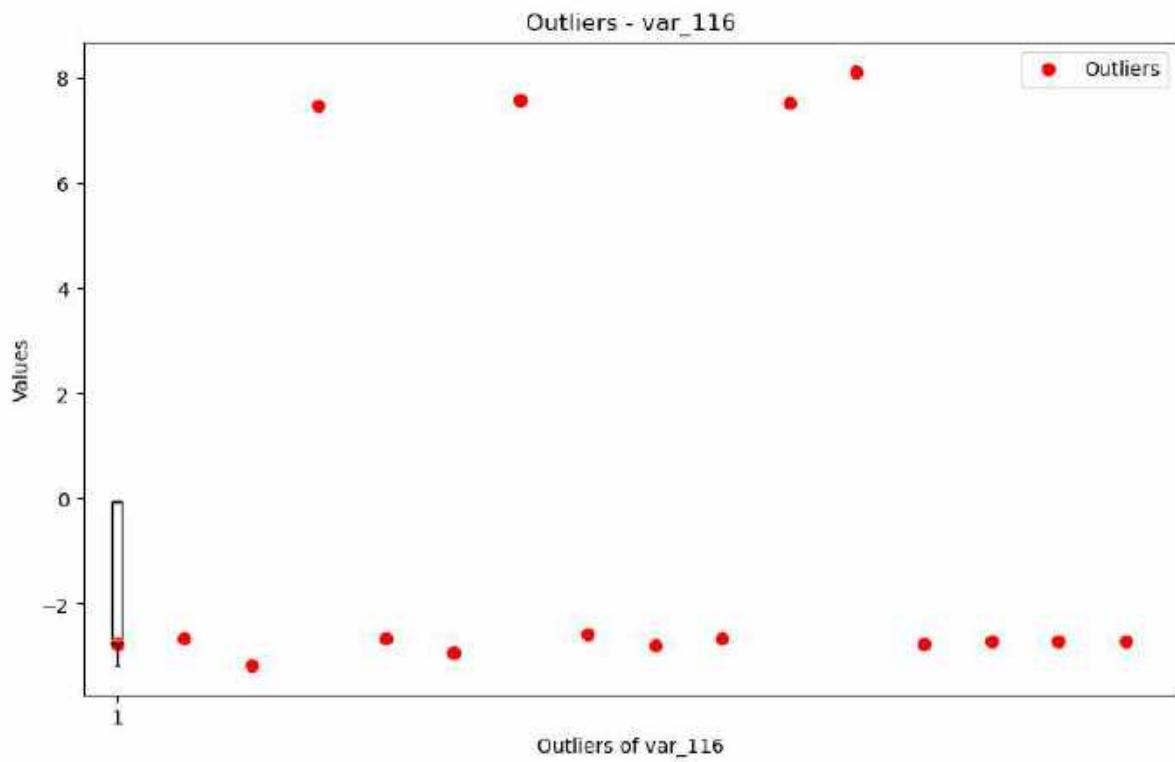
Total outliers in column var\_114: 49

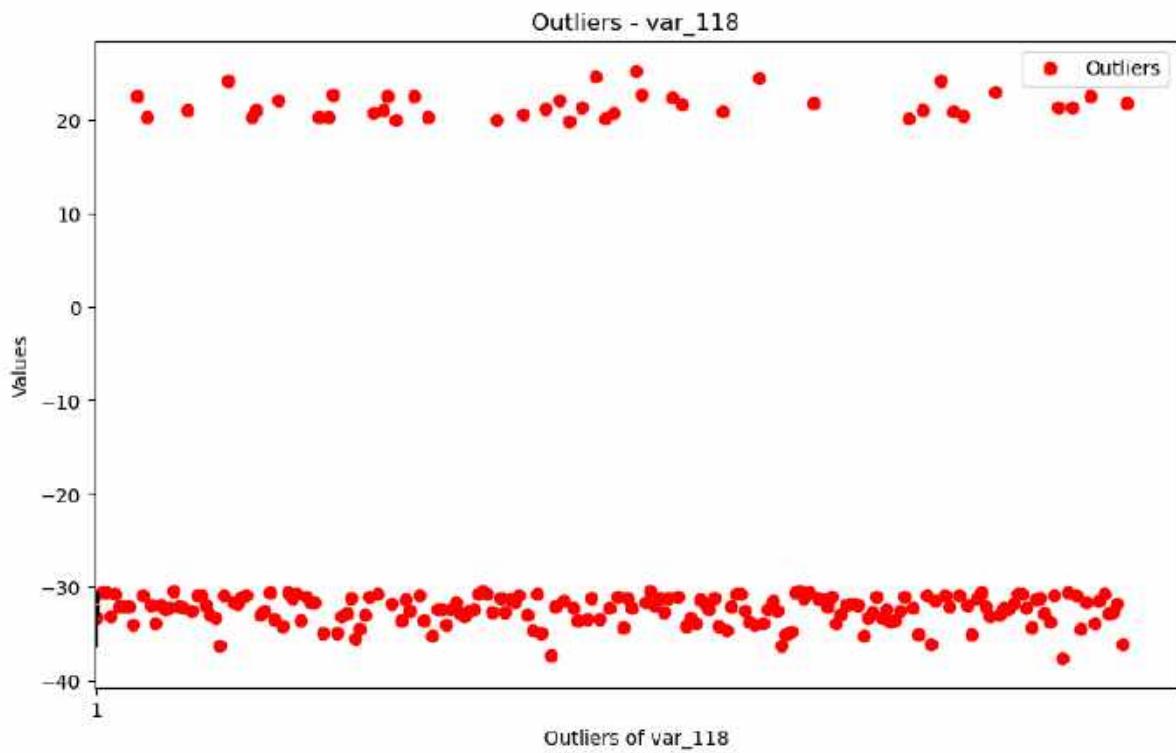
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_115: 177

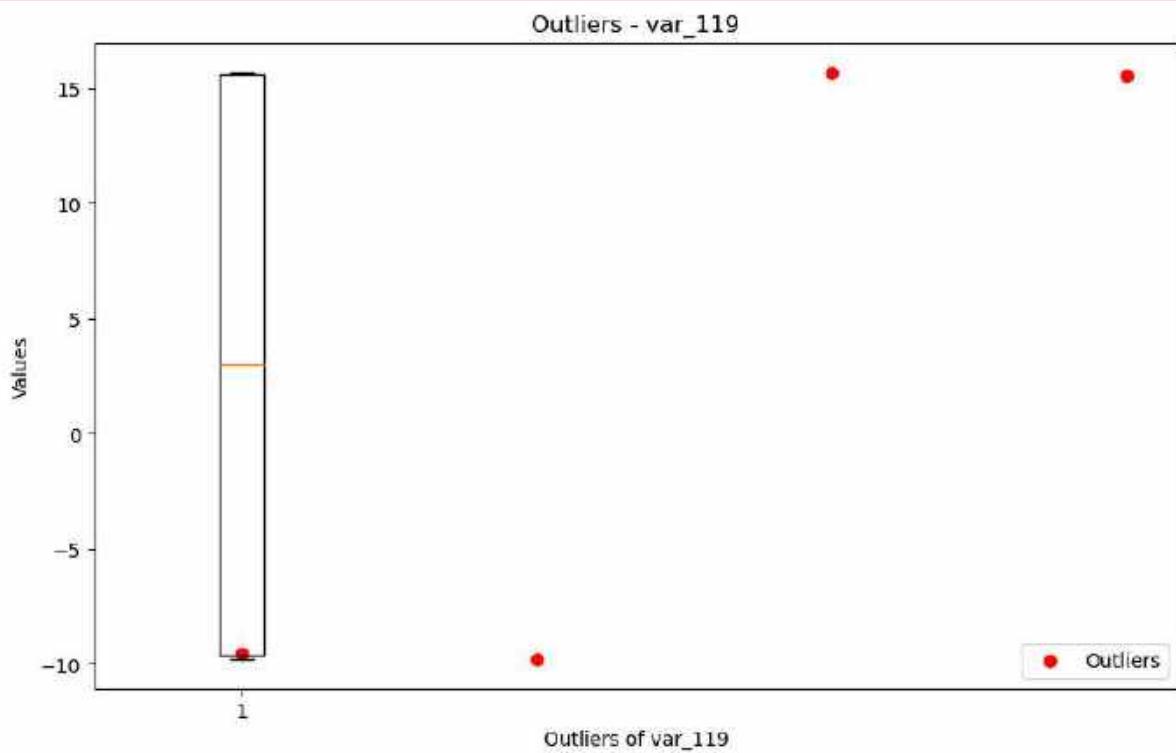
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





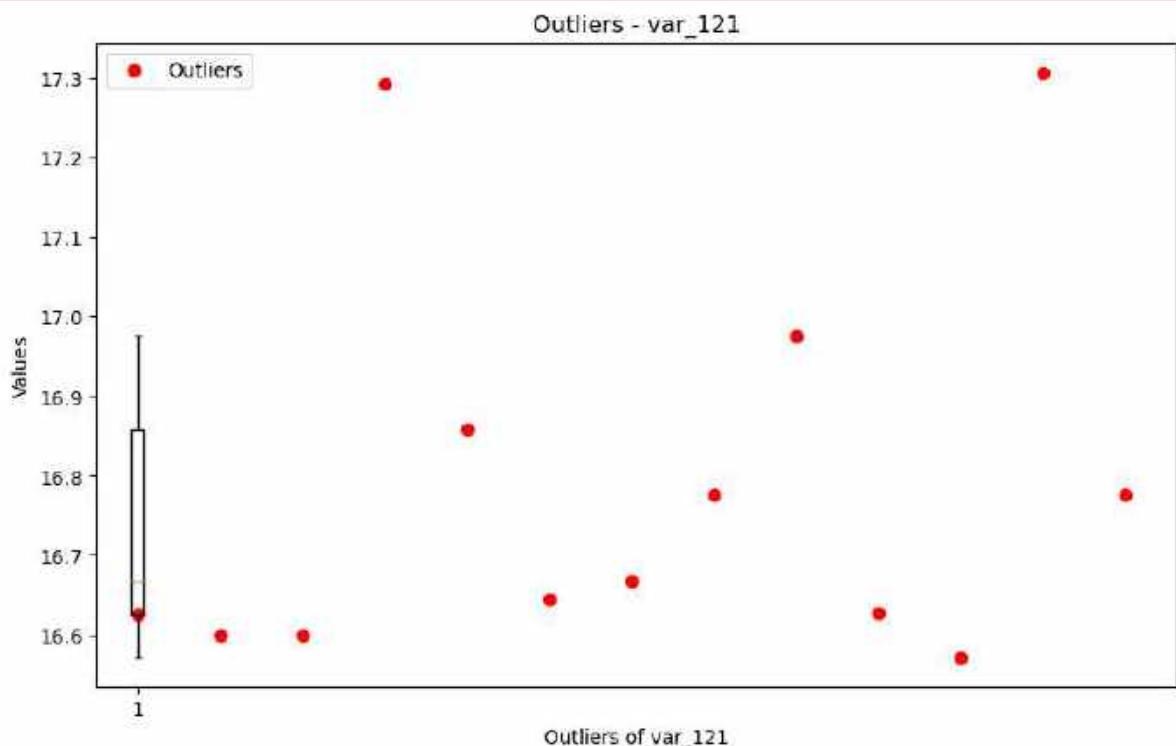
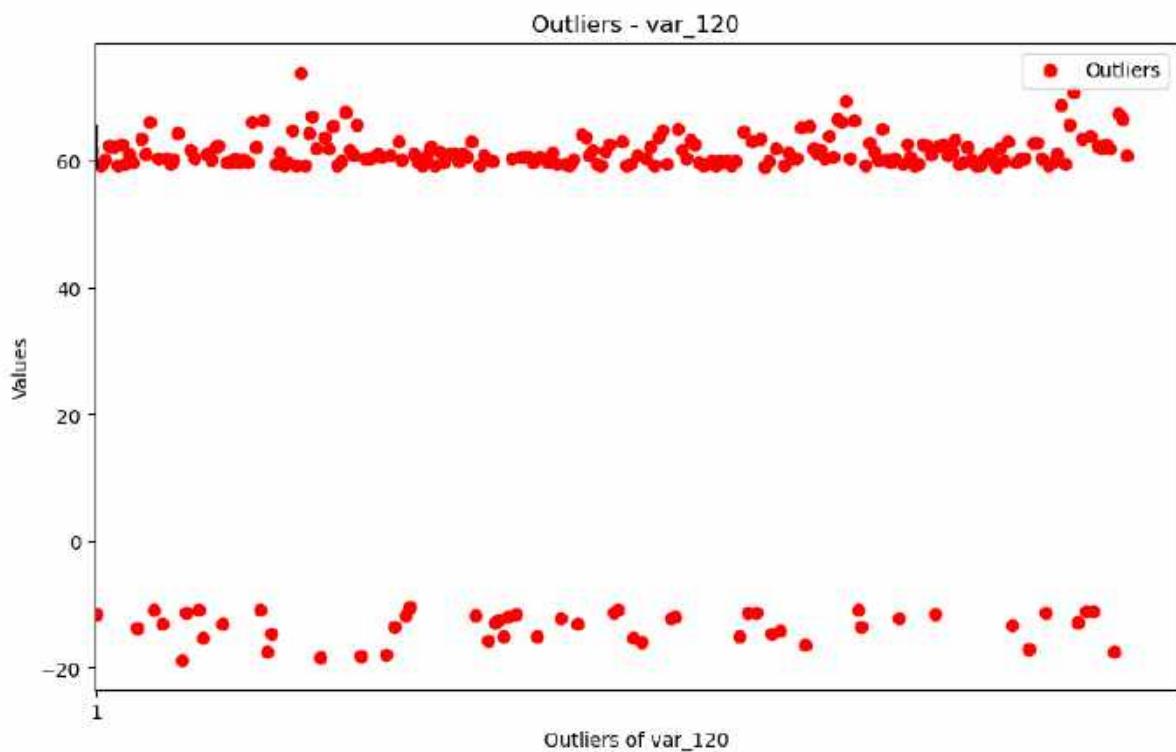
Total outliers in column var\_118: 228

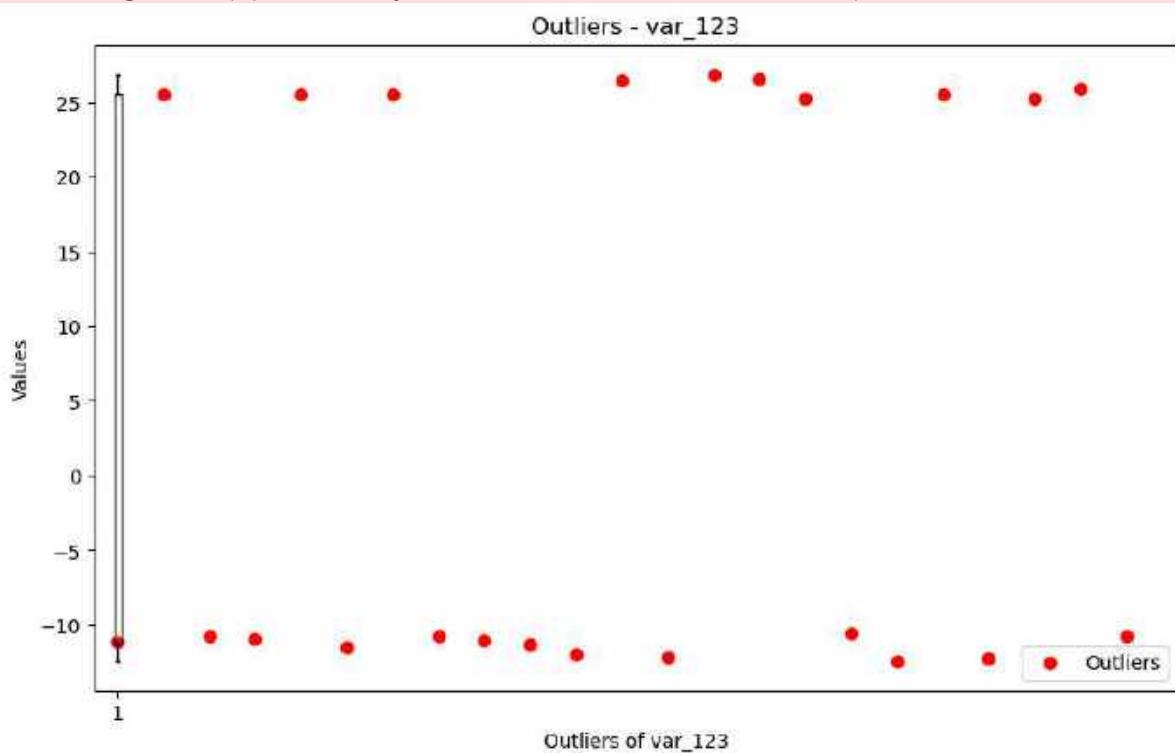
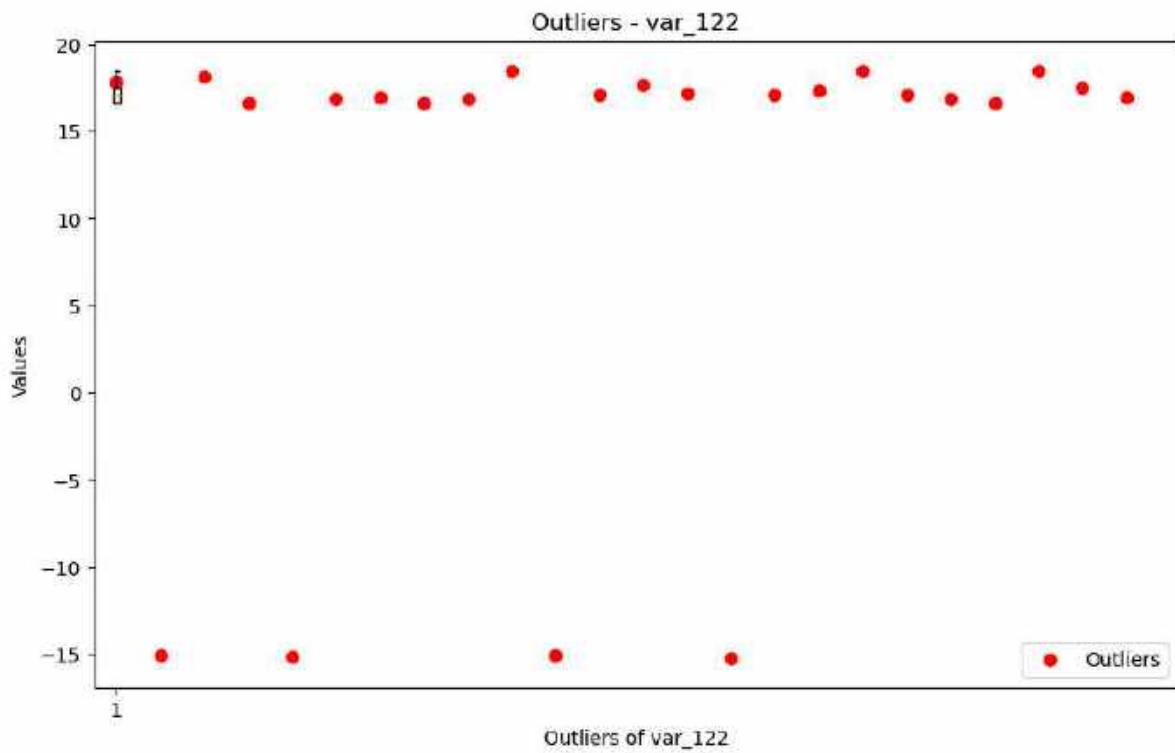
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



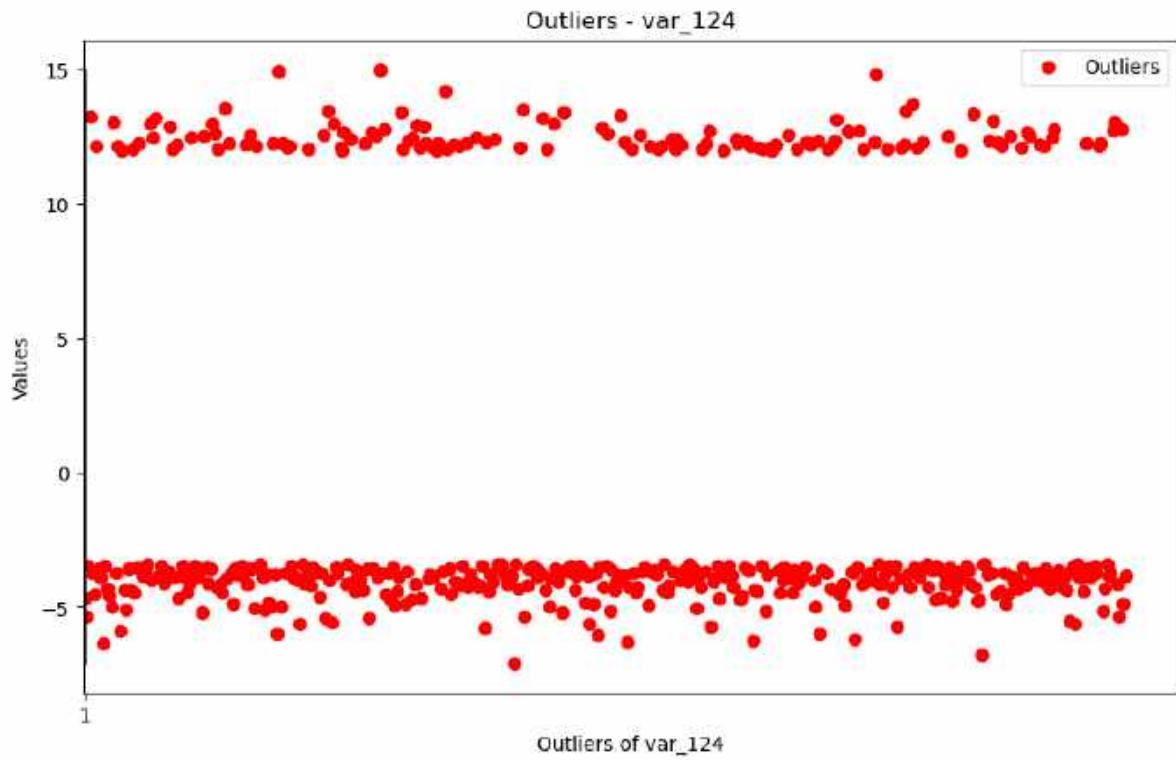
Total outliers in column var\_119: 4

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



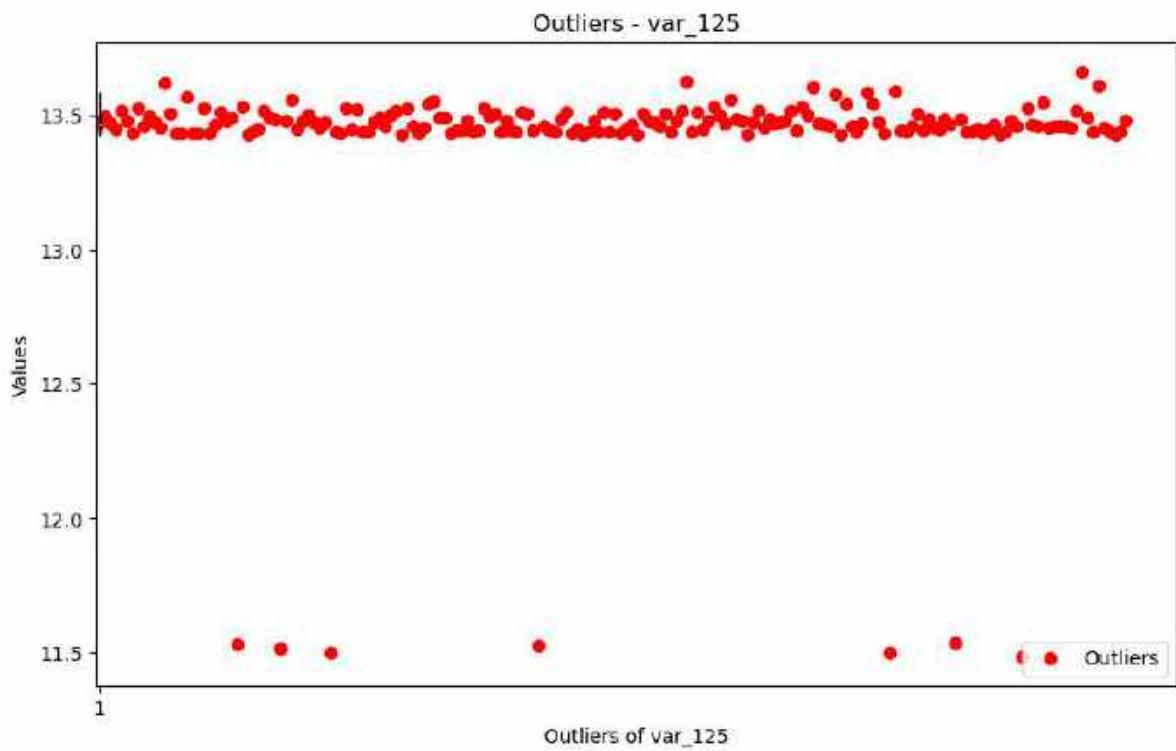


Total outliers in column var\_123: 23



Total outliers in column var\_124: 498

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



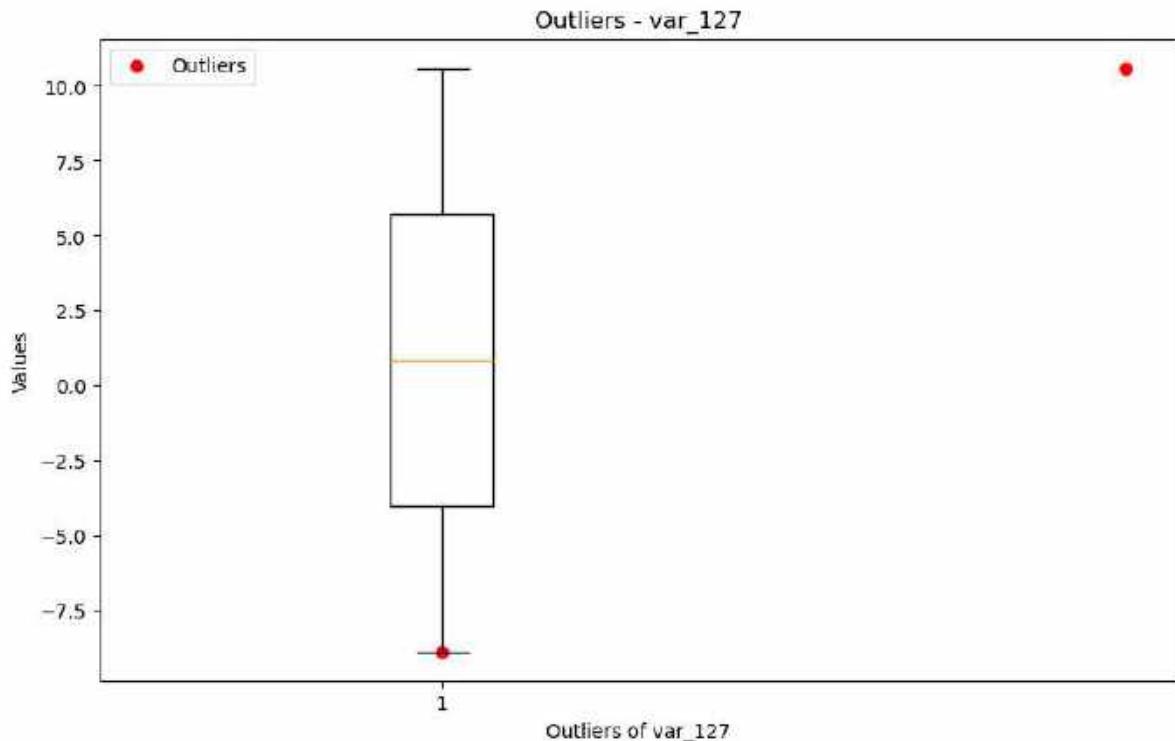
Total outliers in column var\_125: 188

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



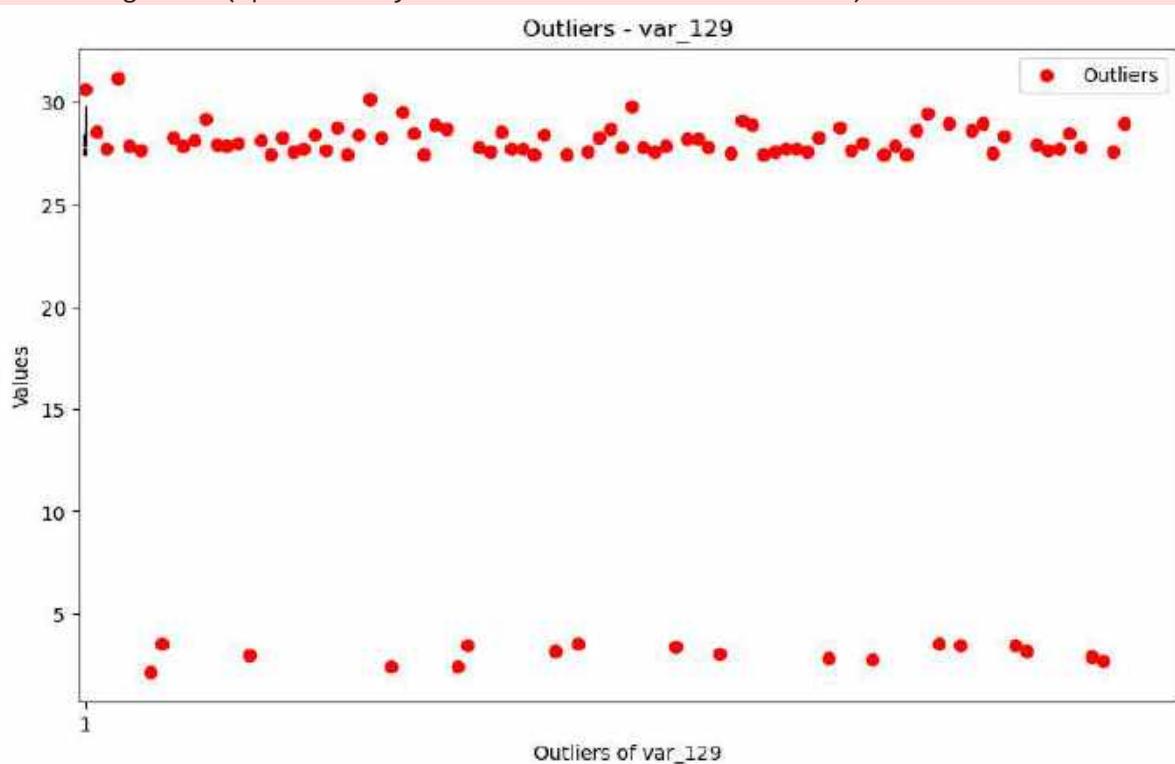
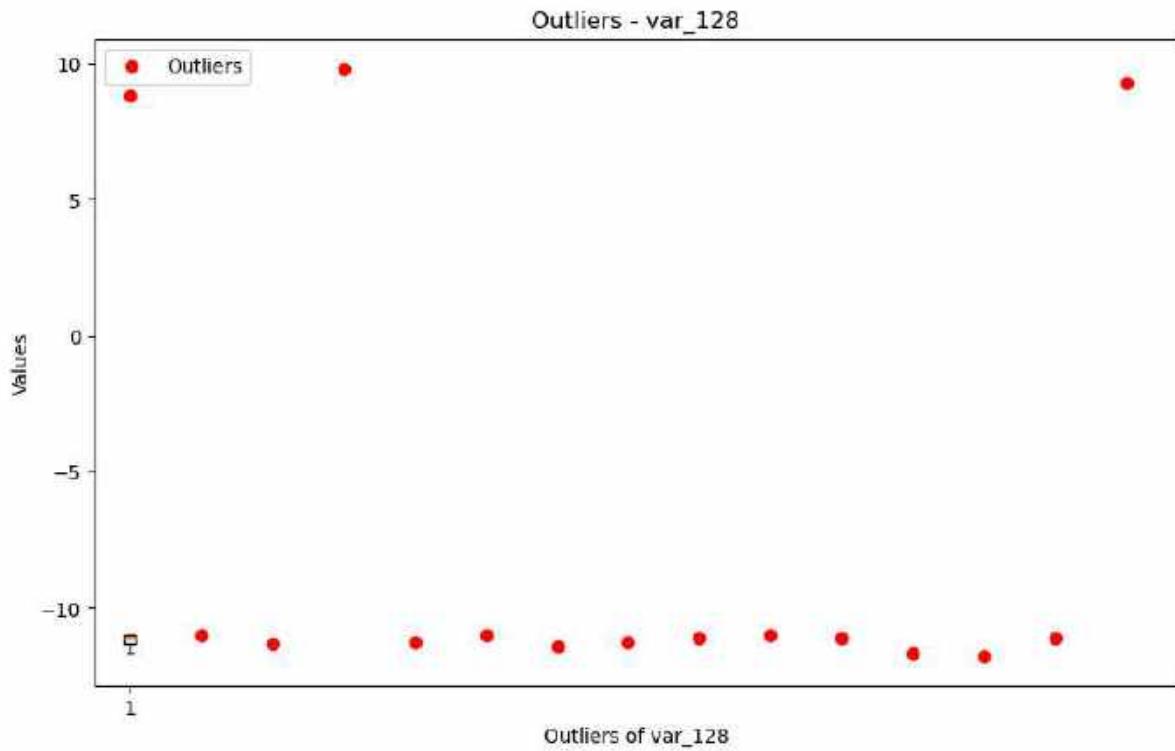
```
Total outliers in column var_126: 0
```

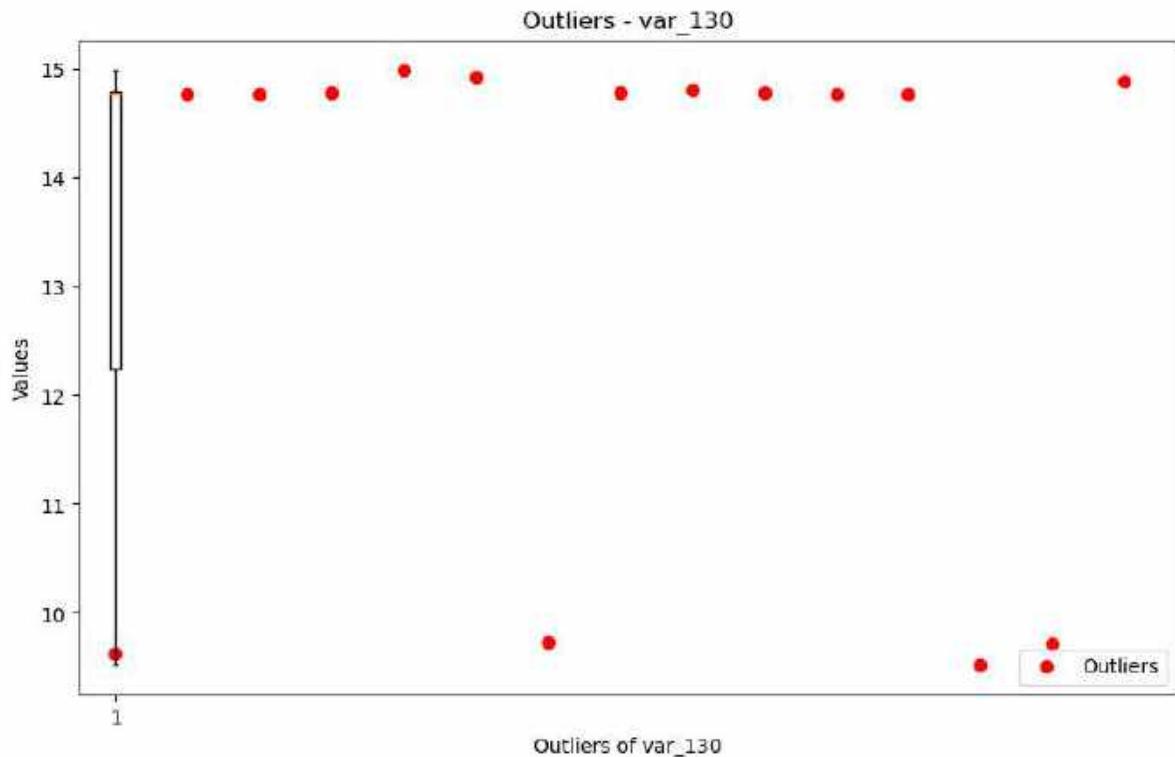
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



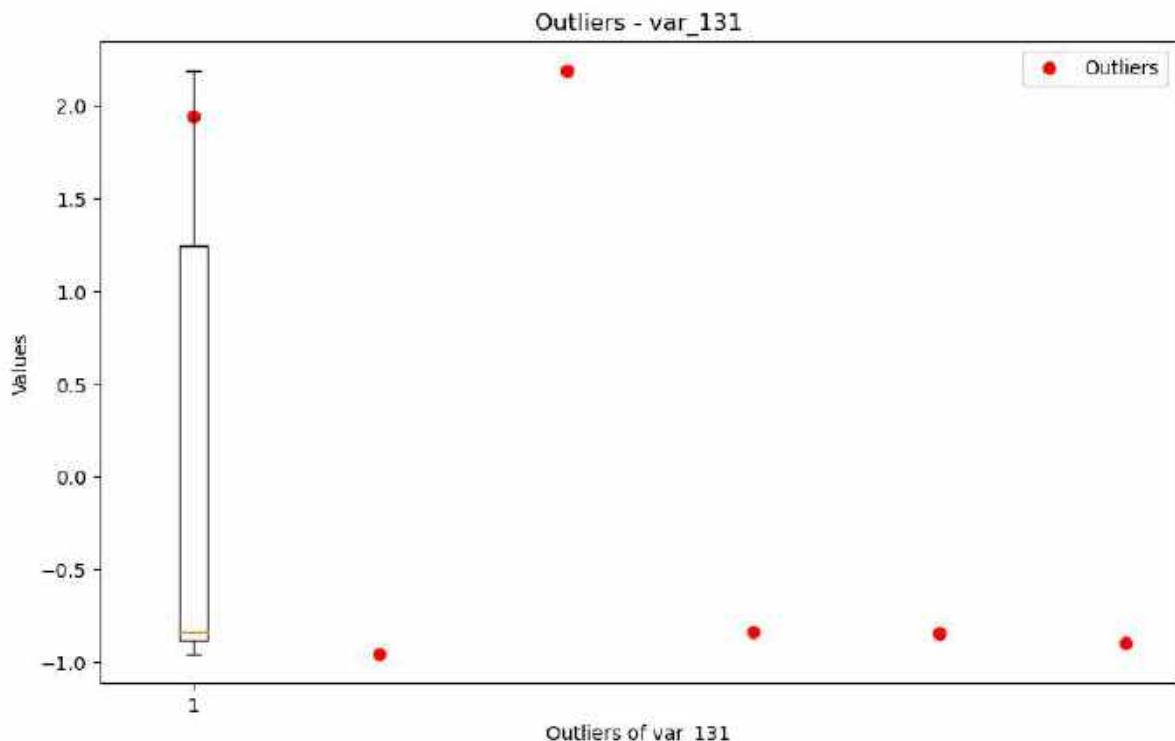
```
Total outliers in column var_127: 2
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

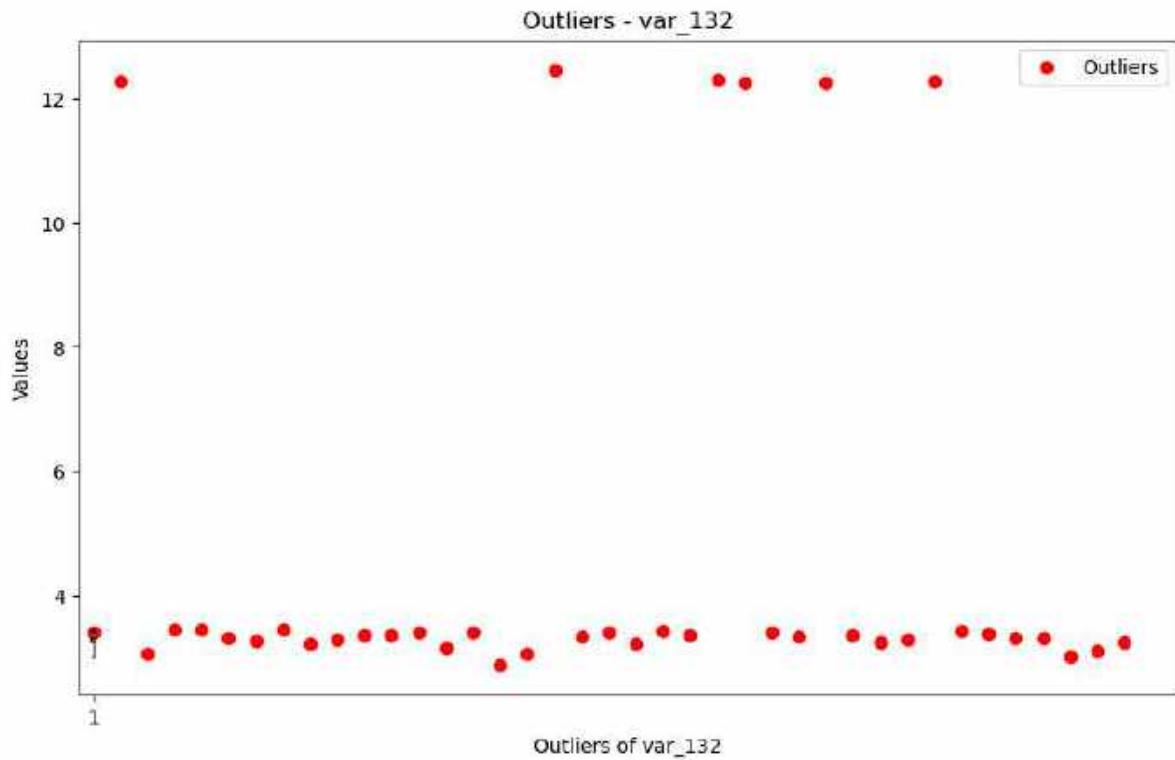




```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_130: 15
```

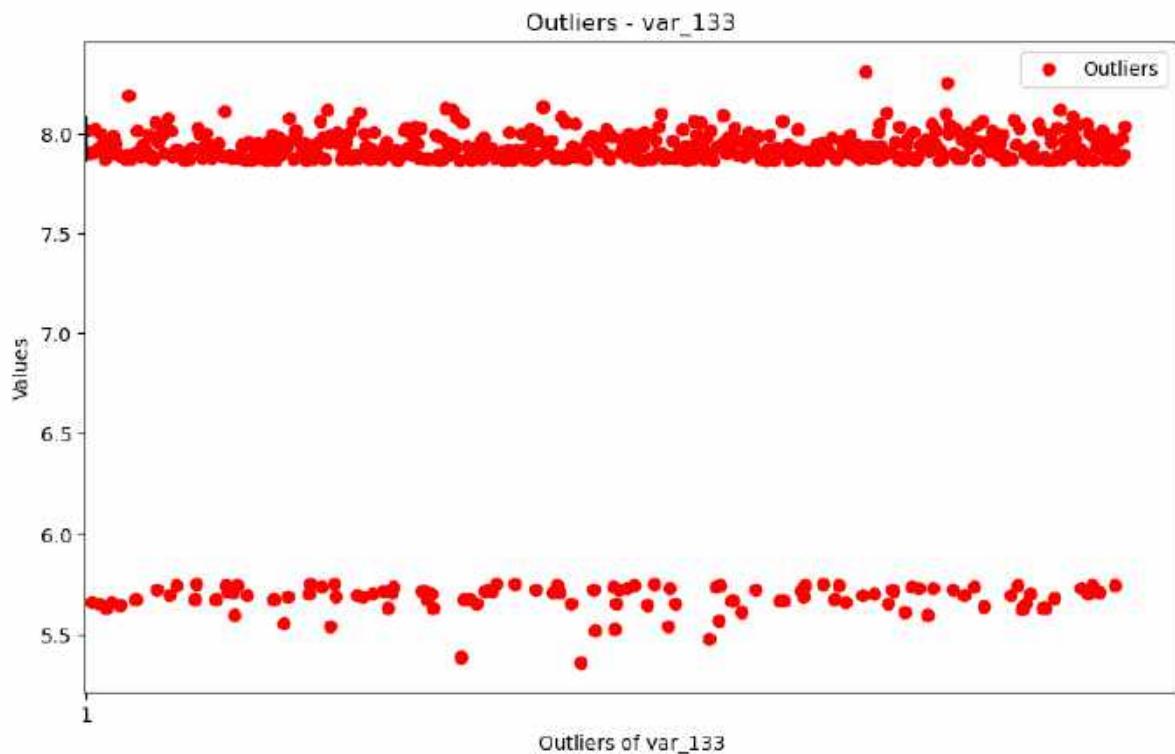


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_131: 6
```



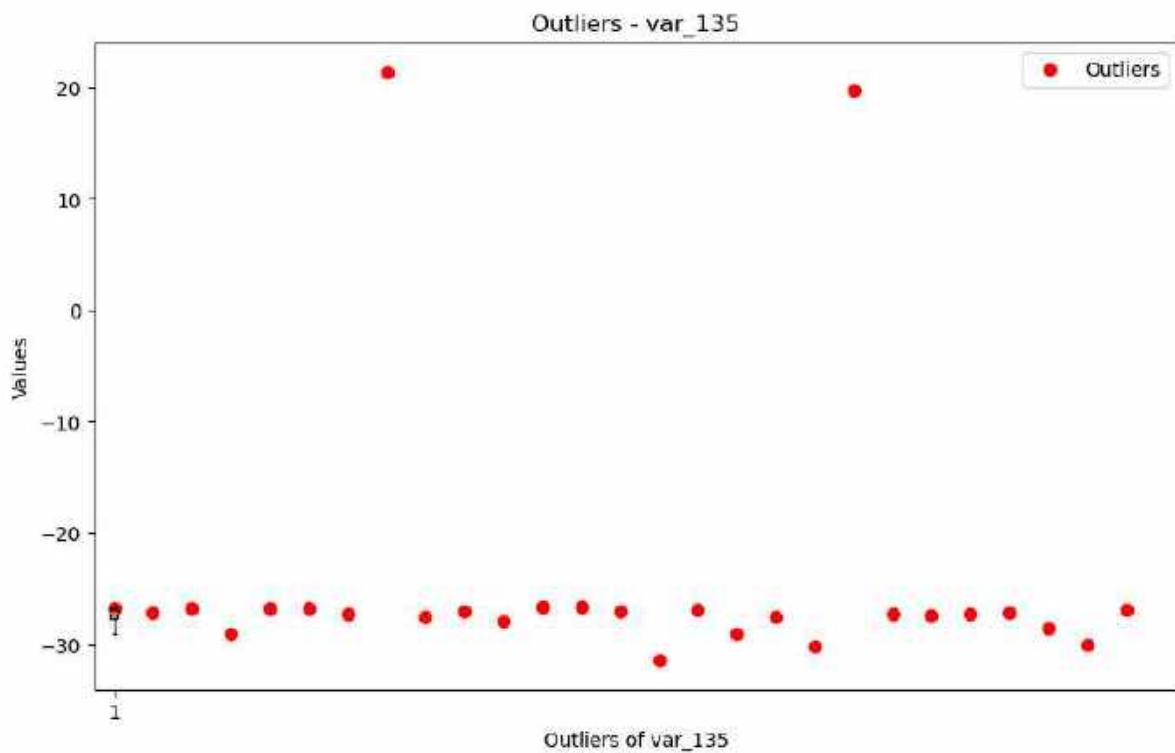
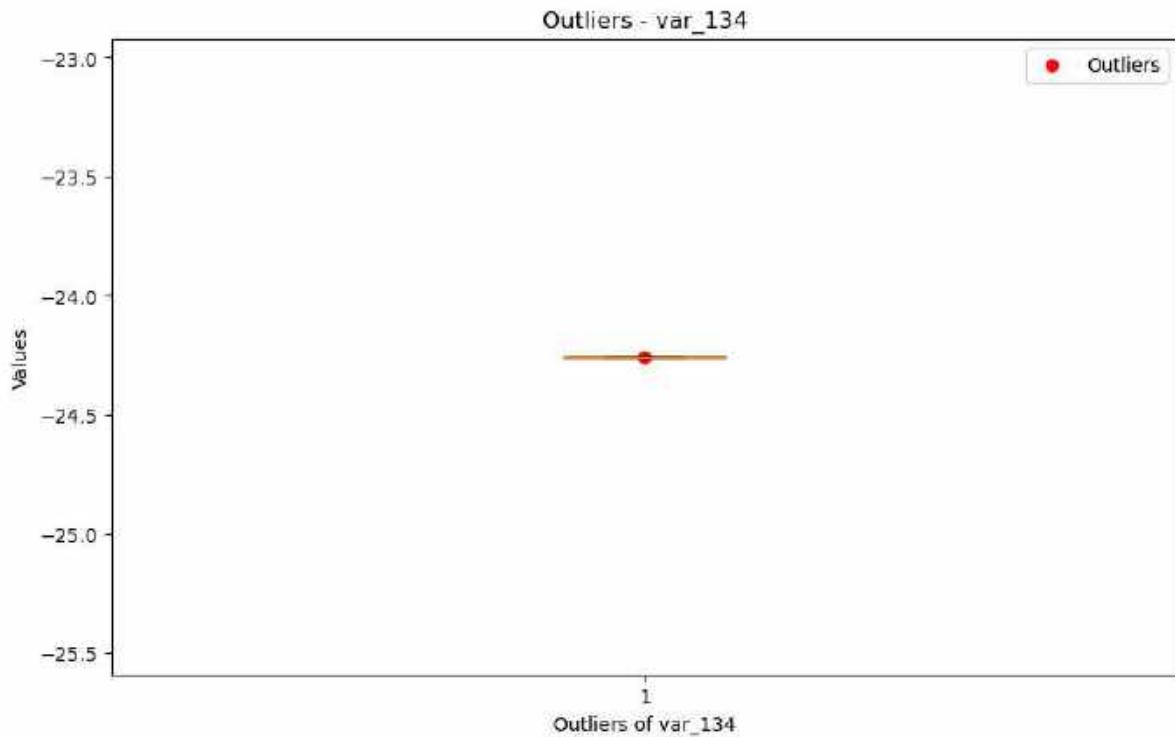
Total outliers in column var\_132: 39

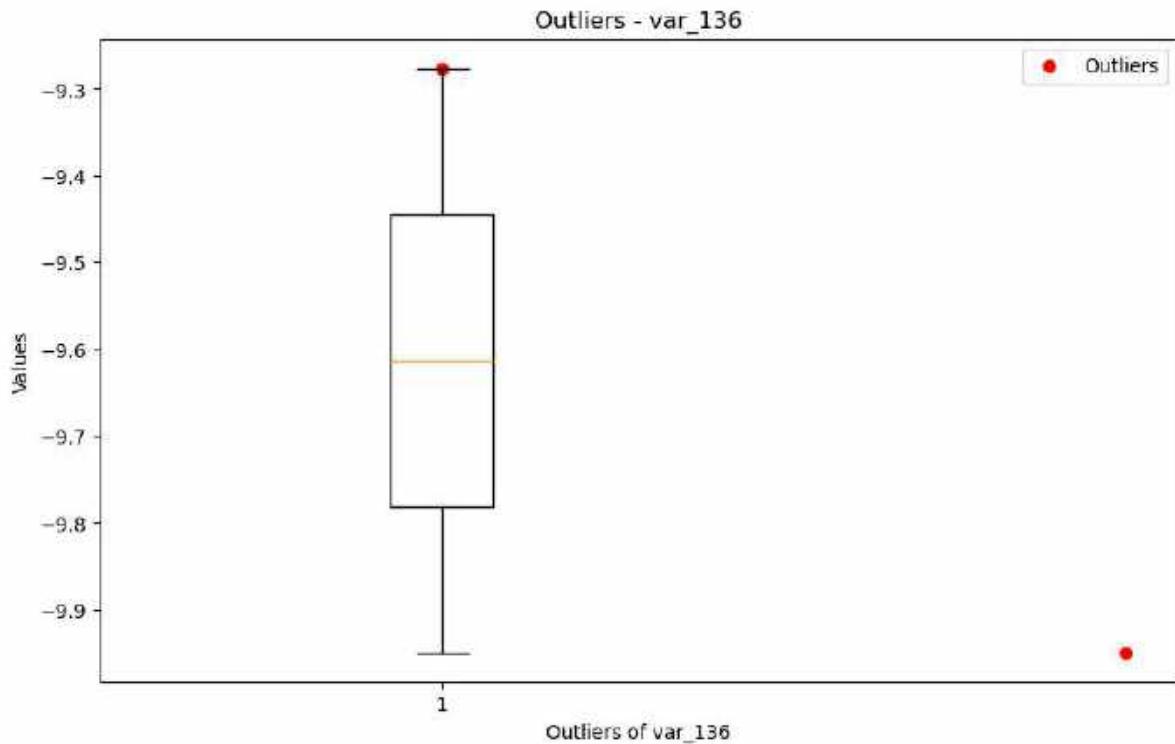
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_133: 589

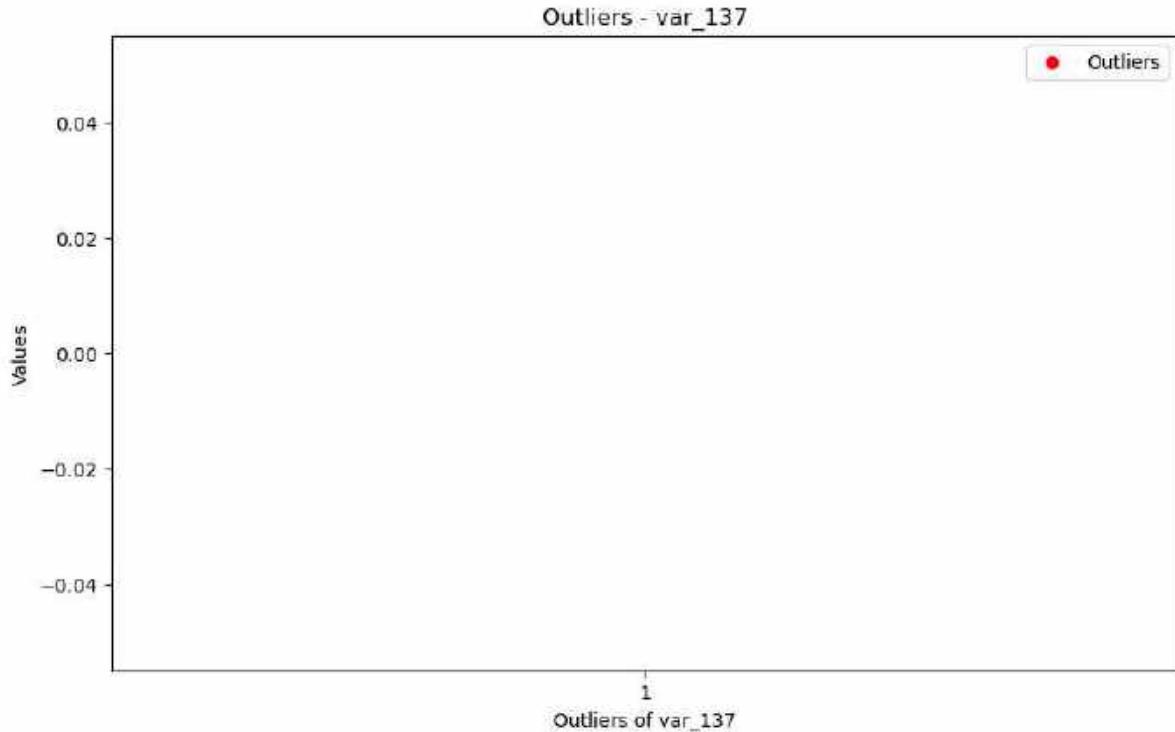
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





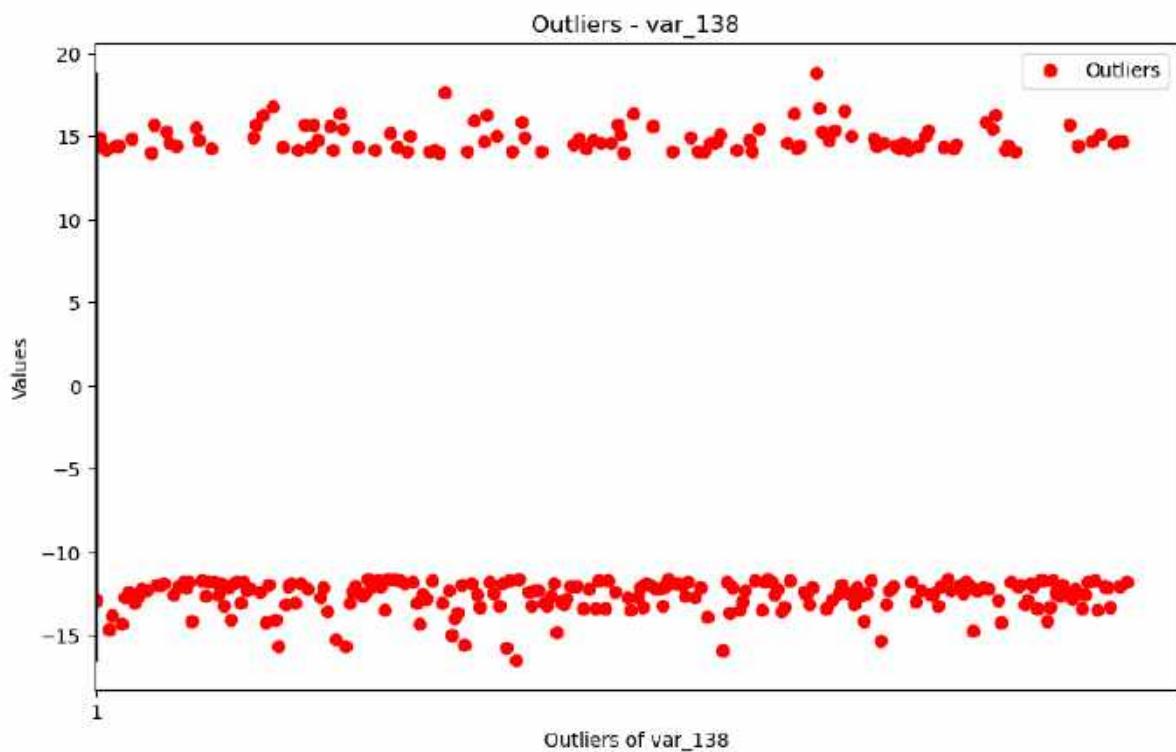
Total outliers in column var\_136: 2

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



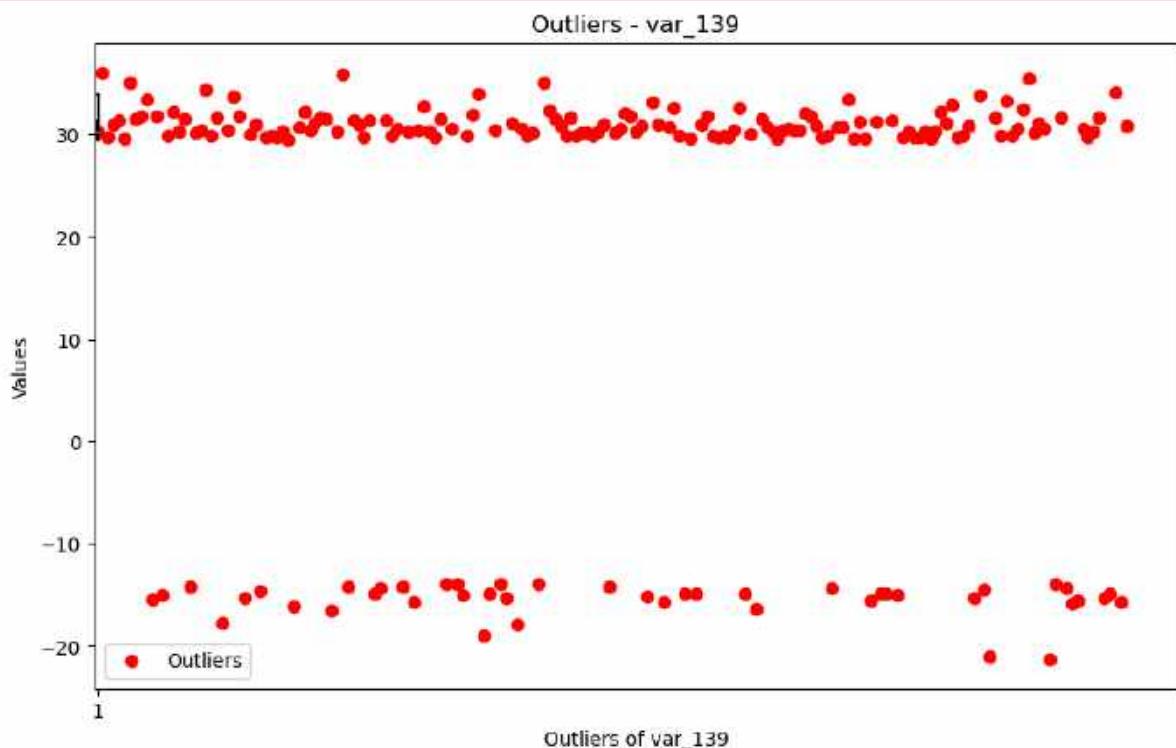
Total outliers in column var\_137: 0

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



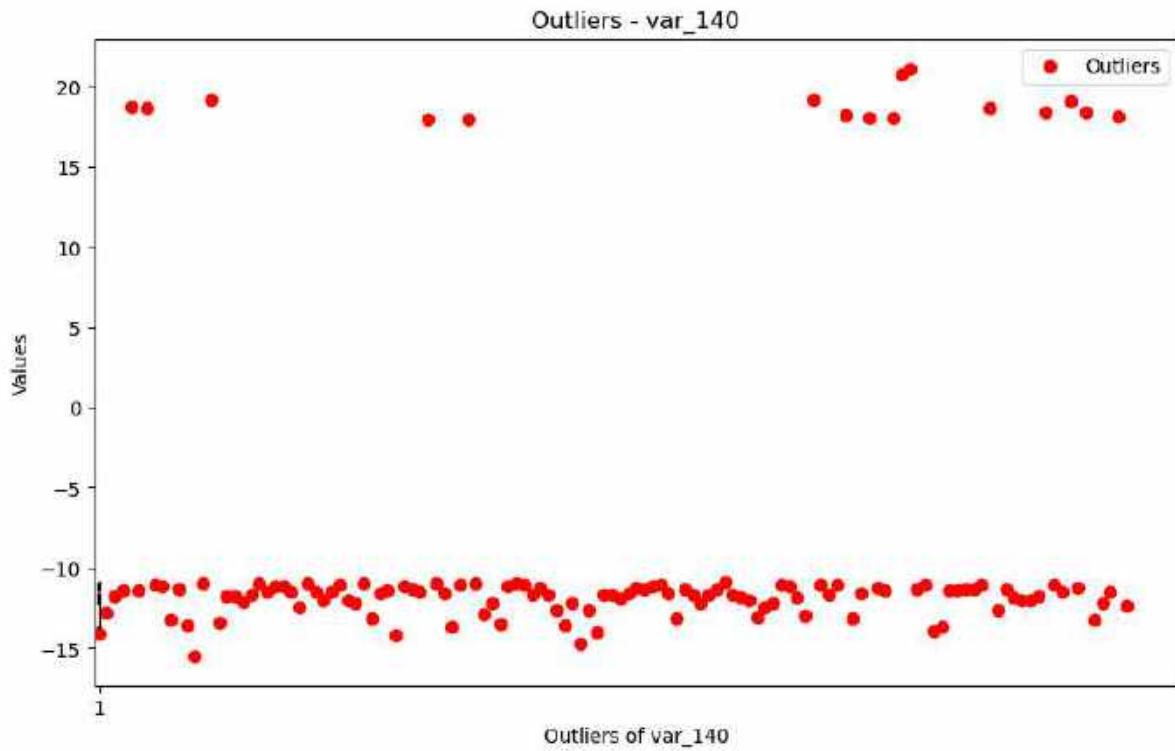
Total outliers in column var\_138: 323

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



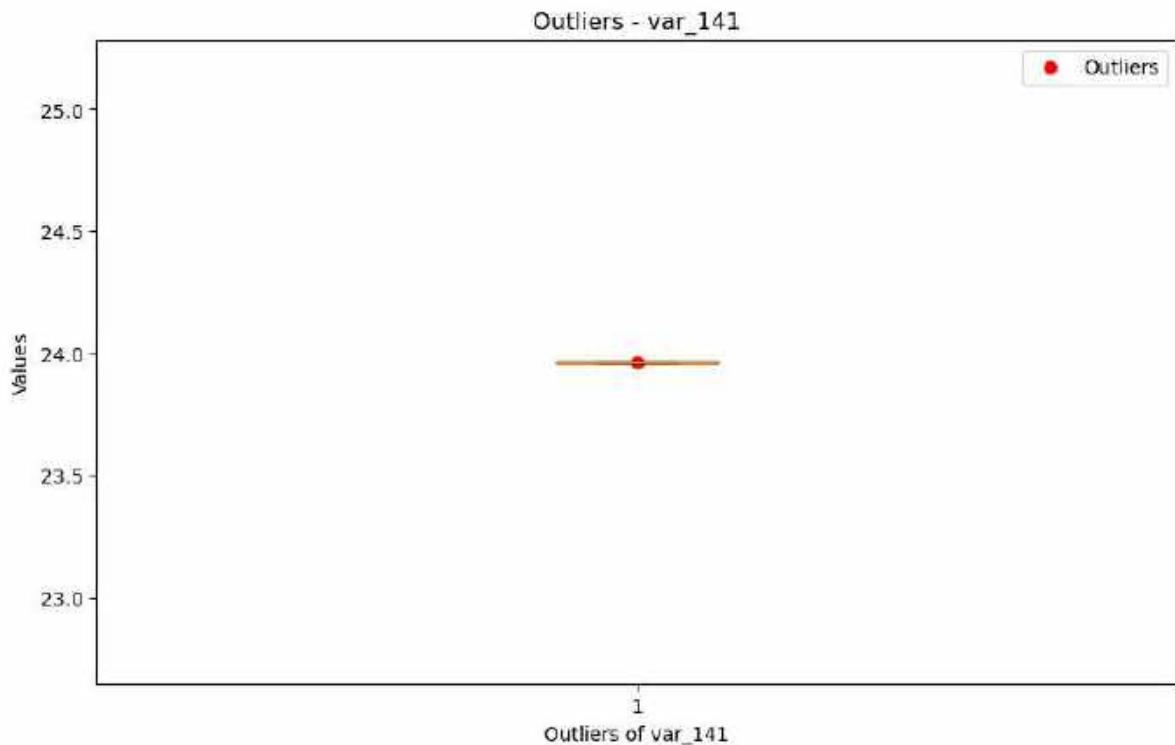
Total outliers in column var\_139: 190

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



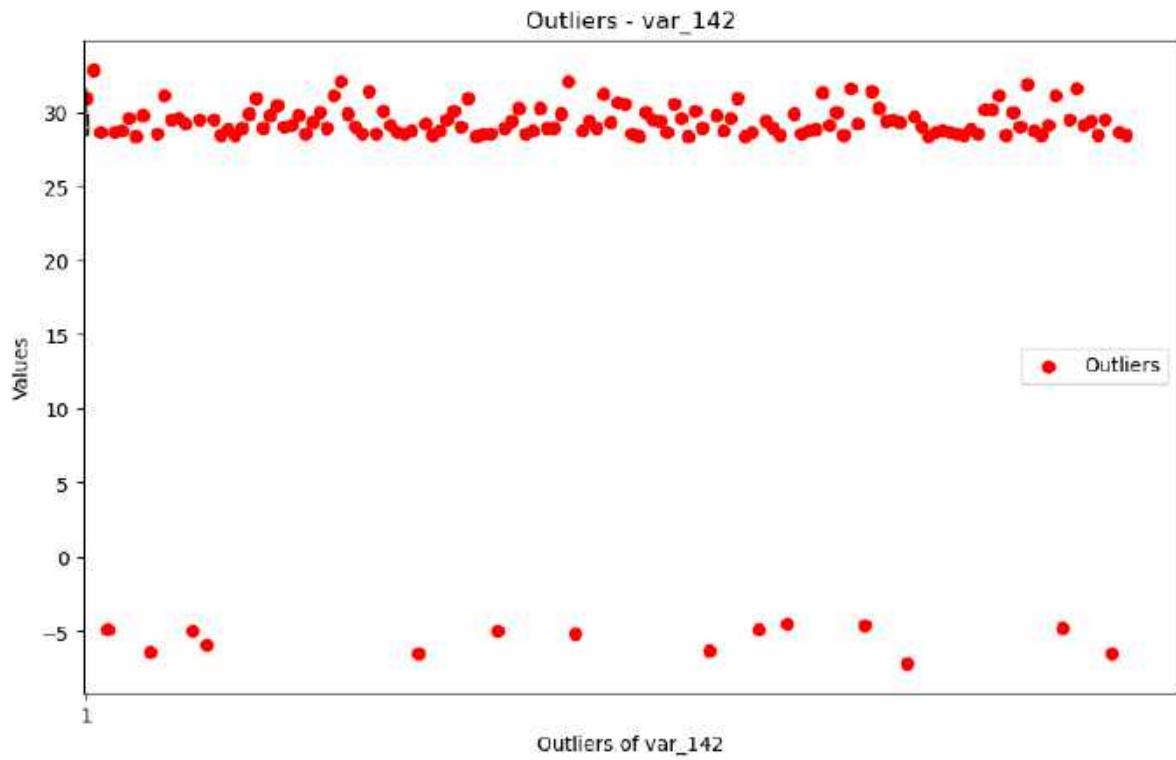
Total outliers in column var\_140: 129

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

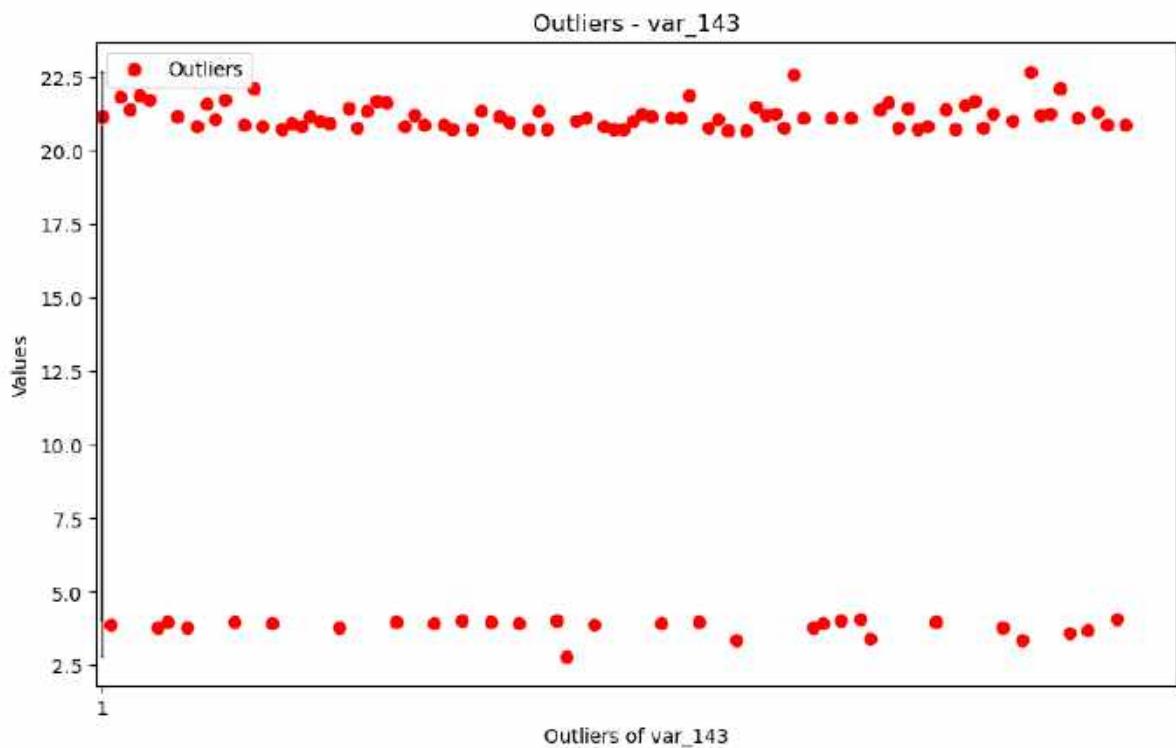


Total outliers in column var\_141: 1

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

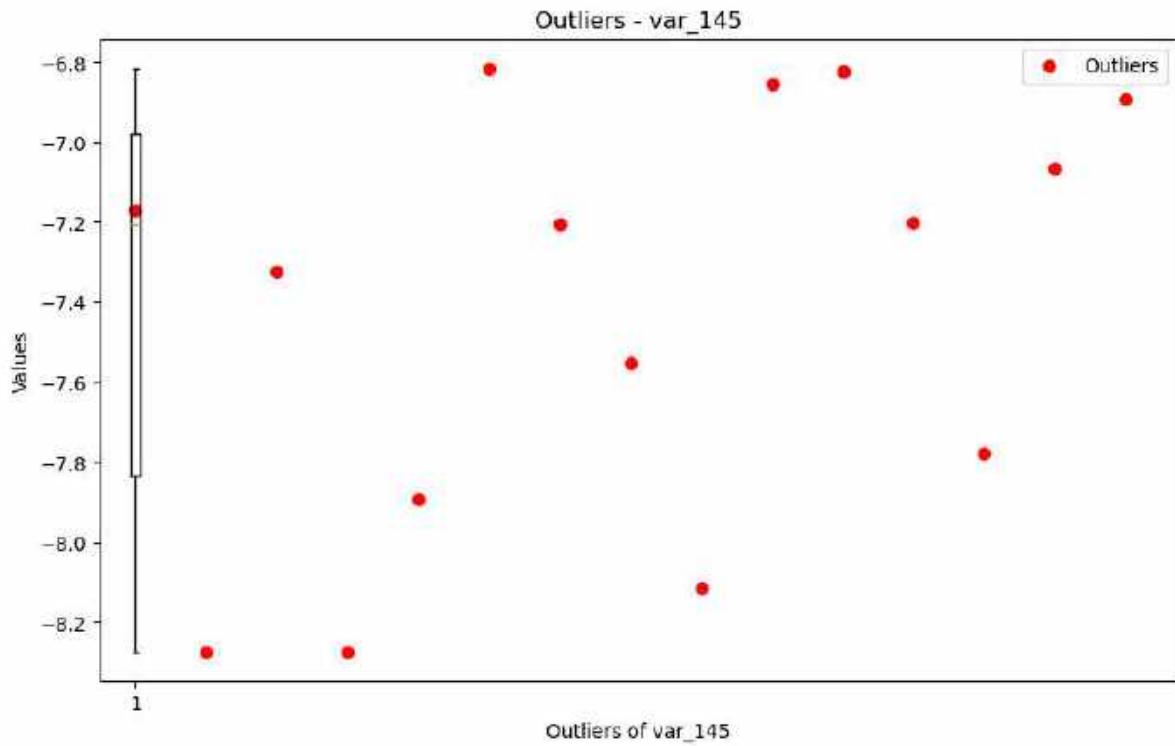
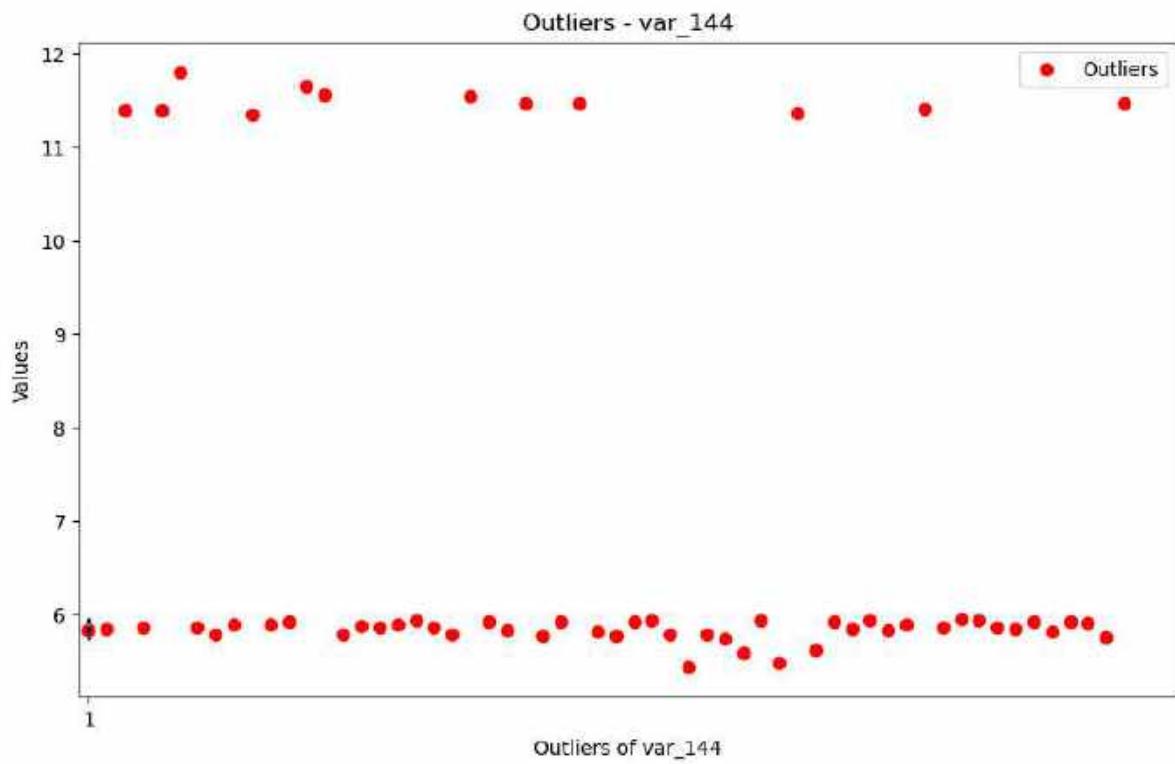


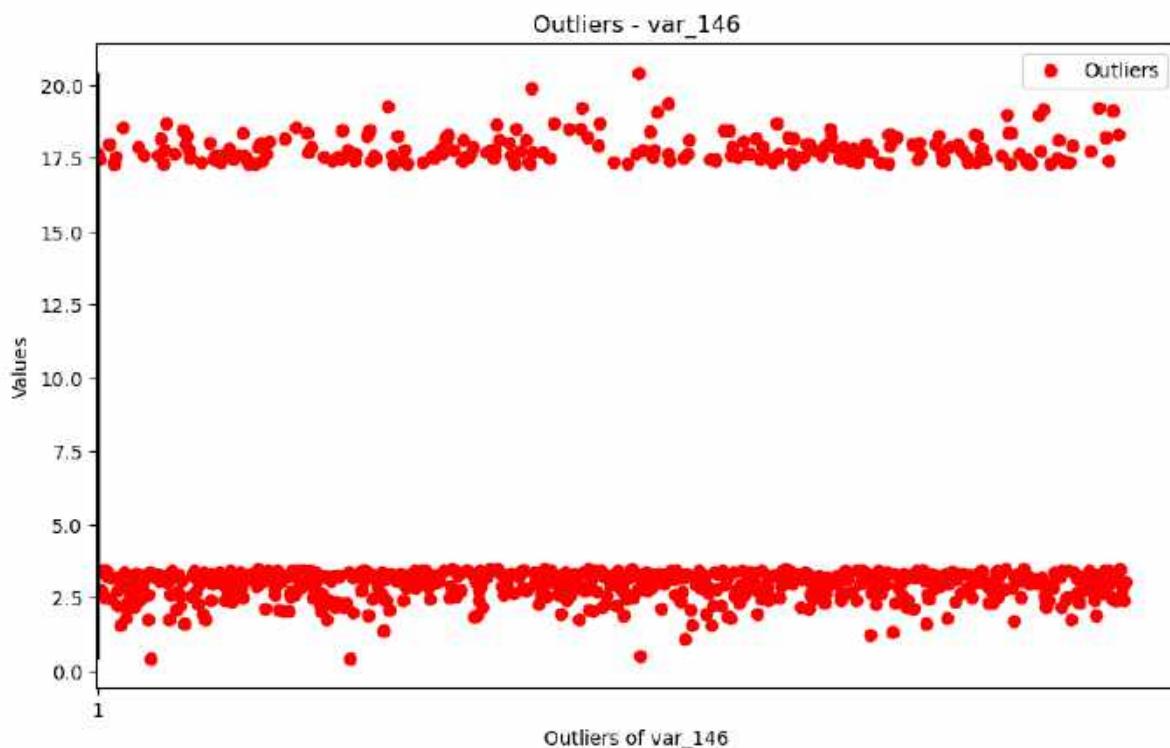
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_142: 148
```



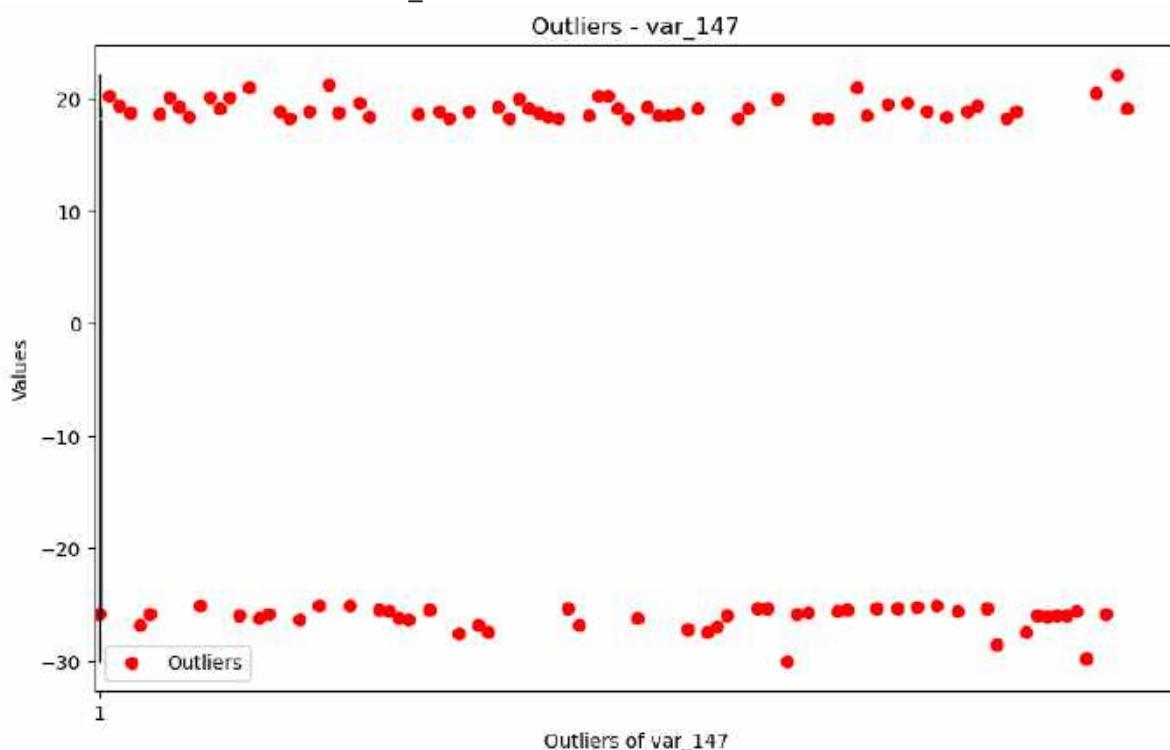
```
Total outliers in column var_143: 109
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



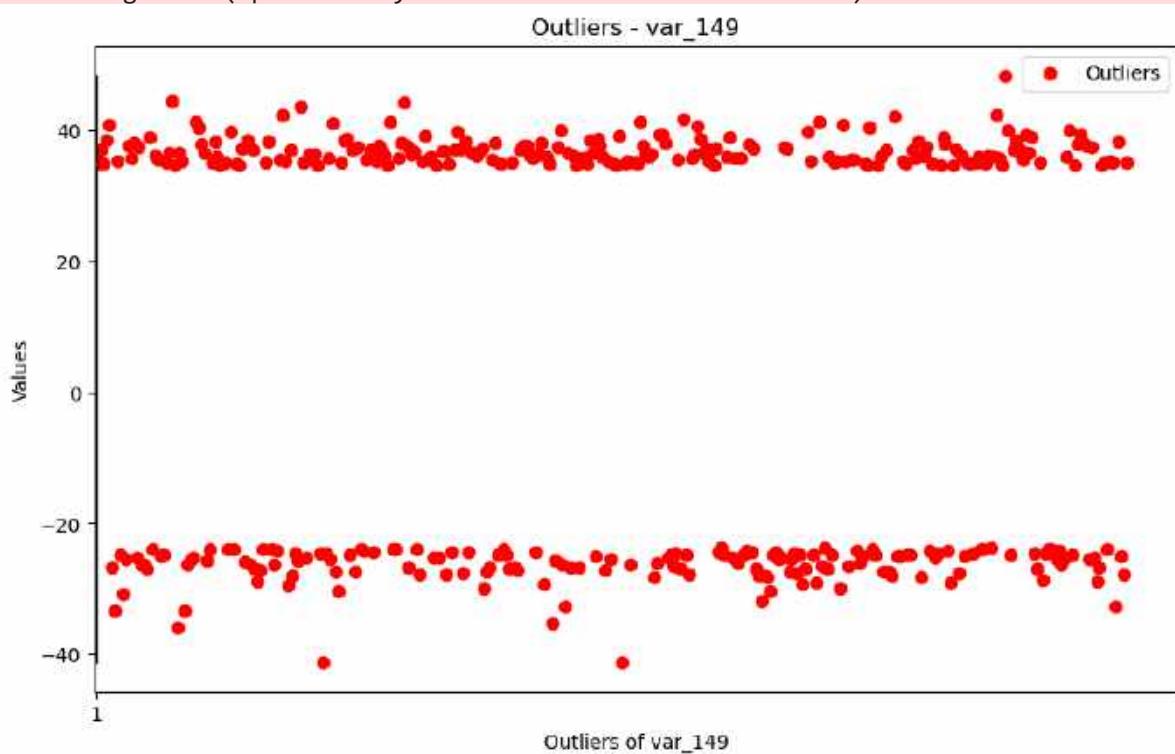
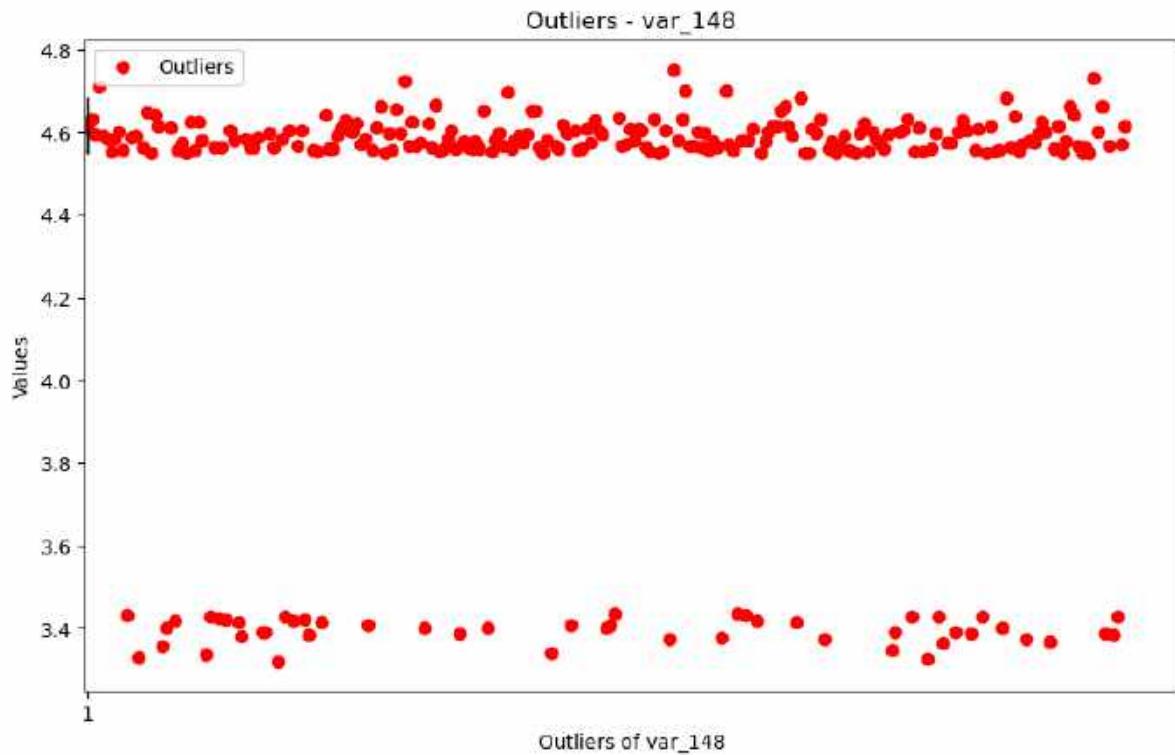


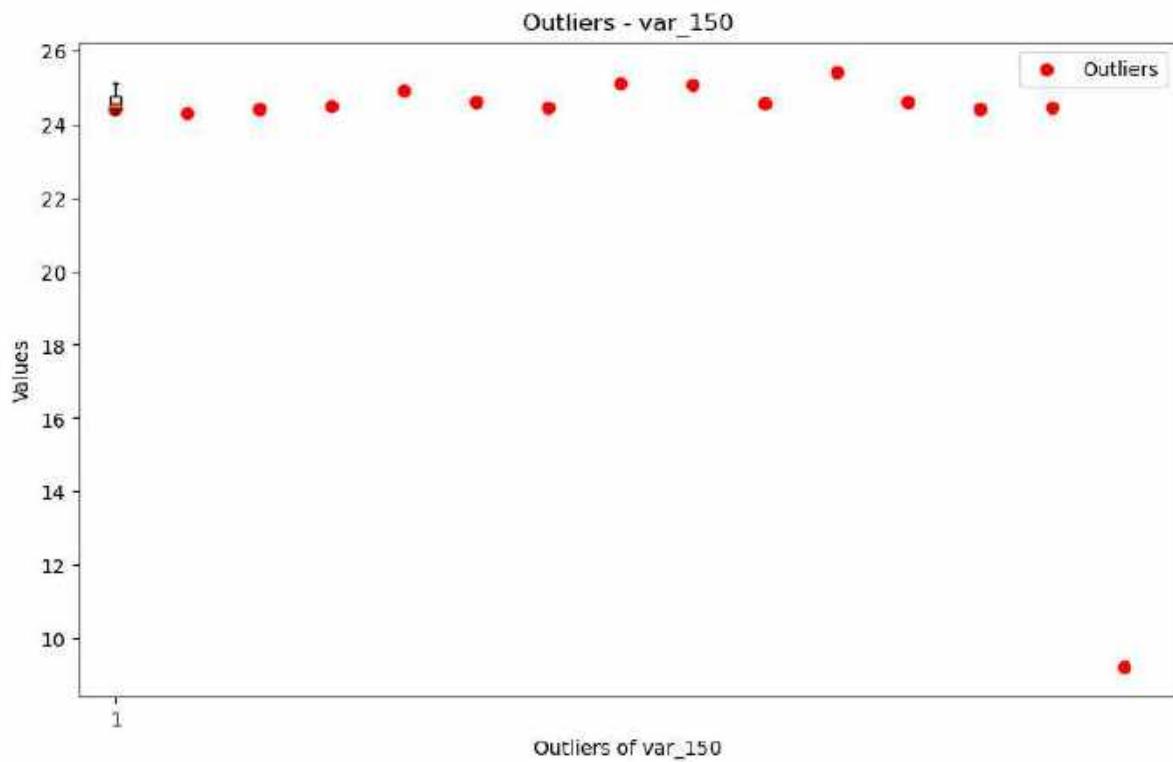
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_146: 804
```



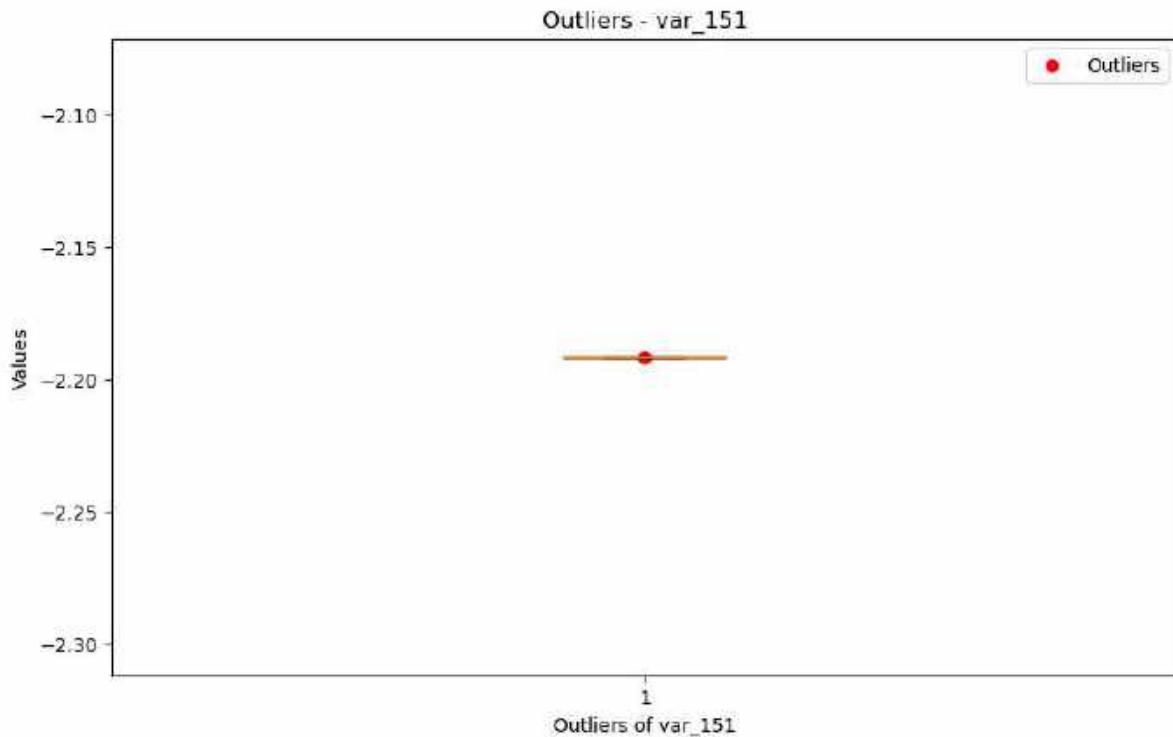
```
Total outliers in column var_147: 104
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

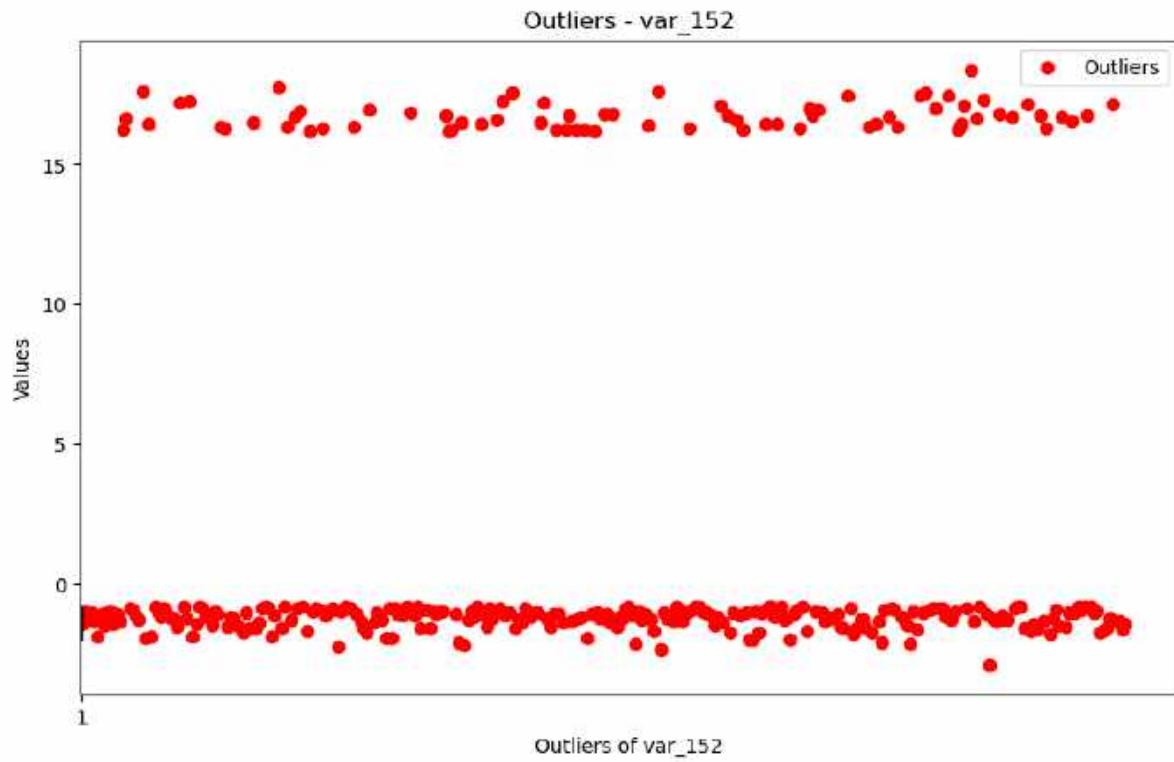




```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_150: 15
```

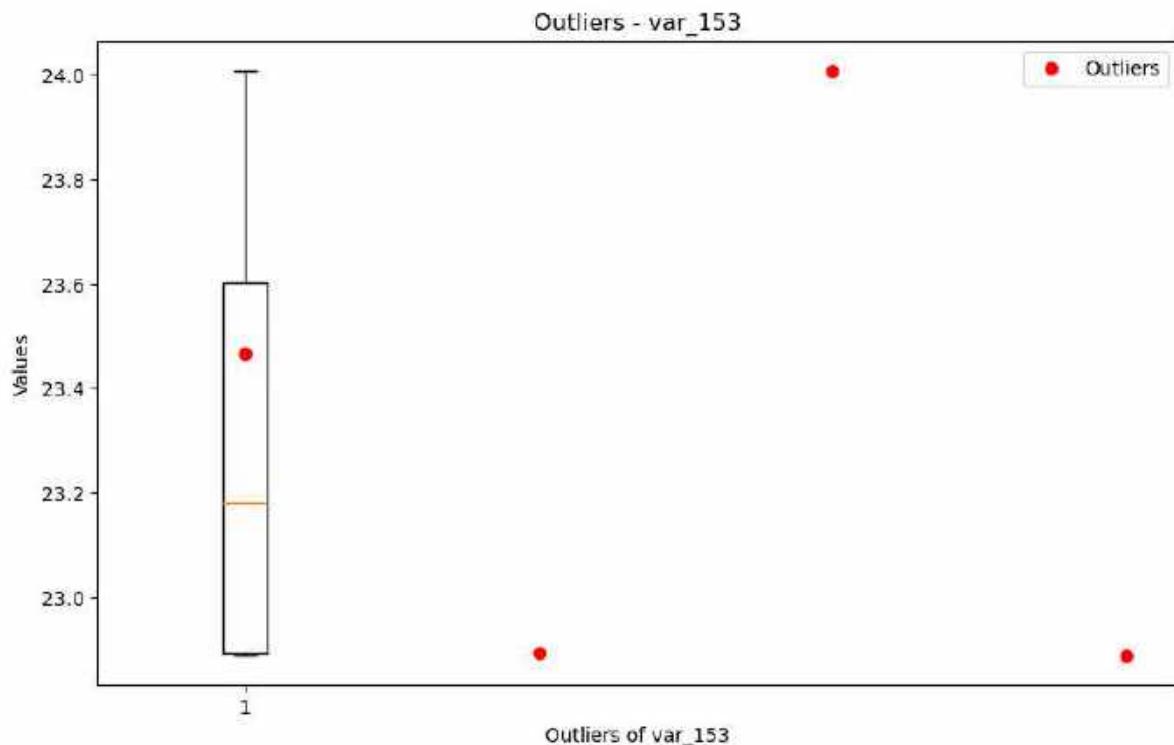


```
Total outliers in column var_151: 1  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



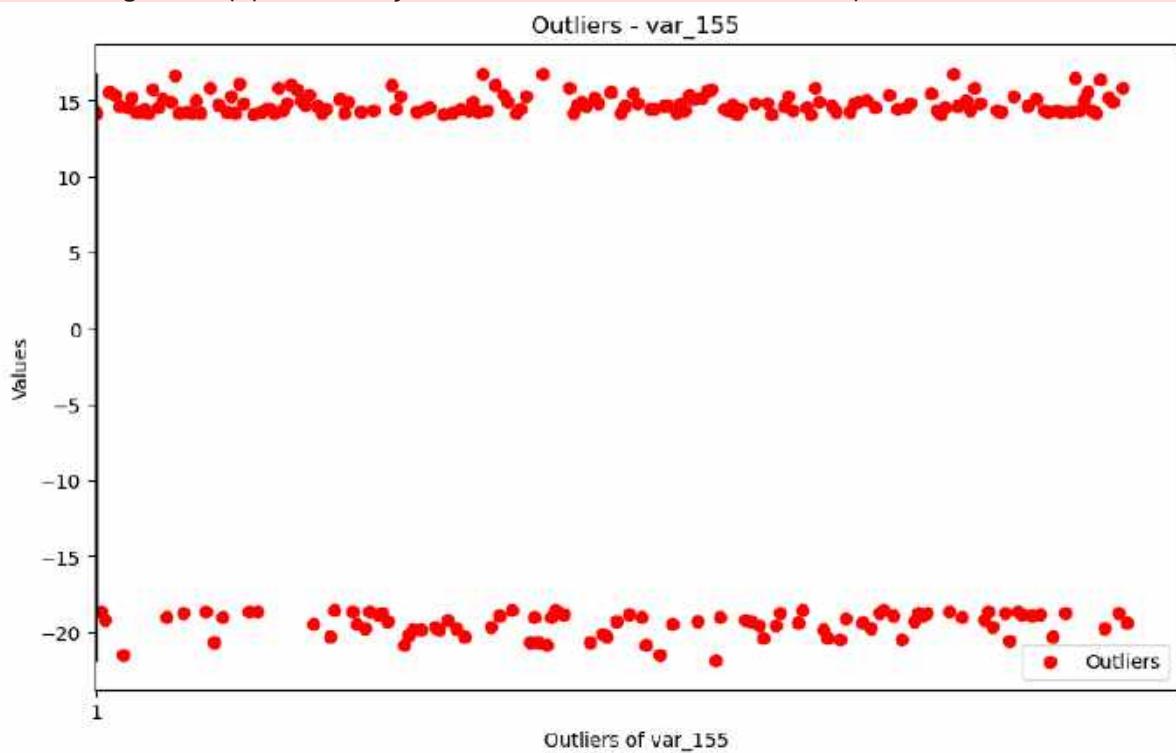
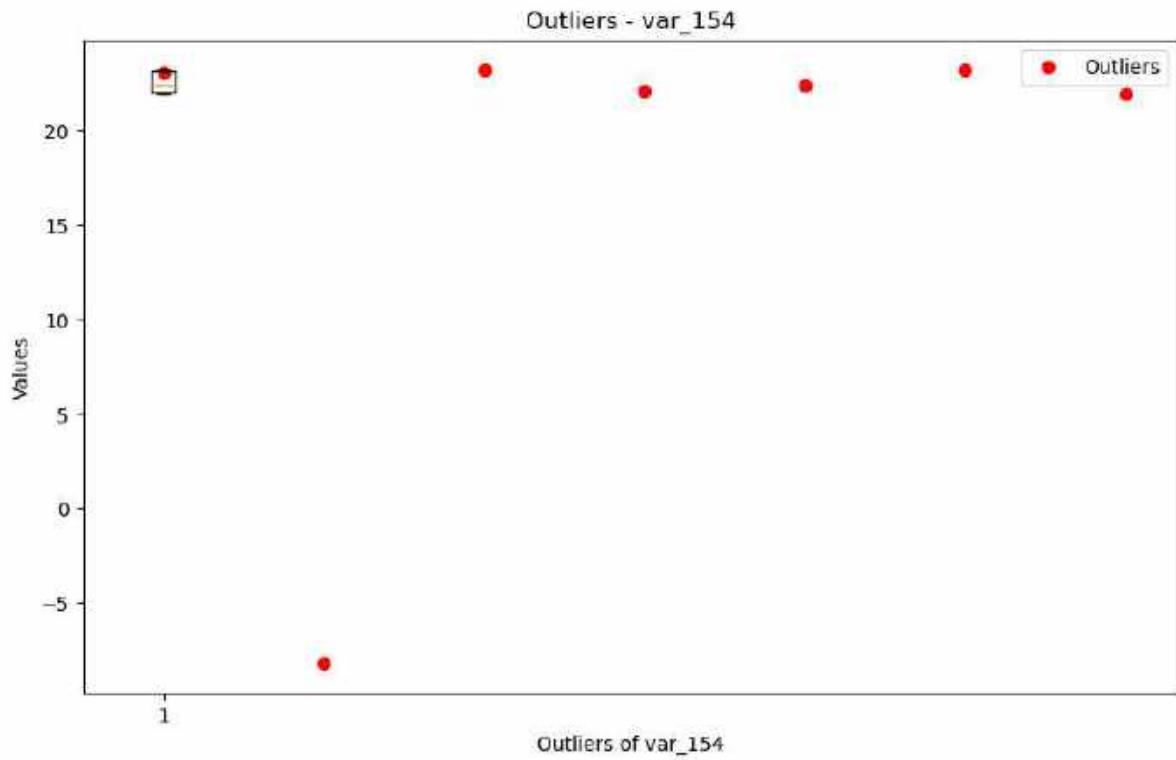
Total outliers in column var\_152: 331

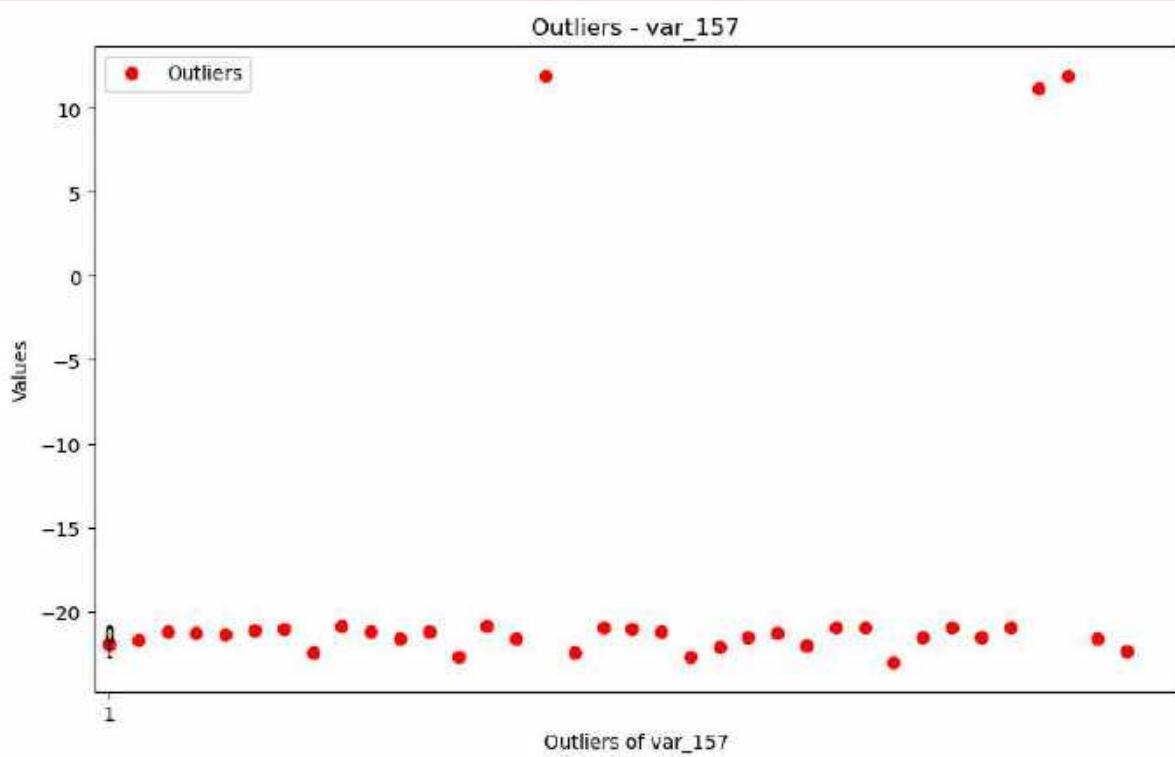
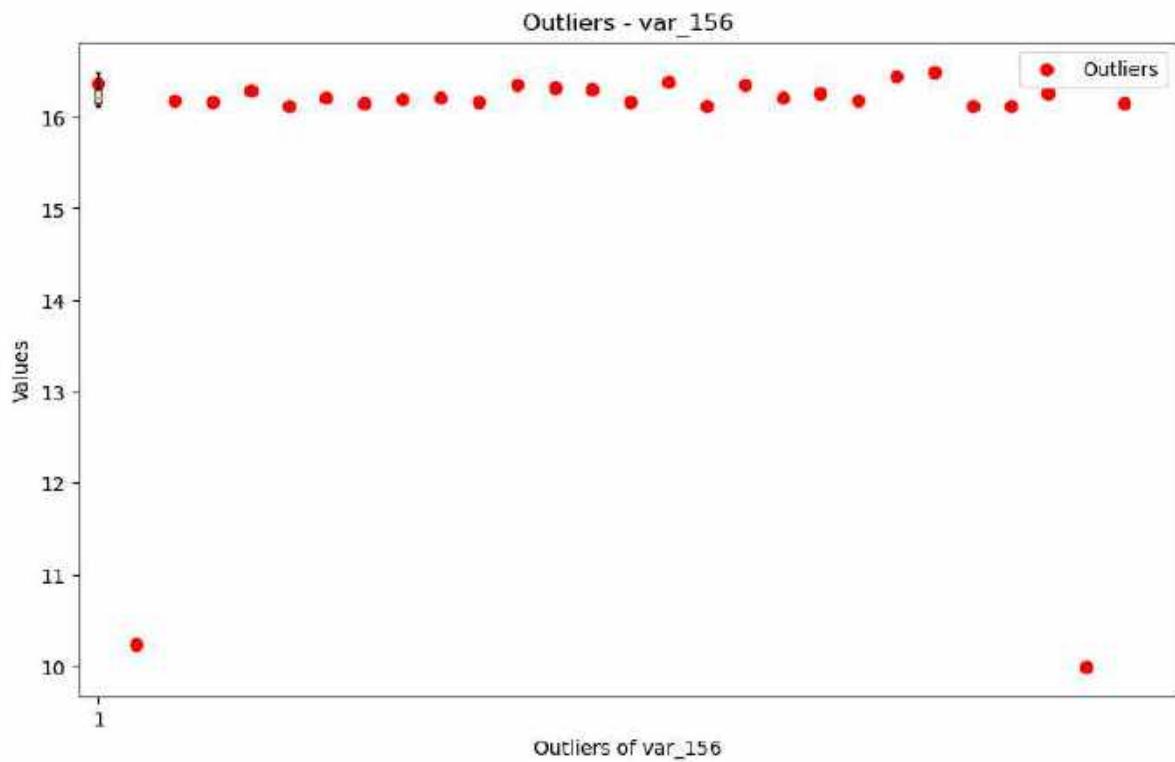
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

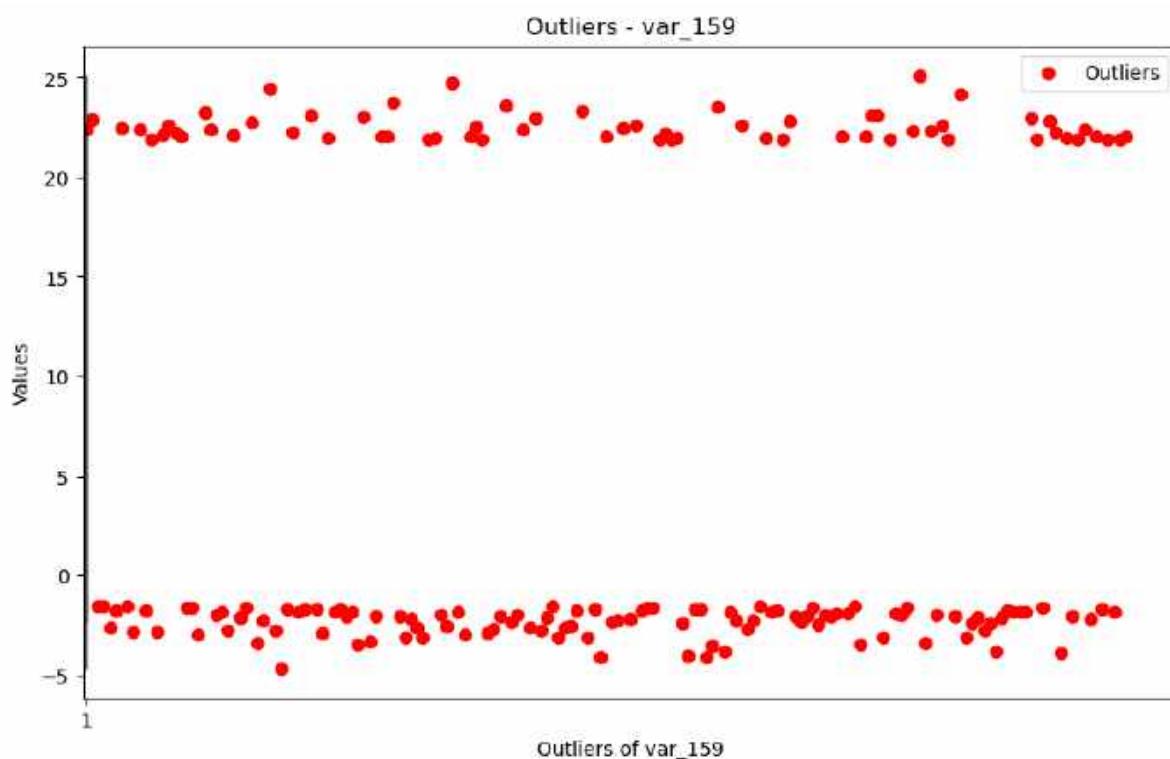
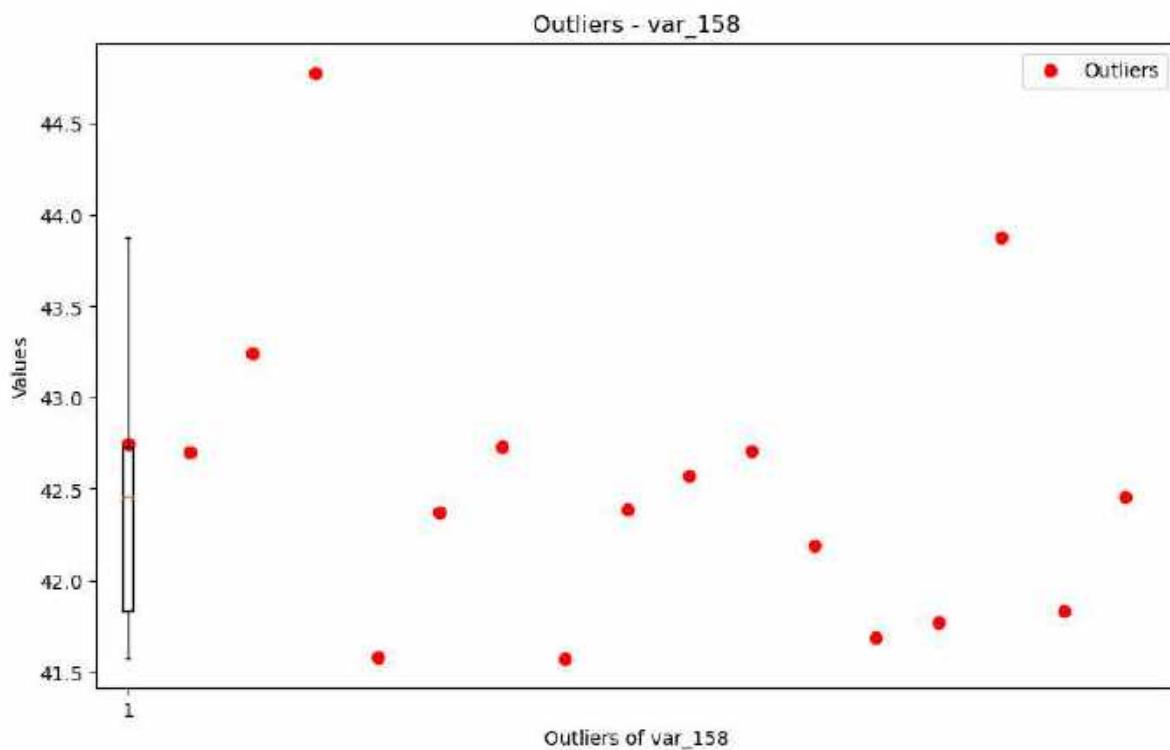


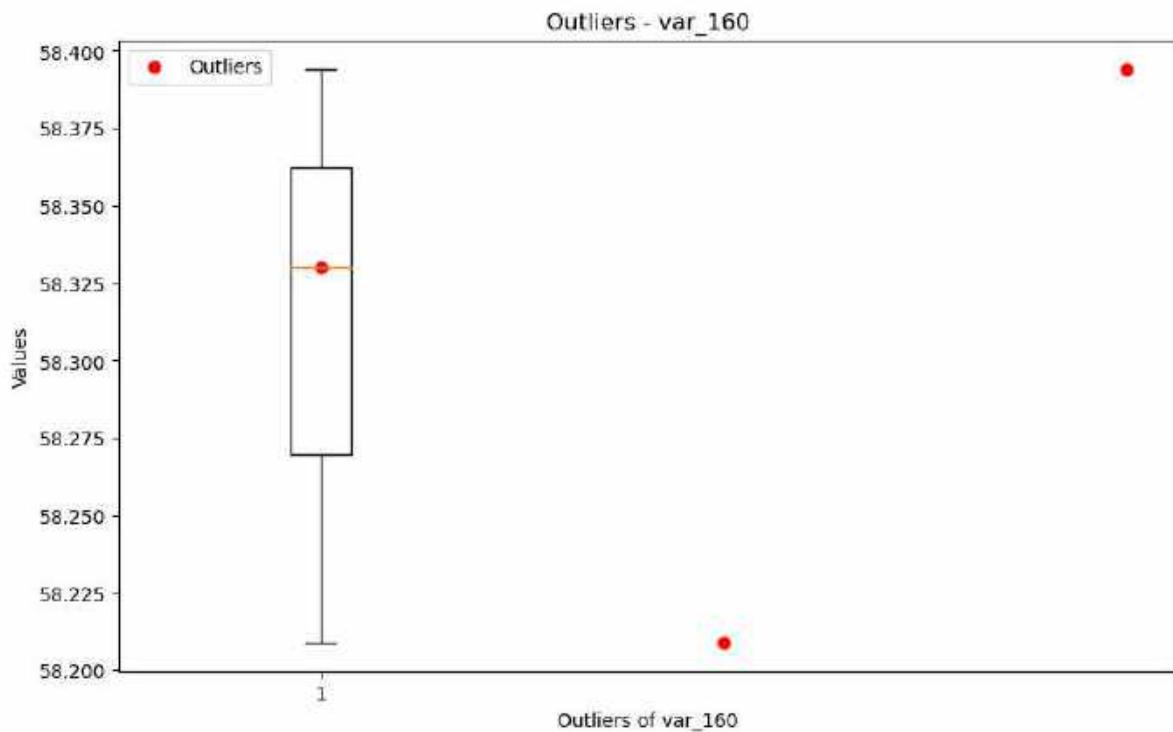
Total outliers in column var\_153: 4

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

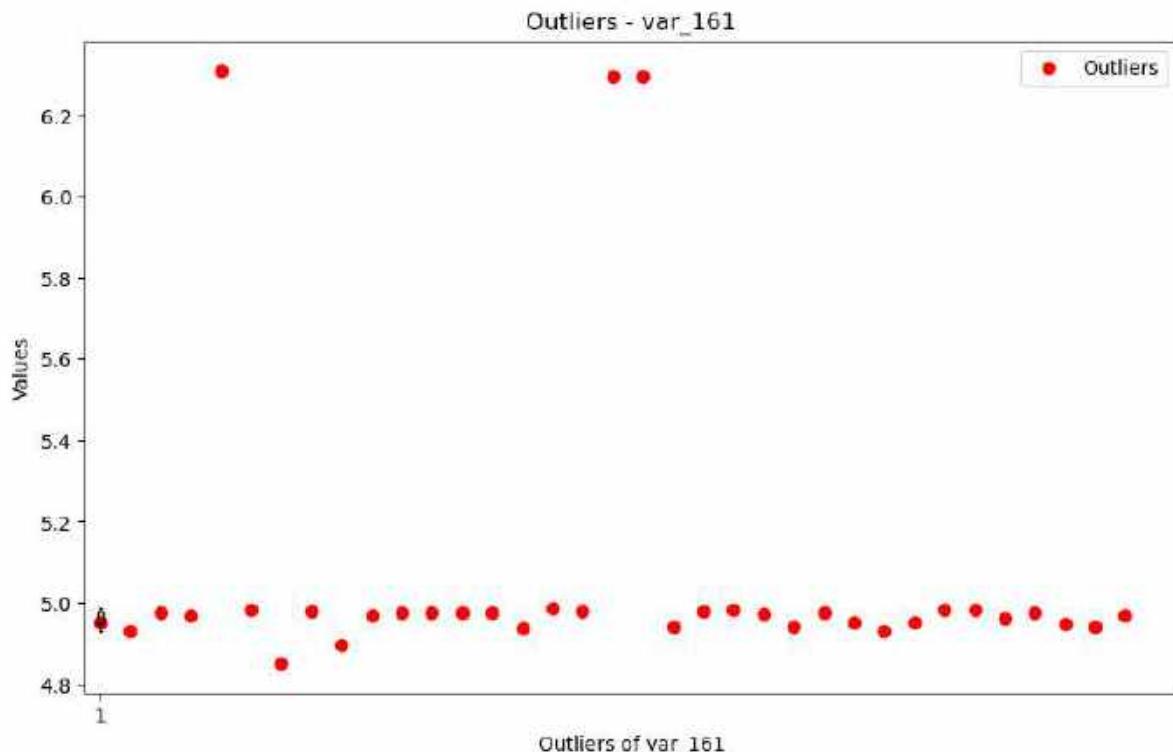




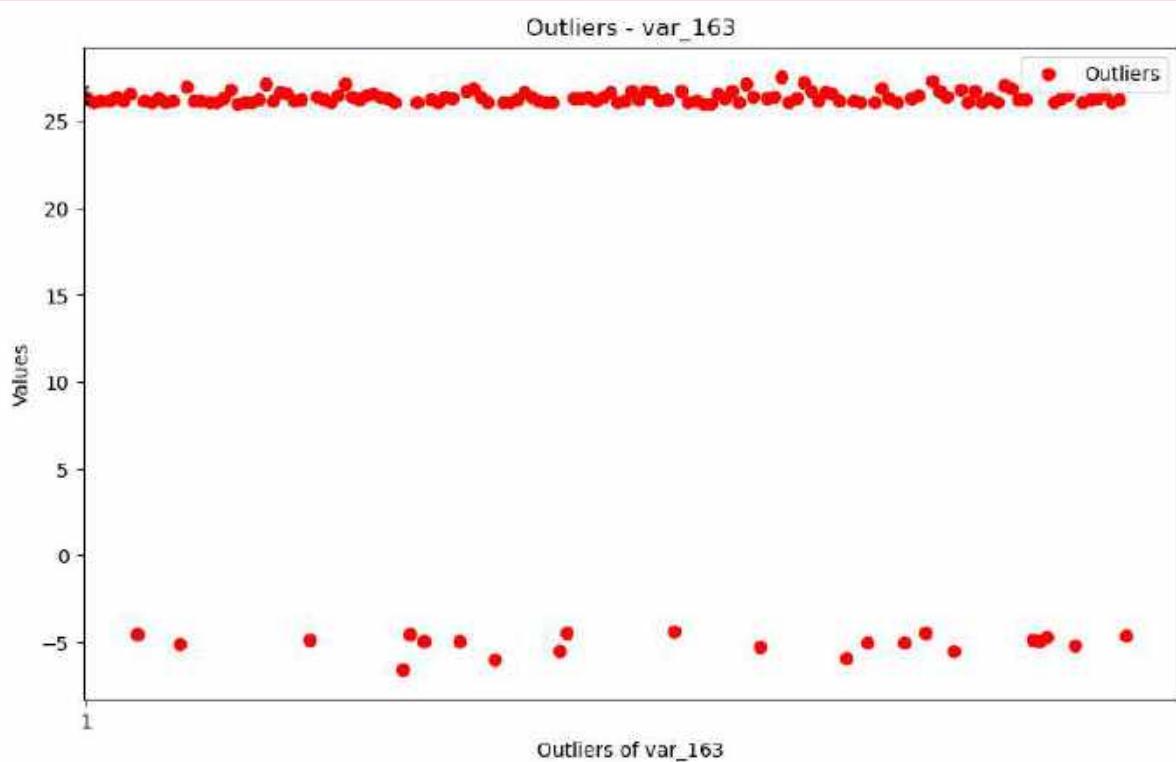
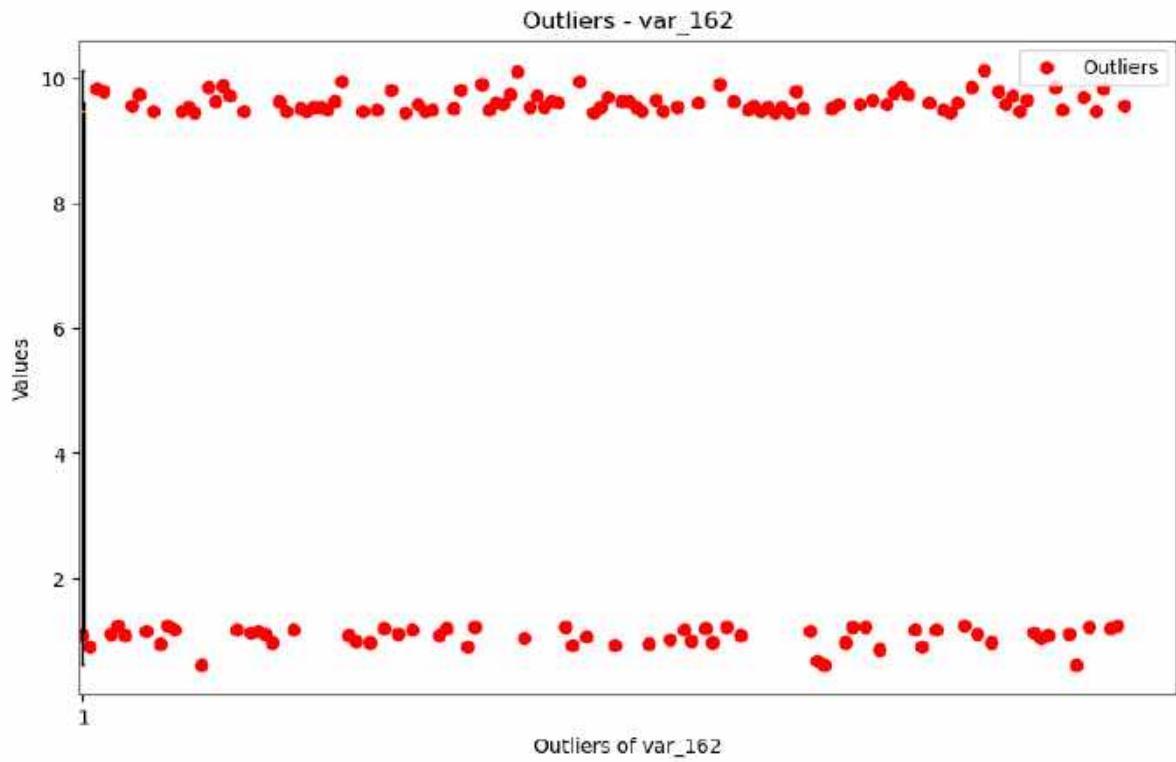


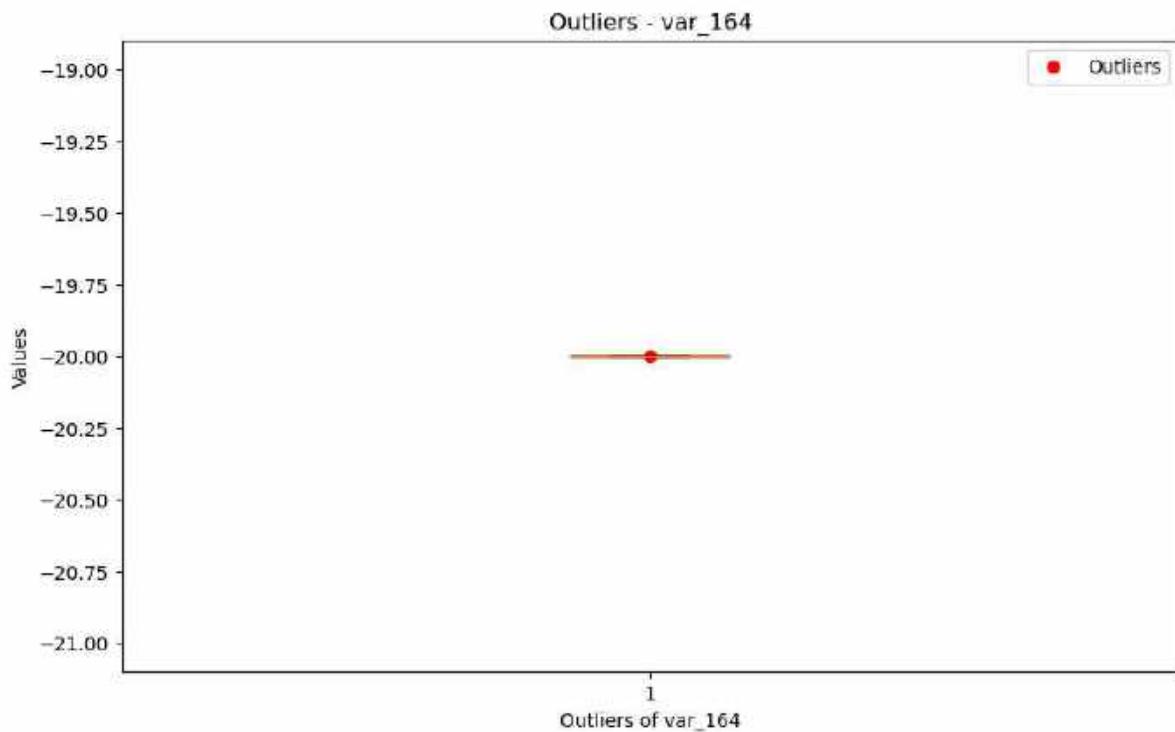


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_160: 3
```



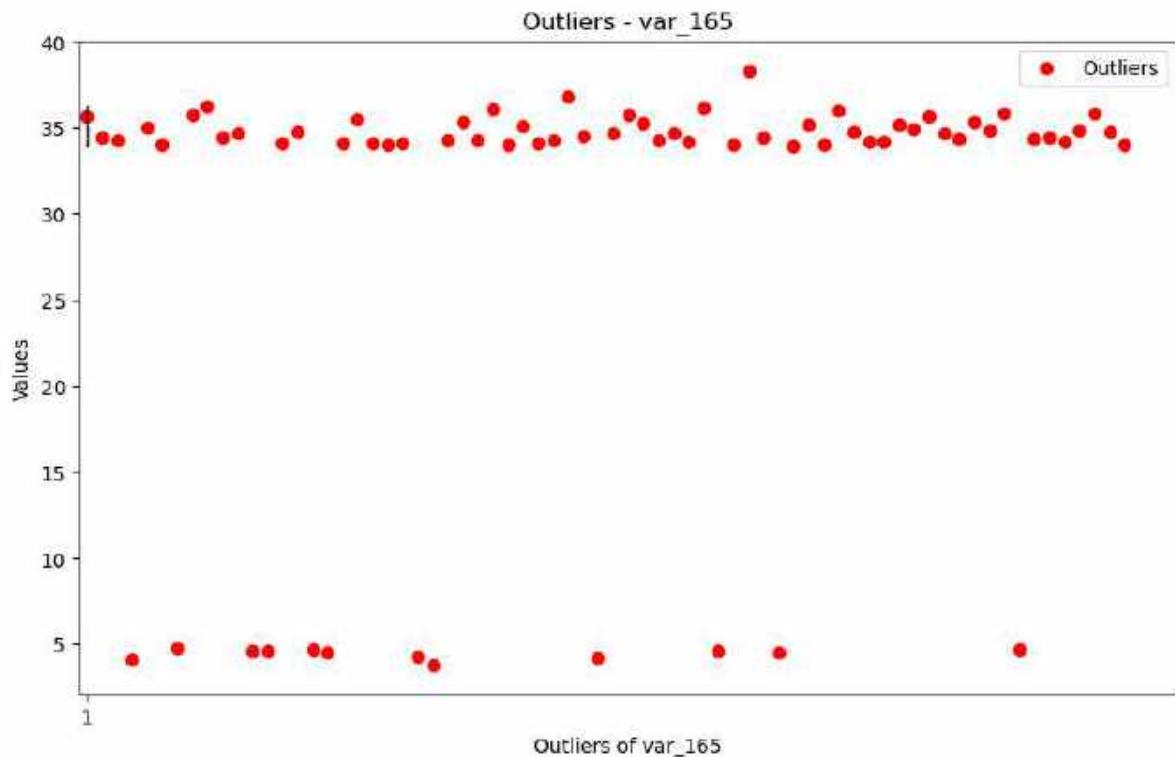
```
Total outliers in column var_161: 35  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





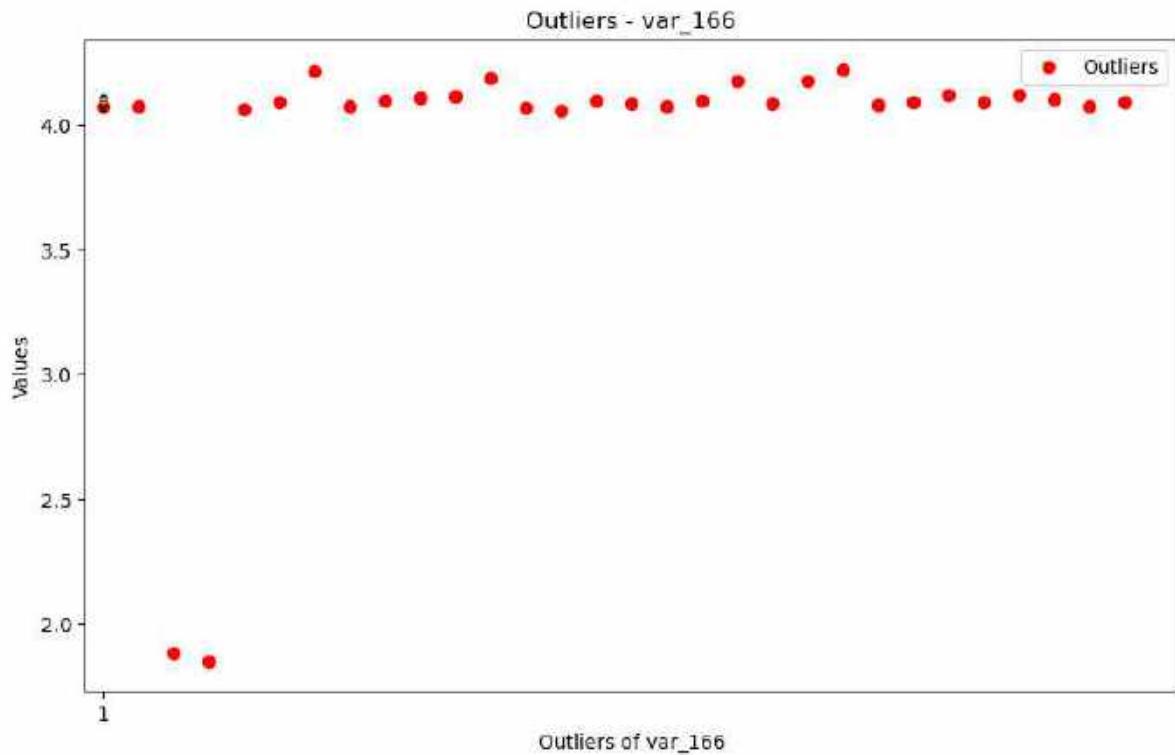
Total outliers in column var\_164: 1

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



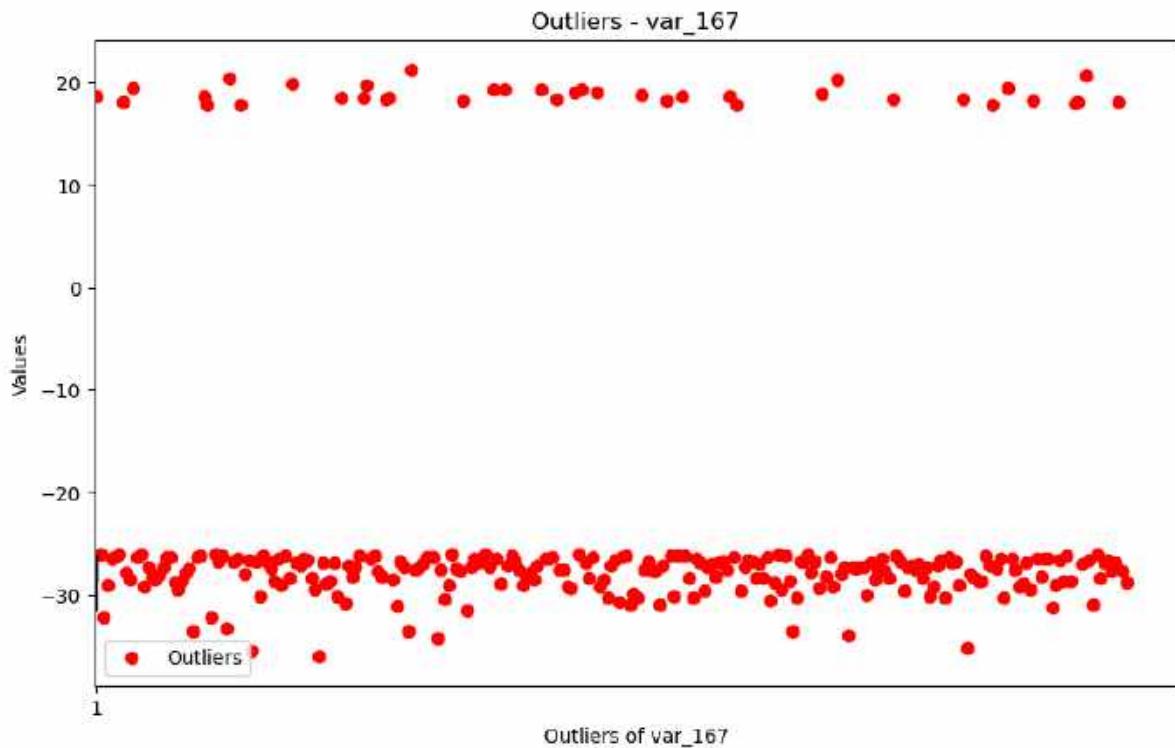
Total outliers in column var\_165: 70

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



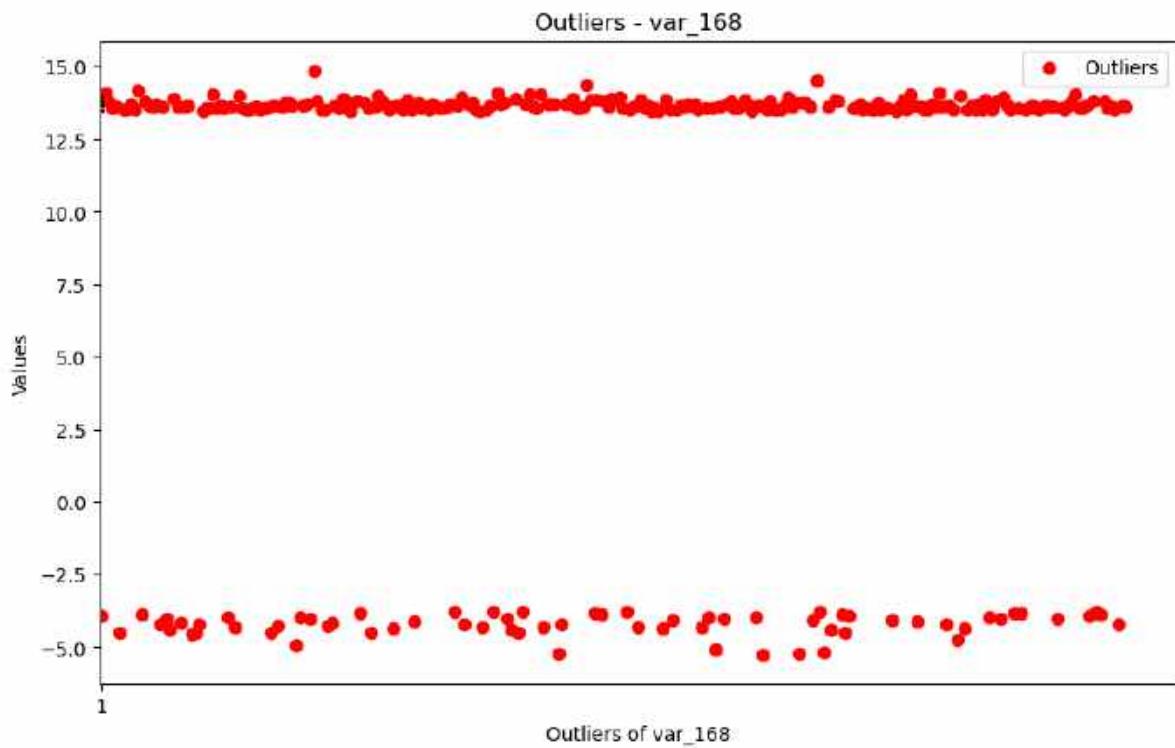
Total outliers in column var\_166: 30

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



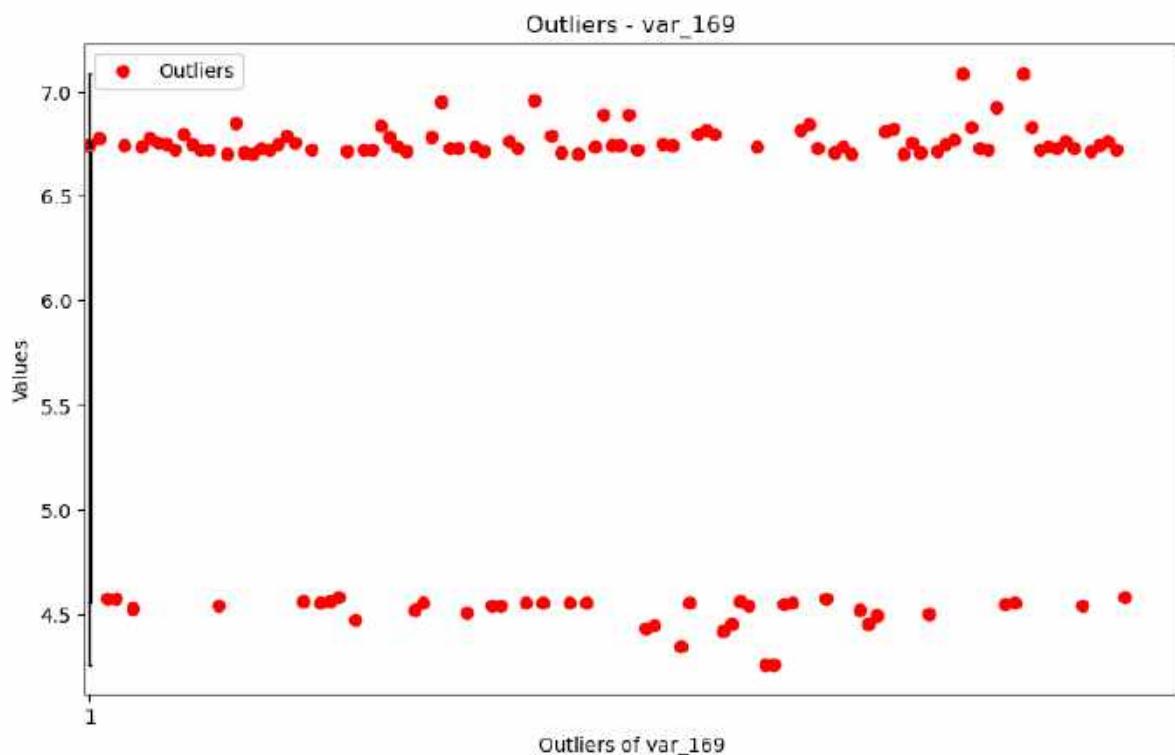
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_167: 279



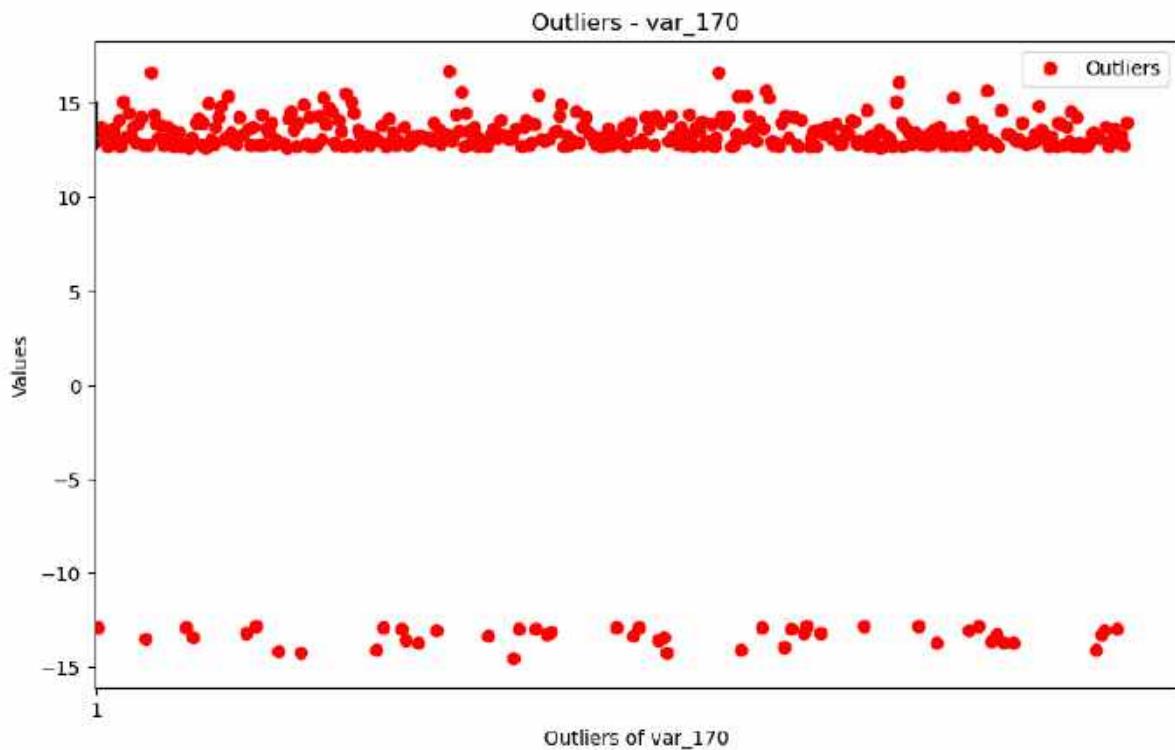
Total outliers in column var\_168: 286

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



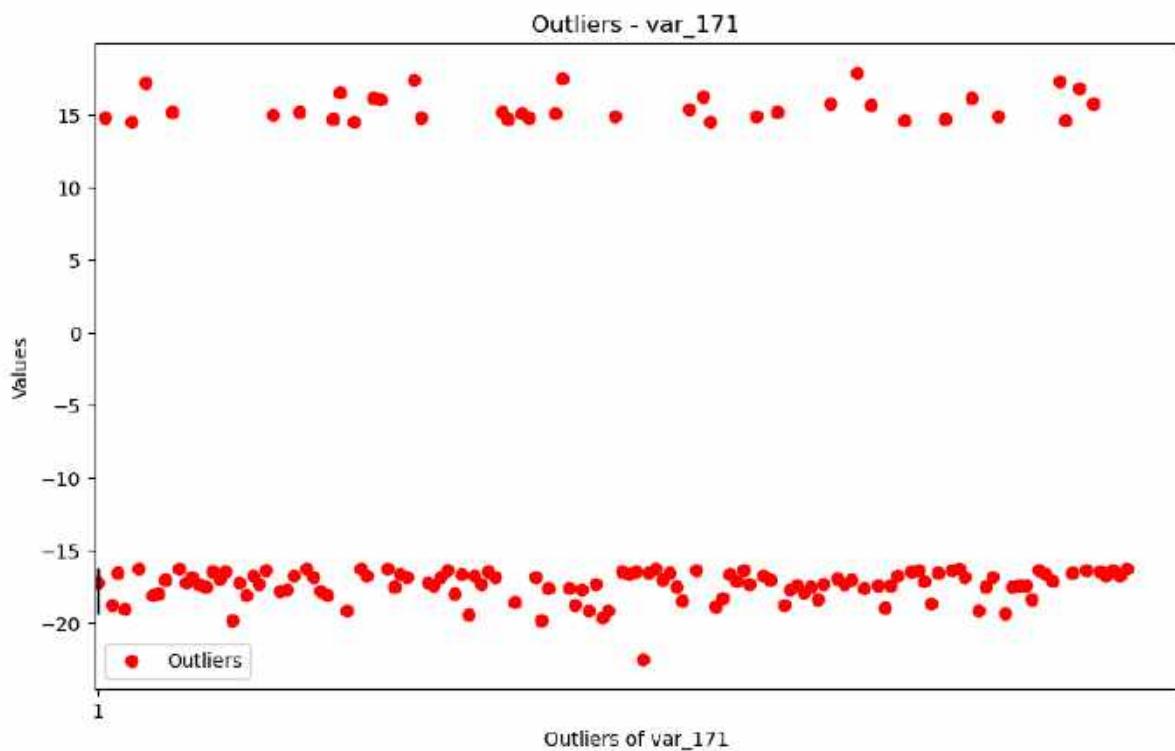
Total outliers in column var\_169: 122

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



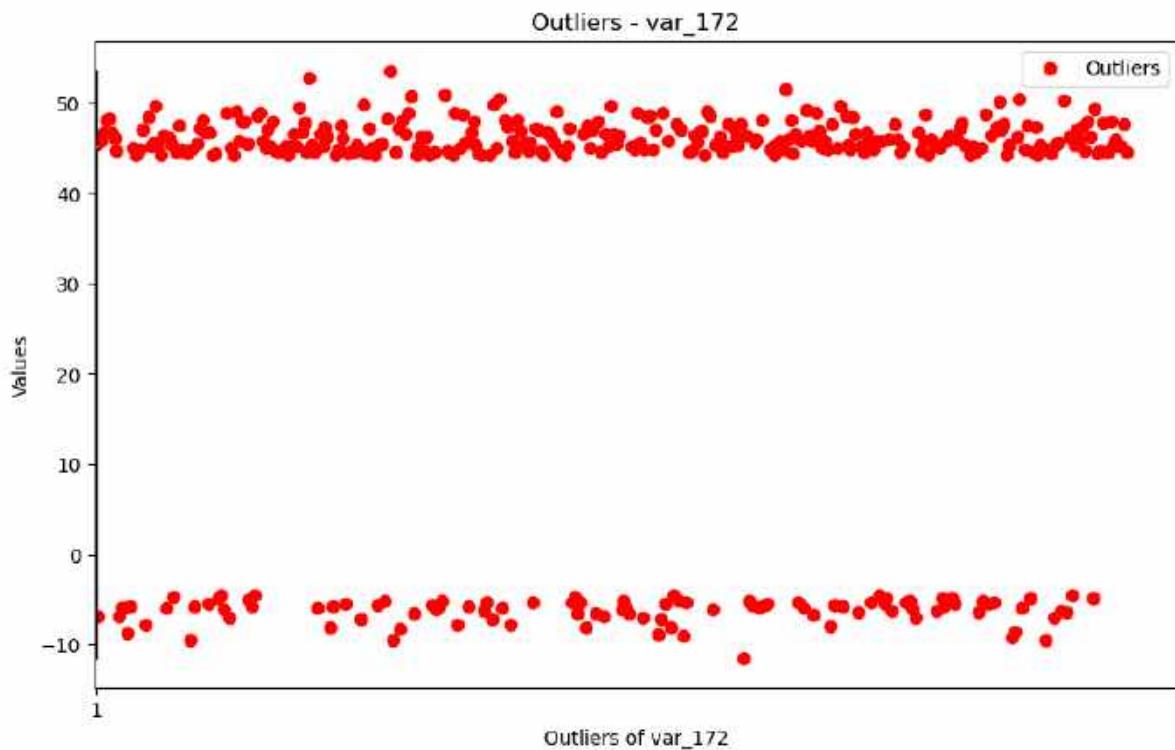
Total outliers in column var\_170: 413

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



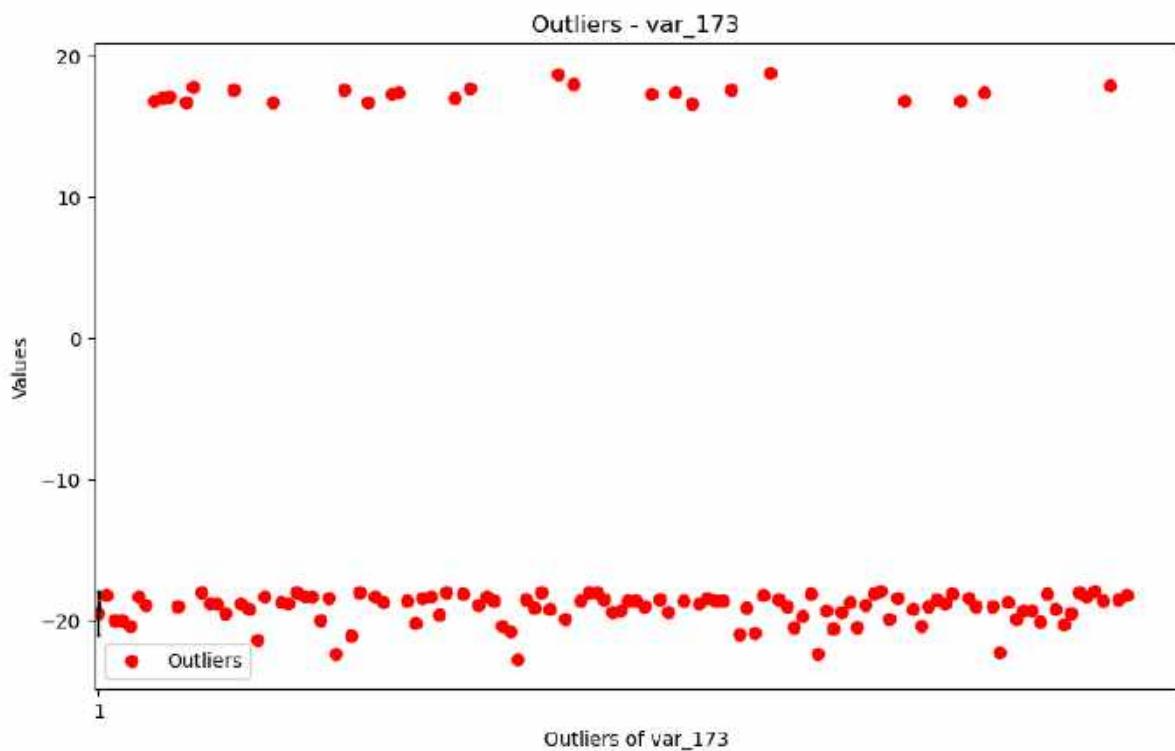
Total outliers in column var\_171: 154

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



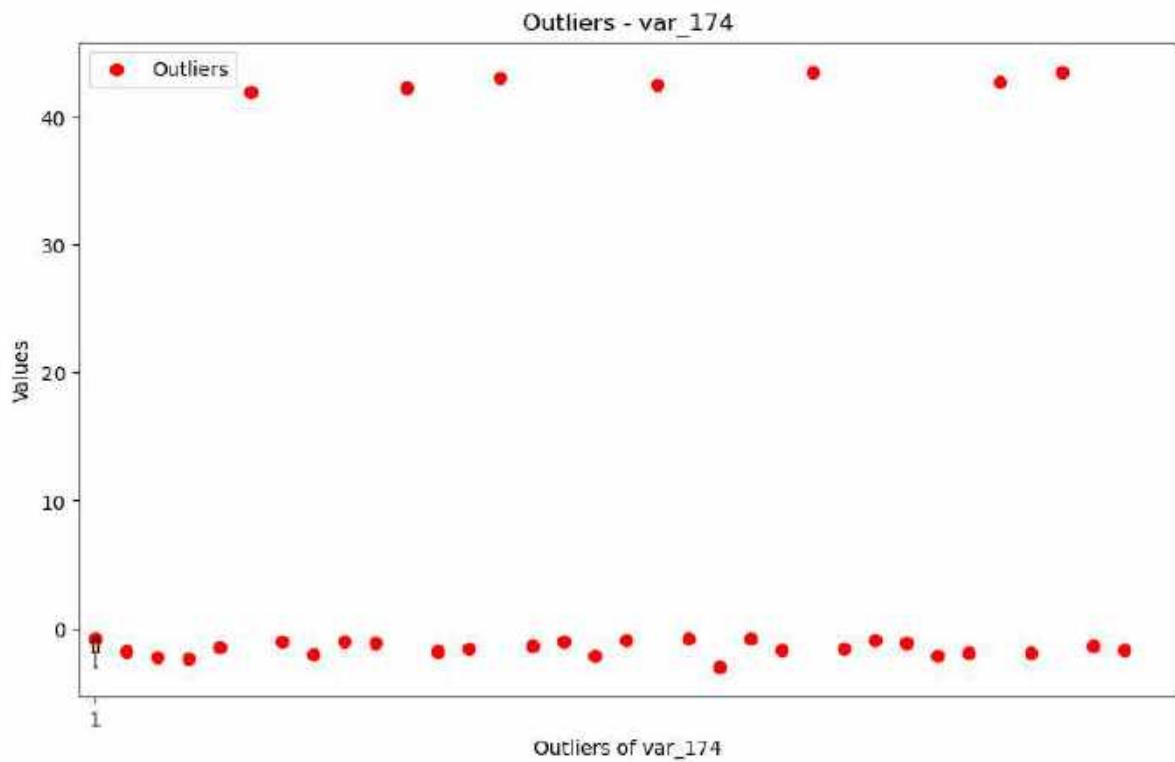
Total outliers in column var\_172: 397

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



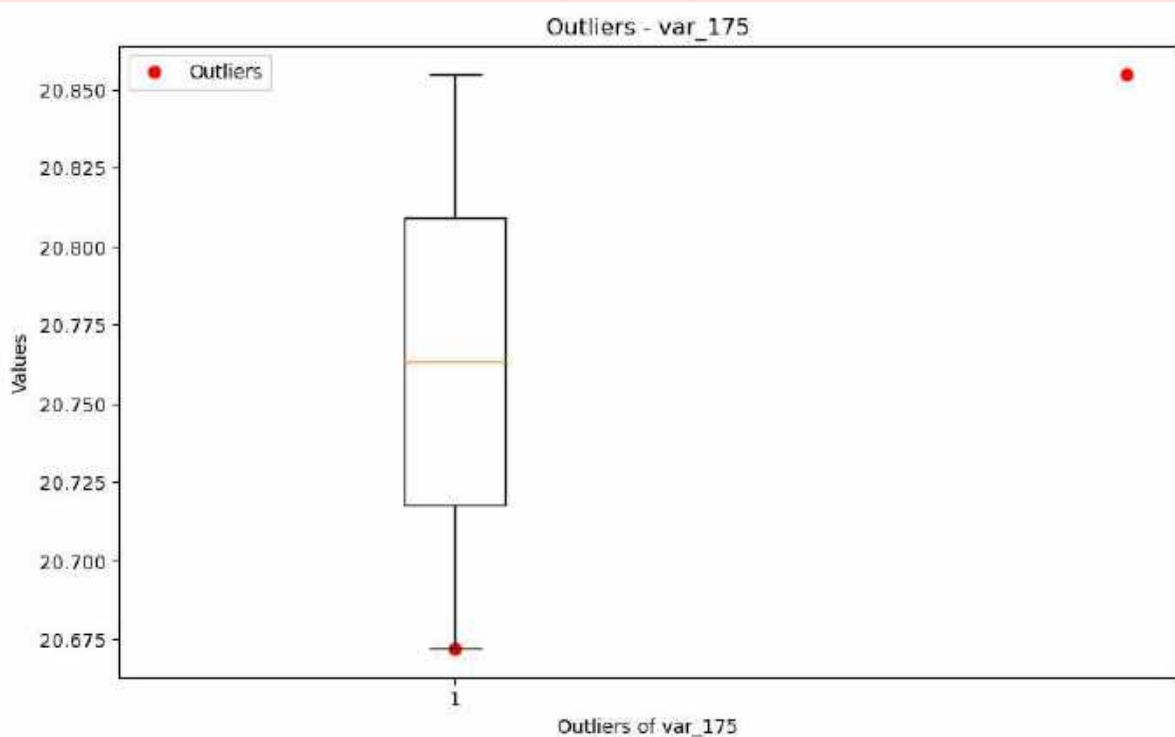
Total outliers in column var\_173: 131

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



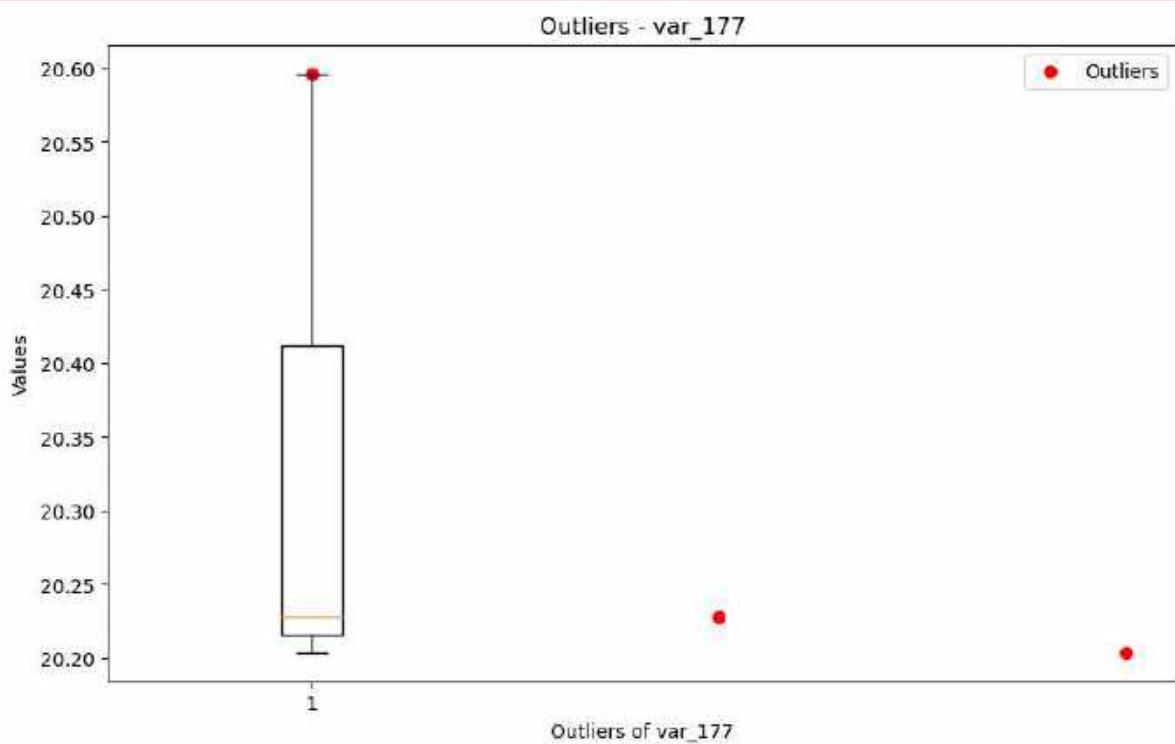
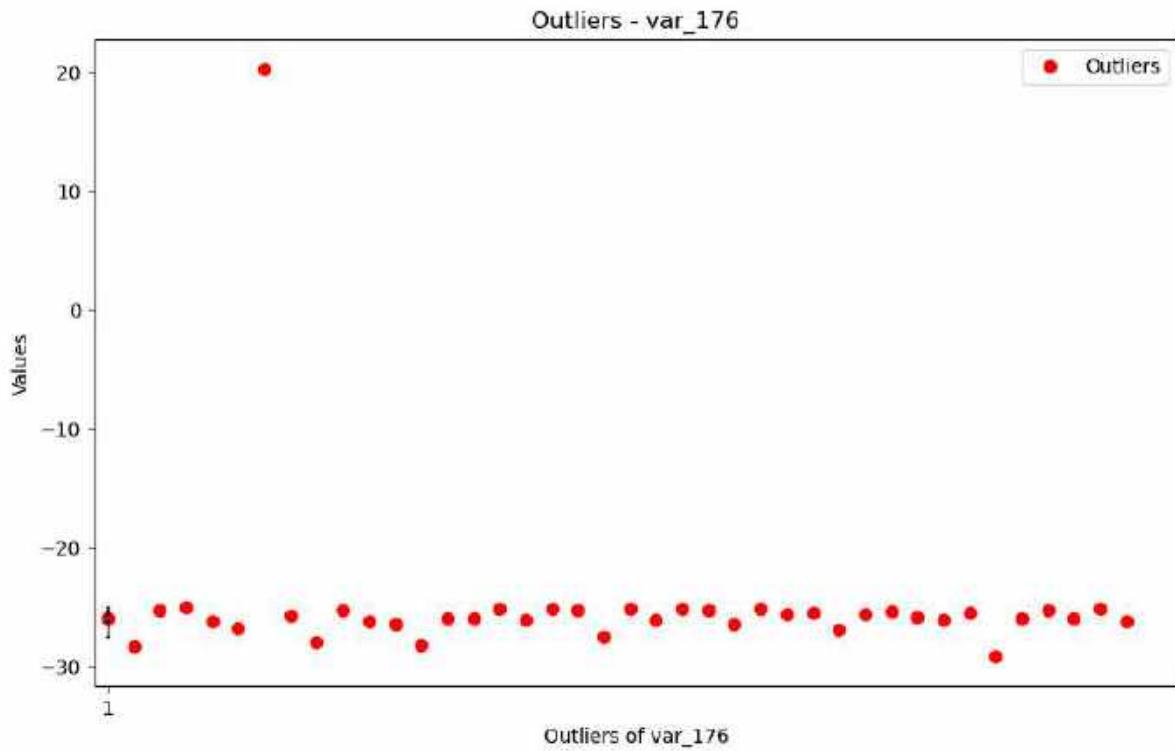
Total outliers in column var\_174: 34

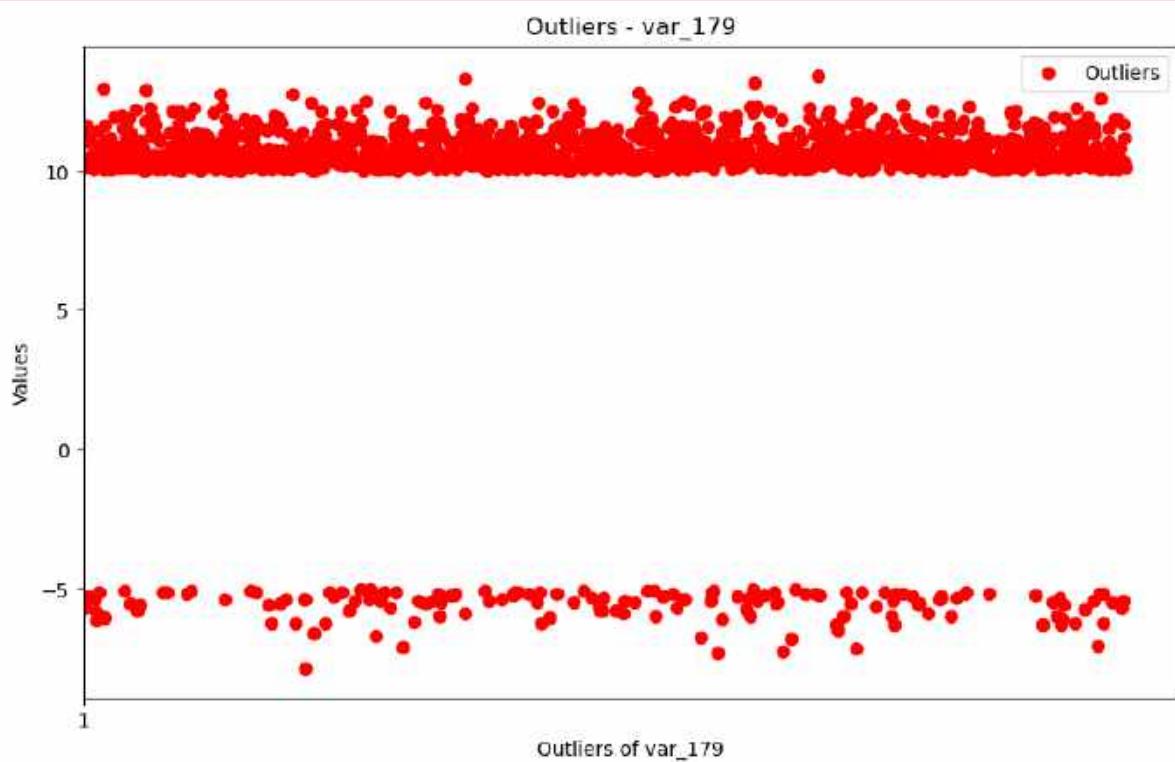
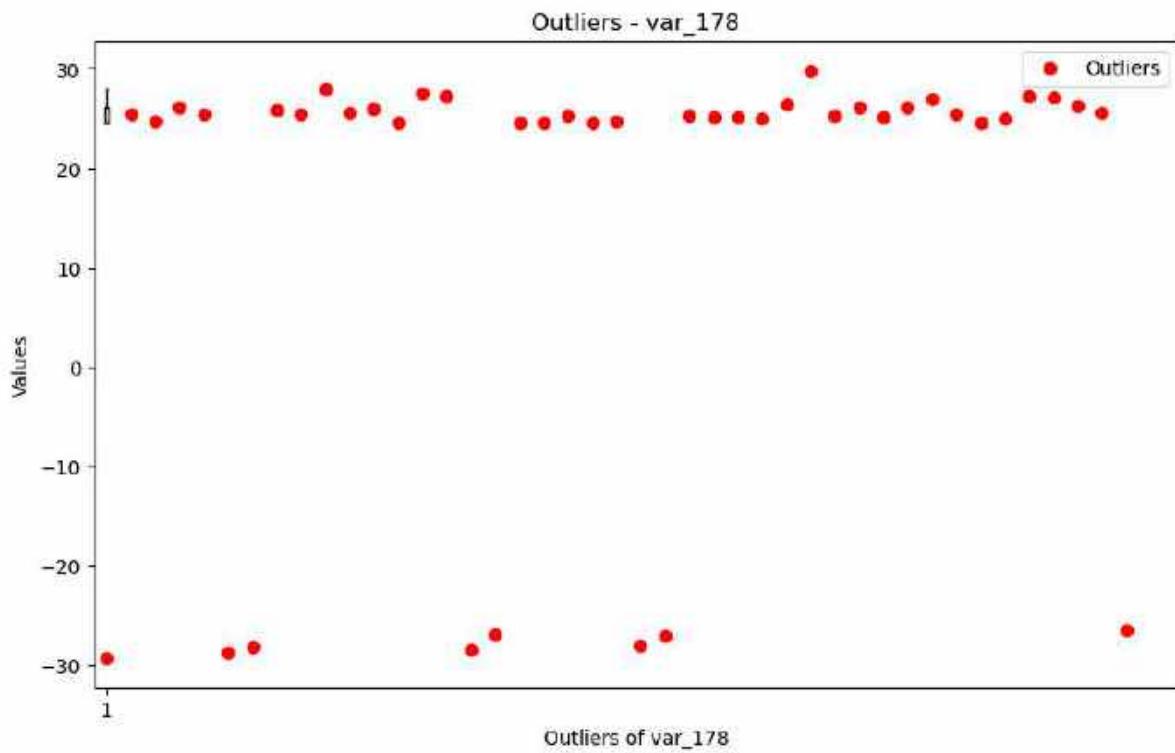
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

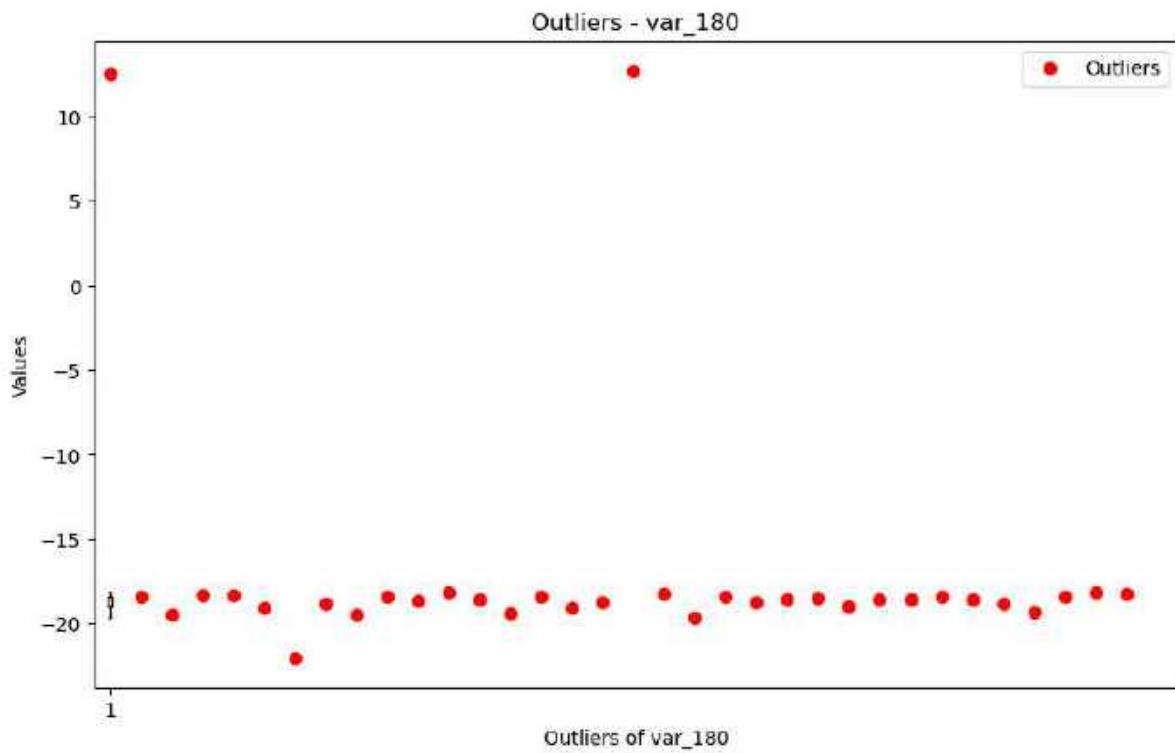


Total outliers in column var\_175: 2

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

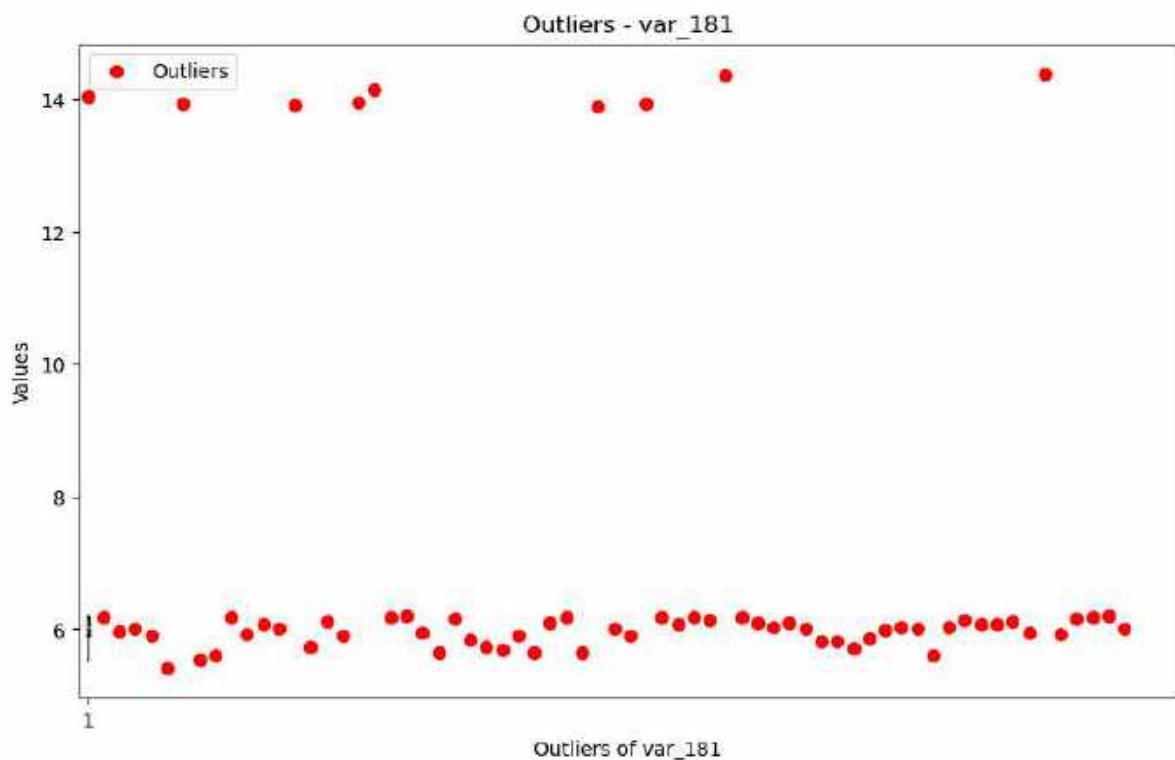






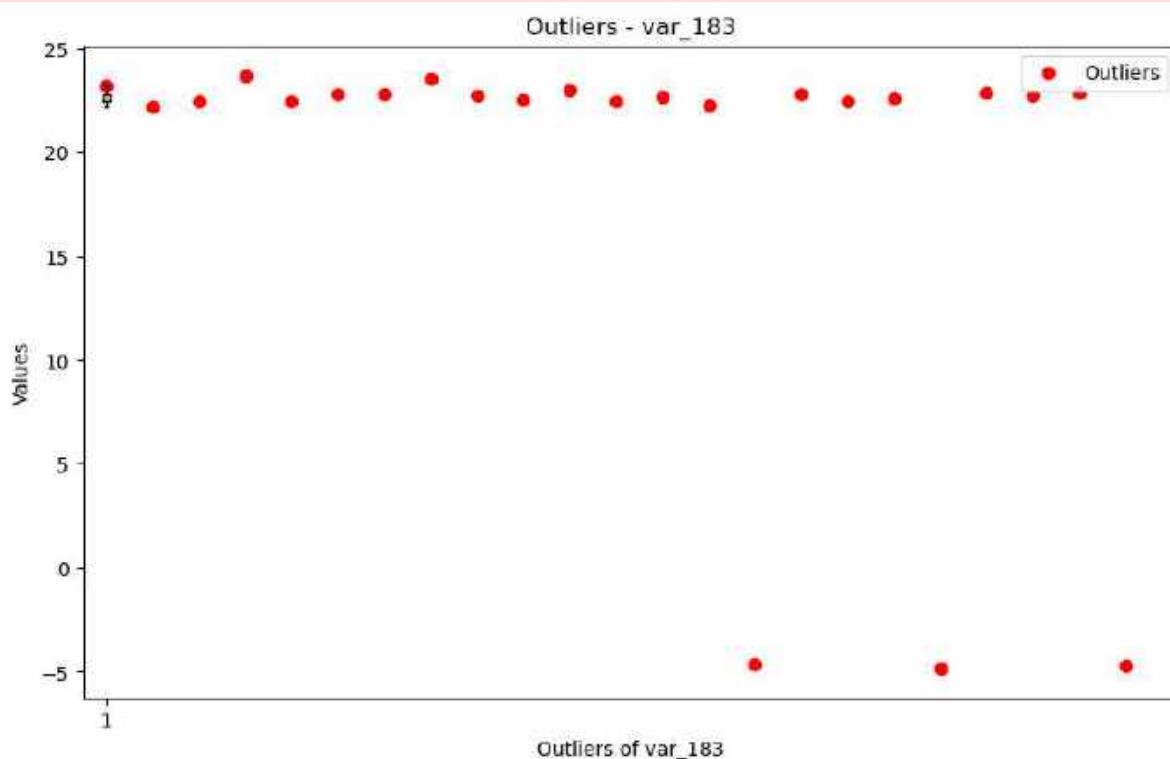
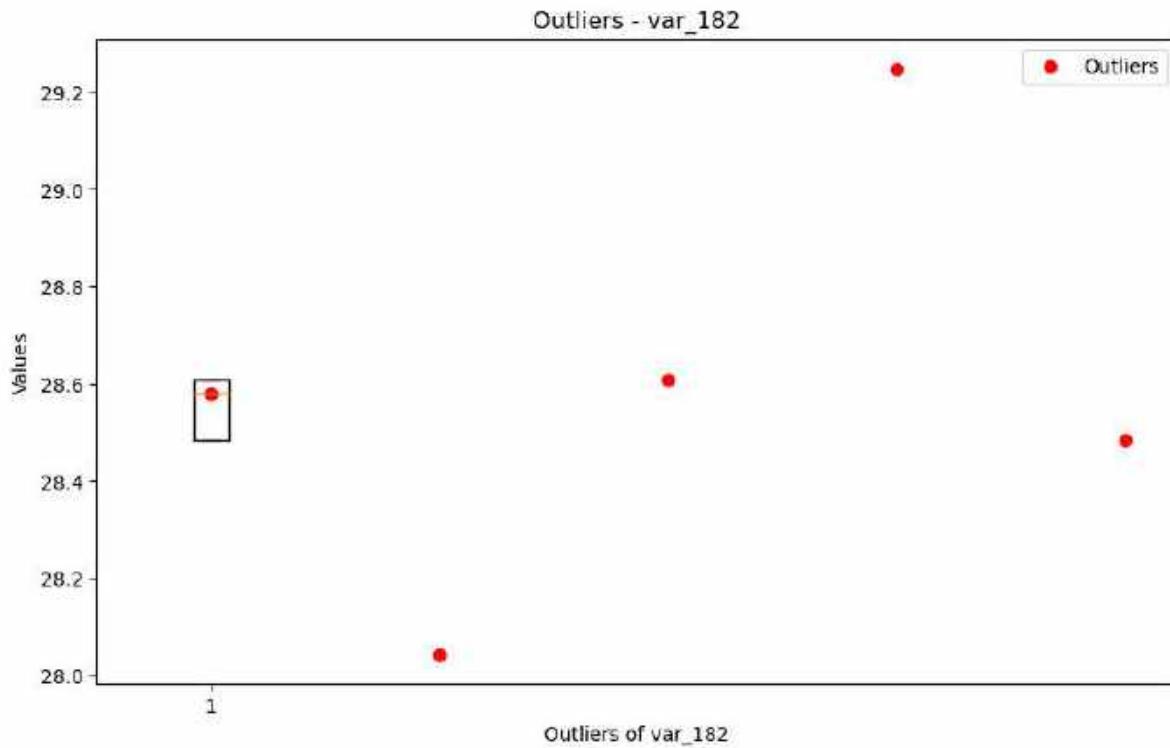
Total outliers in column var\_180: 34

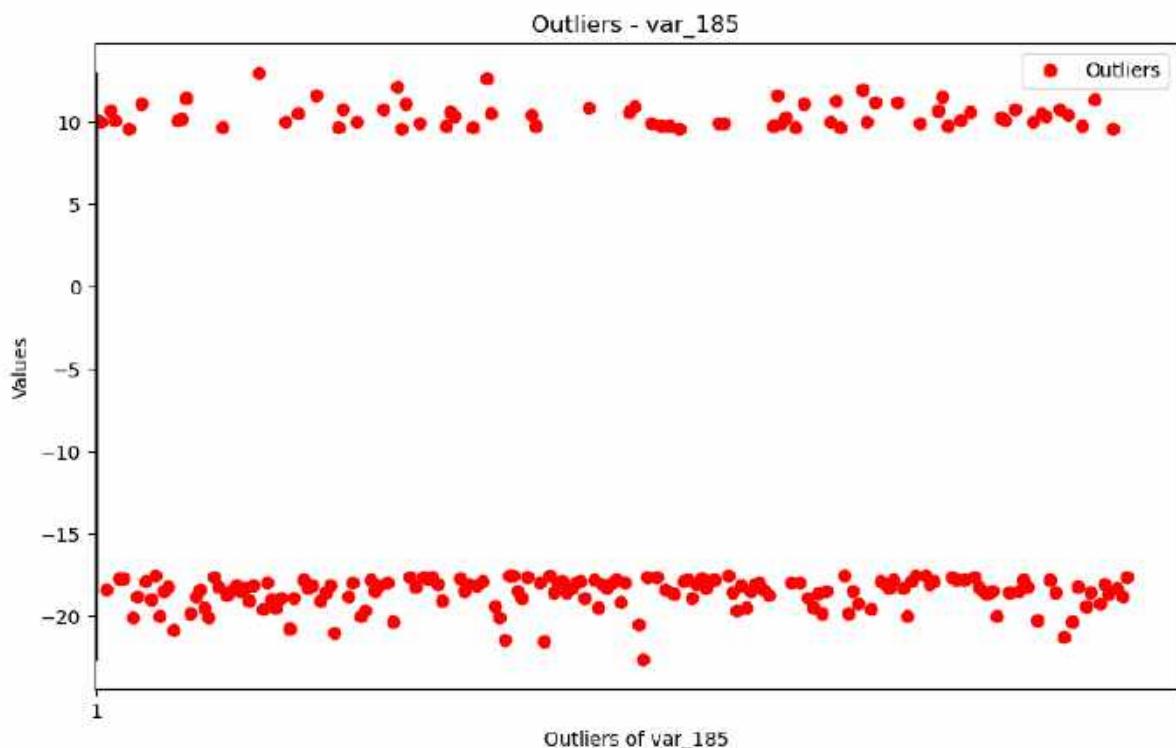
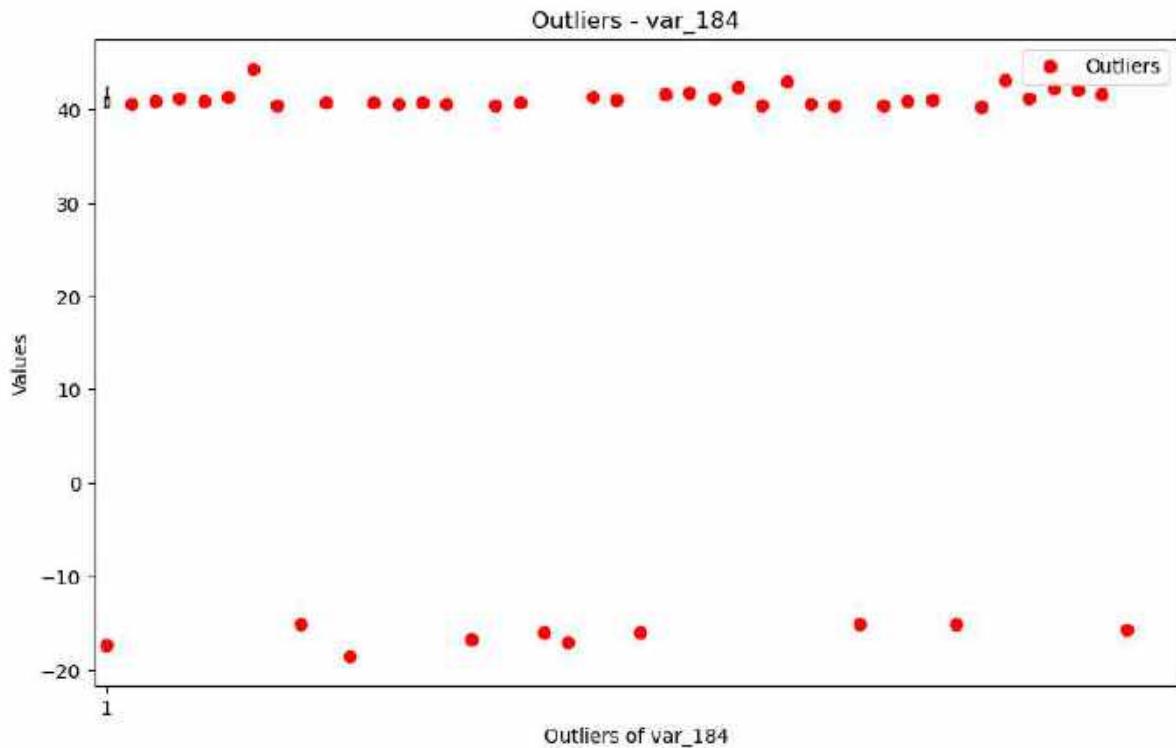
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

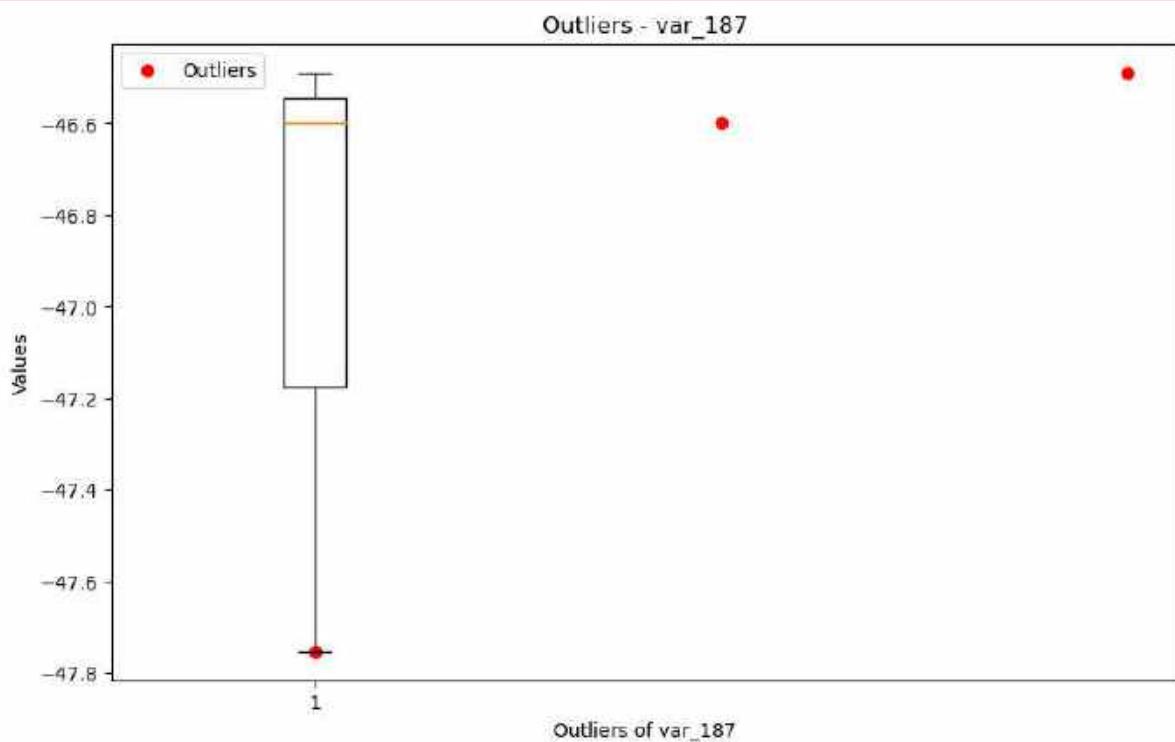
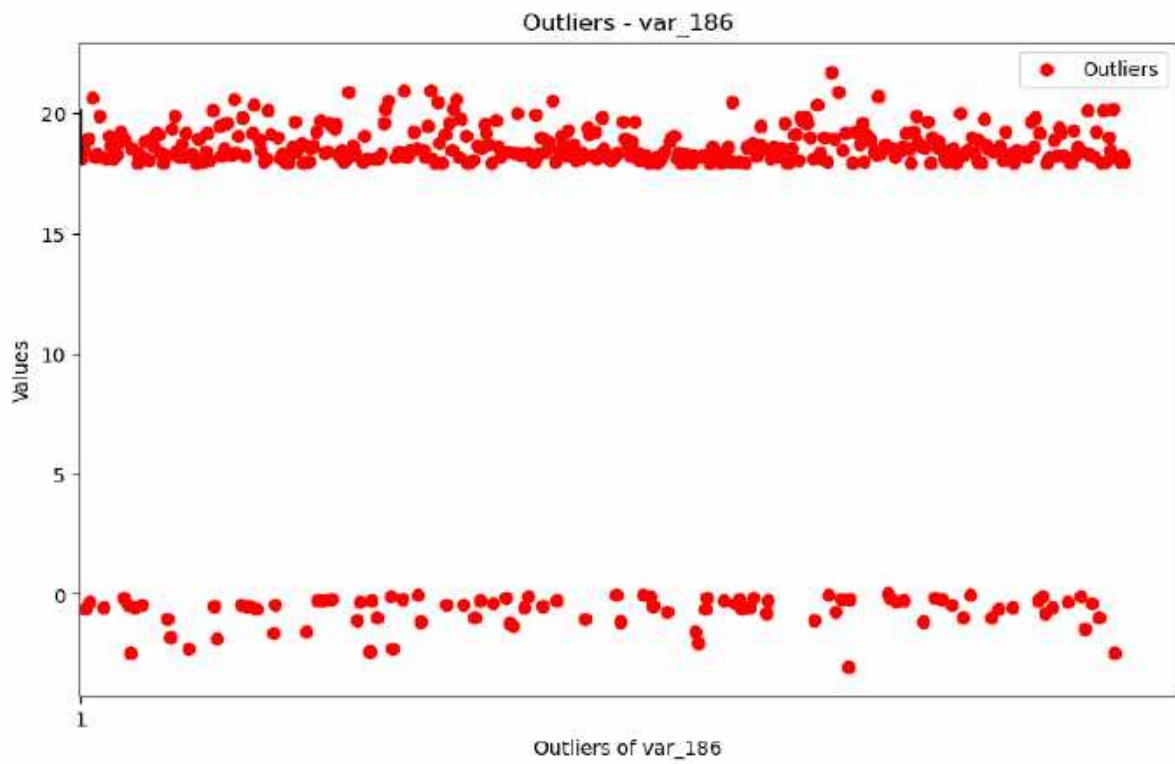


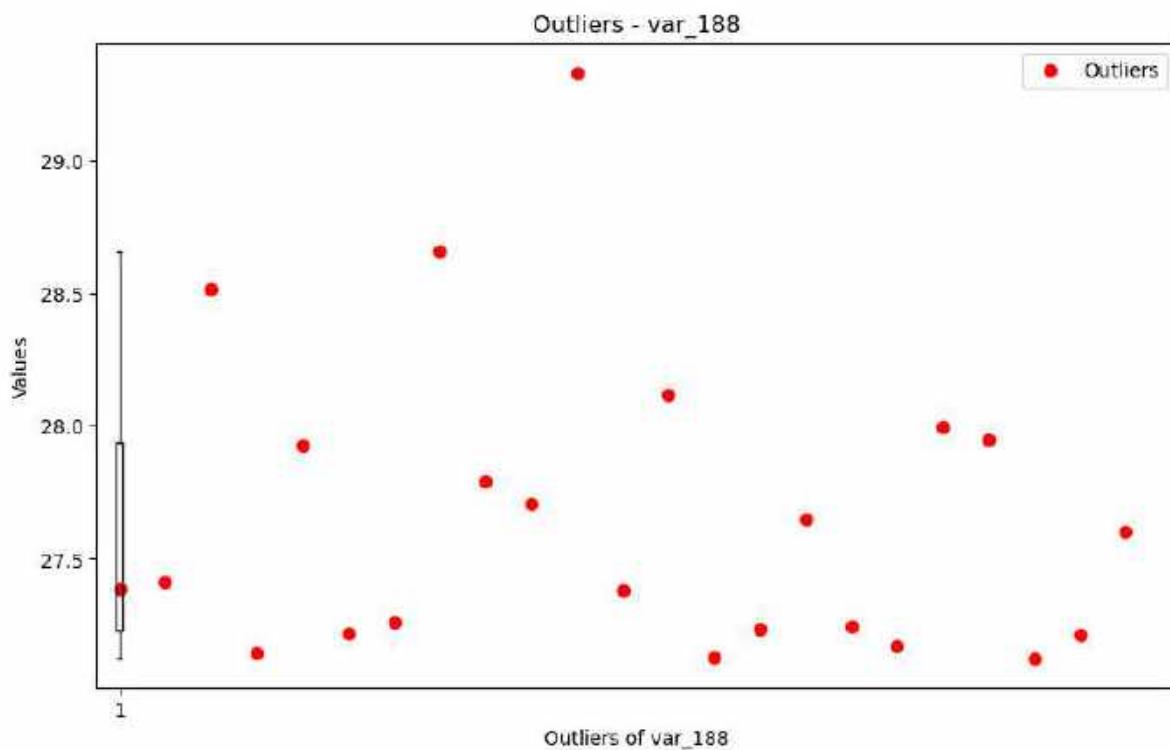
Total outliers in column var\_181: 66

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



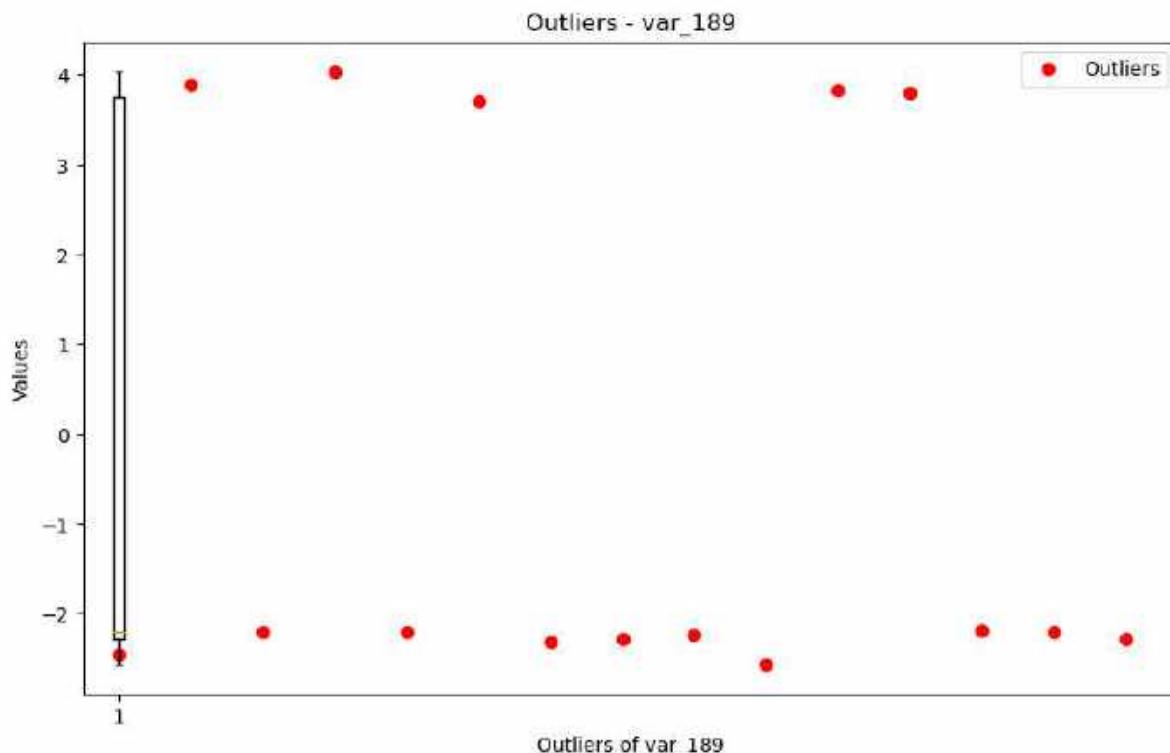






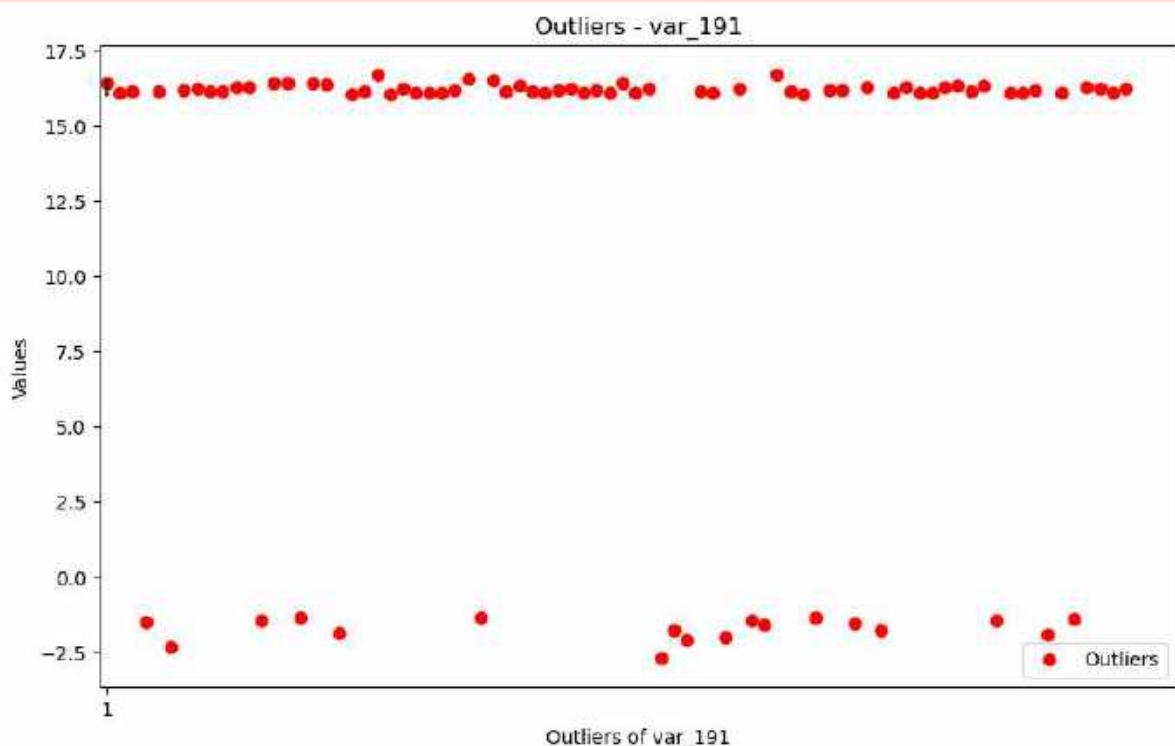
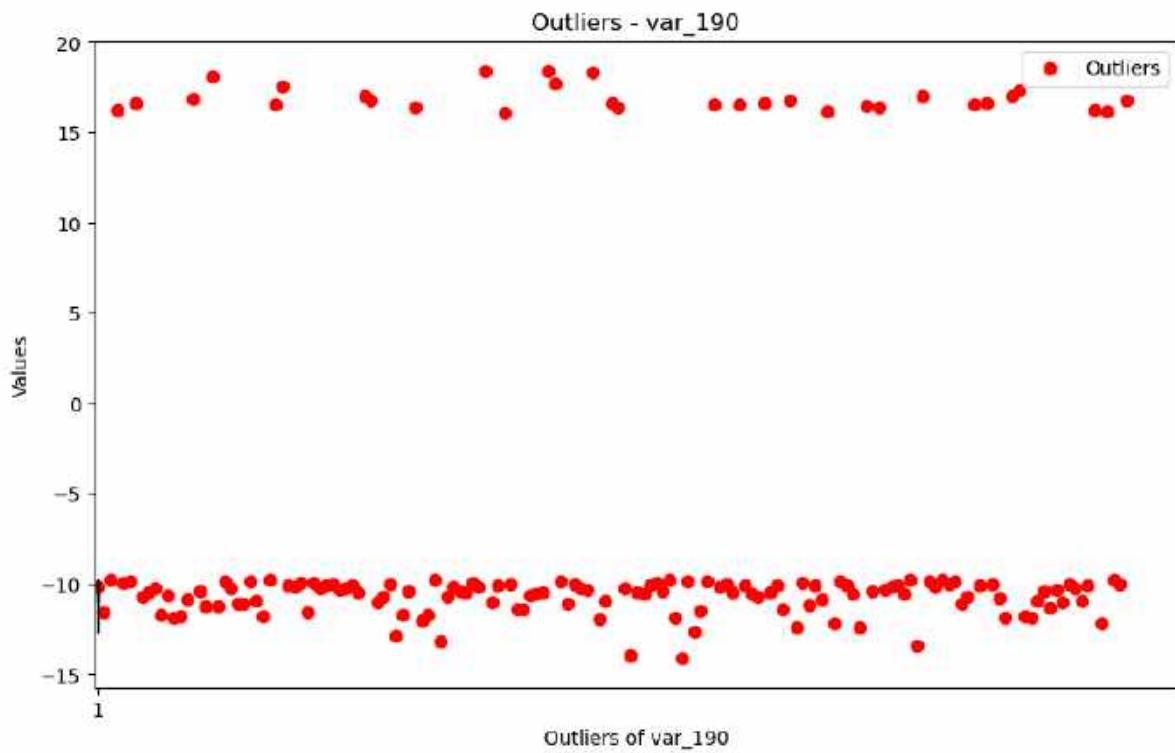
Total outliers in column var\_188: 23

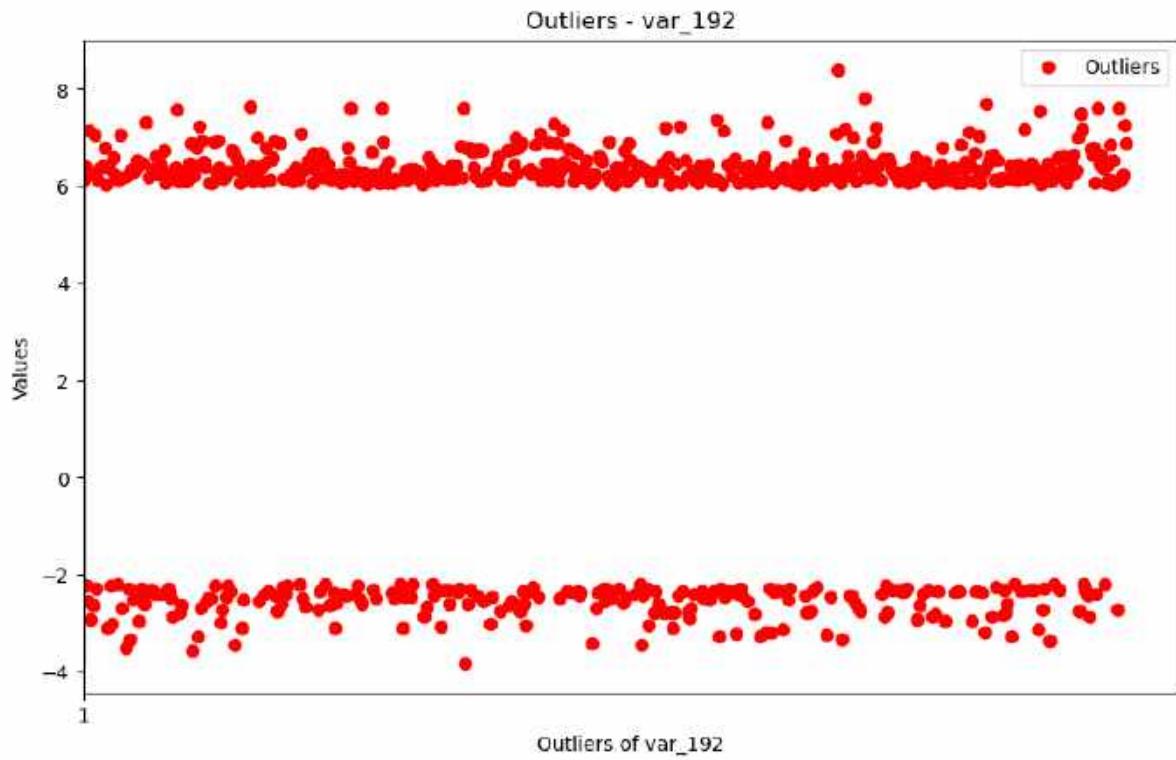
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



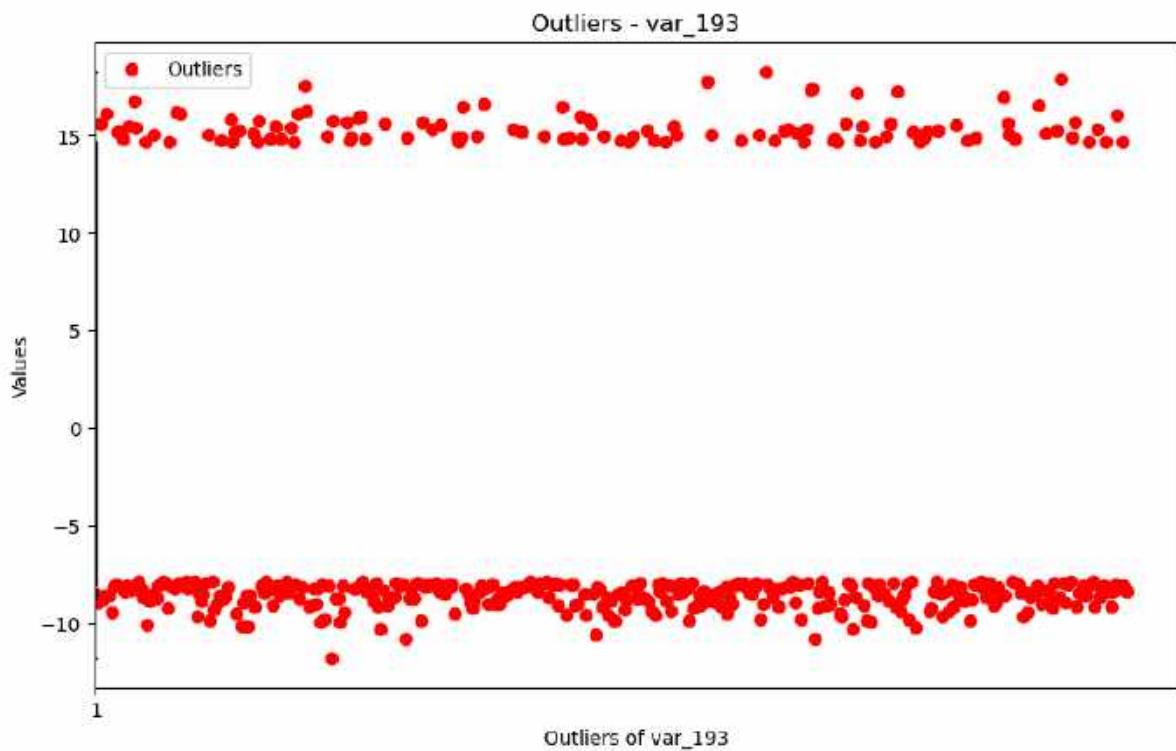
Total outliers in column var\_189: 15

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



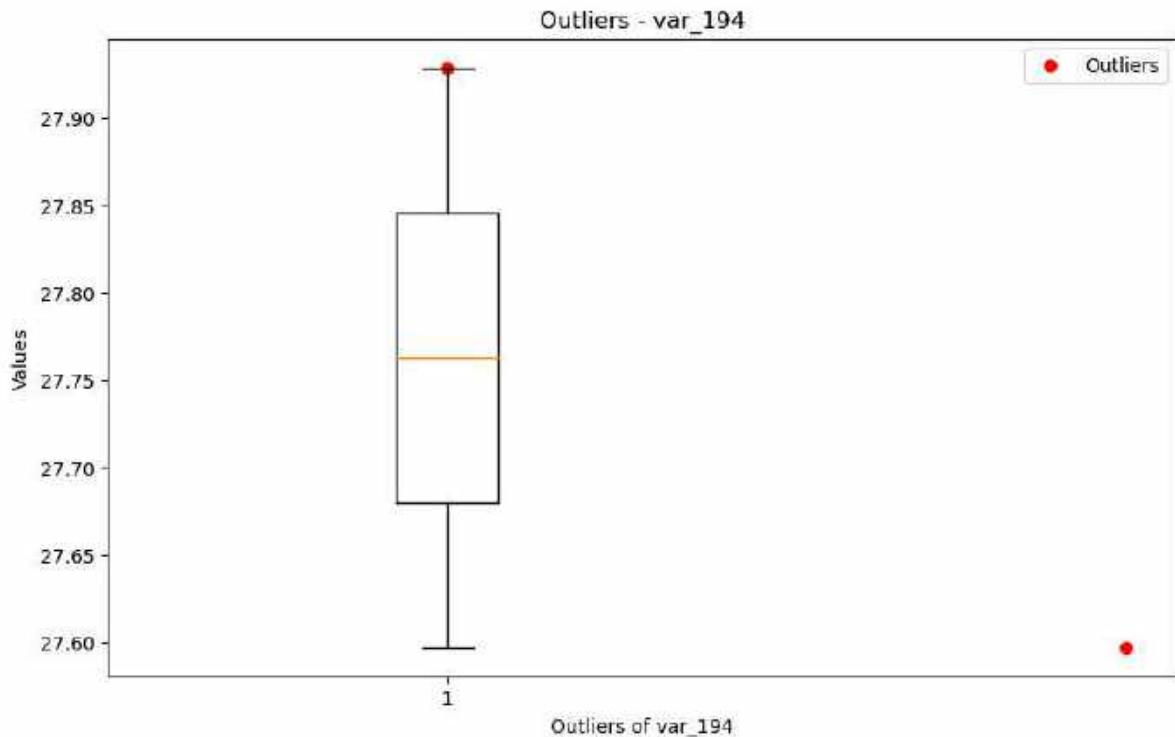


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_192: 733
```



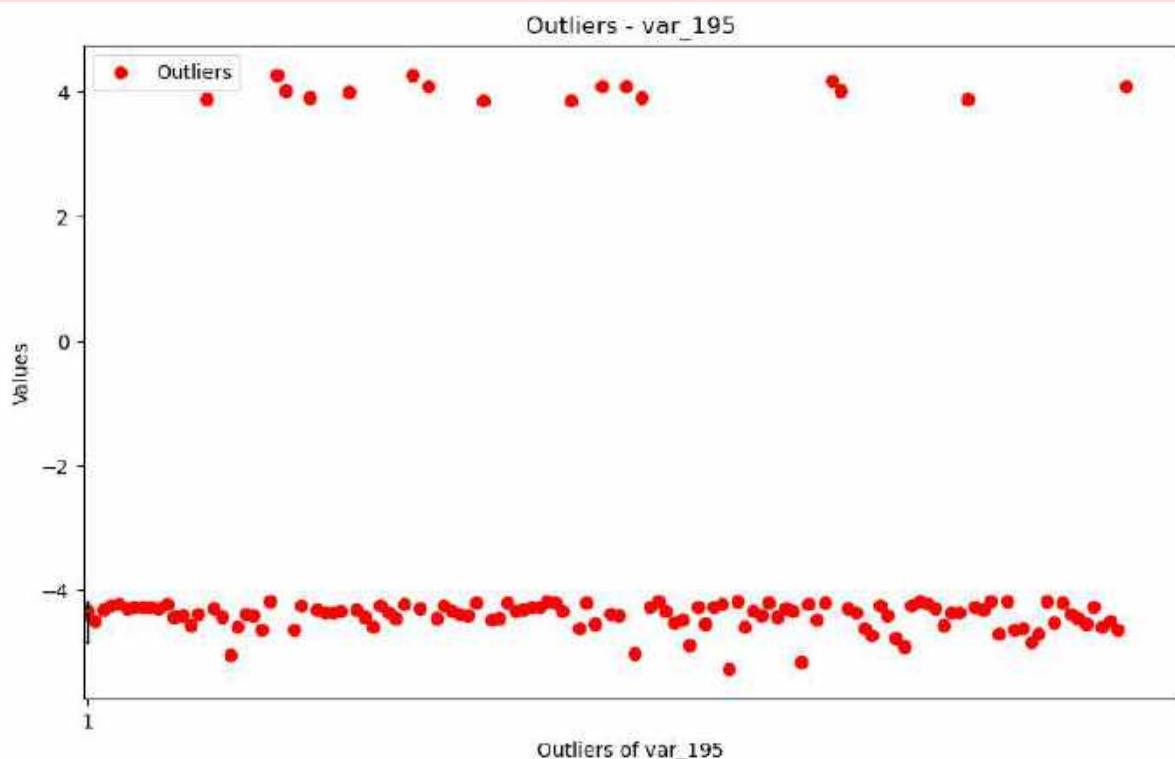
```
Total outliers in column var_193: 461
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



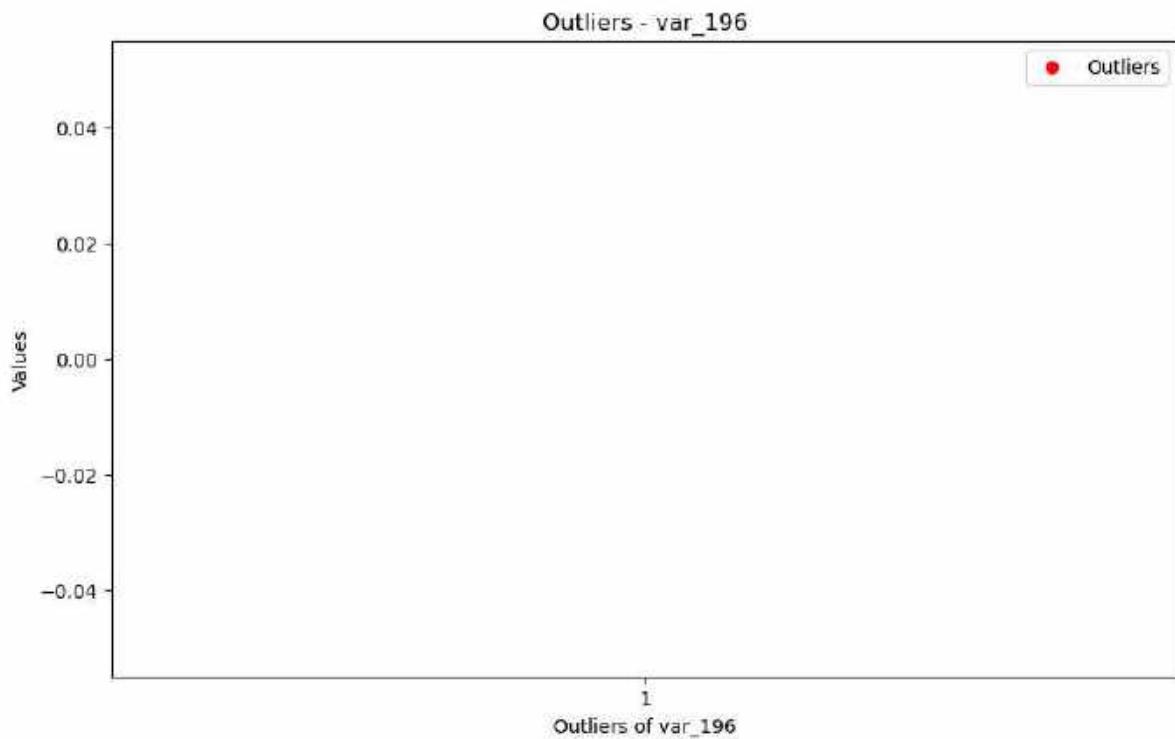
Total outliers in column var\_194: 2

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



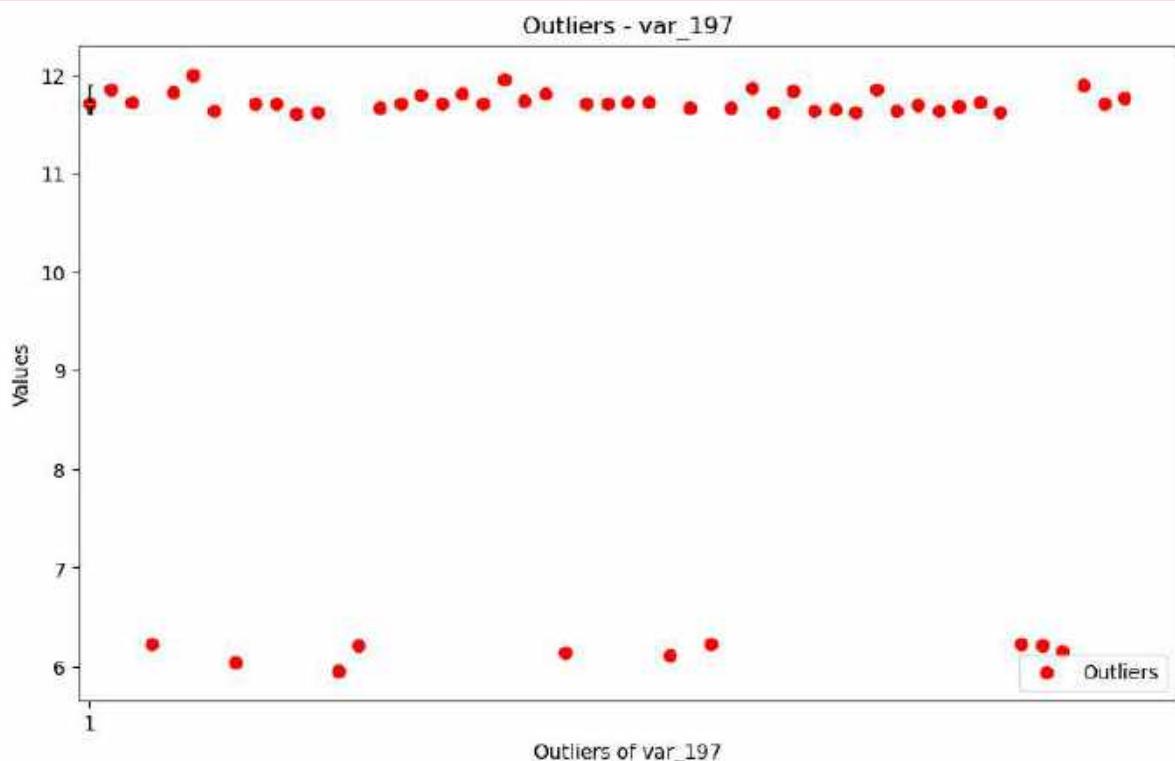
Total outliers in column var\_195: 132

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



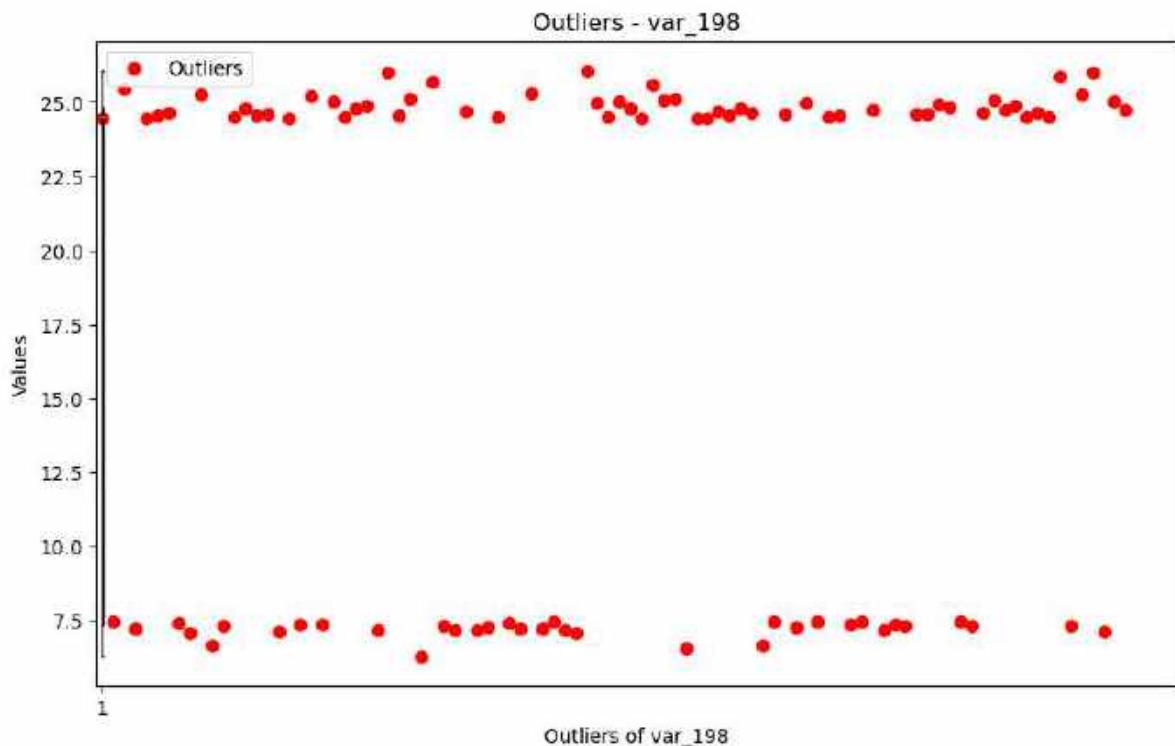
```
Total outliers in column var_196: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

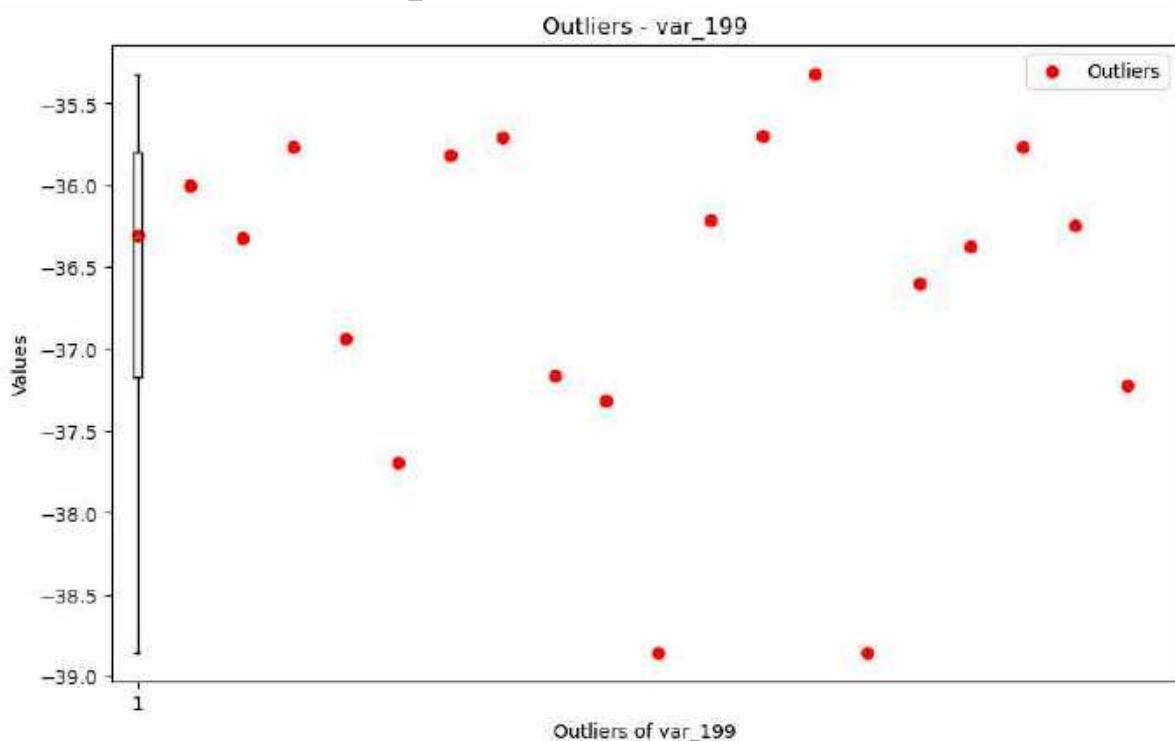


```
Total outliers in column var_197: 51
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



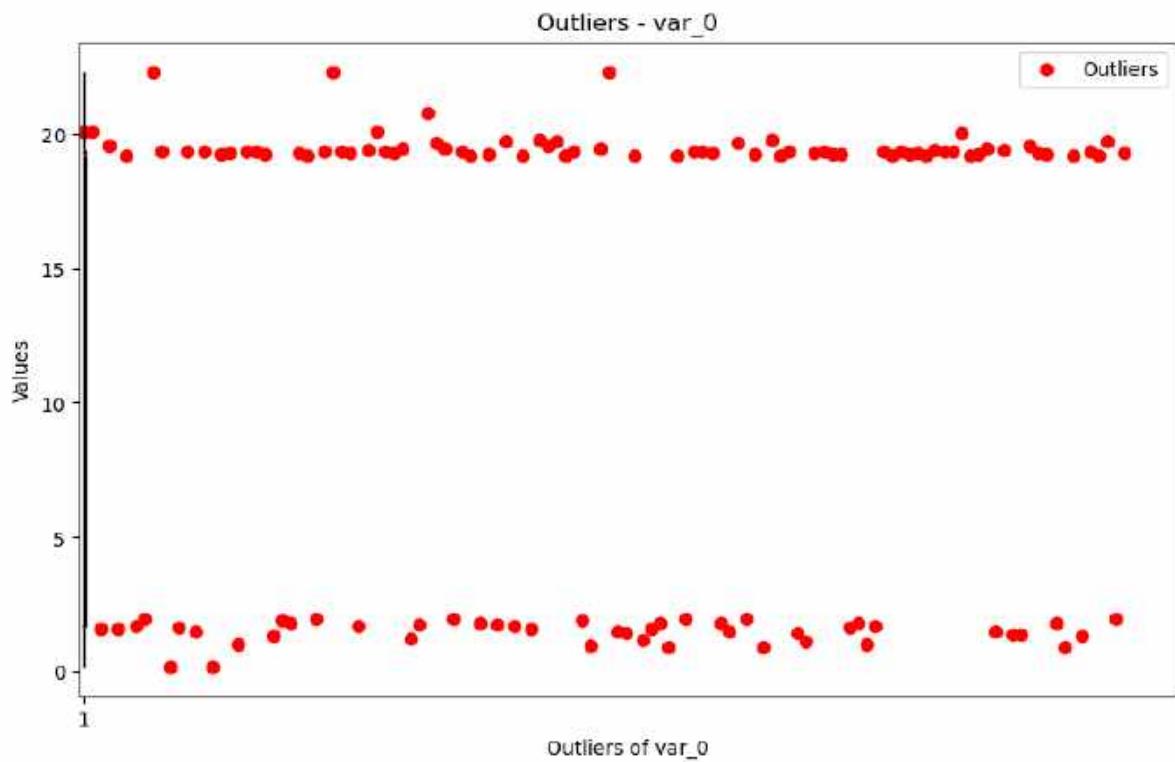
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:
p-value may not be accurate for N > 5000.
warnings.warn("p-value may not be accurate for N > 5000.")
Total outliers in column var_198: 94
```



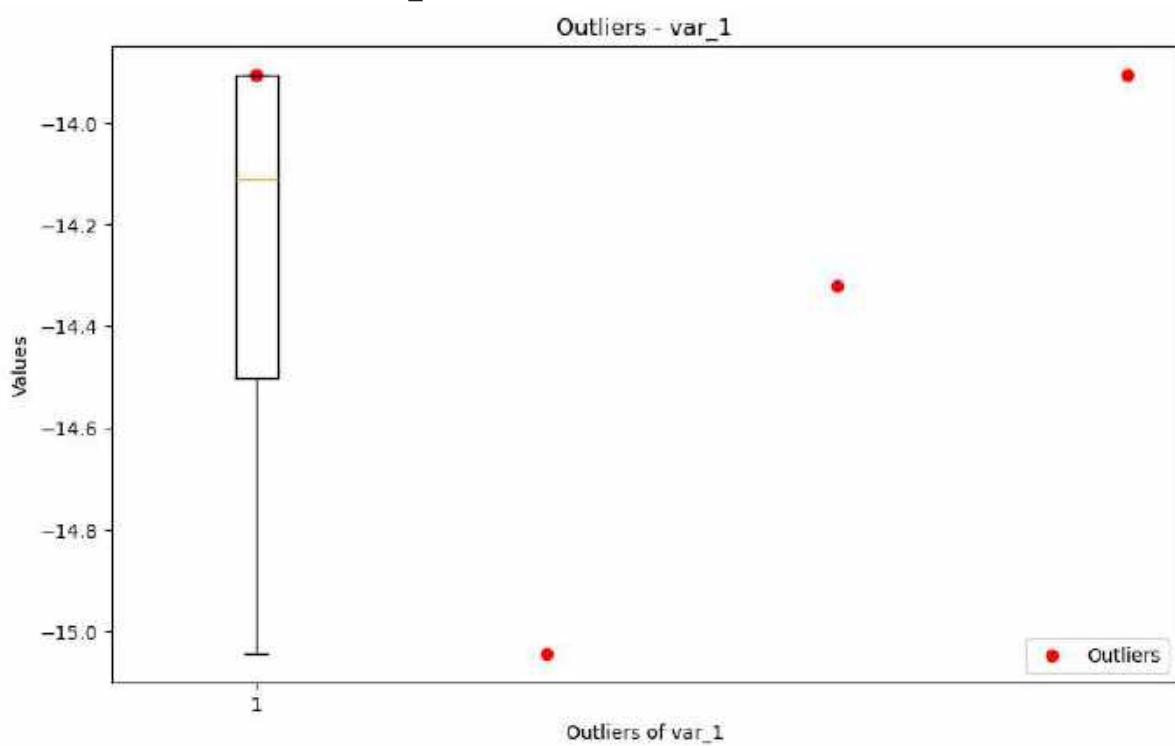
Total outliers in column var\_199: 20

```
In [38]: # Detect outliers for each column in the test set
for column in numeric_columns_test:
    outliers = detect_outliers(data_test_set[column], column)
    visualize_outliers(column, outliers)
```

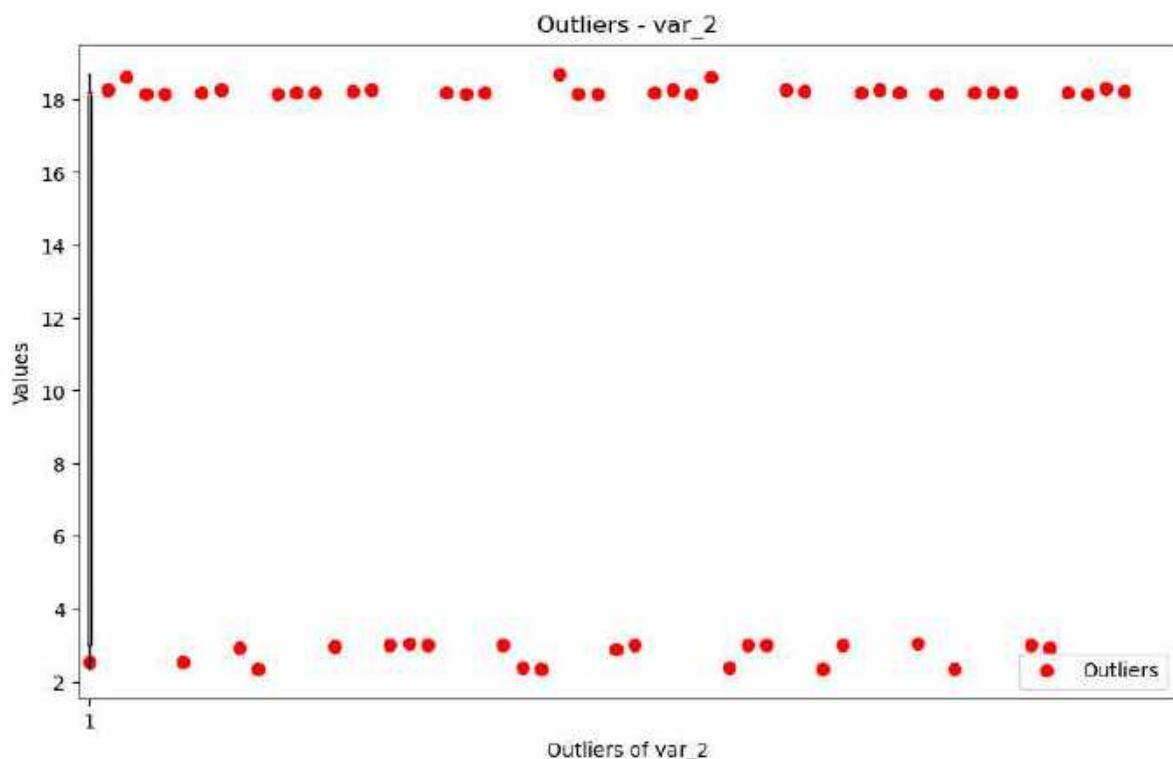
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:
p-value may not be accurate for N > 5000.
warnings.warn("p-value may not be accurate for N > 5000.")
```



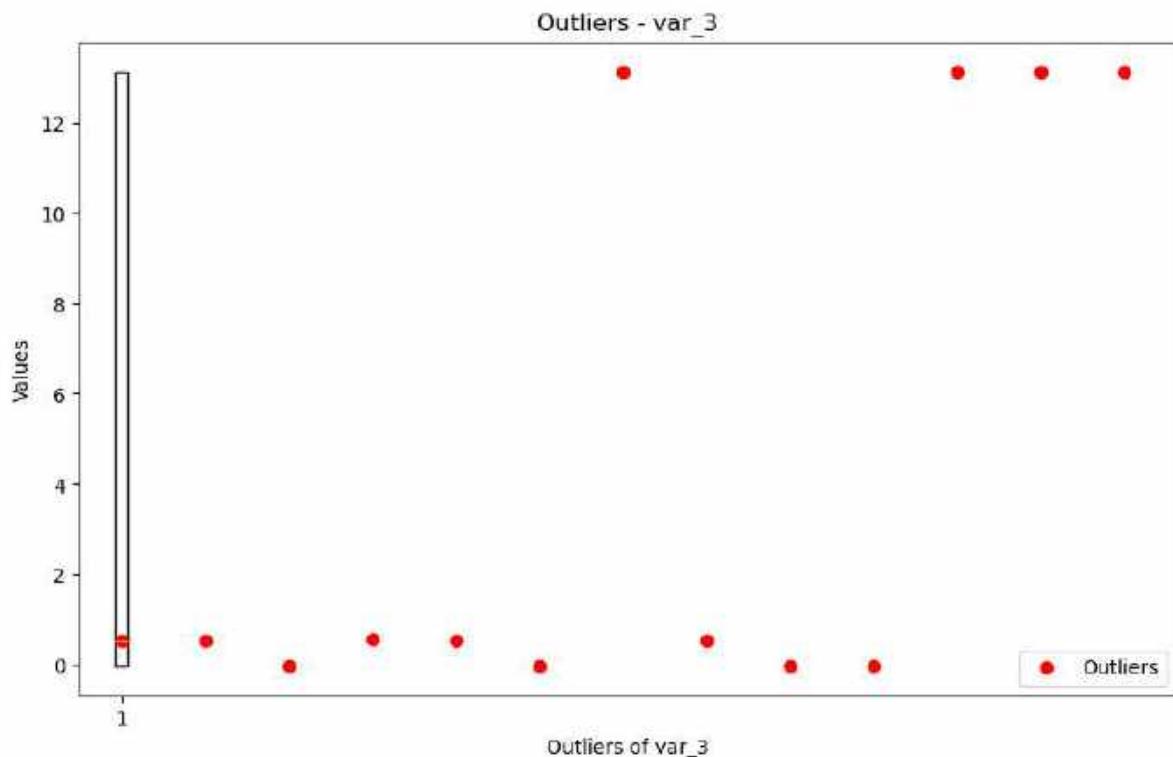
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_0: 122
```



```
Total outliers in column var_1: 4  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

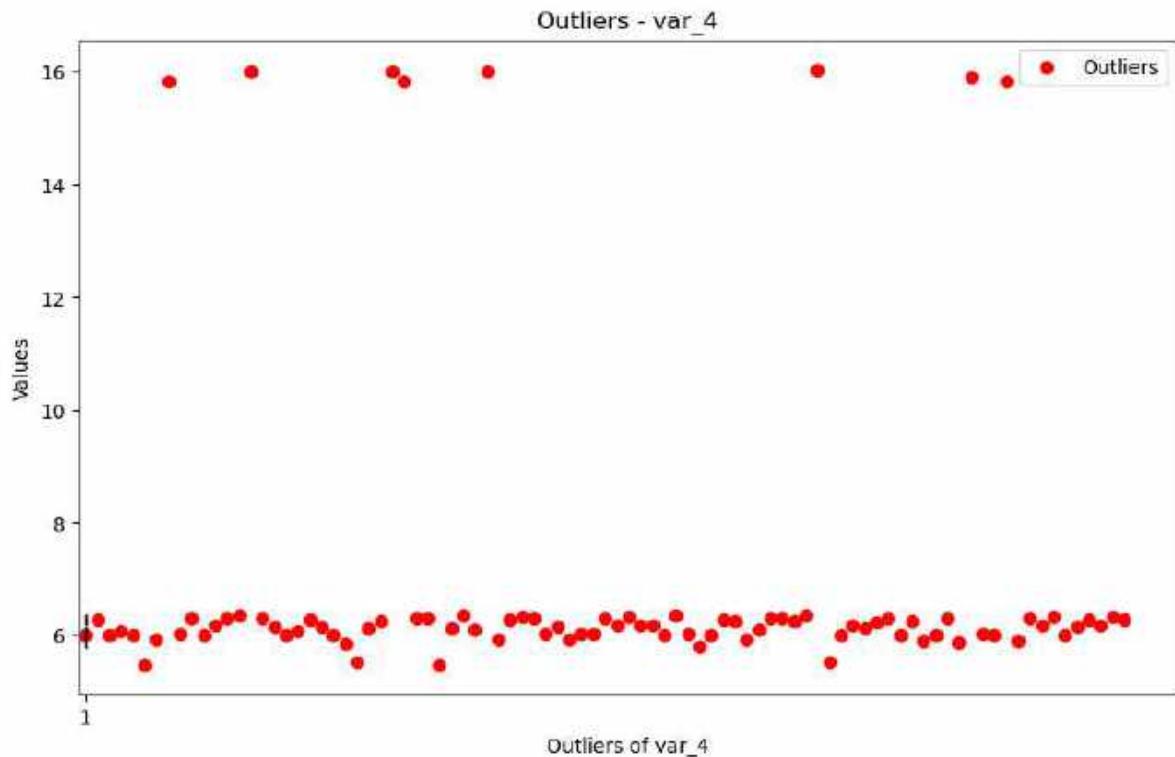


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_2: 56
```



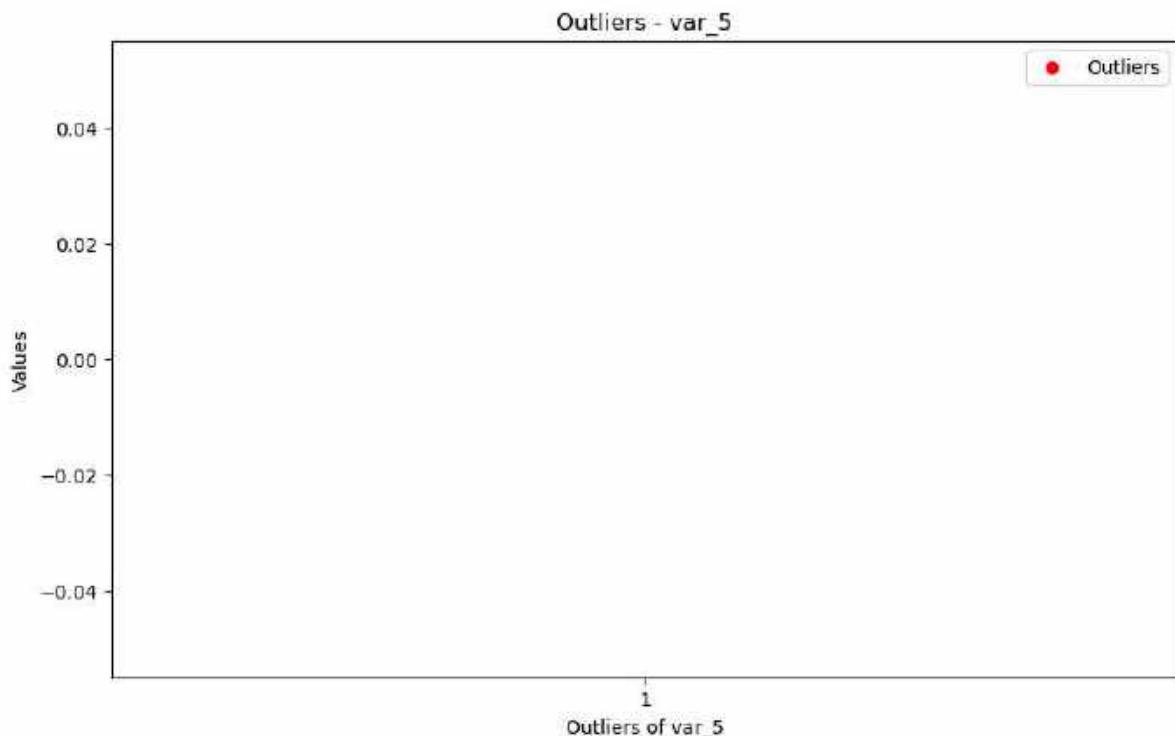
```
Total outliers in column var_3: 13
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



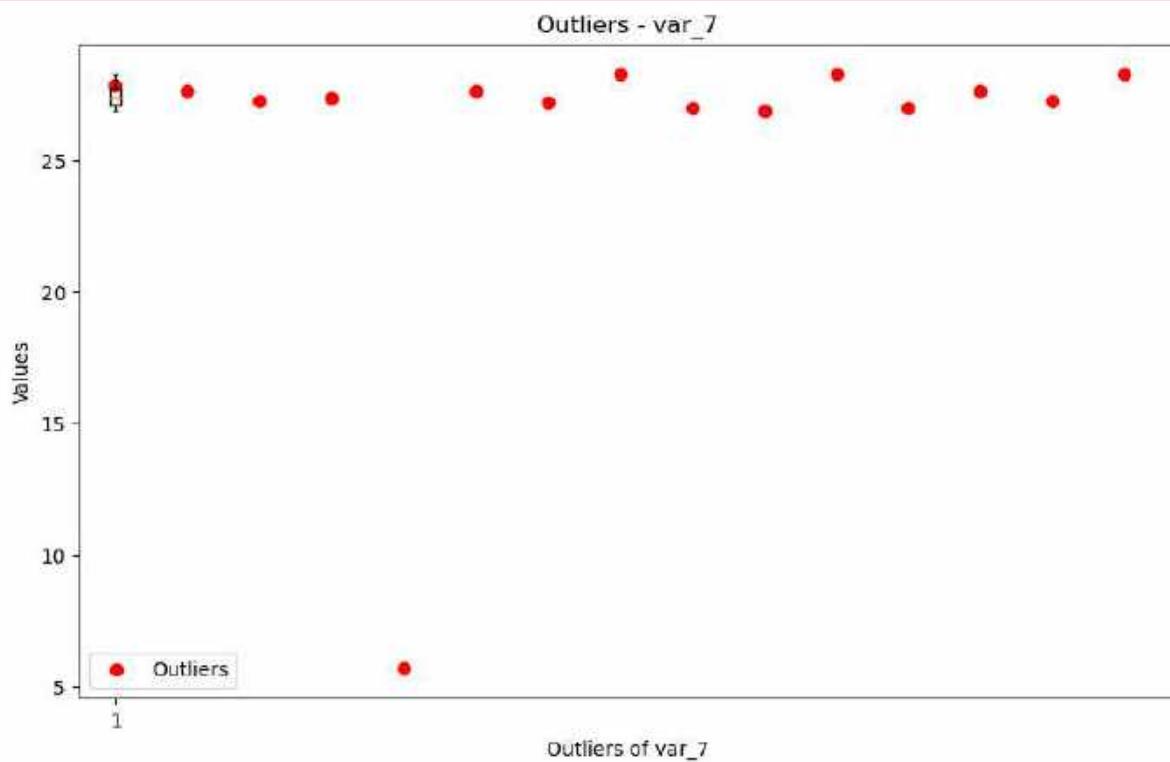
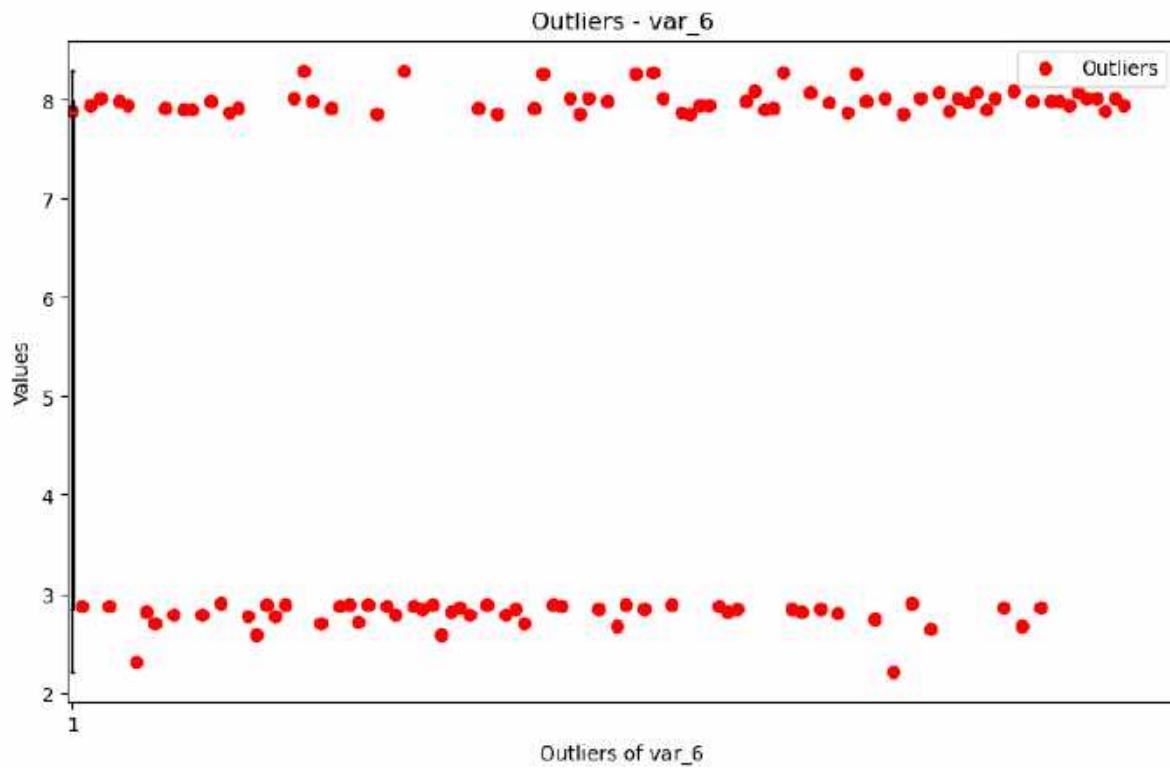
Total outliers in column var\_4: 89

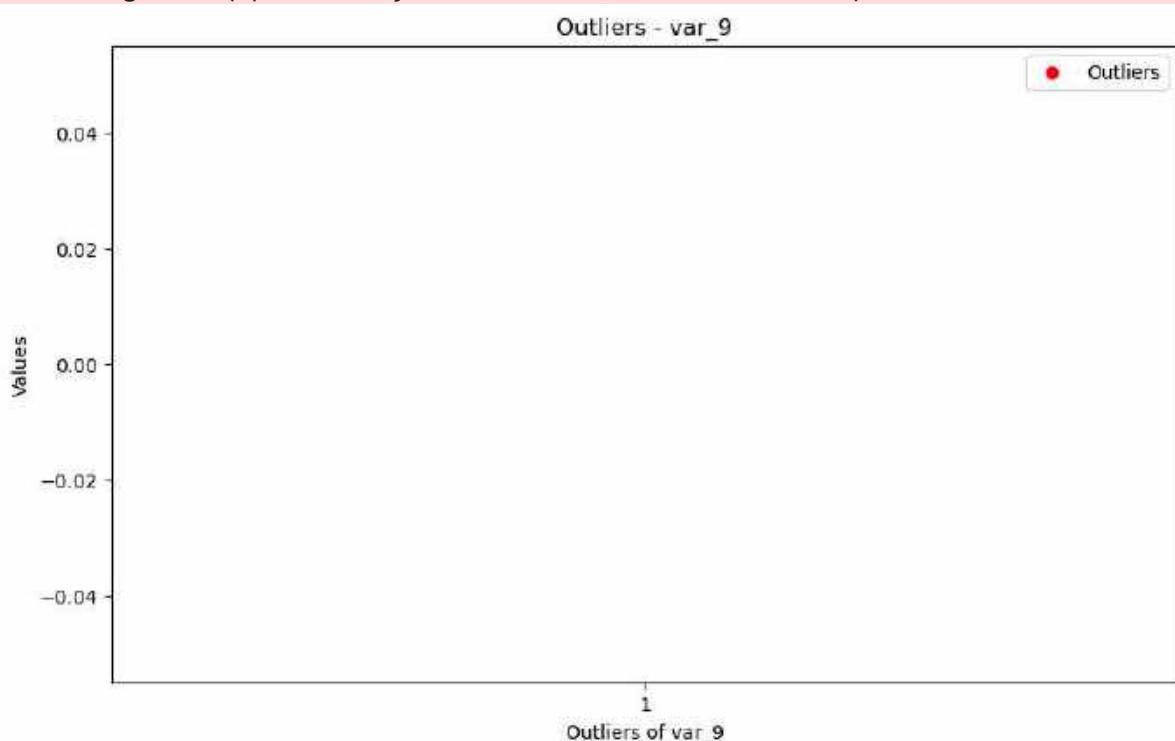
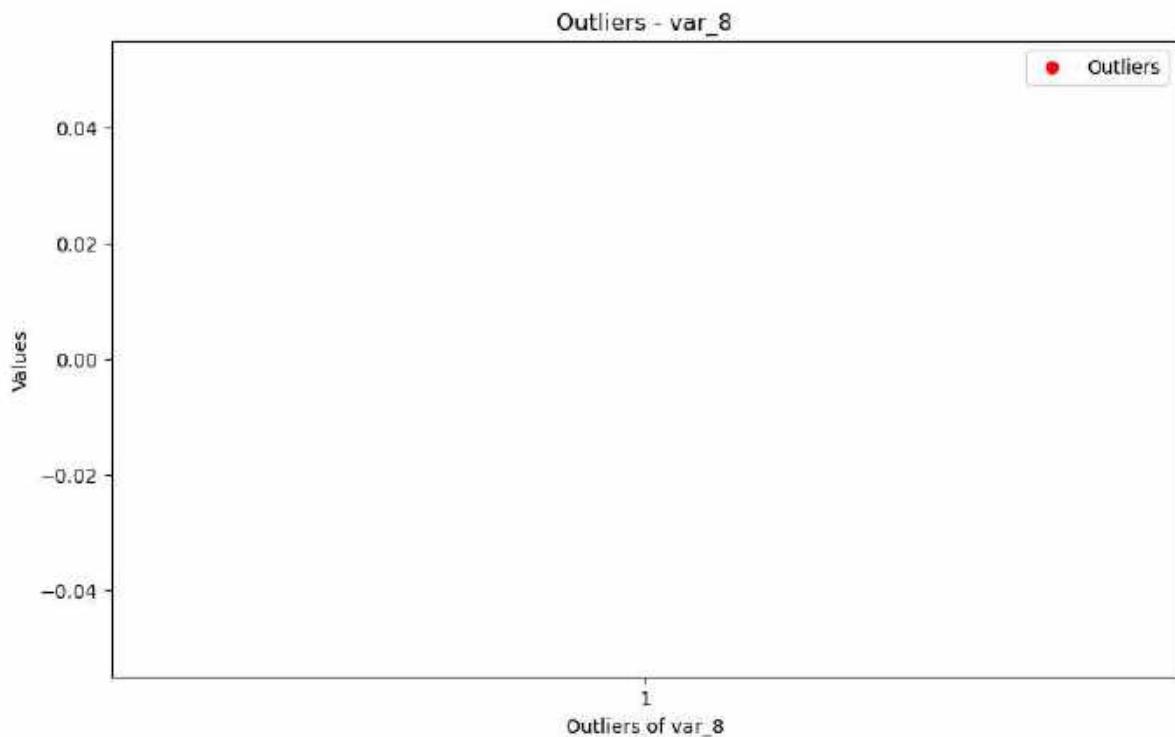
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

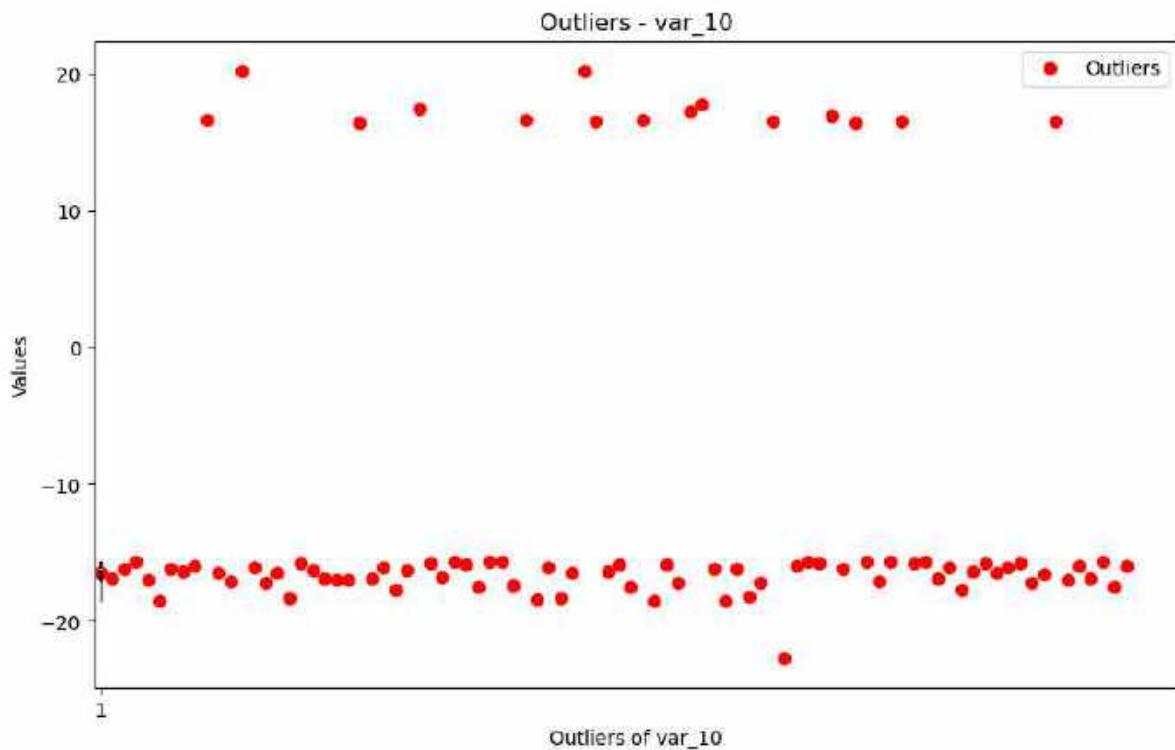


Total outliers in column var\_5: 0

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

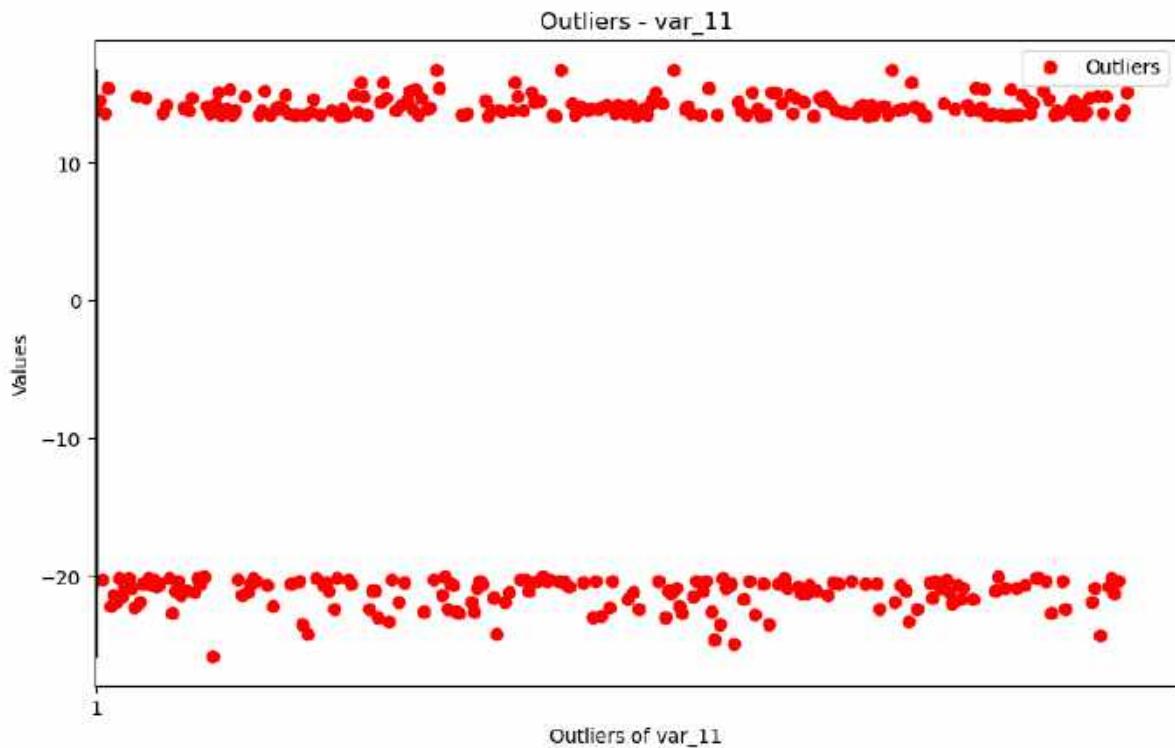






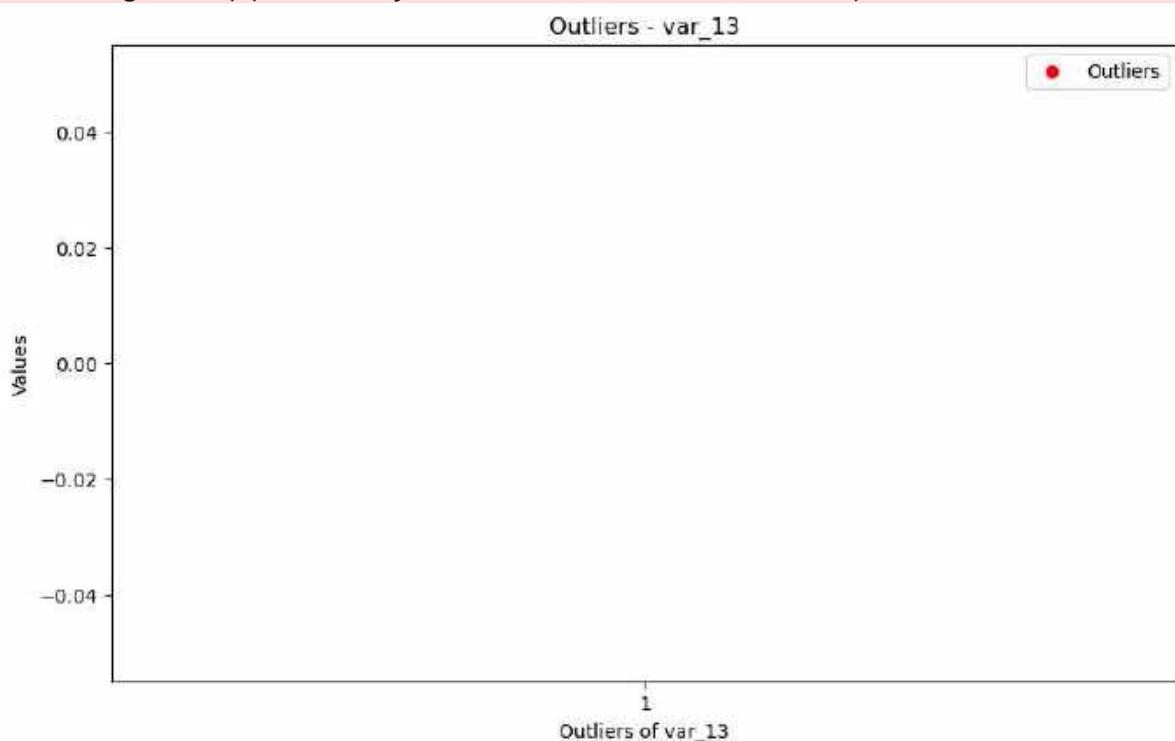
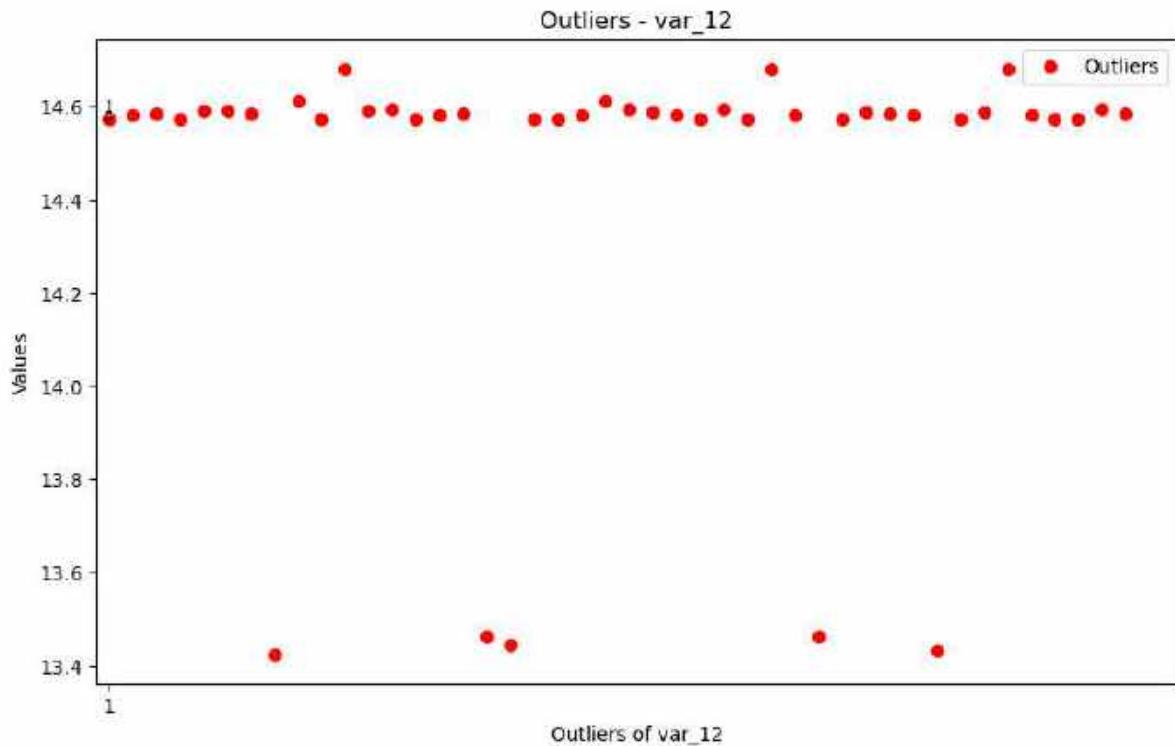
Total outliers in column var\_10: 88

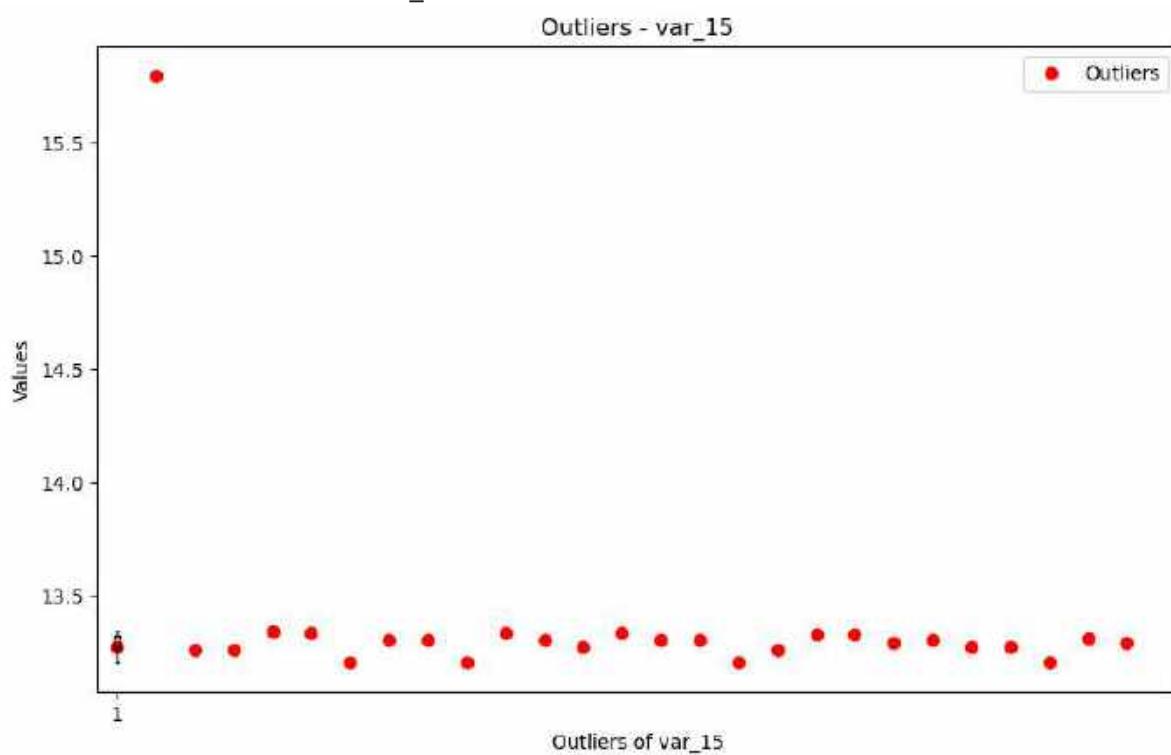
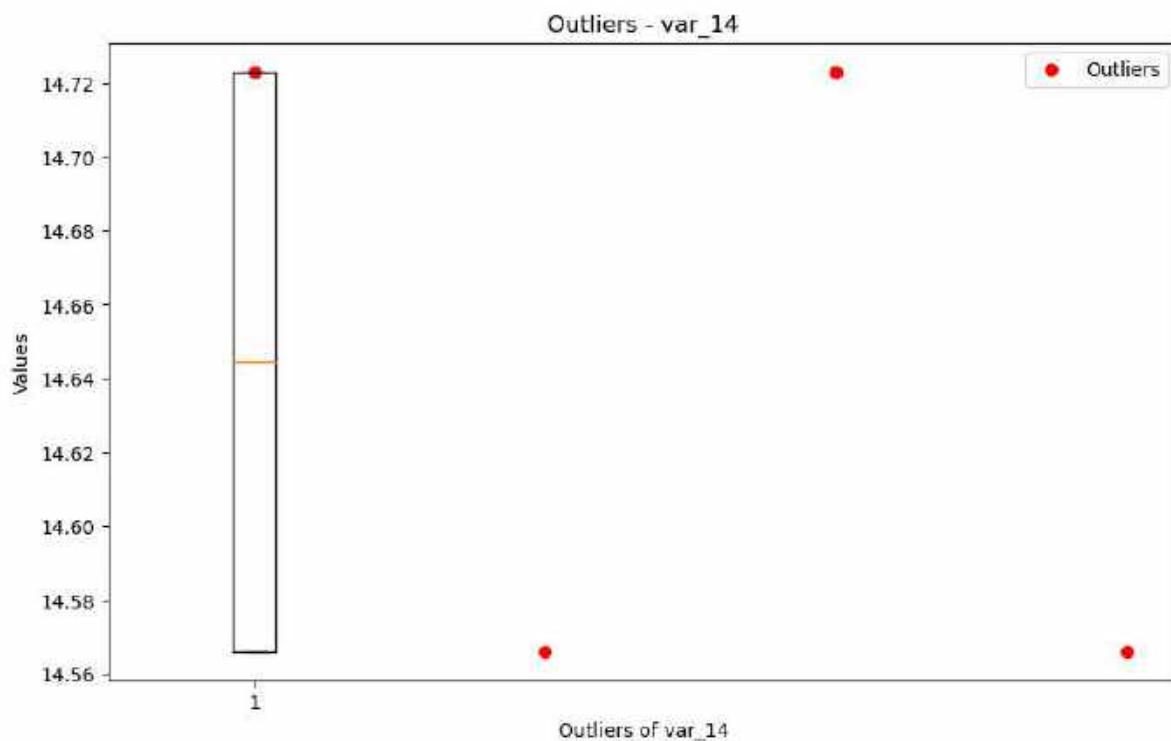
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

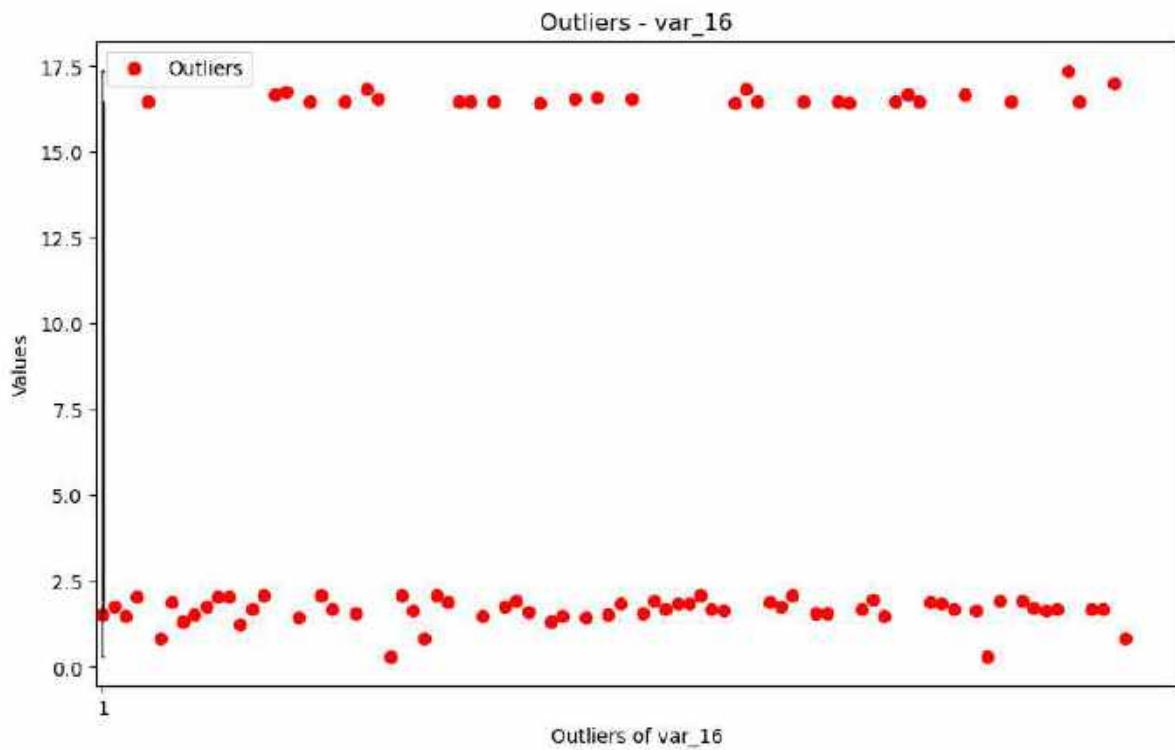


Total outliers in column var\_11: 356

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

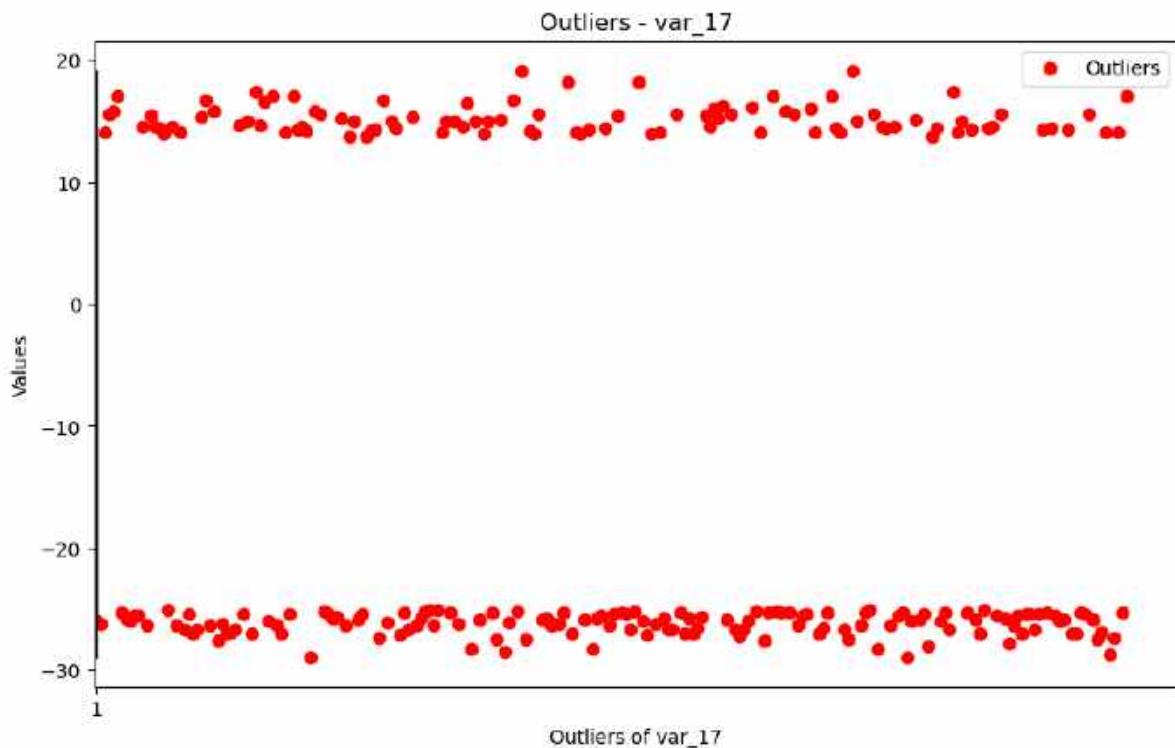






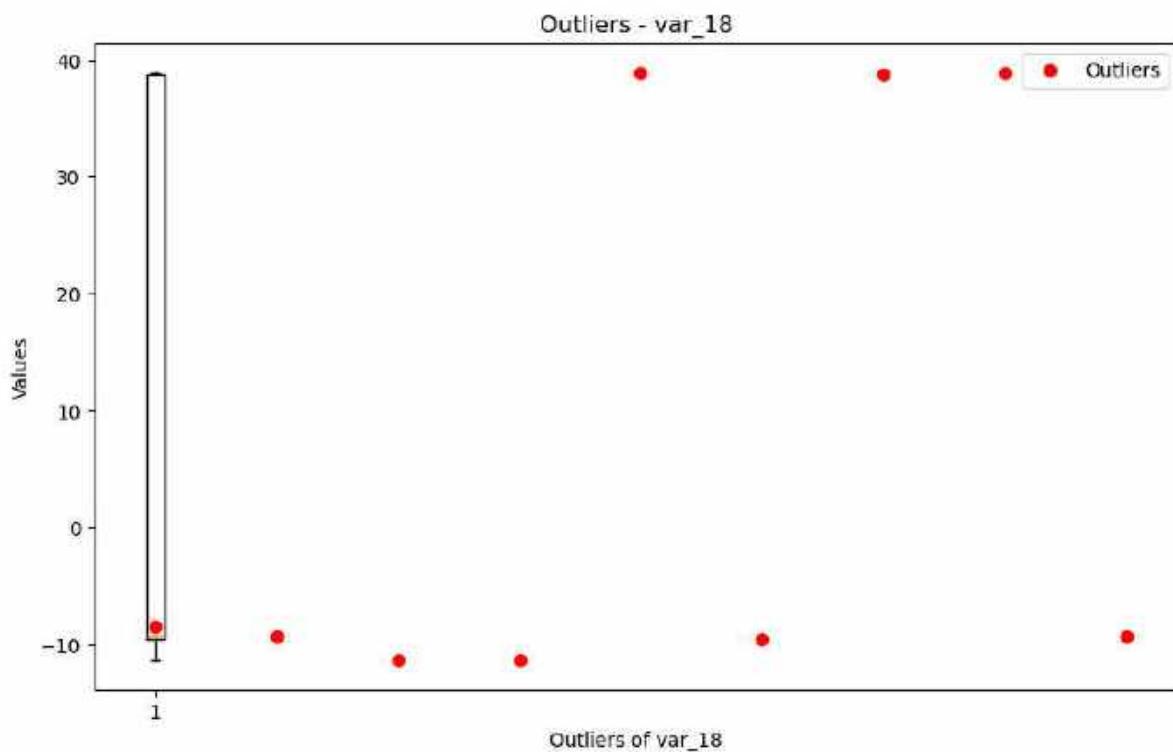
Total outliers in column var\_16: 90

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



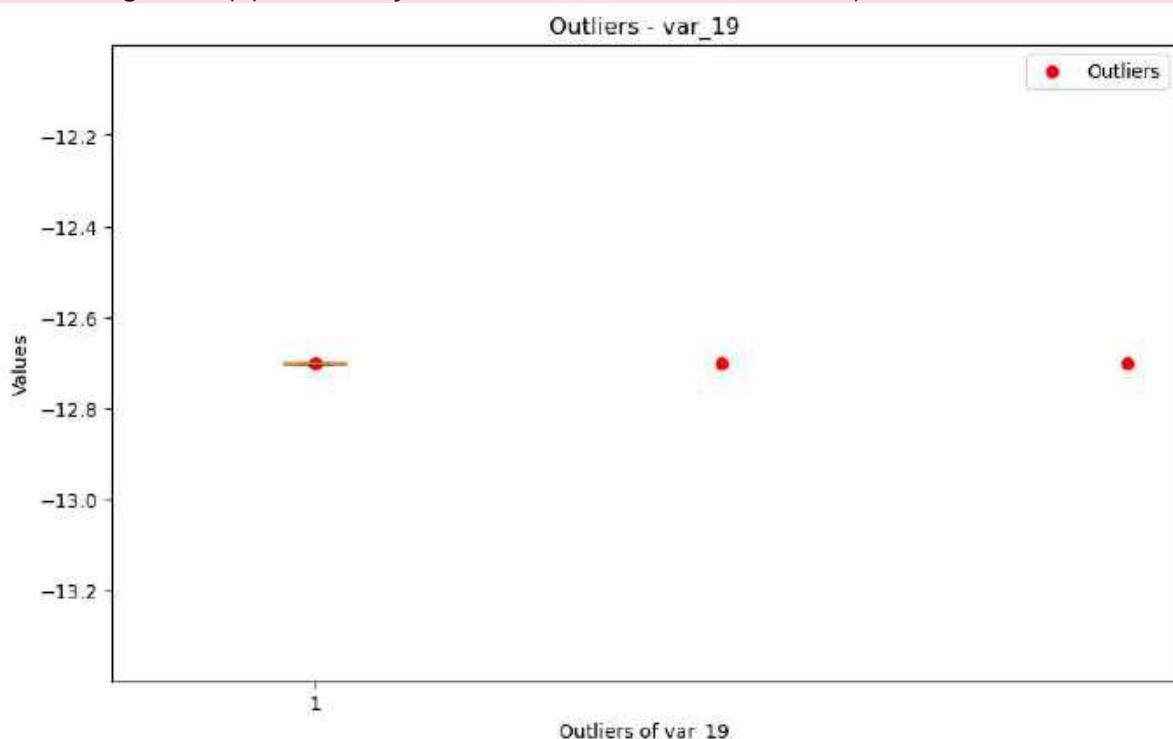
Total outliers in column var\_17: 246

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



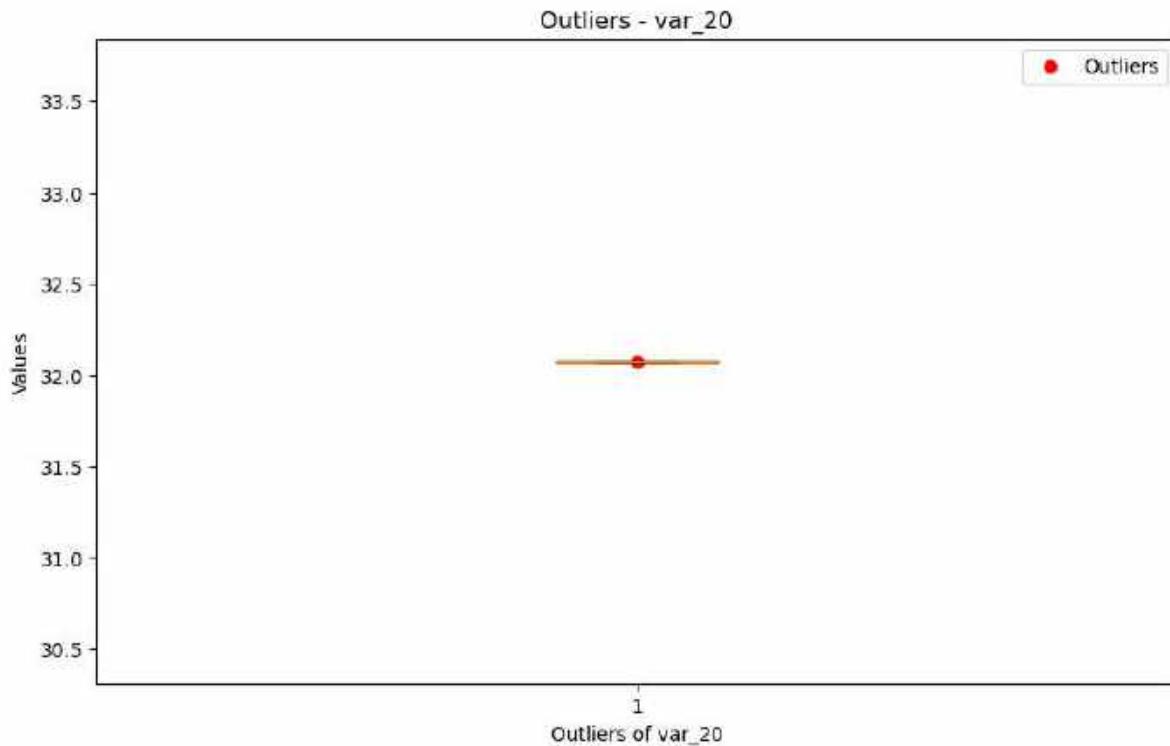
Total outliers in column var\_18: 9

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



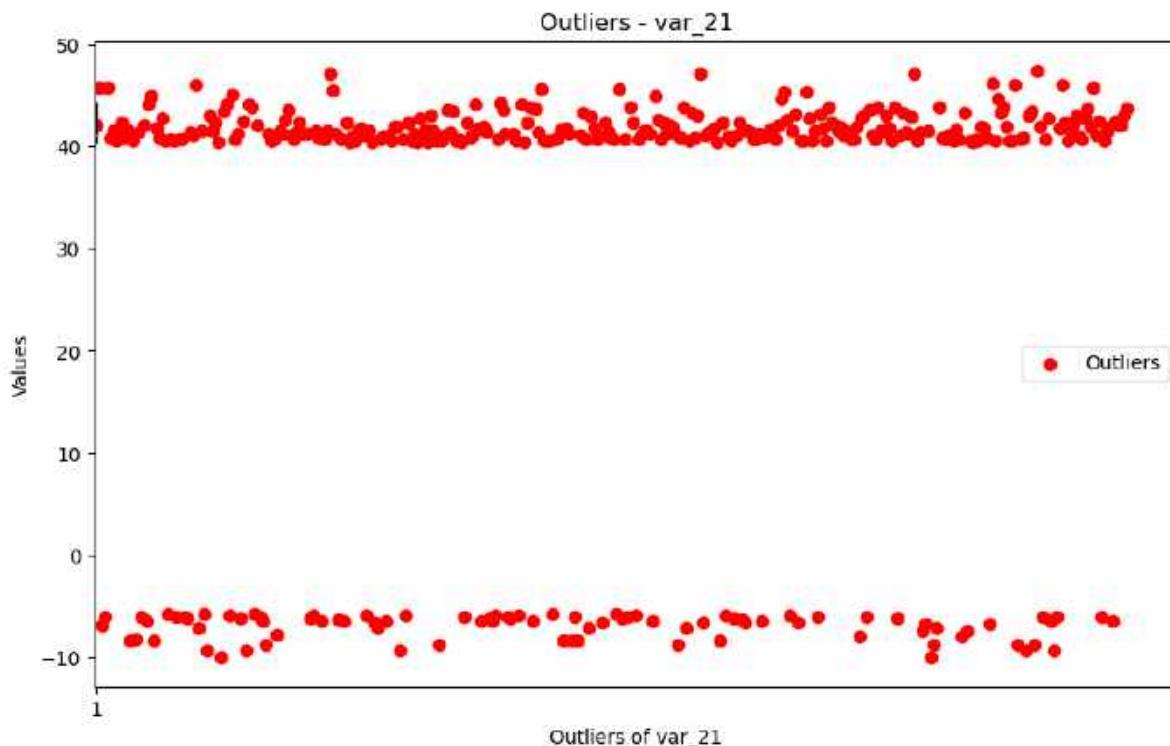
Total outliers in column var\_19: 3

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



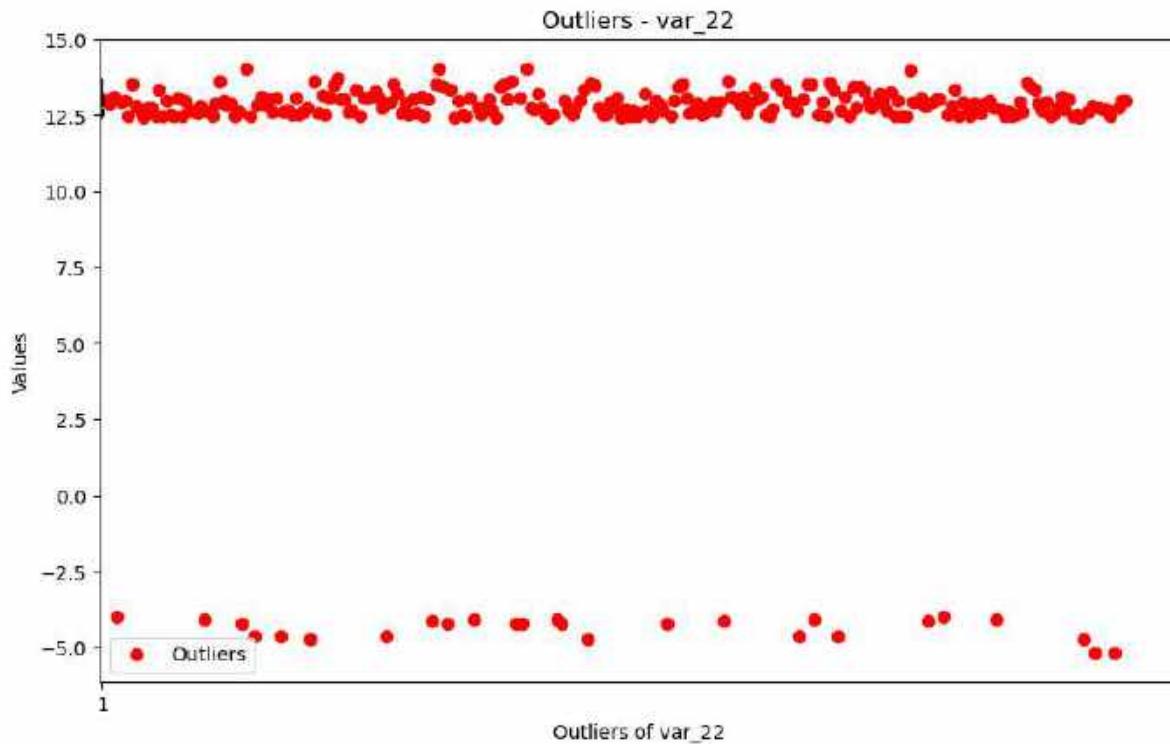
Total outliers in column var\_20: 1

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



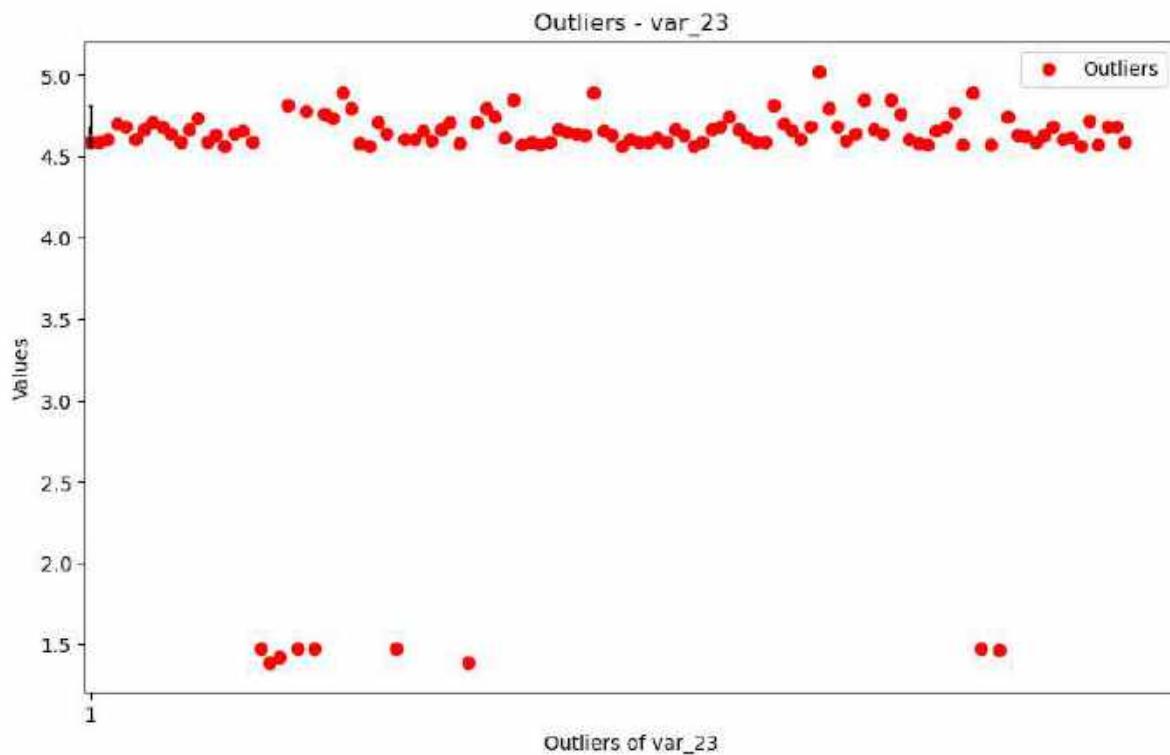
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_21: 371



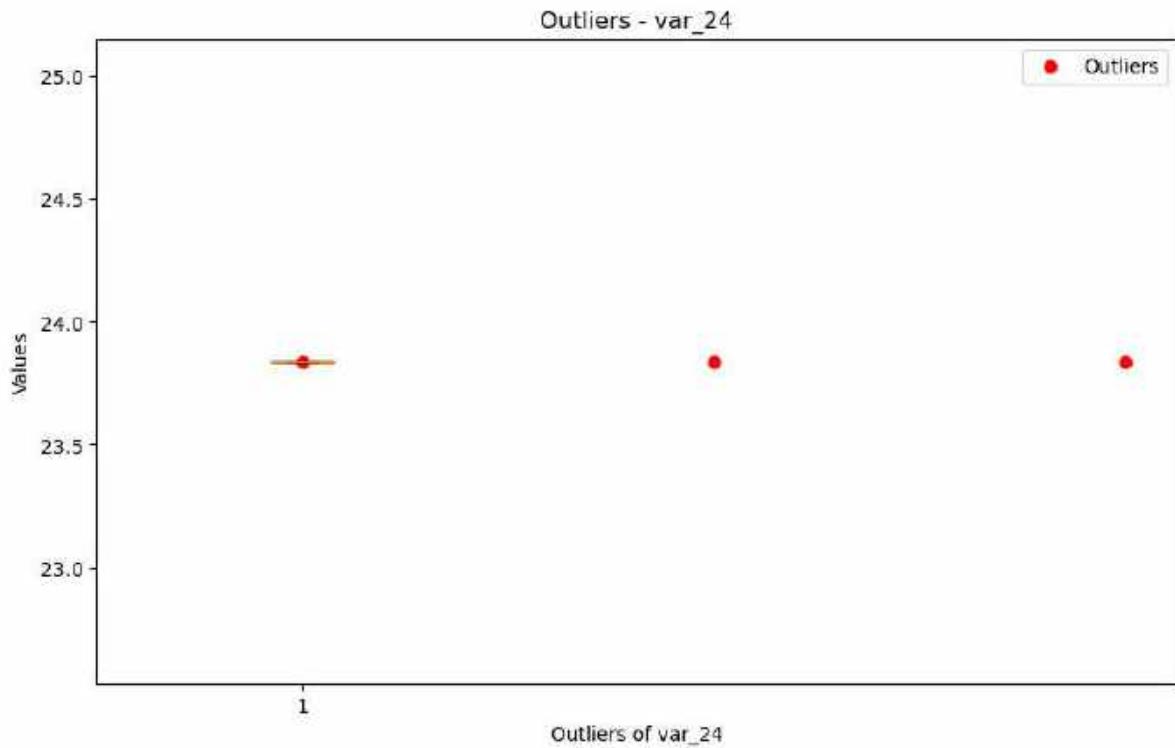
Total outliers in column var\_22: 271

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



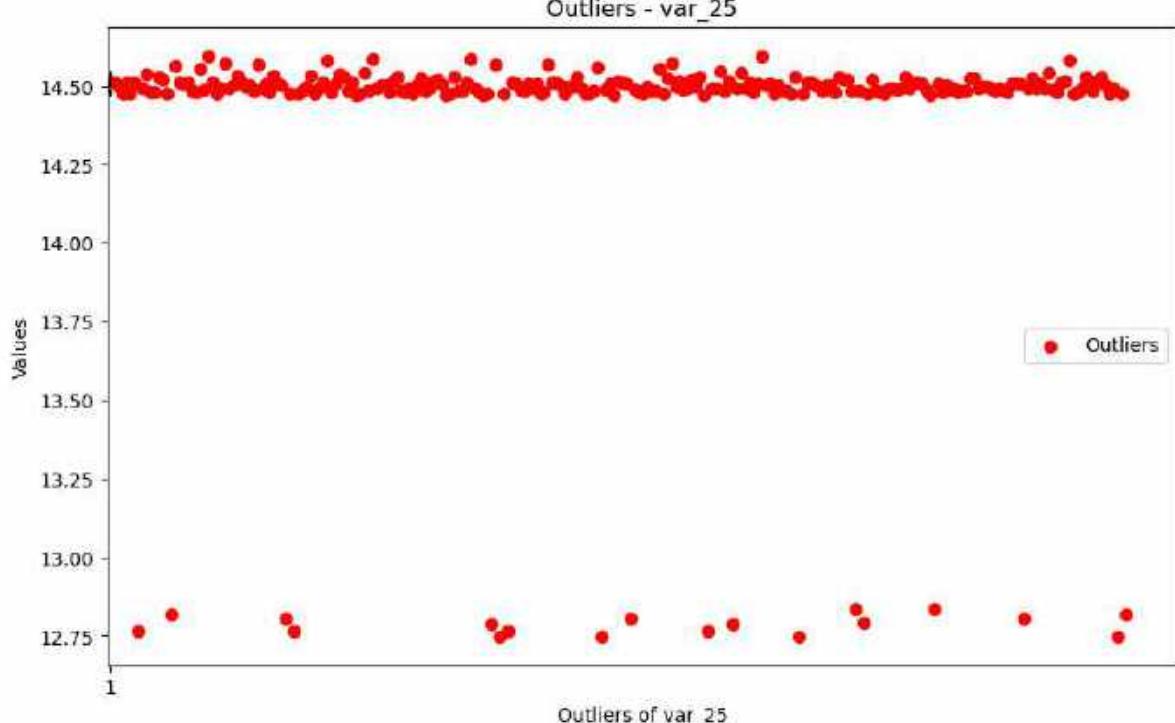
Total outliers in column var\_23: 116

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



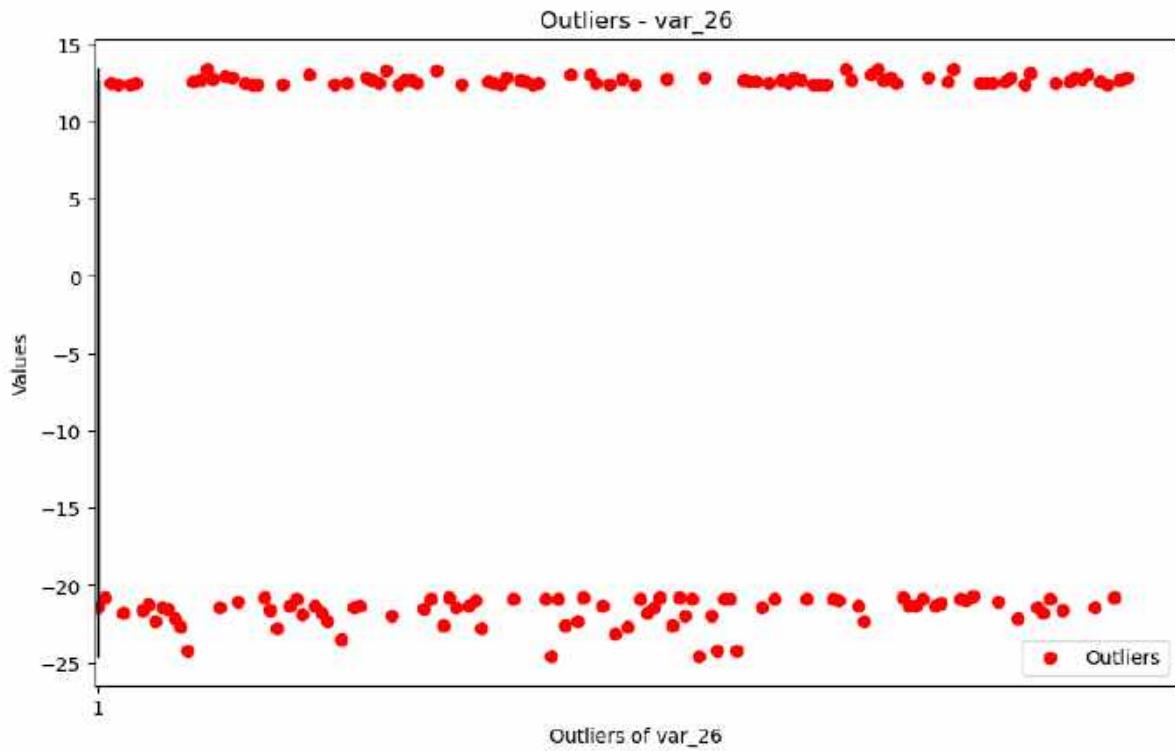
Total outliers in column var\_24: 3

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

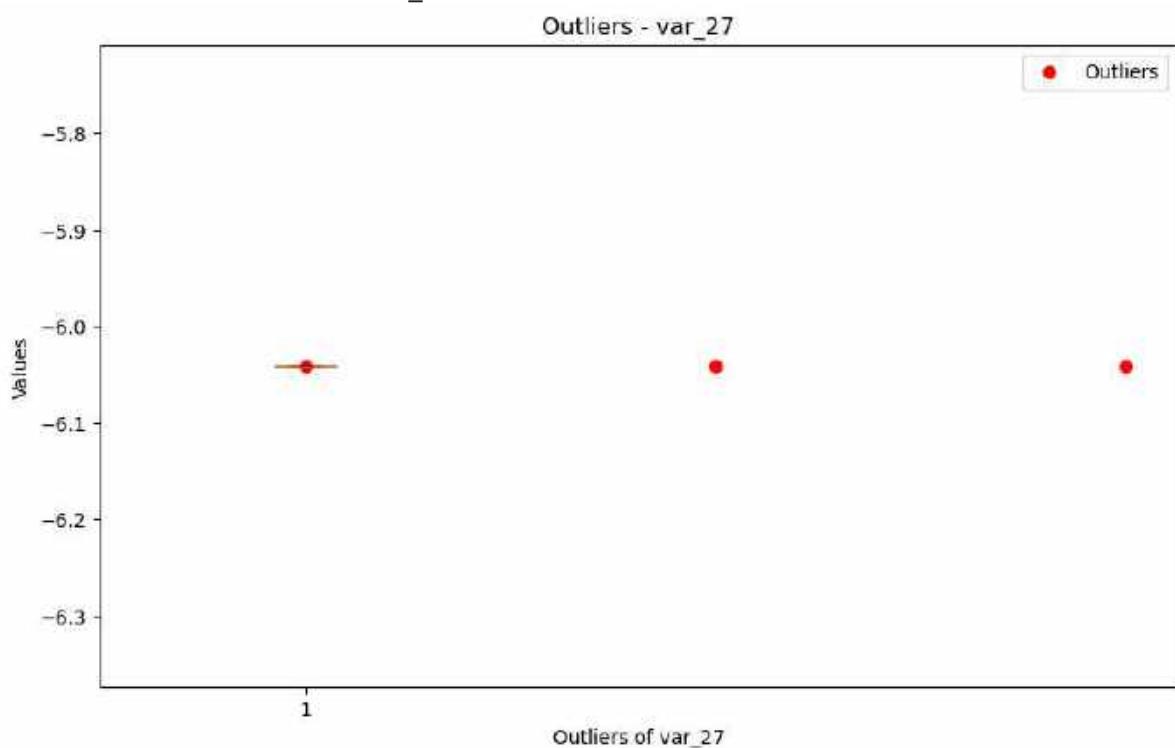


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_25: 249

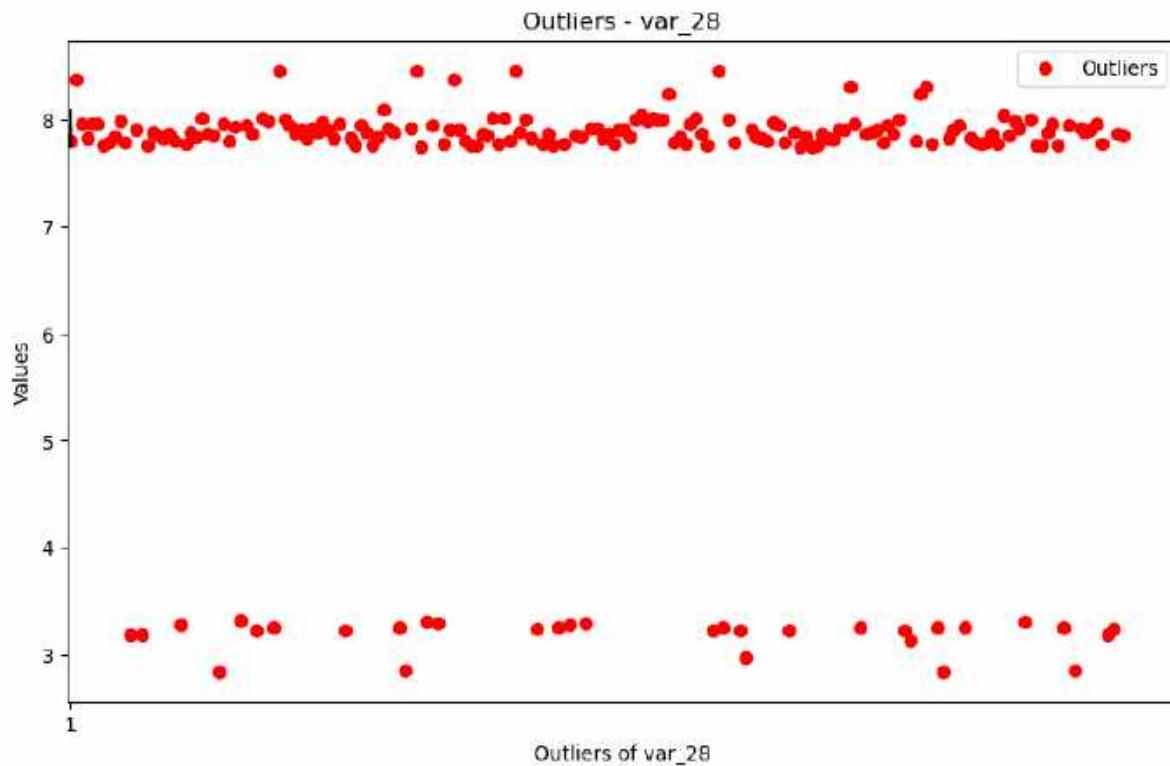


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_26: 162
```



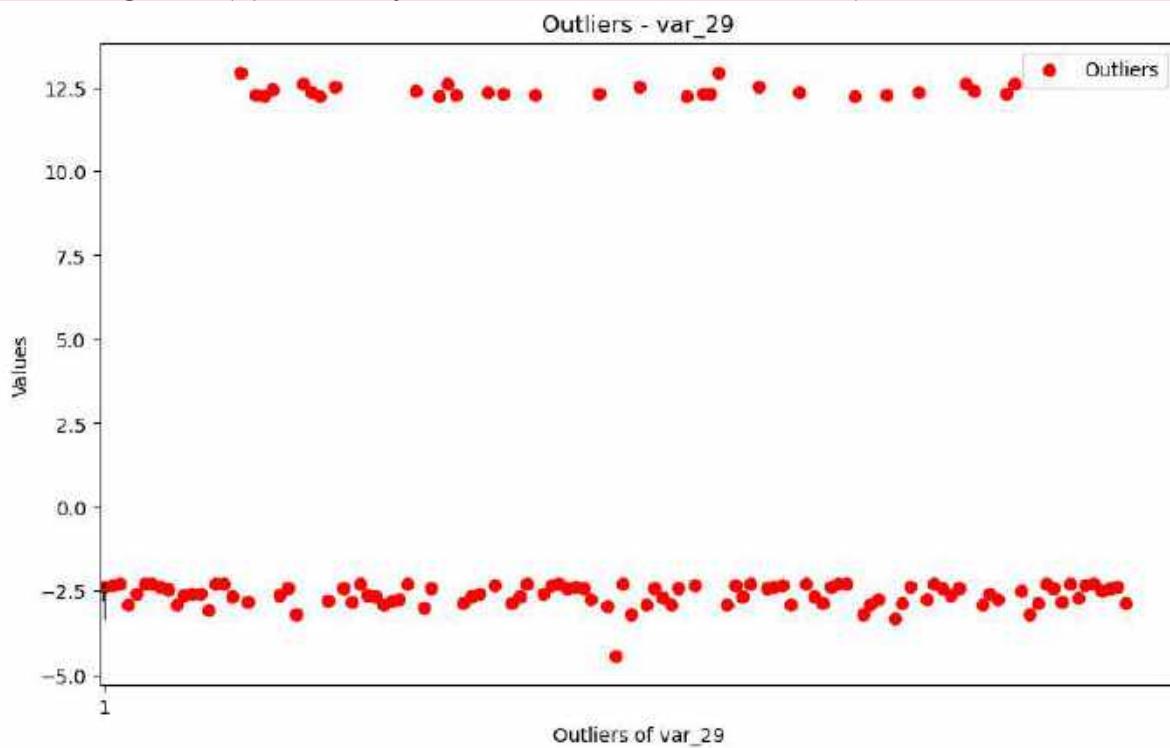
```
Total outliers in column var_27: 3
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



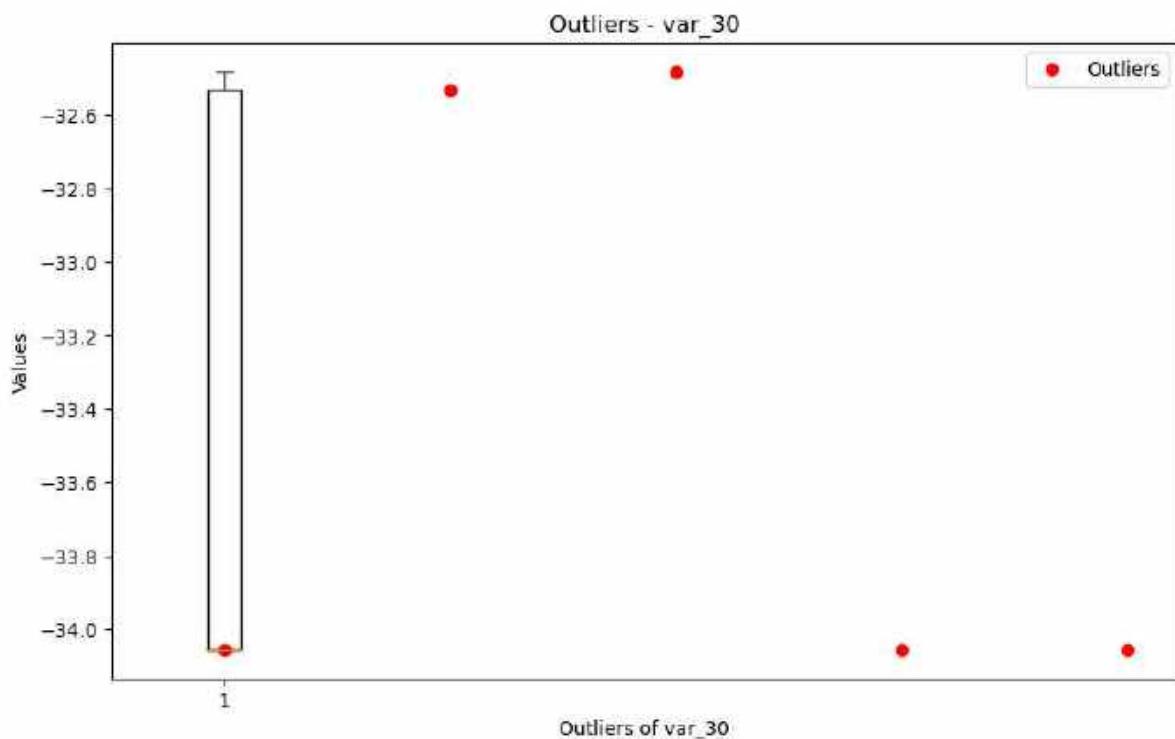
Total outliers in column var\_28: 193

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



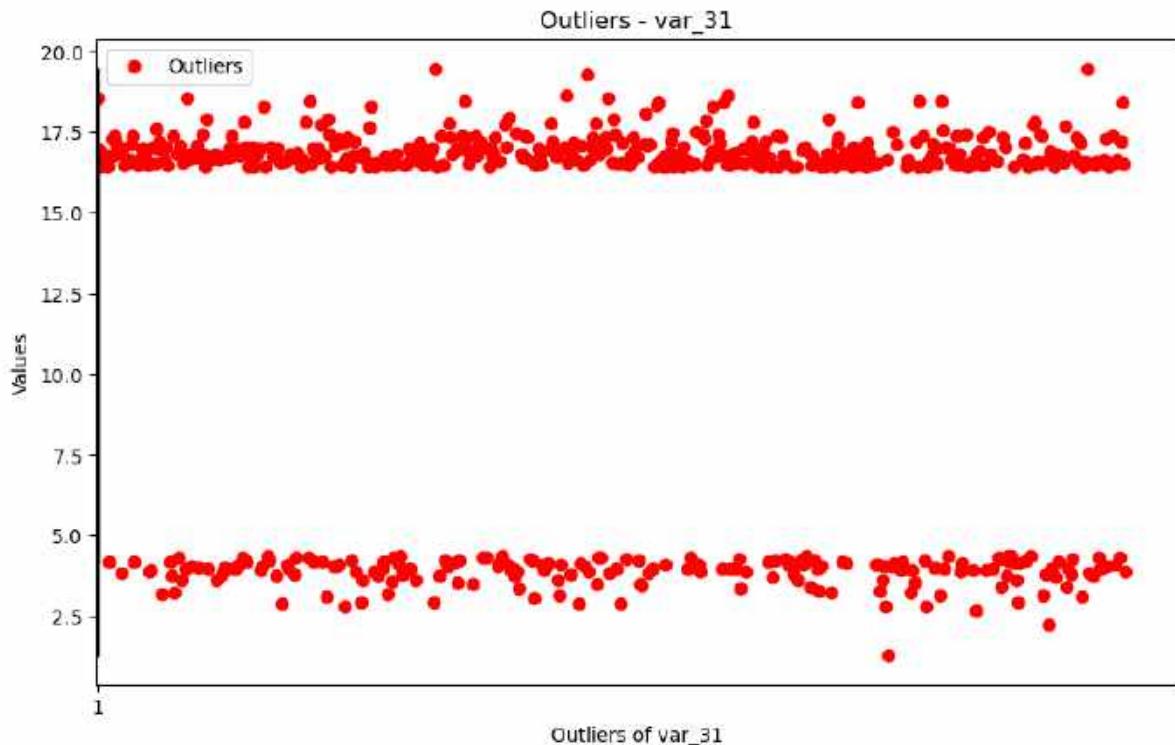
Total outliers in column var\_29: 129

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



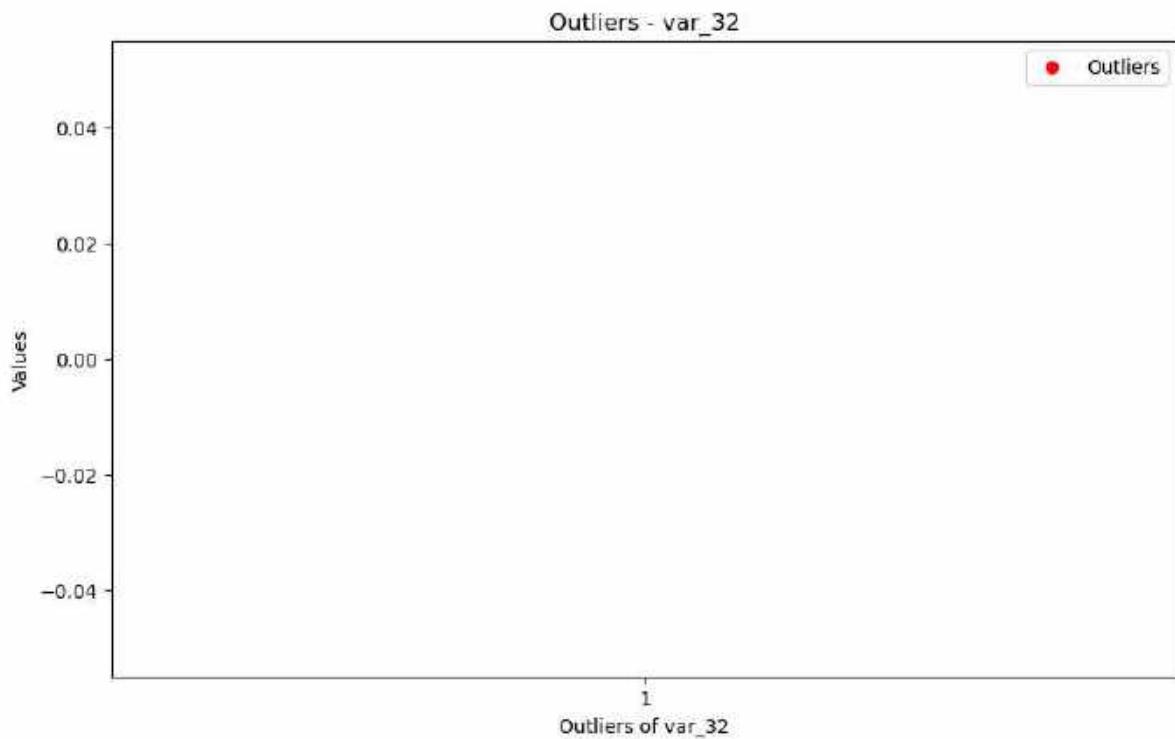
Total outliers in column var\_30: 5

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



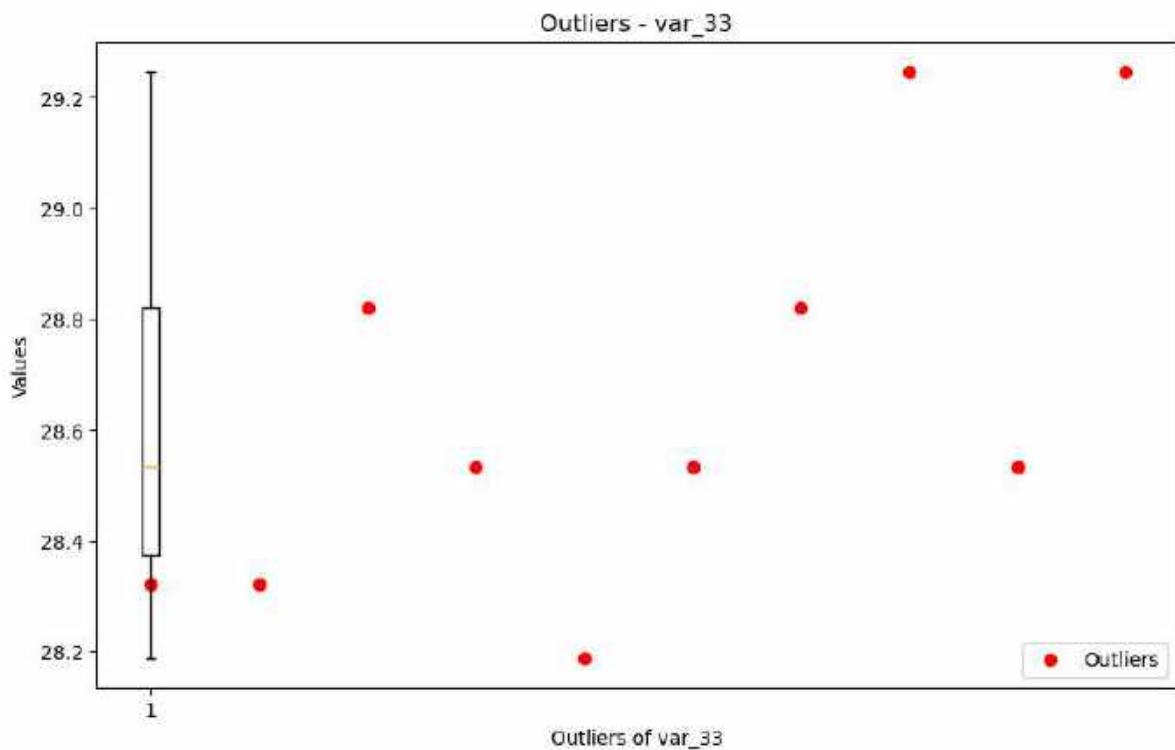
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_31: 613



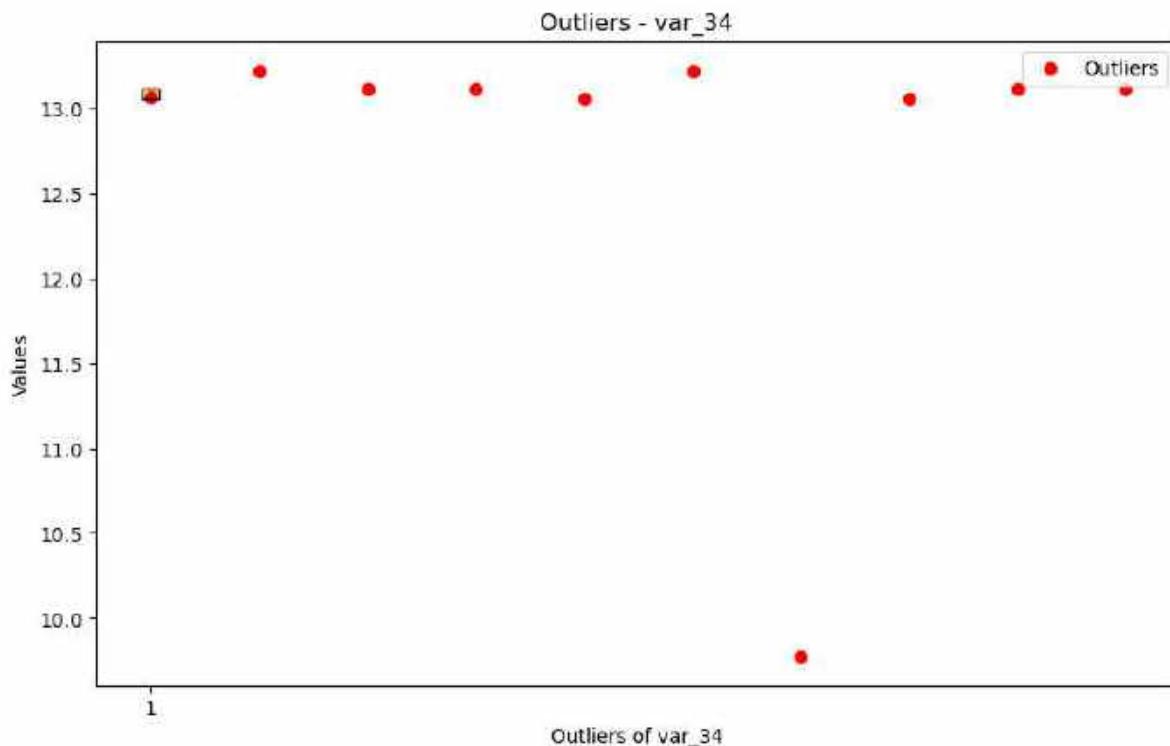
Total outliers in column var\_32: 0

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



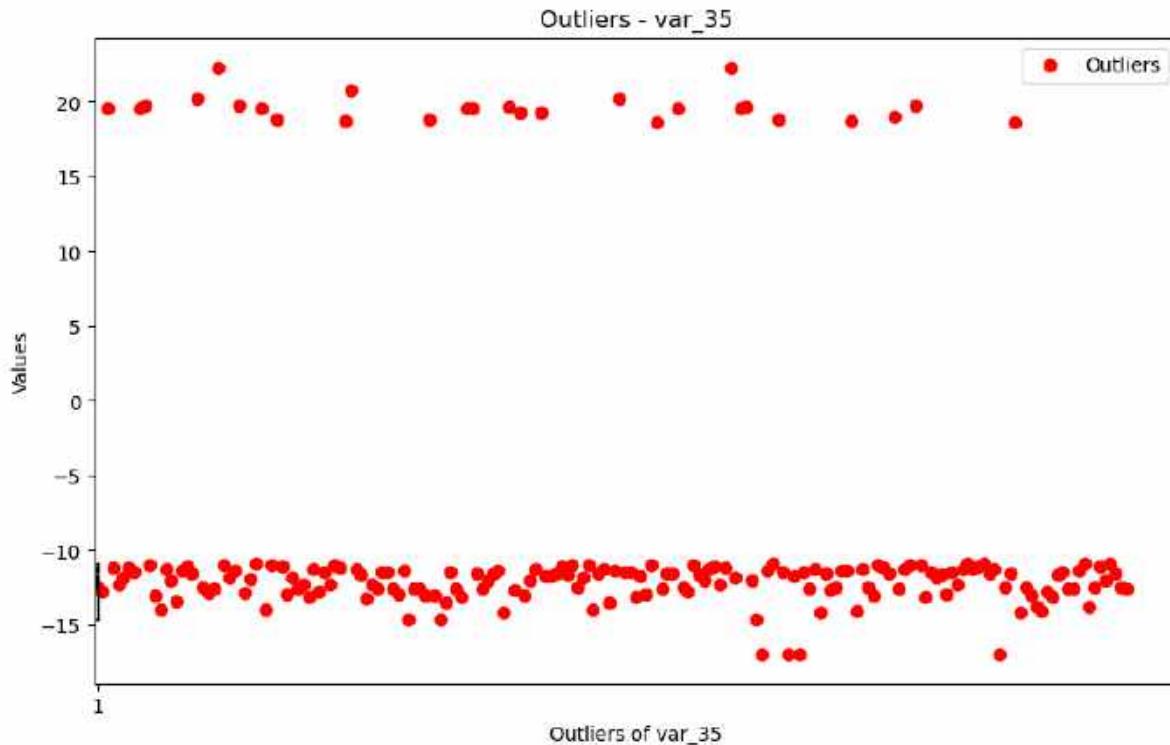
Total outliers in column var\_33: 10

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



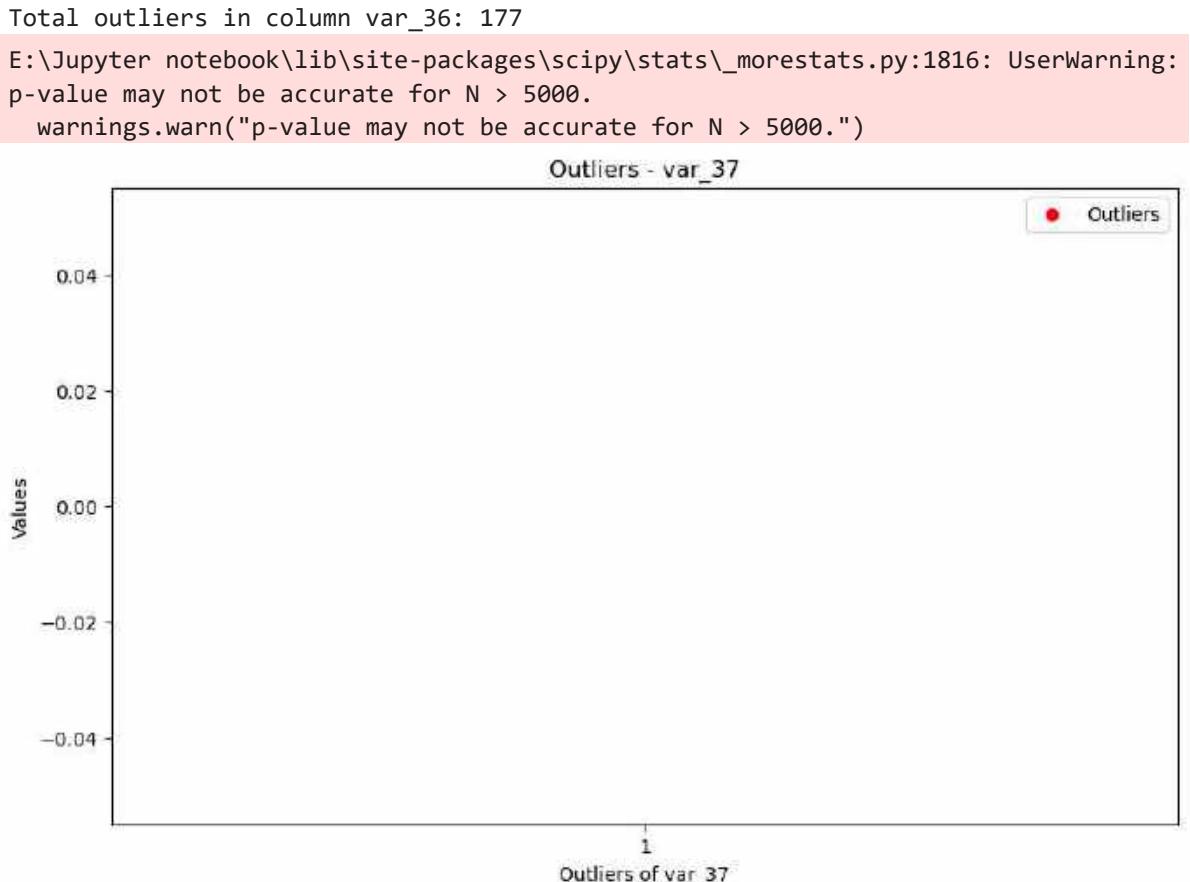
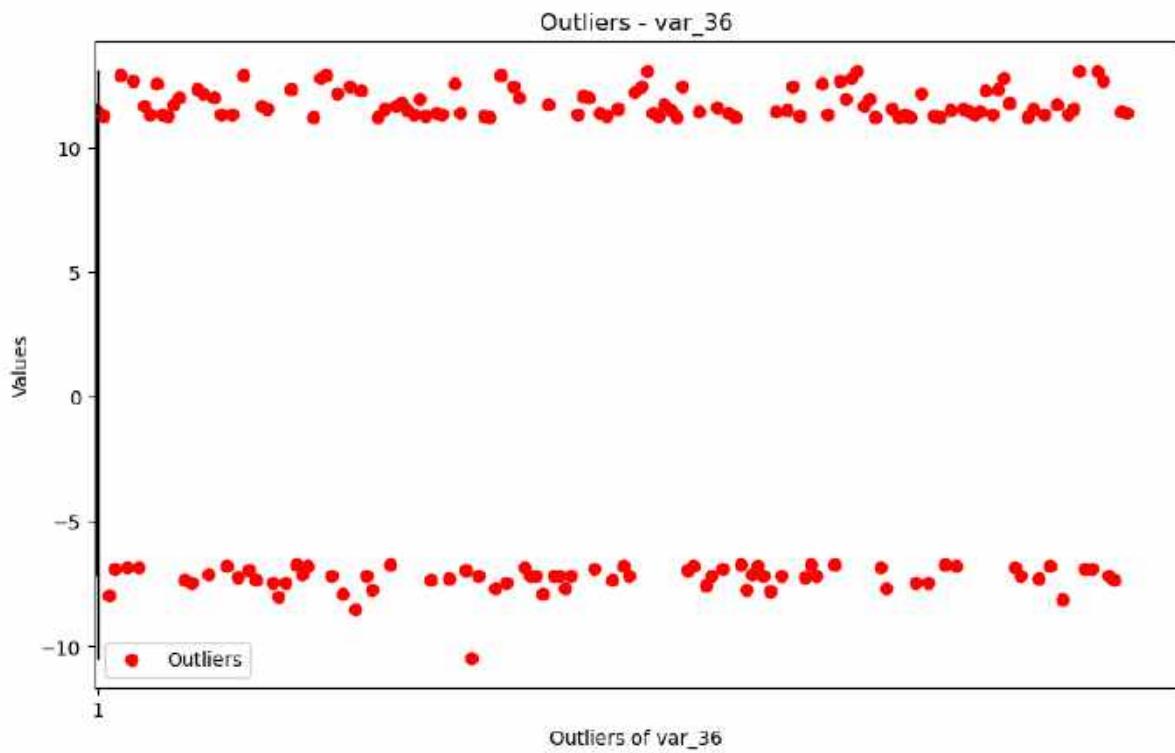
```
Total outliers in column var_34: 10
```

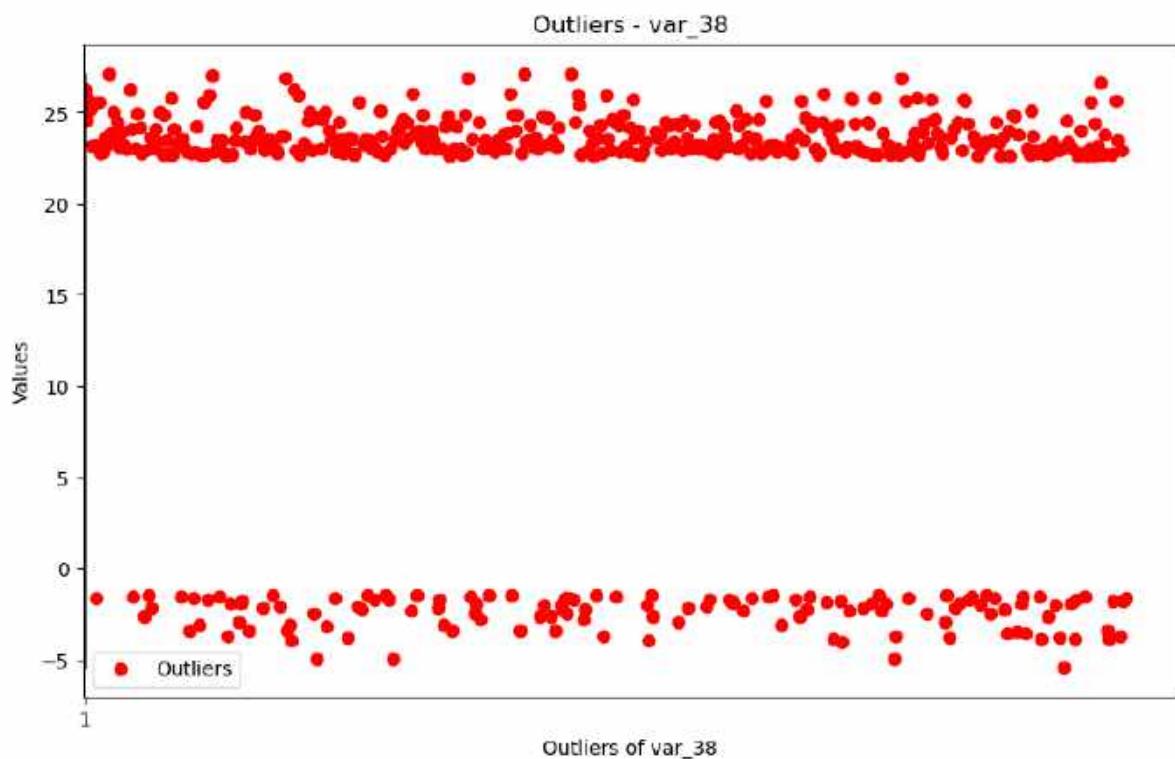
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



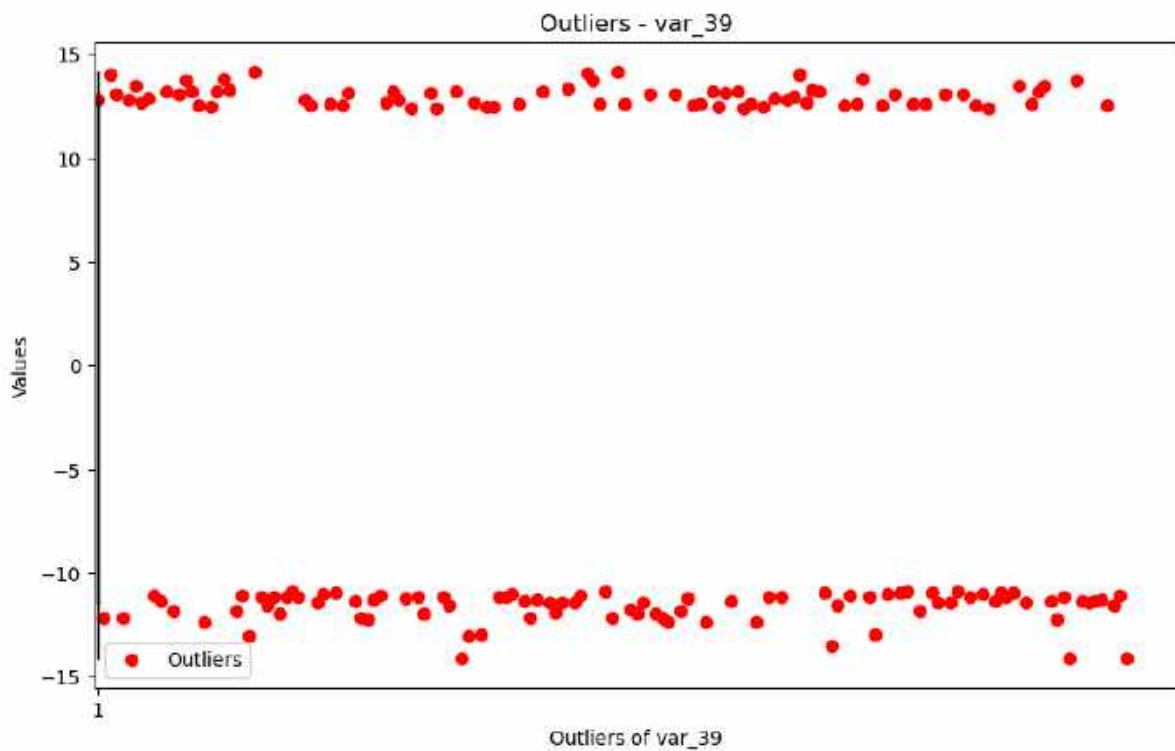
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_35: 196

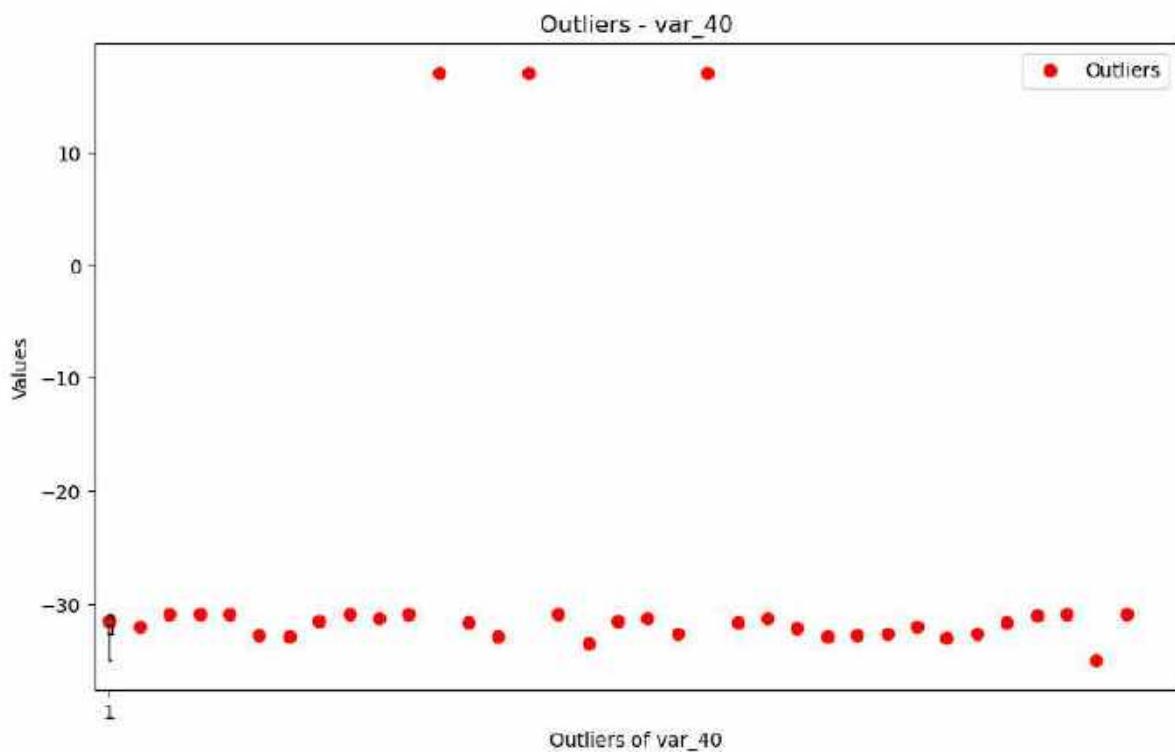




```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_38: 508
```

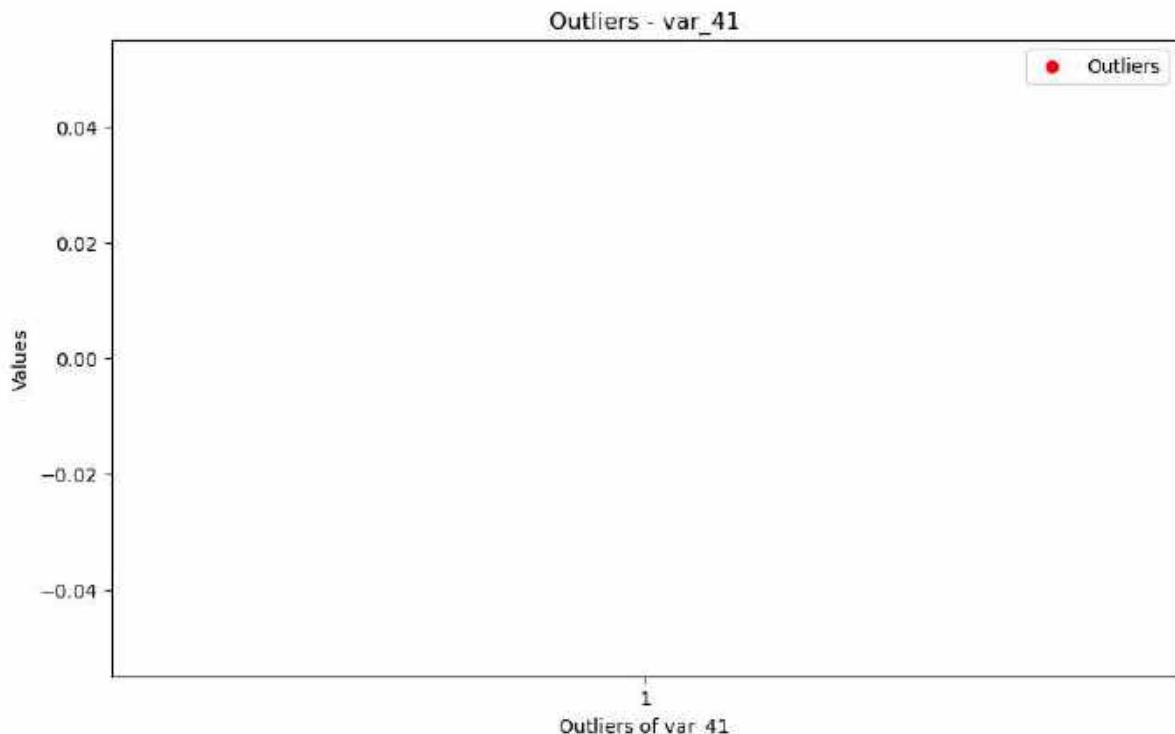


```
Total outliers in column var_39: 165  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



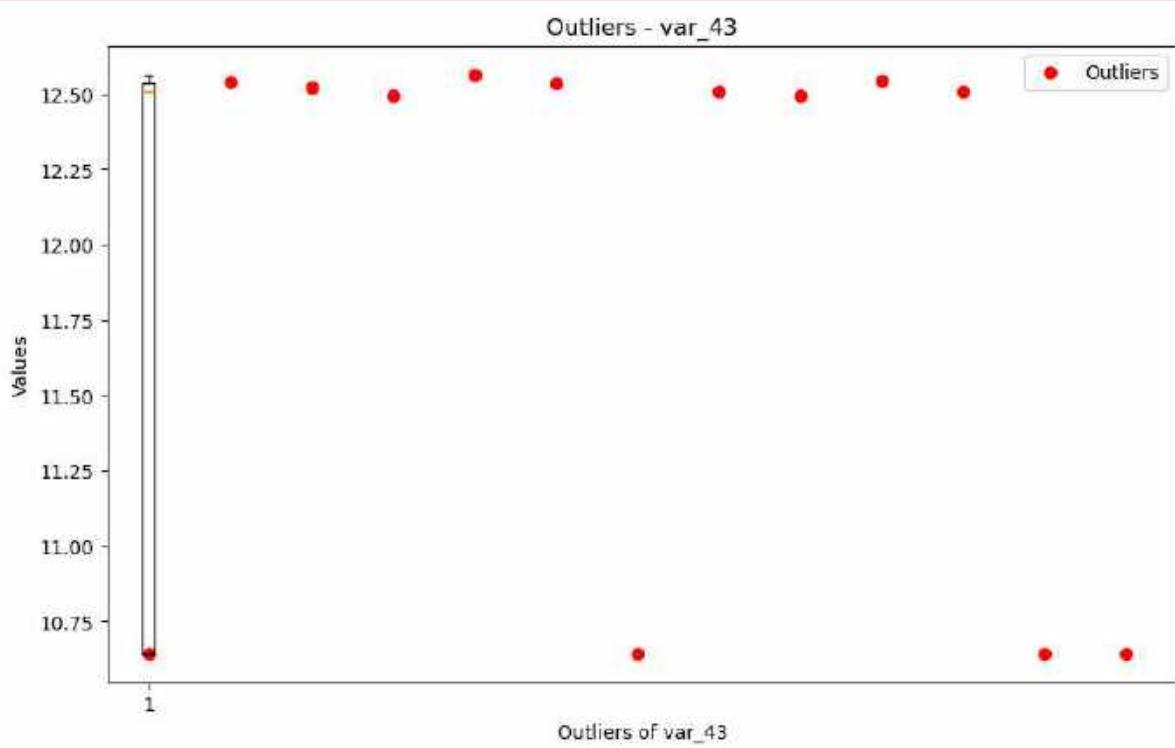
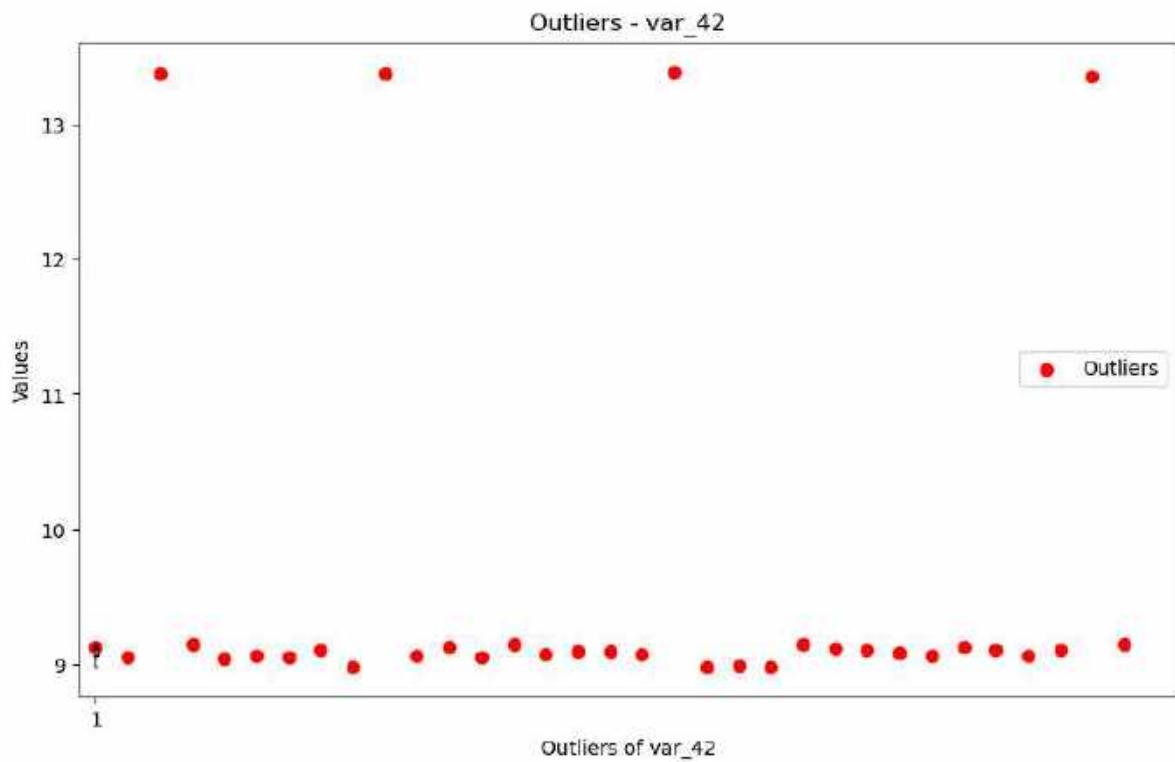
Total outliers in column var\_40: 35

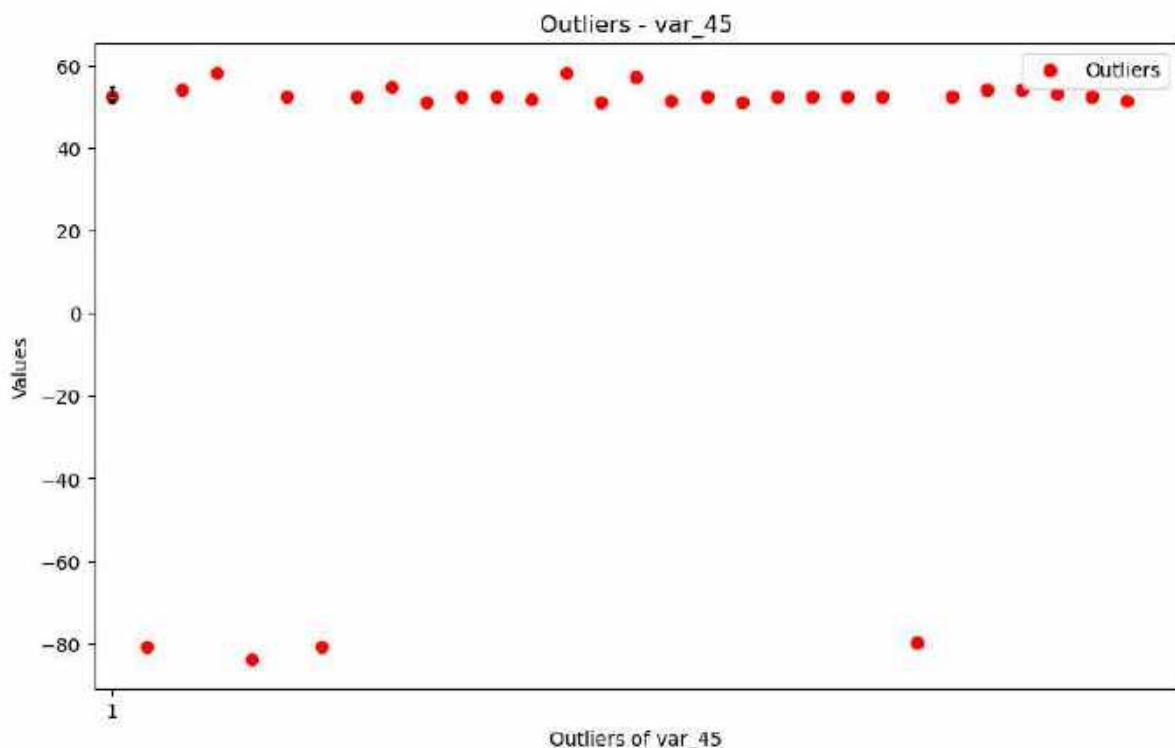
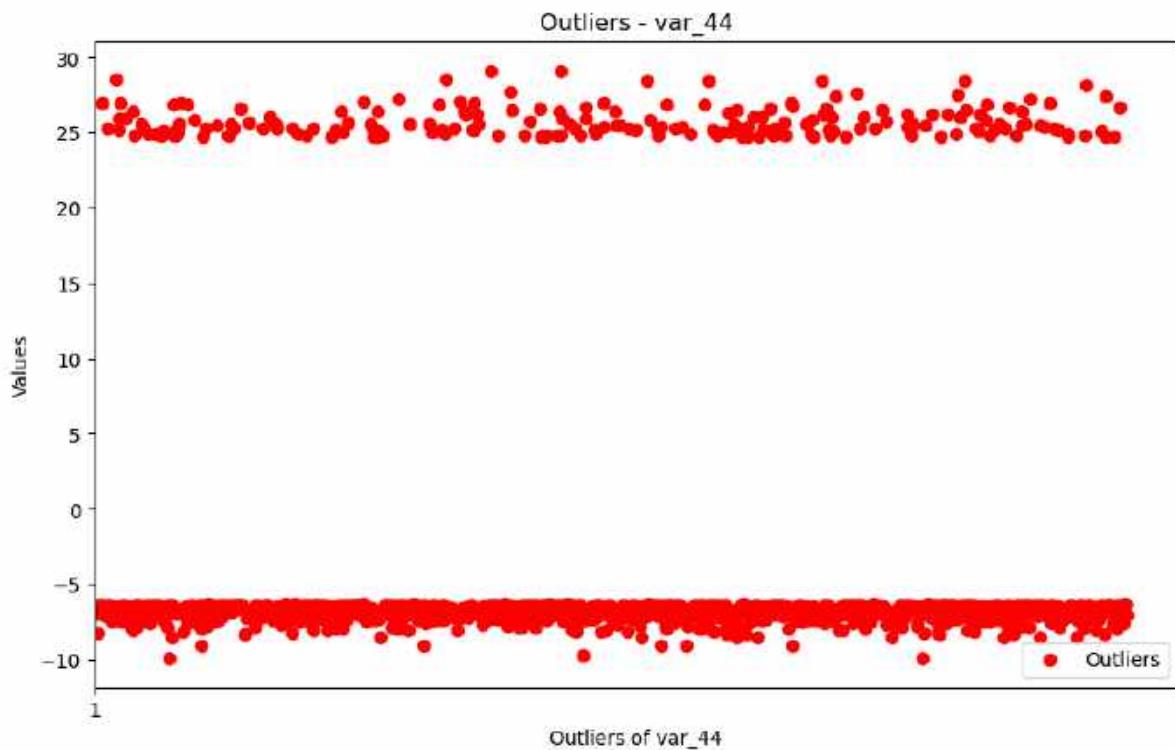
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_41: 0

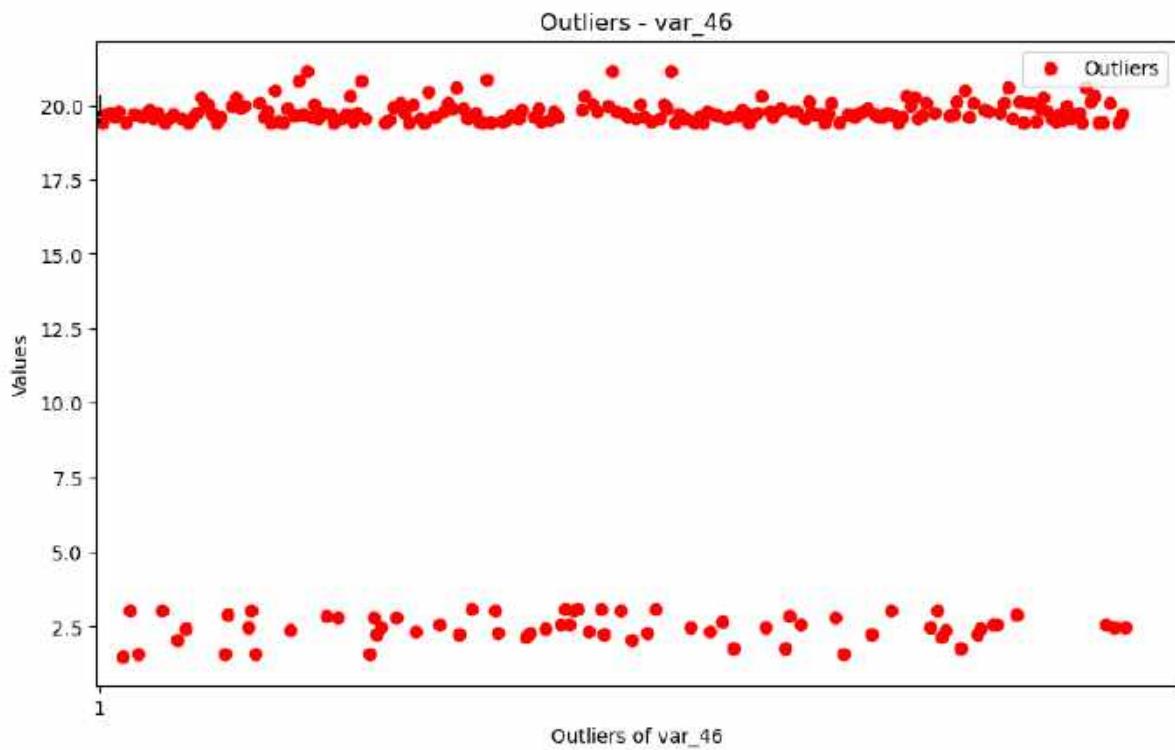
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





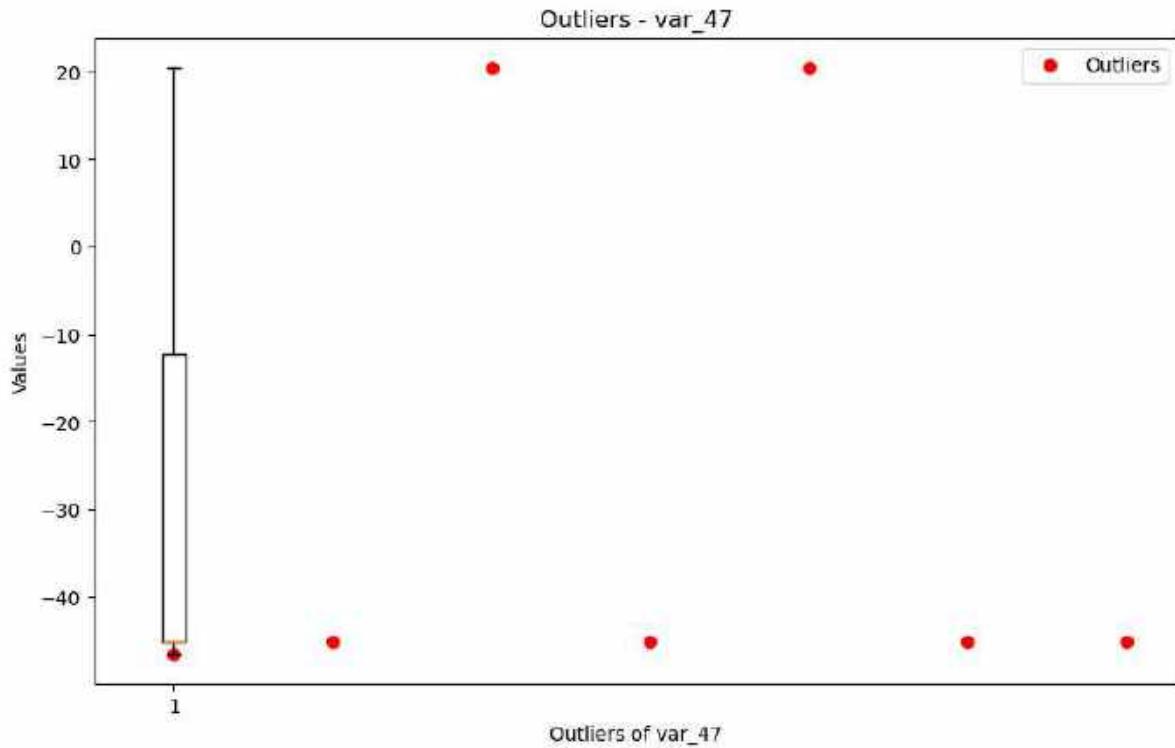
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_45: 30



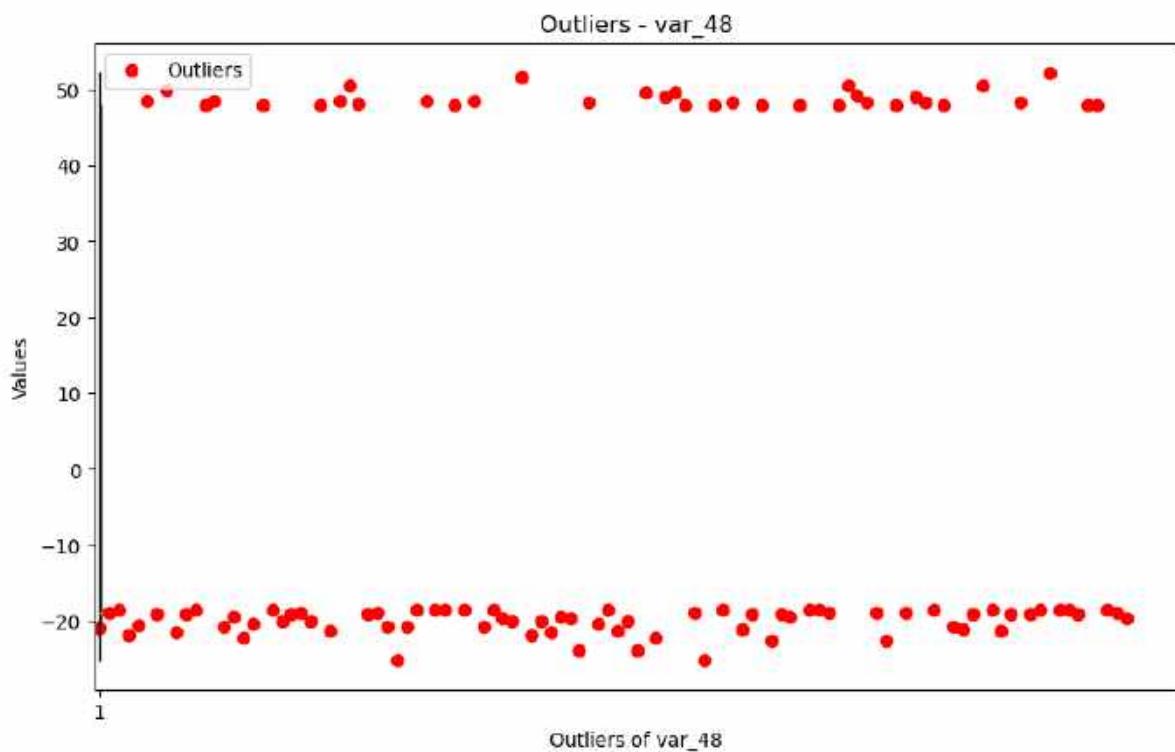
Total outliers in column var\_46: 263

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



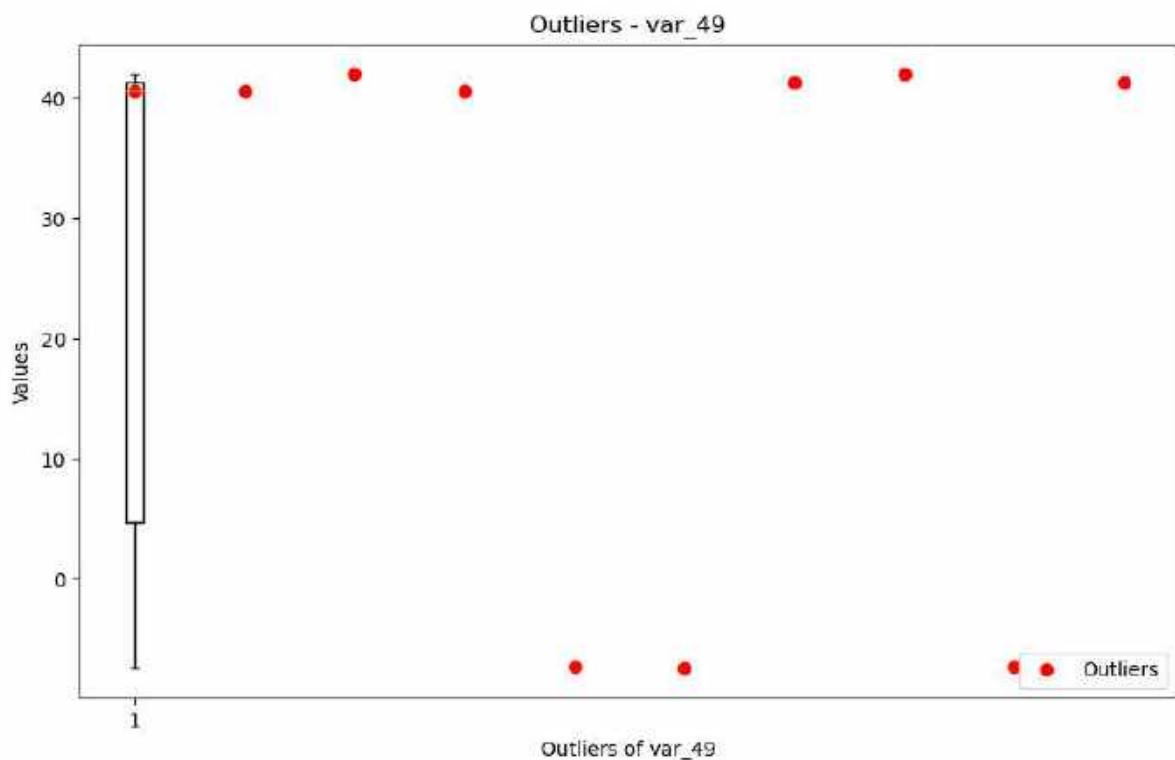
Total outliers in column var\_47: 7

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



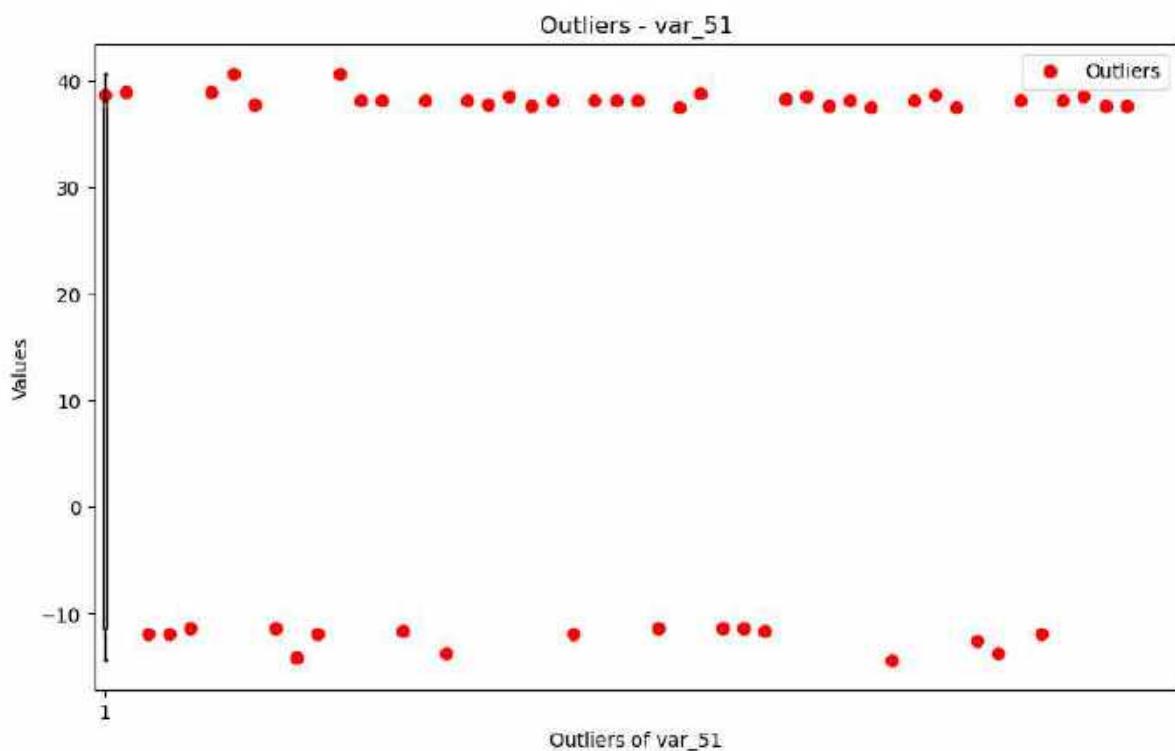
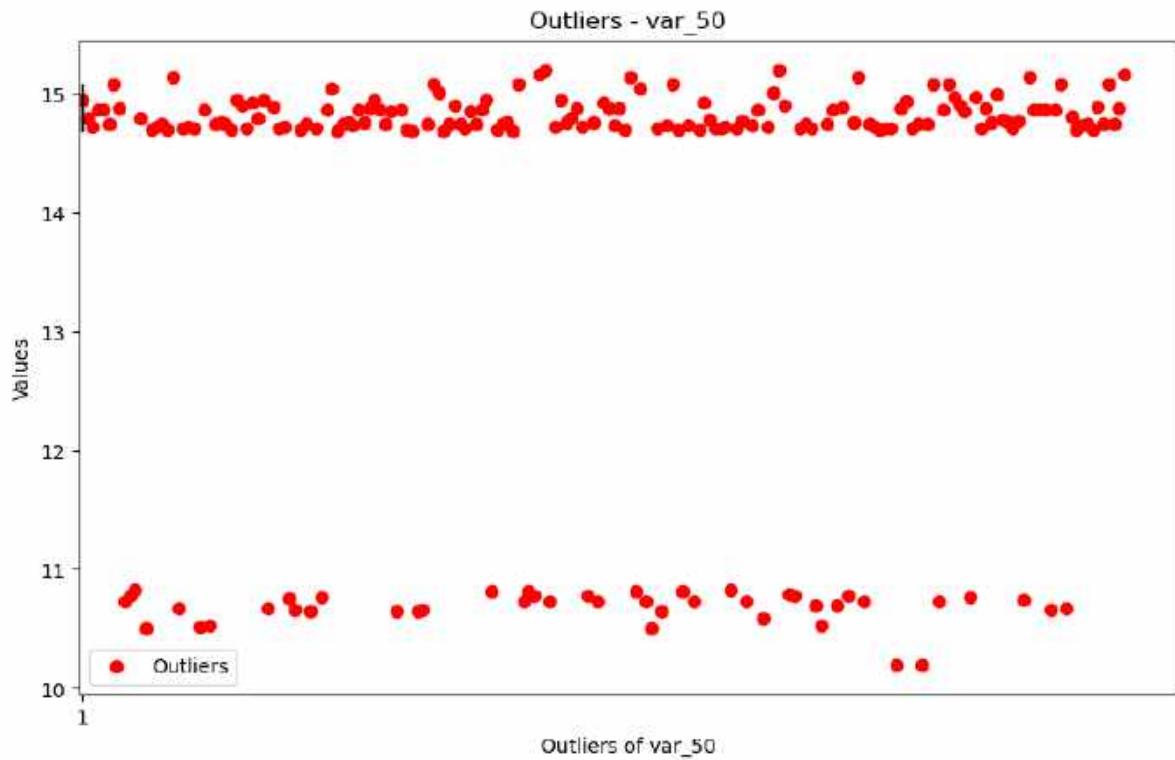
Total outliers in column var\_48: 108

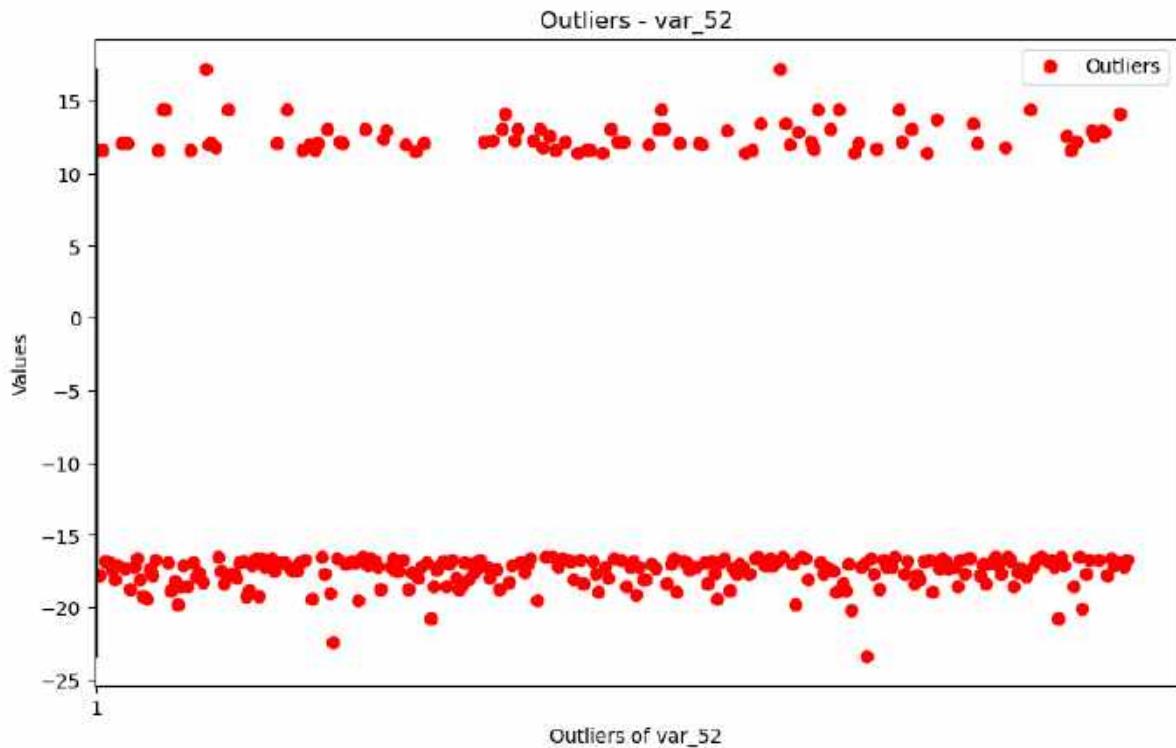
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



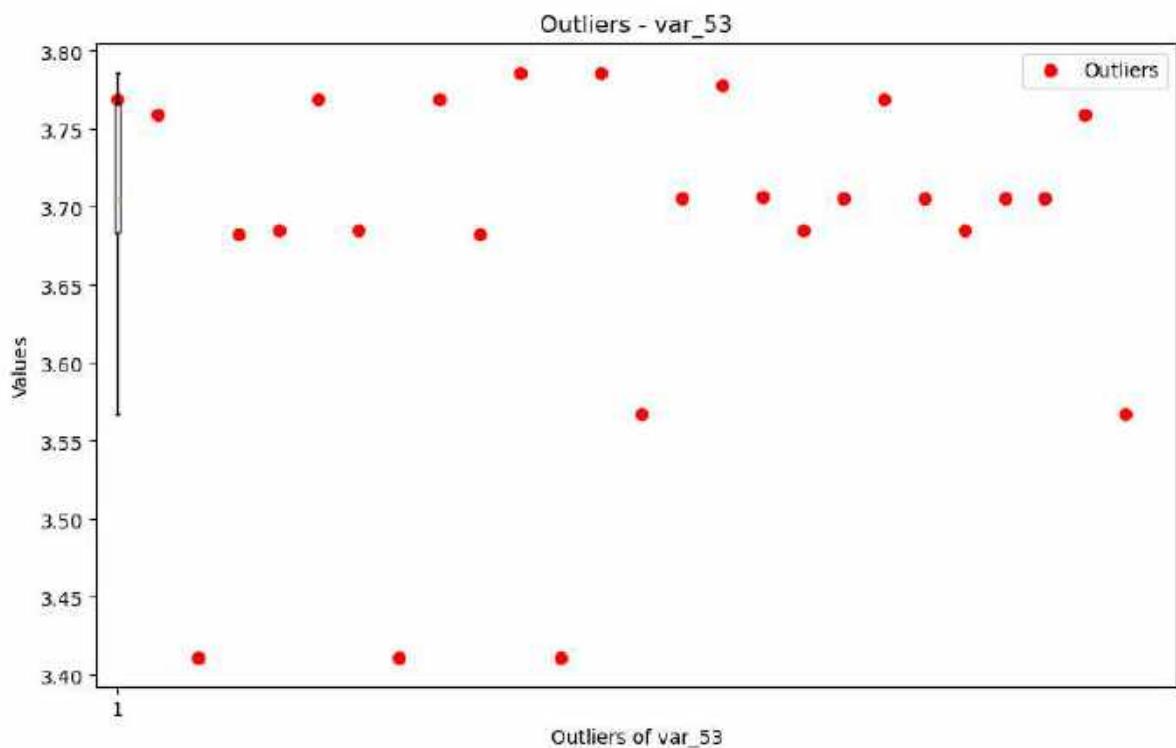
Total outliers in column var\_49: 10

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



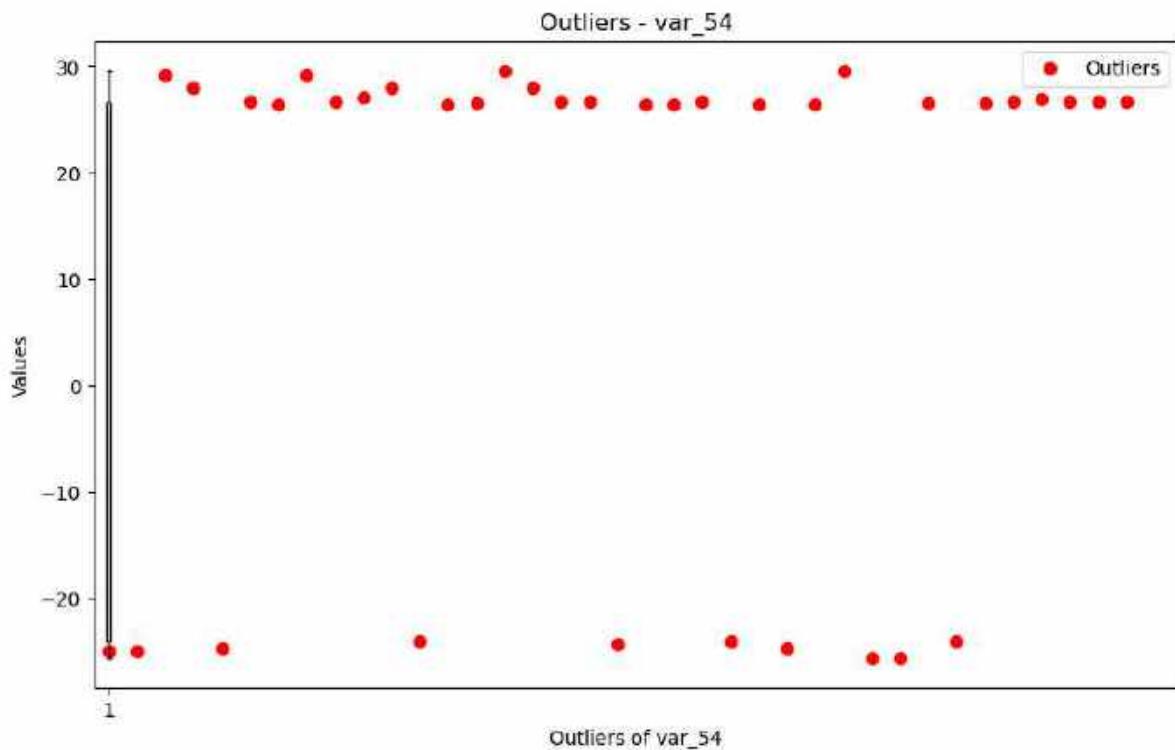


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_52: 331
```



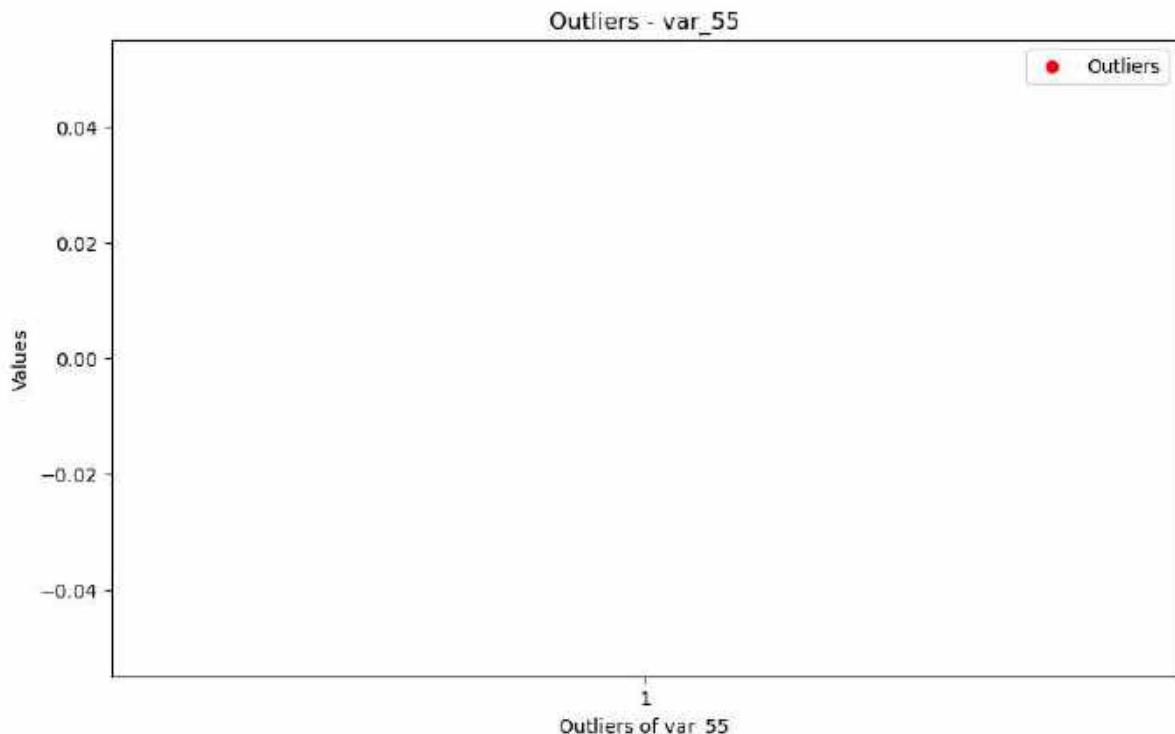
```
Total outliers in column var_53: 26
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



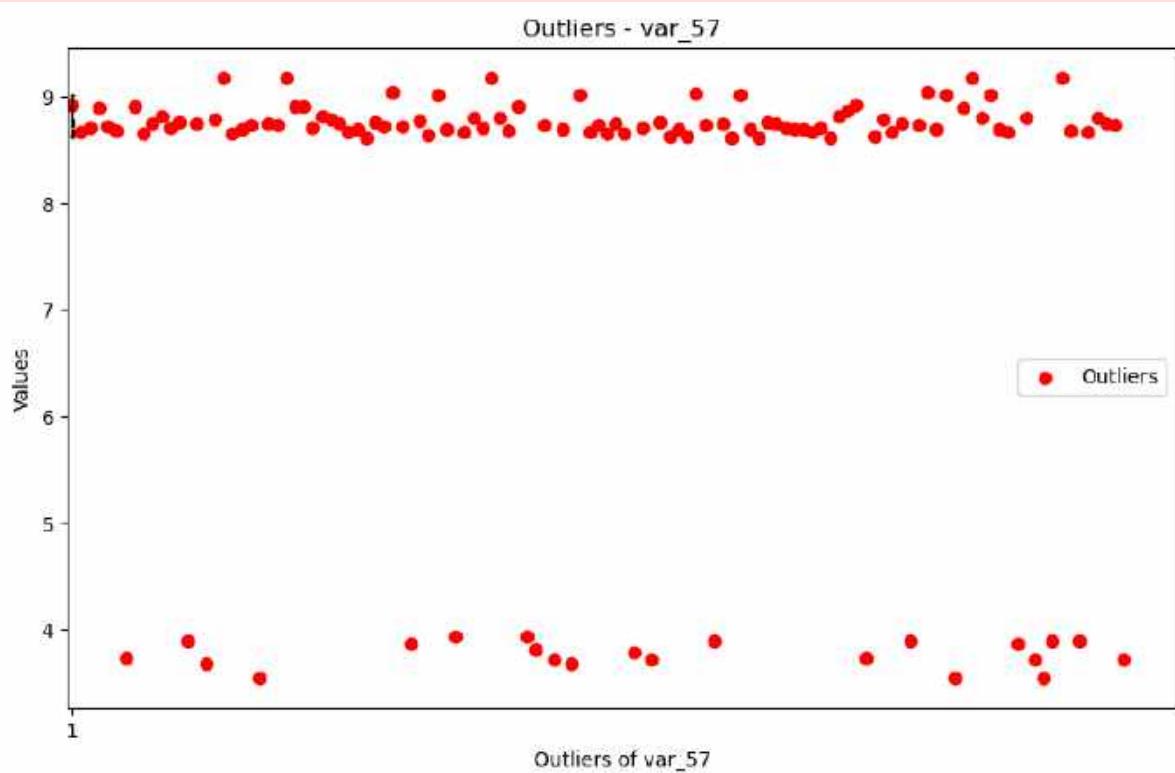
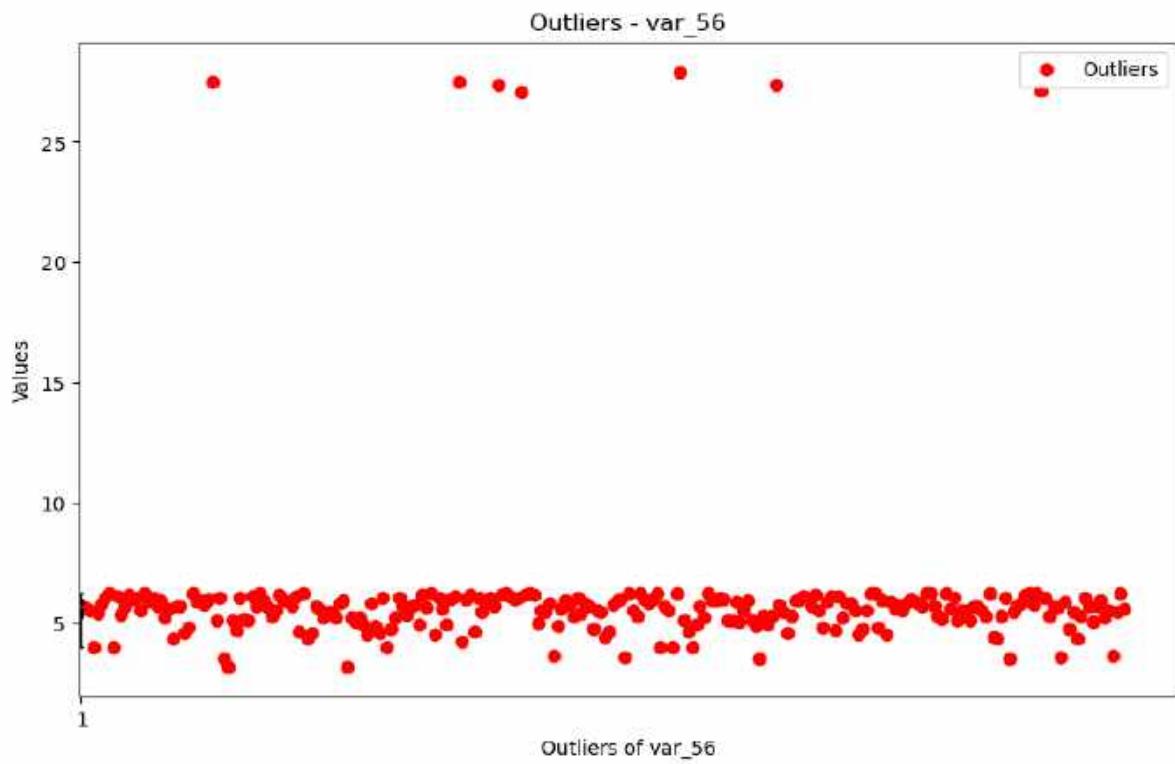
```
Total outliers in column var_54: 37
```

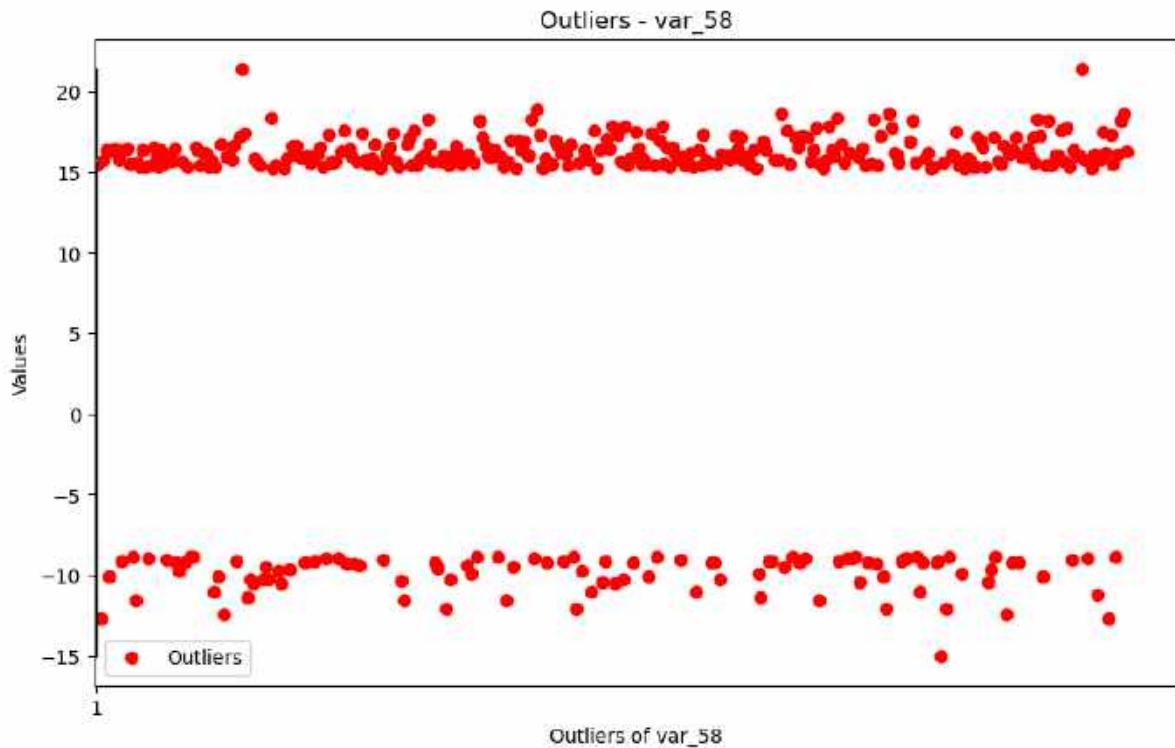
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



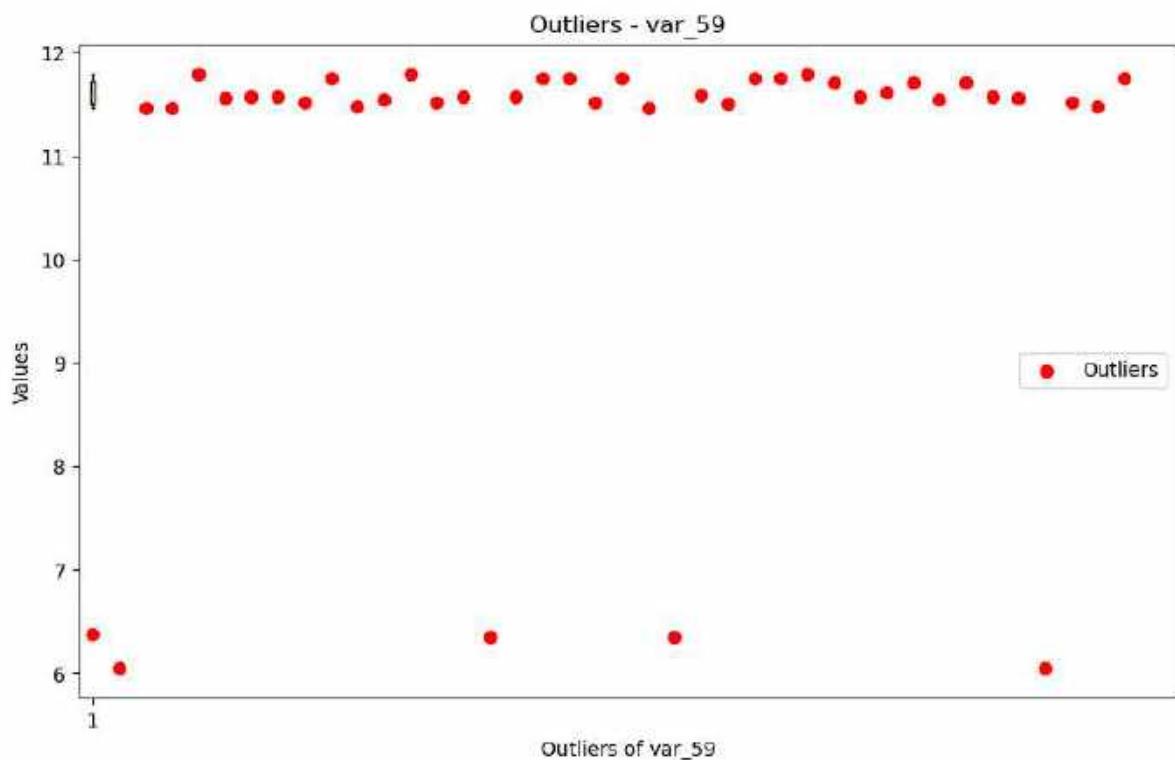
```
Total outliers in column var_55: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

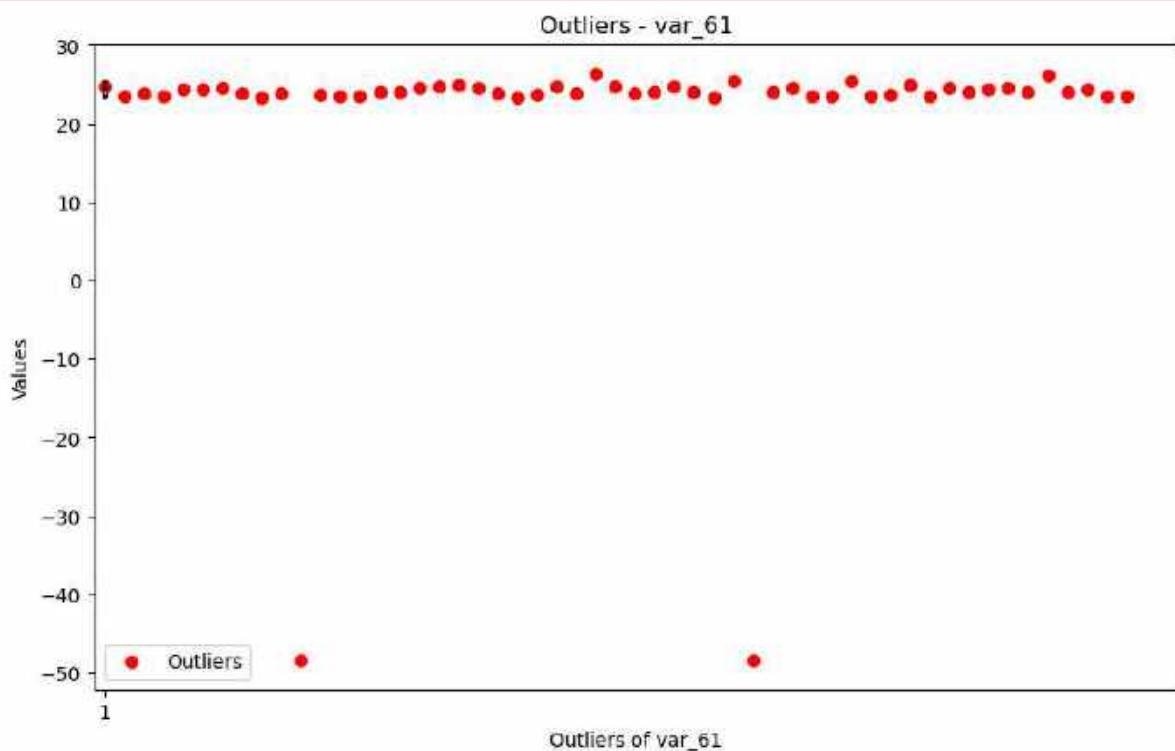
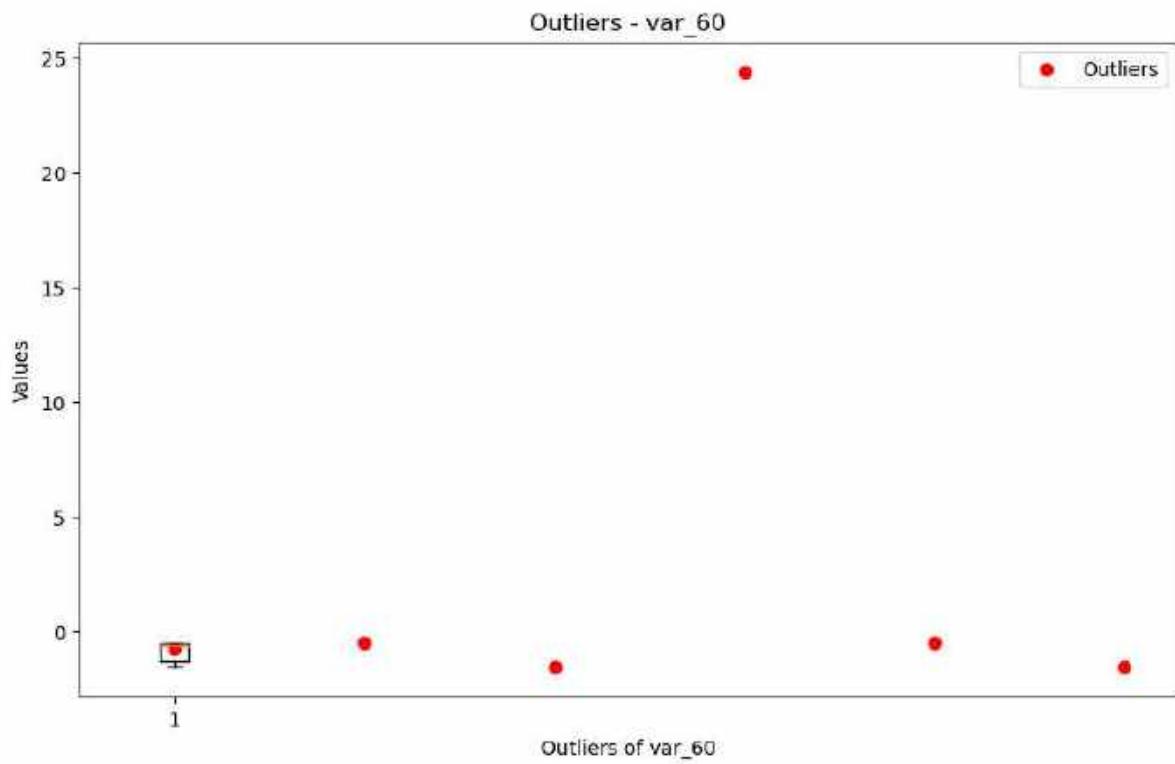


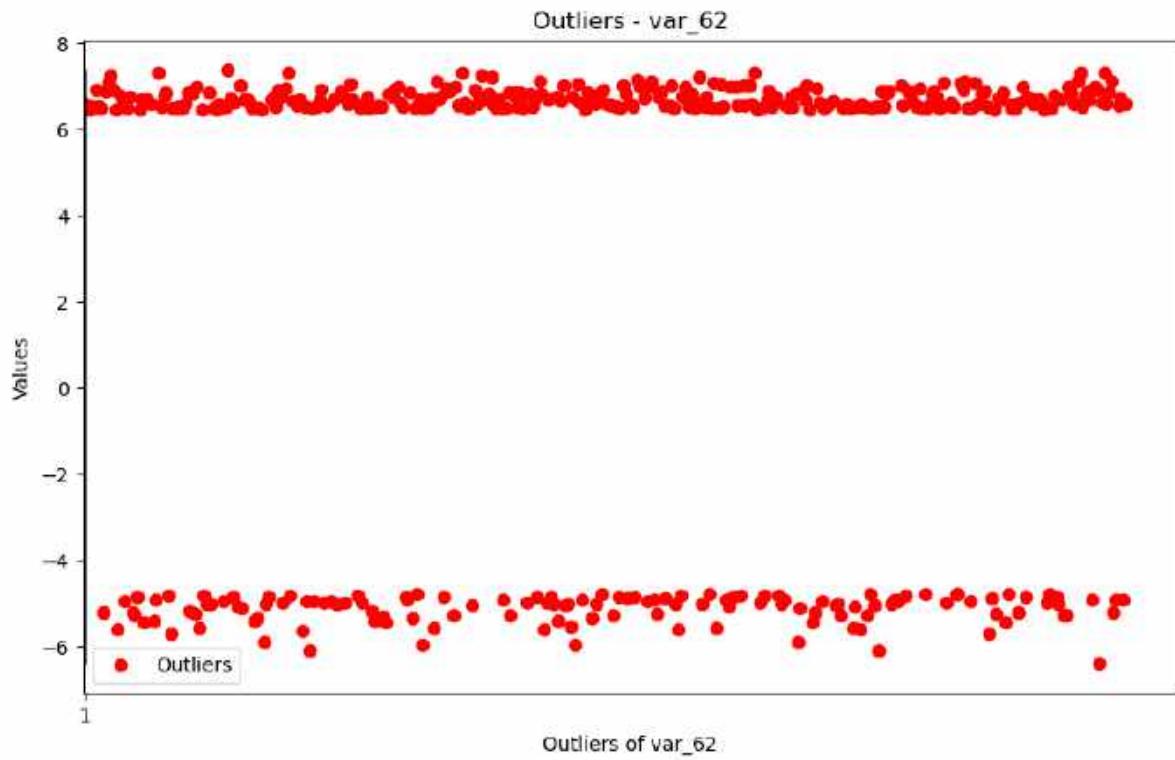


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_58: 396
```



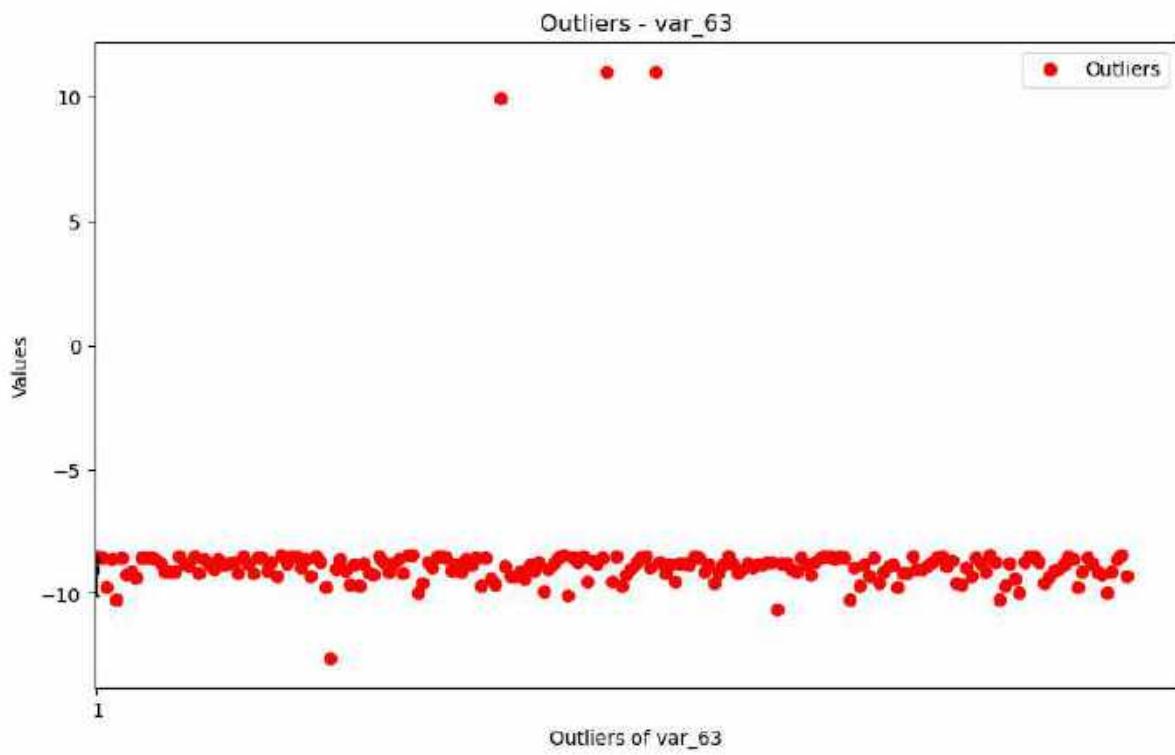
```
Total outliers in column var_59: 40  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





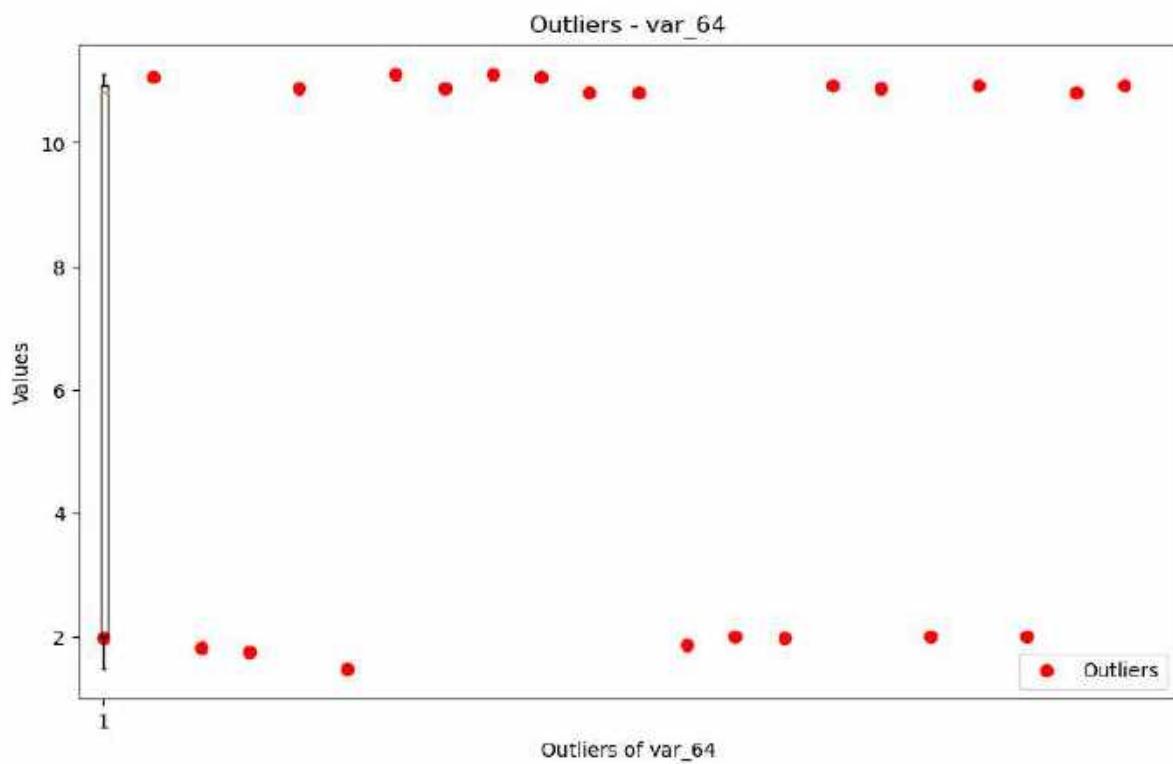
Total outliers in column var\_62: 436

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



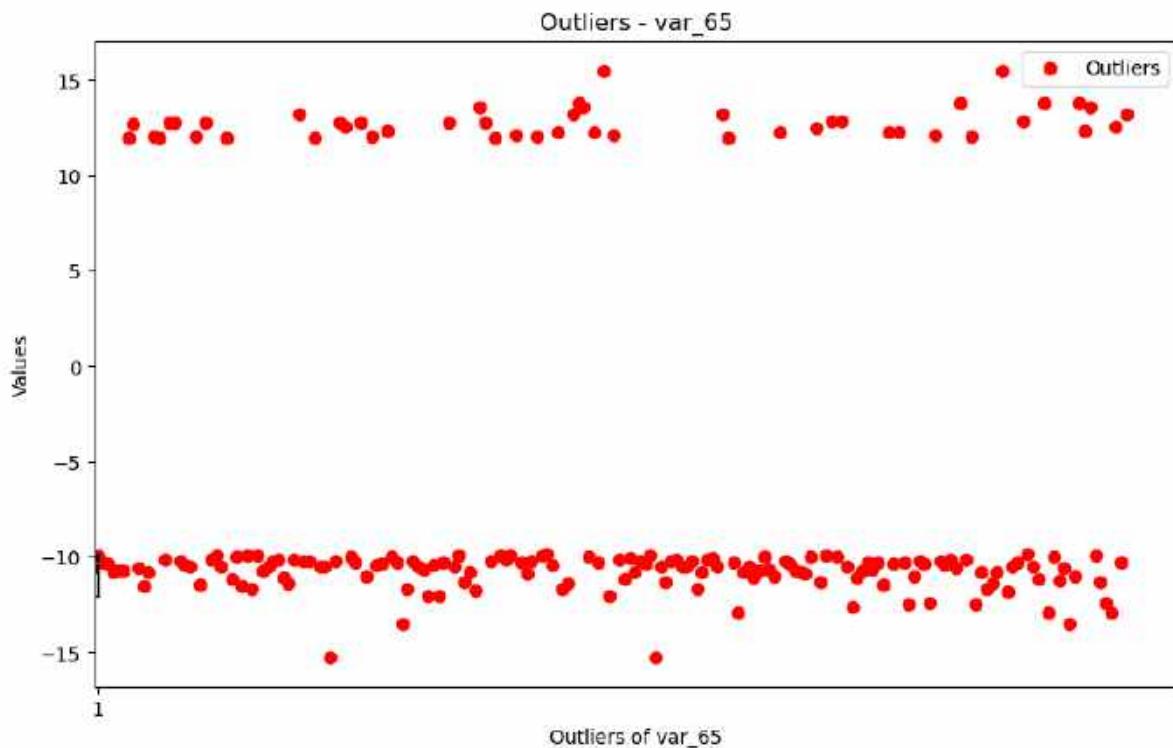
Total outliers in column var\_63: 213

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



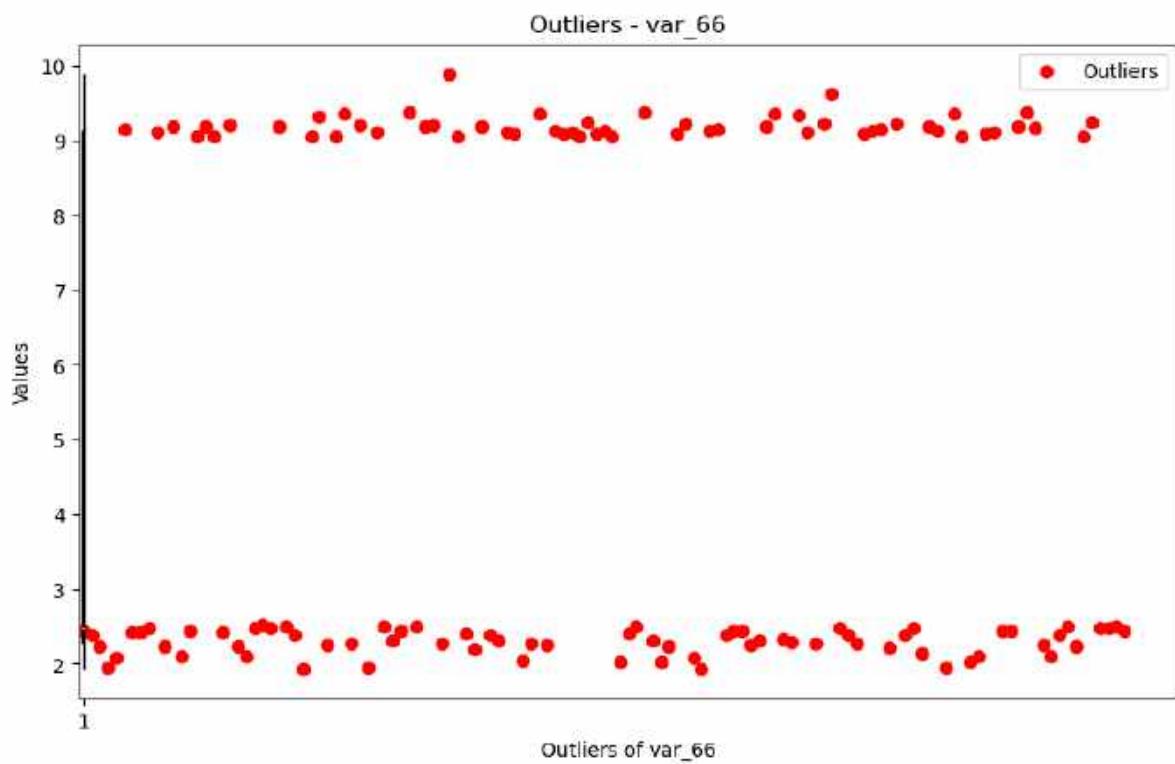
Total outliers in column var\_64: 22

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



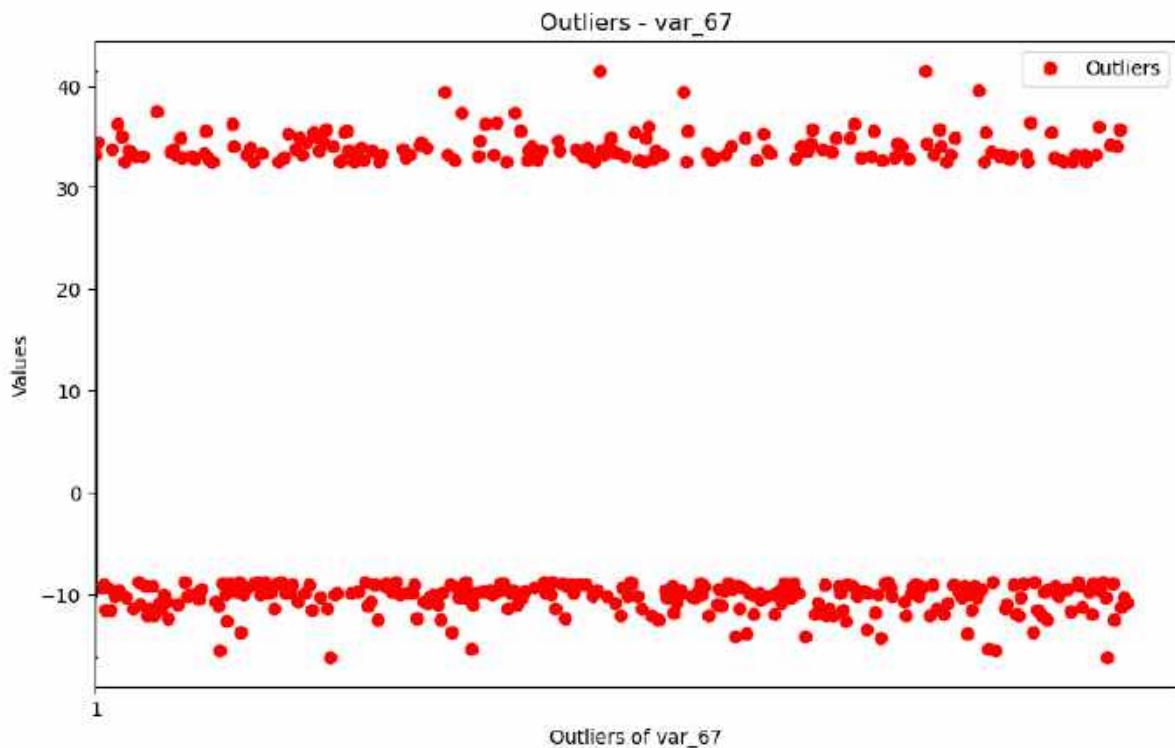
Total outliers in column var\_65: 200

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



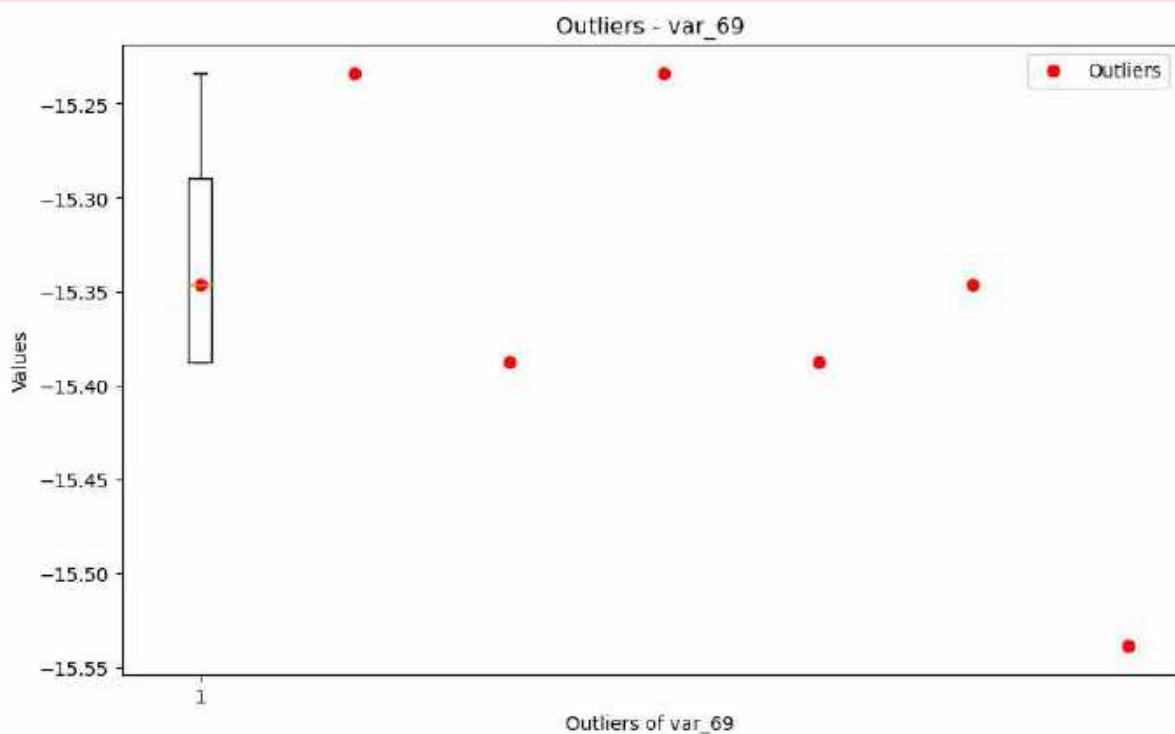
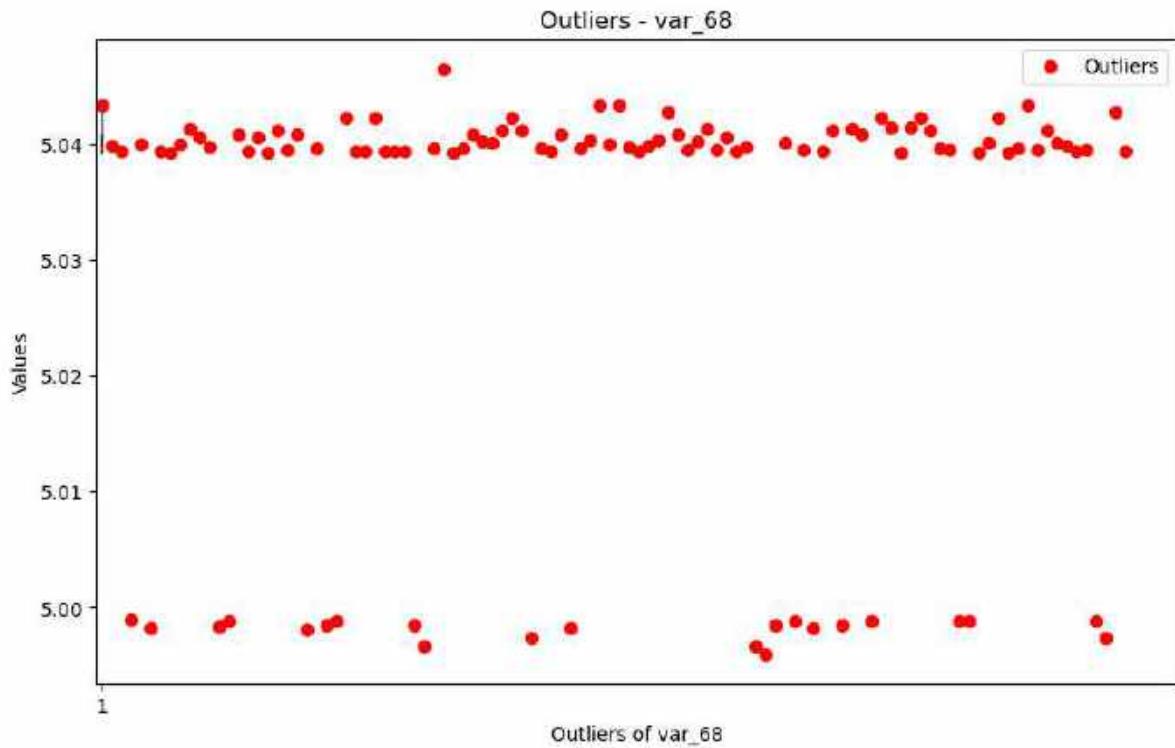
```
Total outliers in column var_66: 129
```

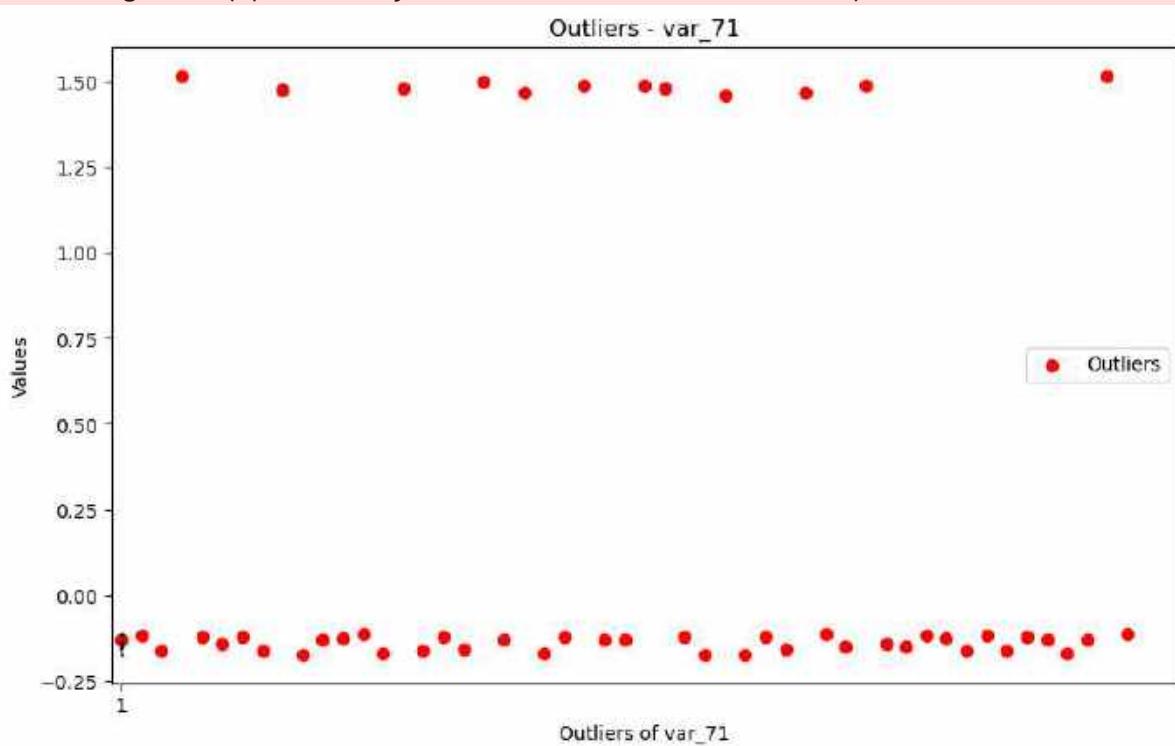
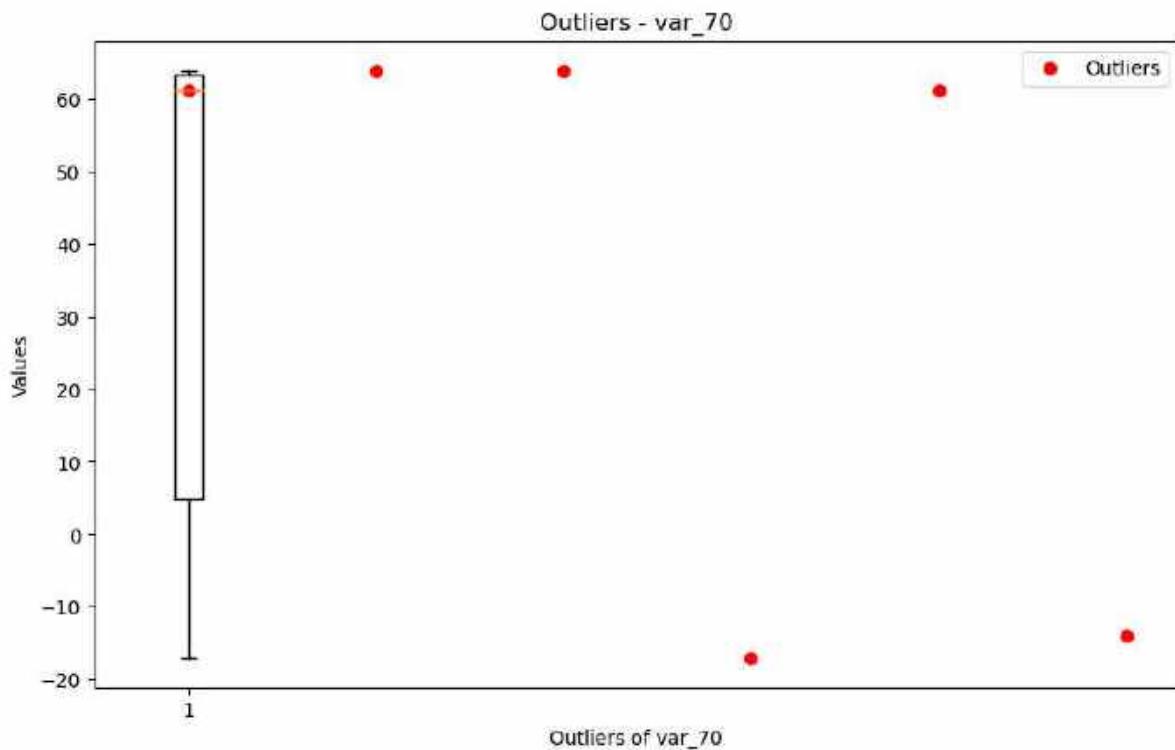
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

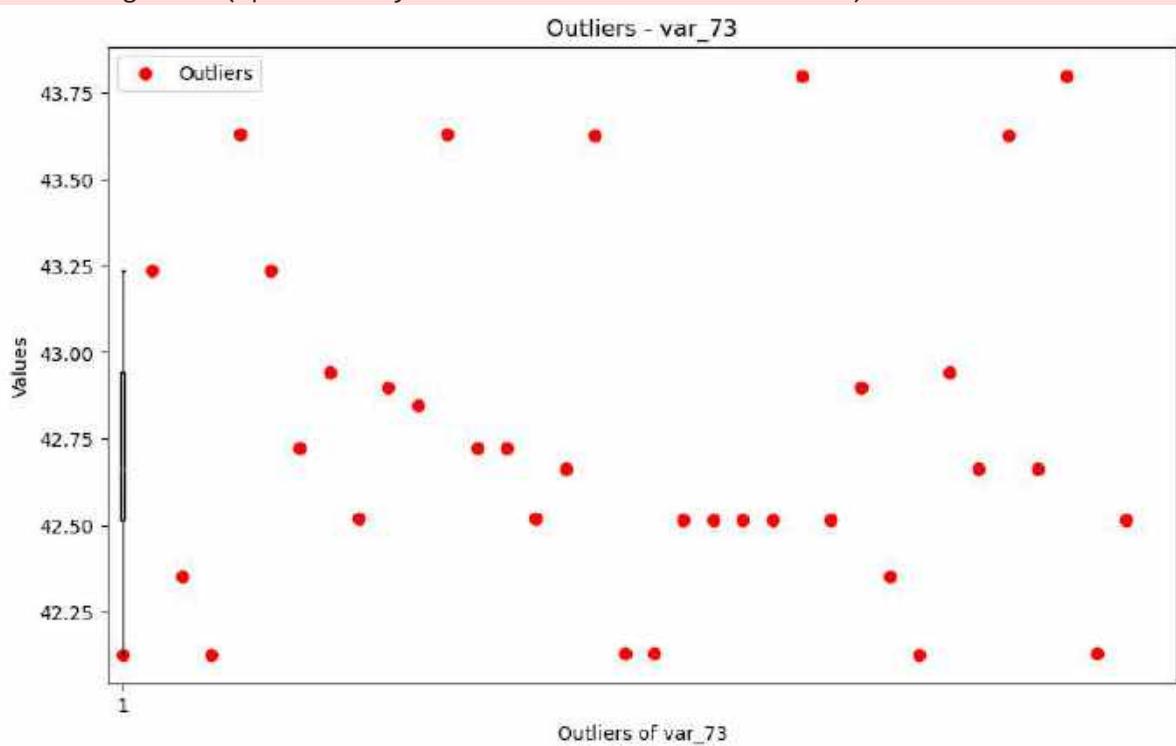
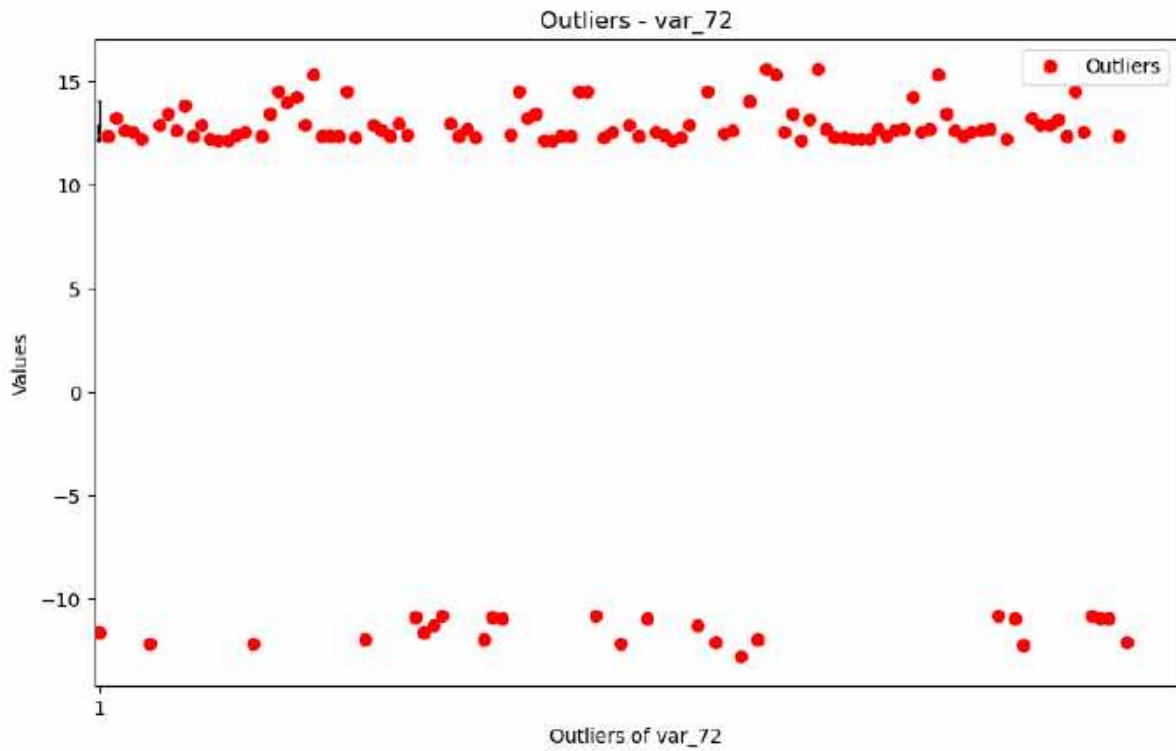


```
Total outliers in column var_67: 441
```

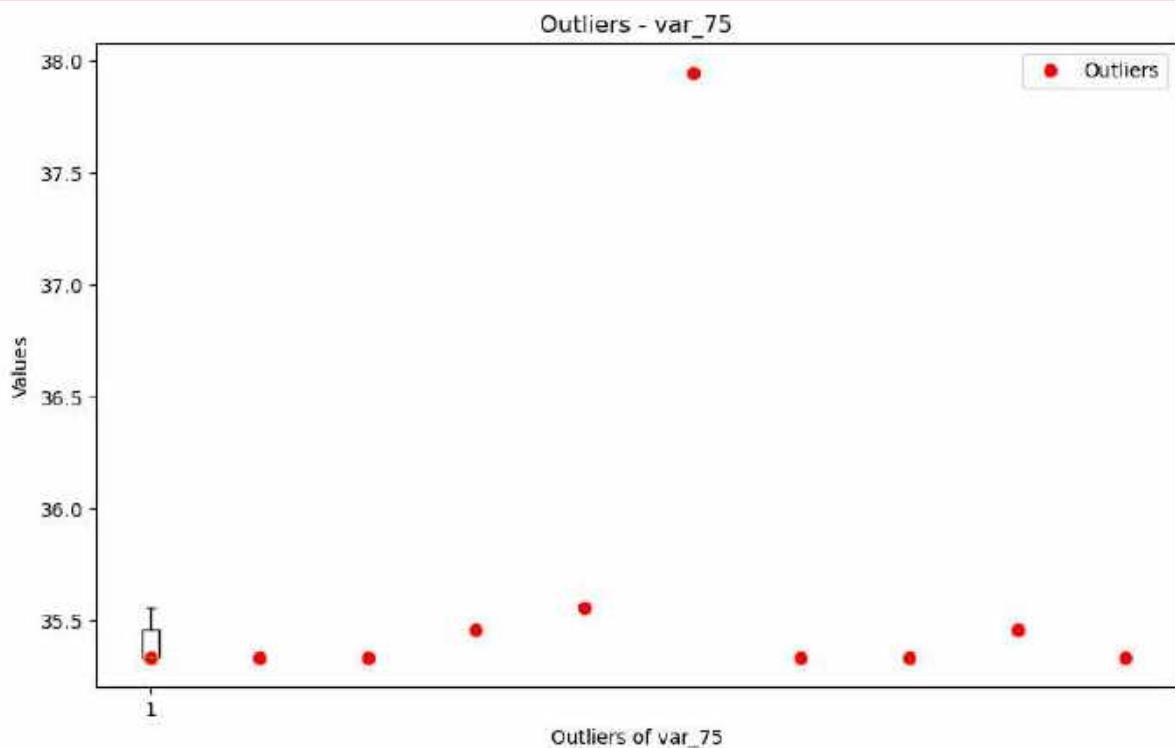
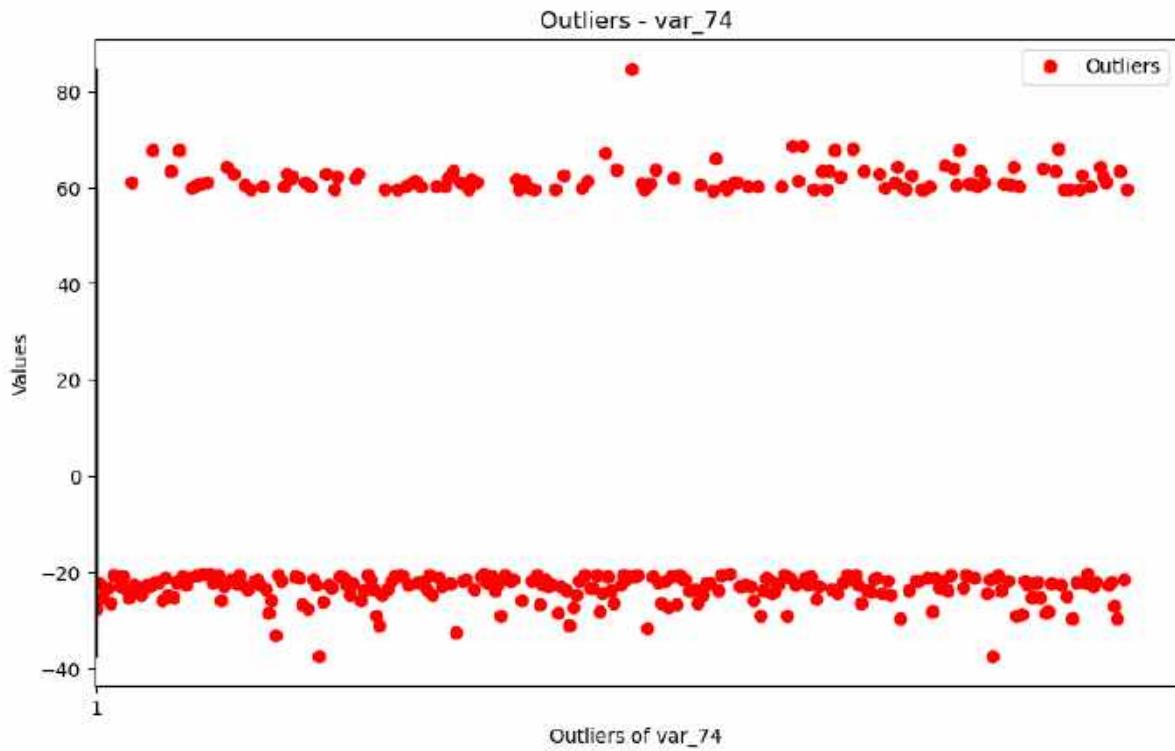
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

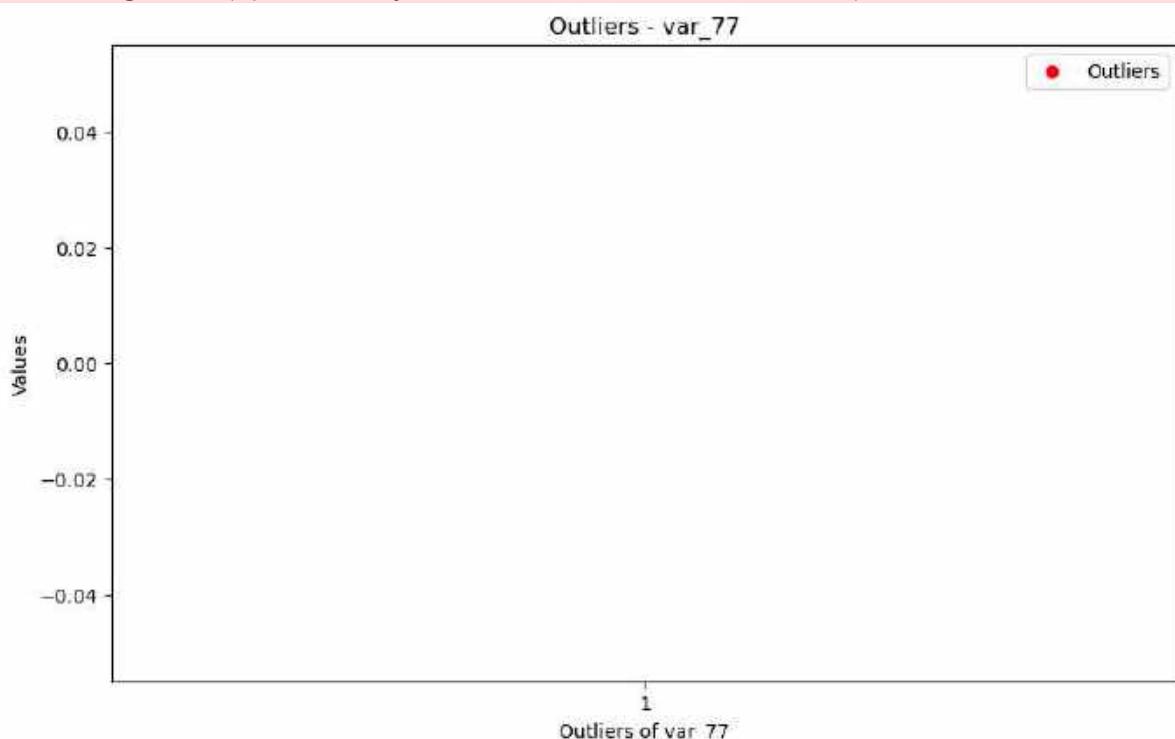
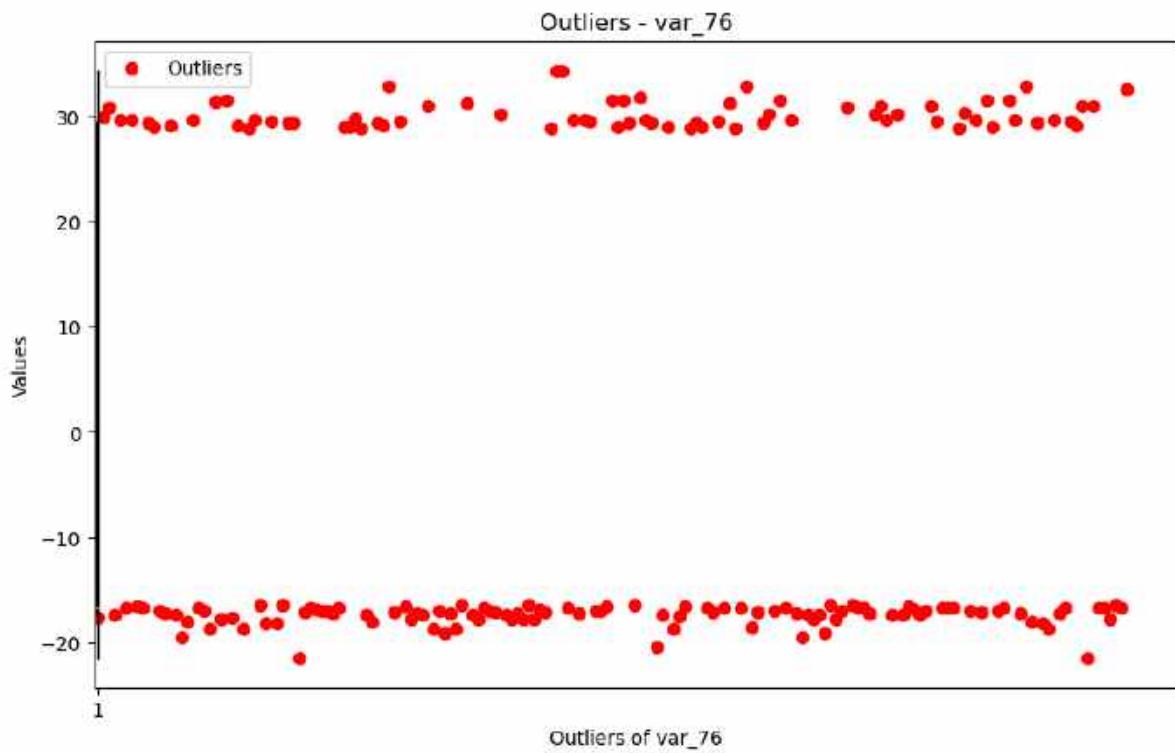


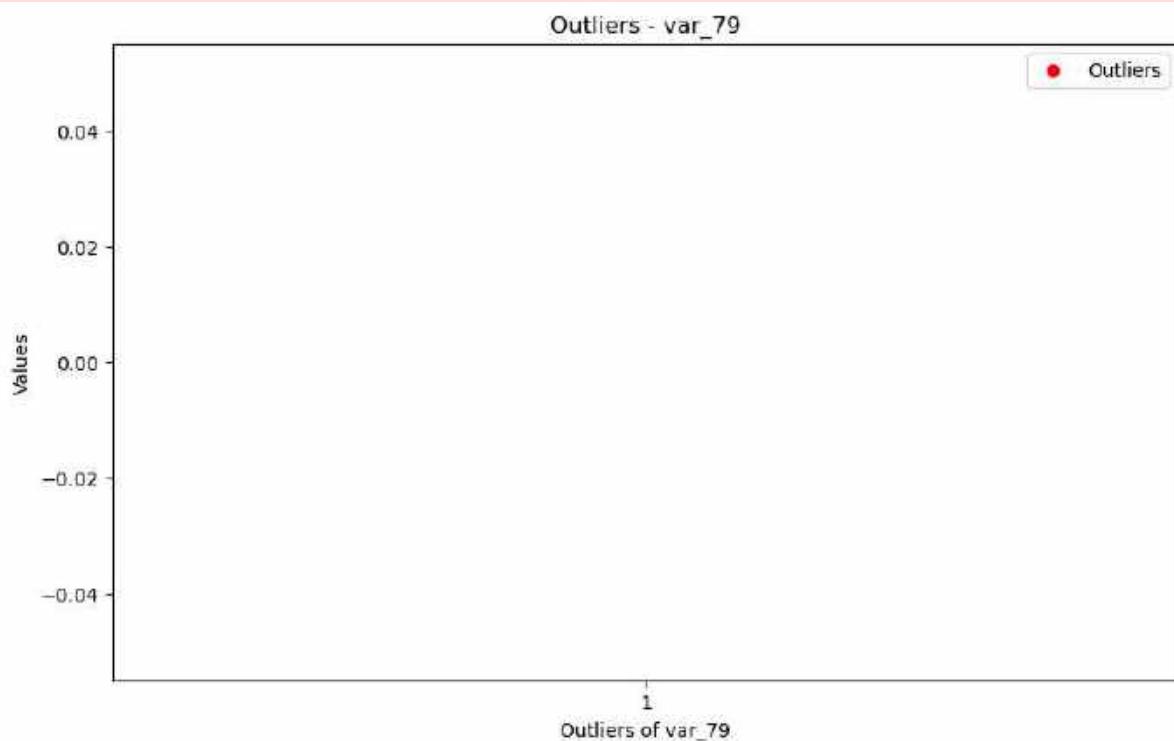
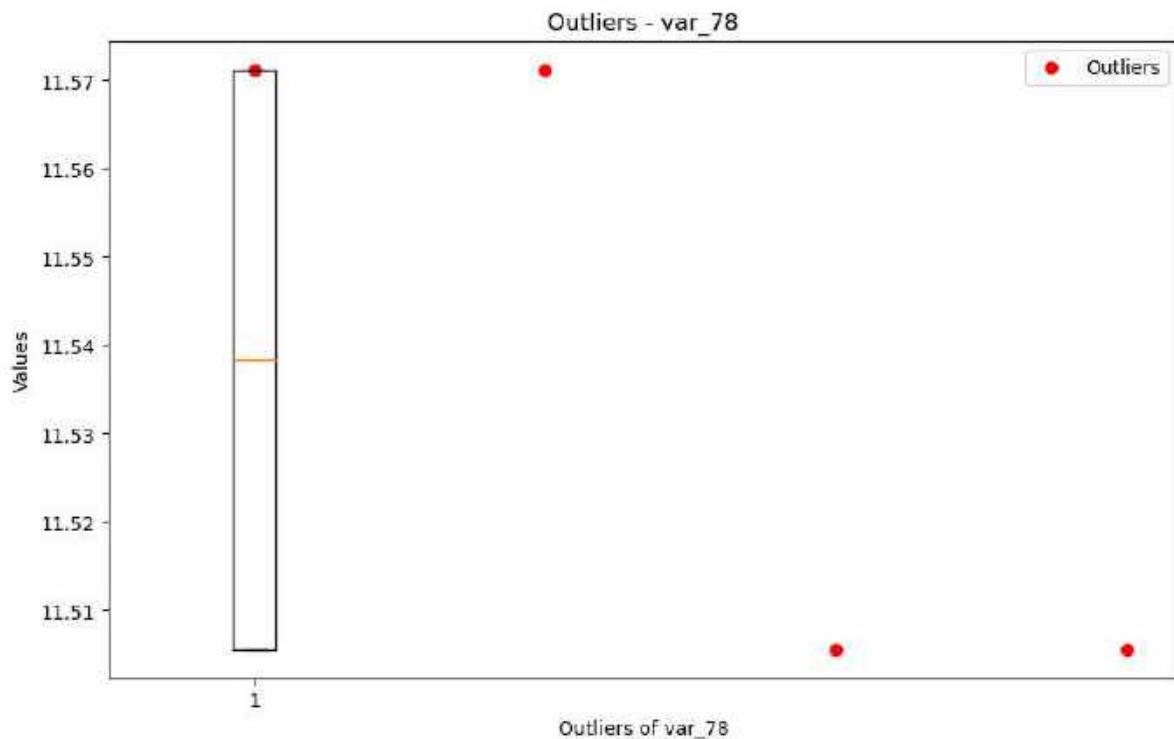




Total outliers in column var\_73: 35

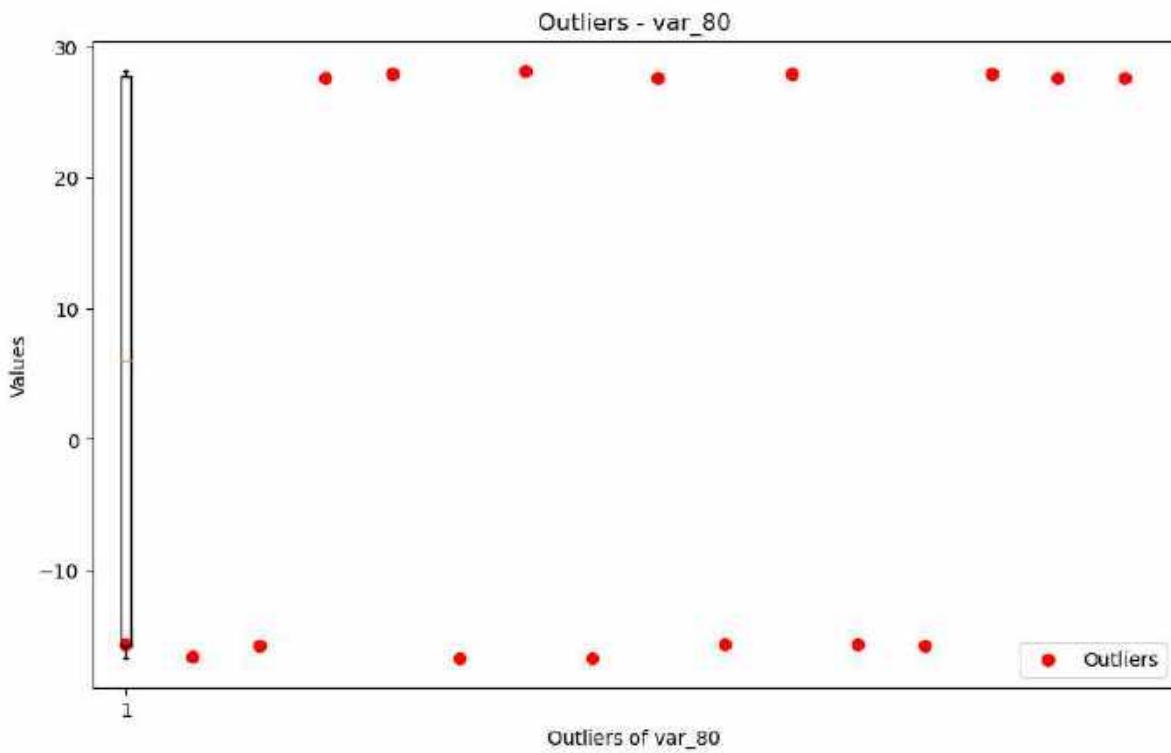






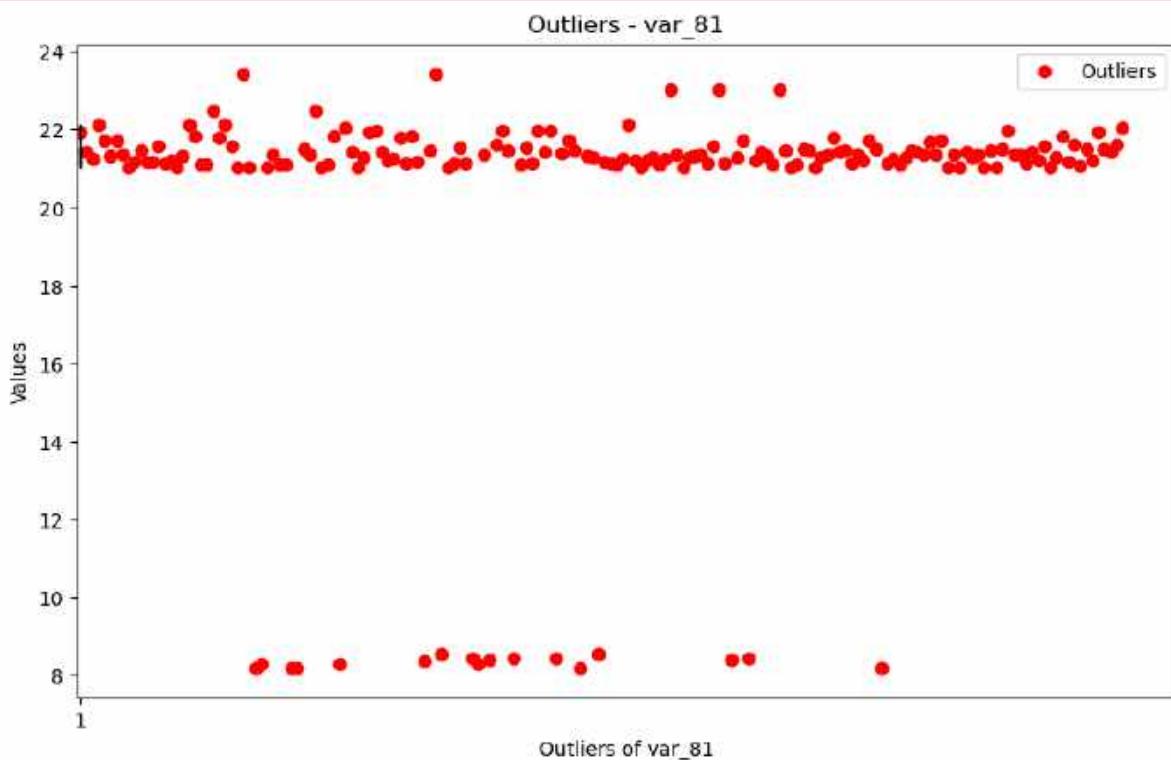
Total outliers in column var\_79: 0

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



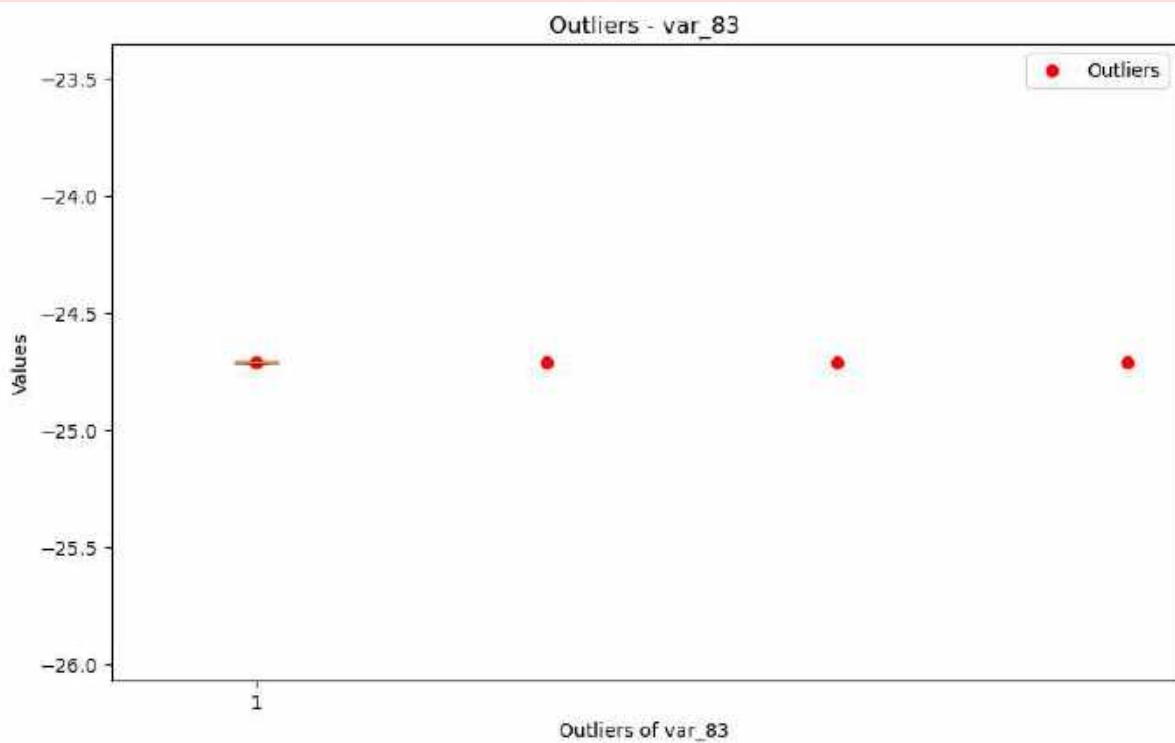
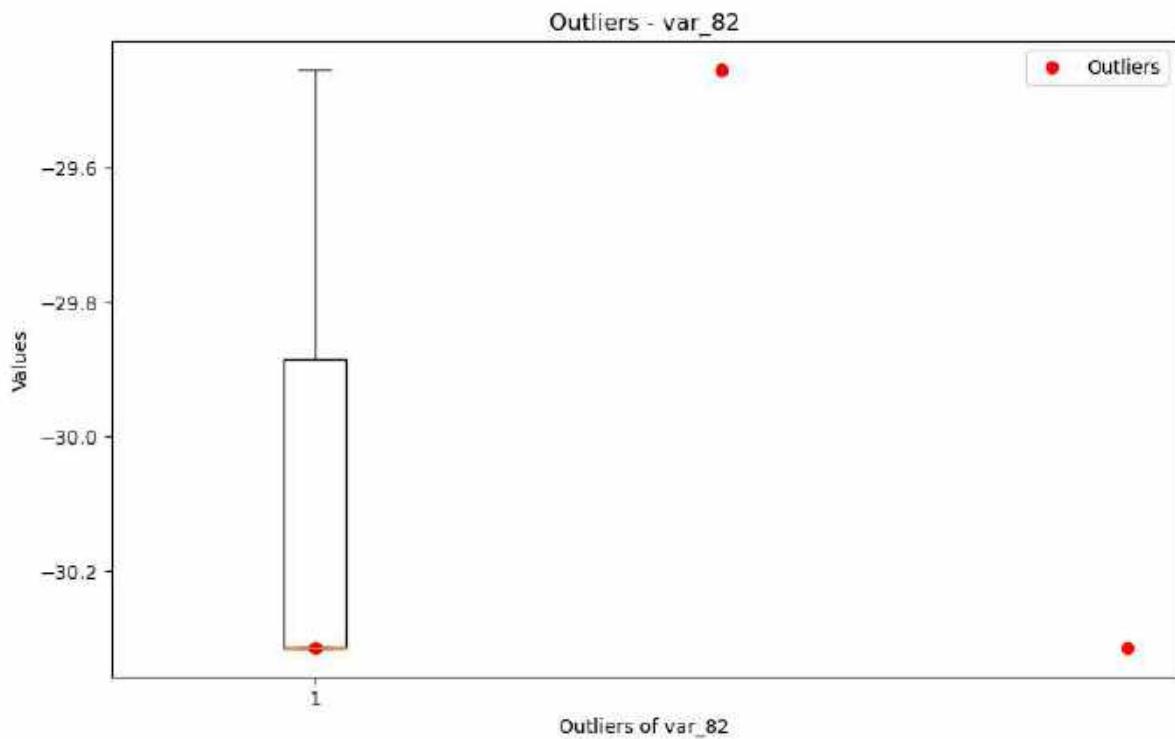
Total outliers in column var\_80: 16

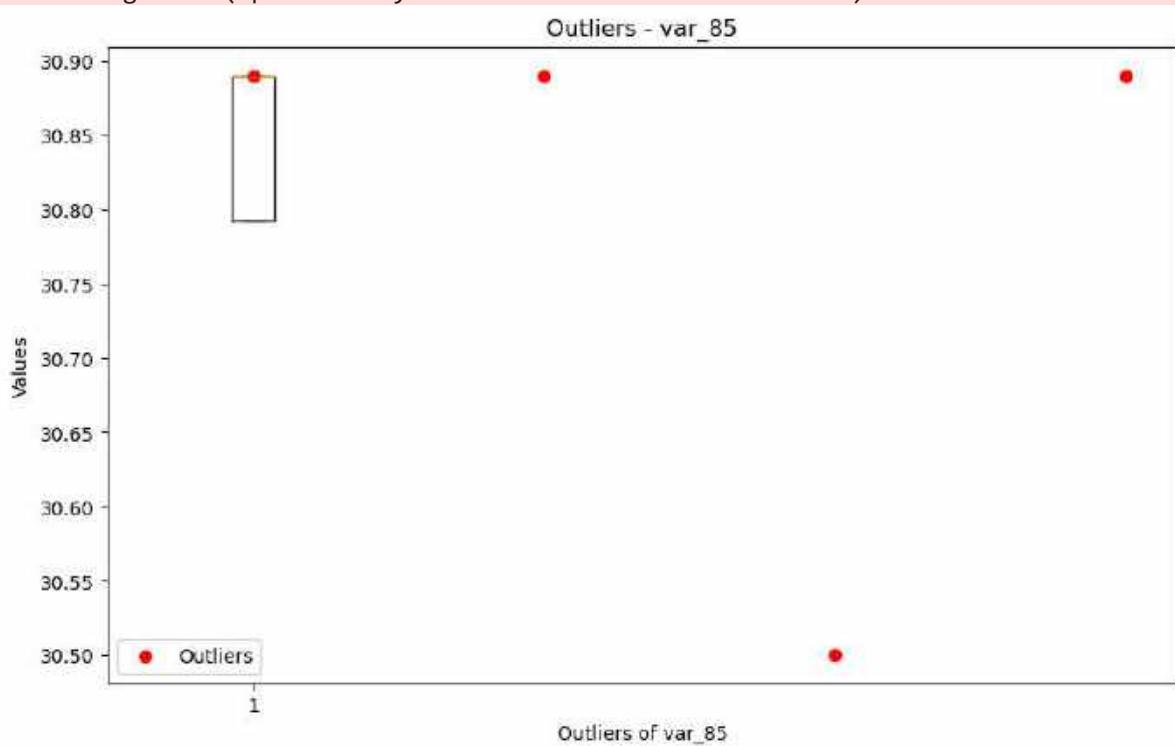
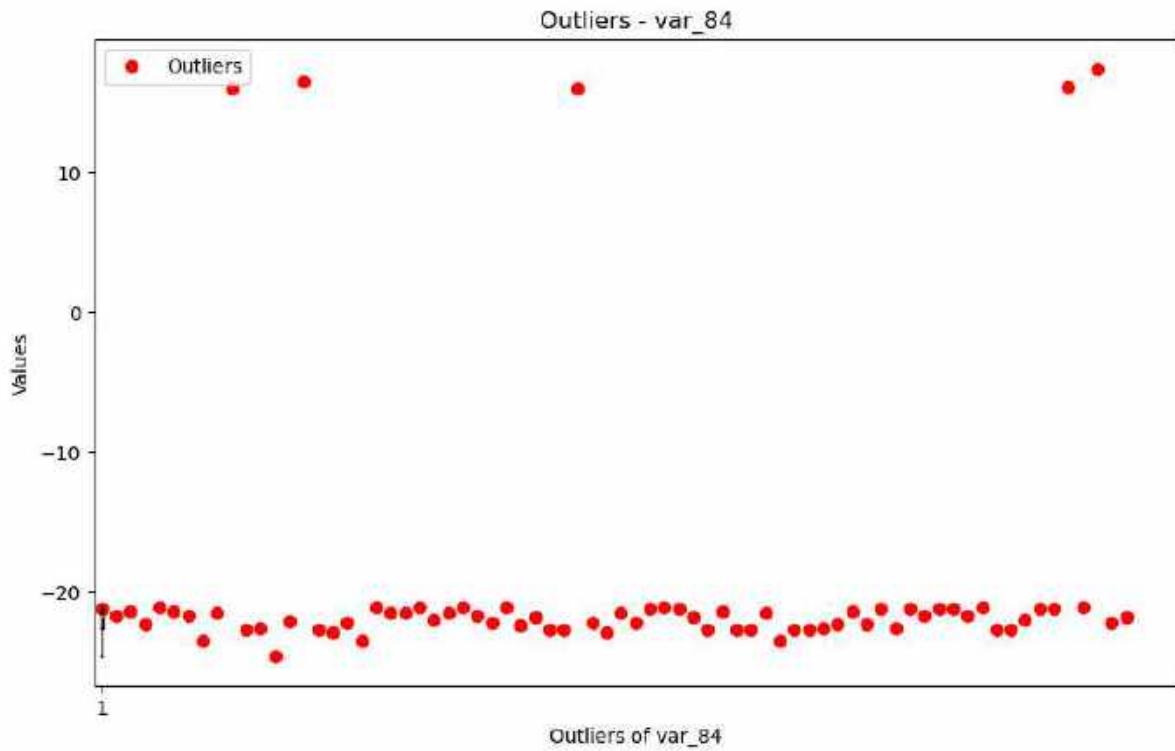
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

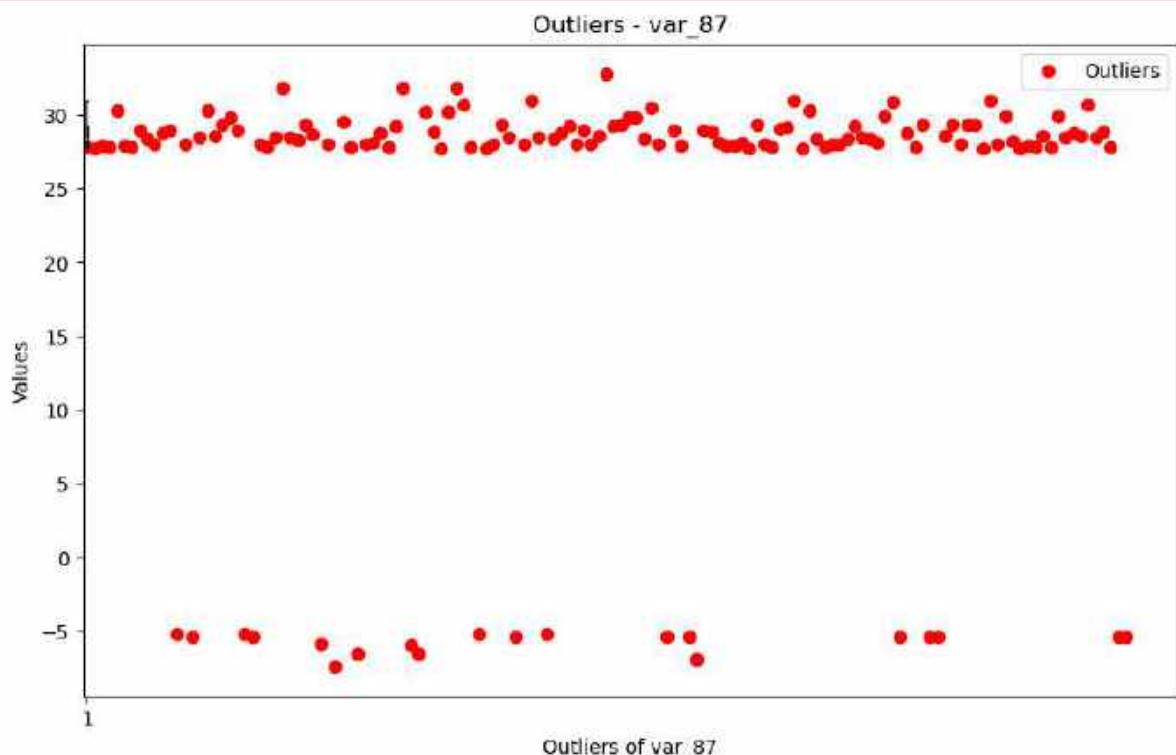
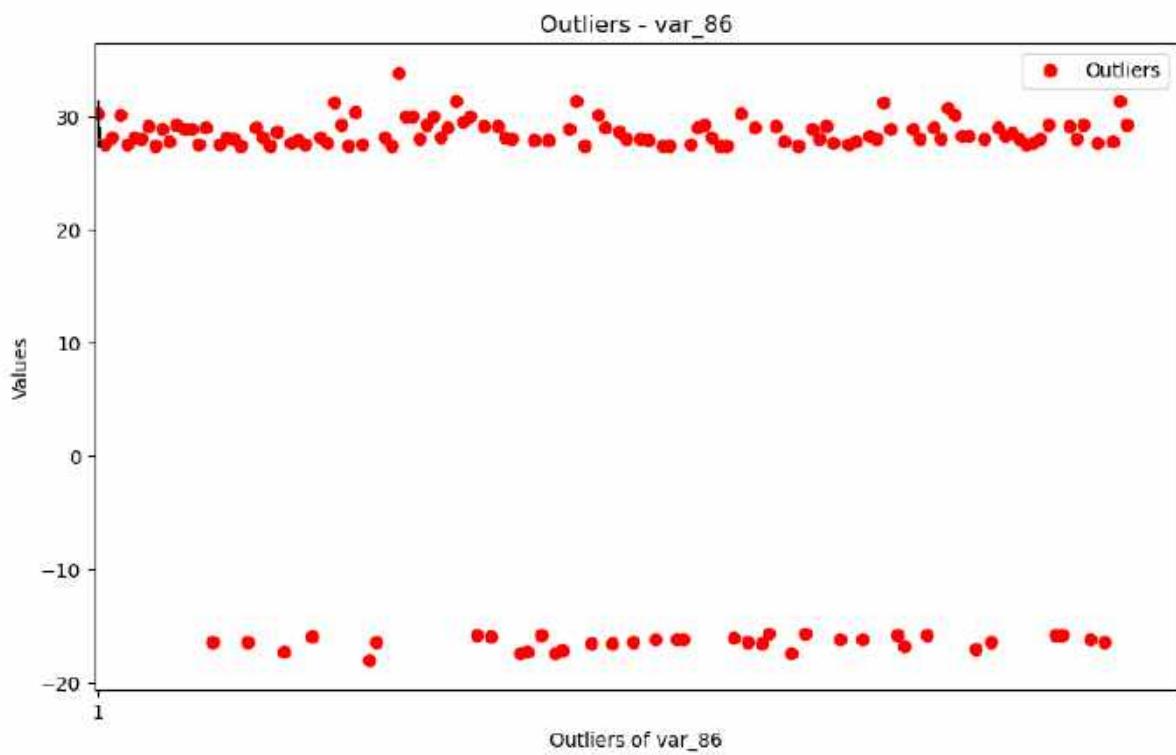


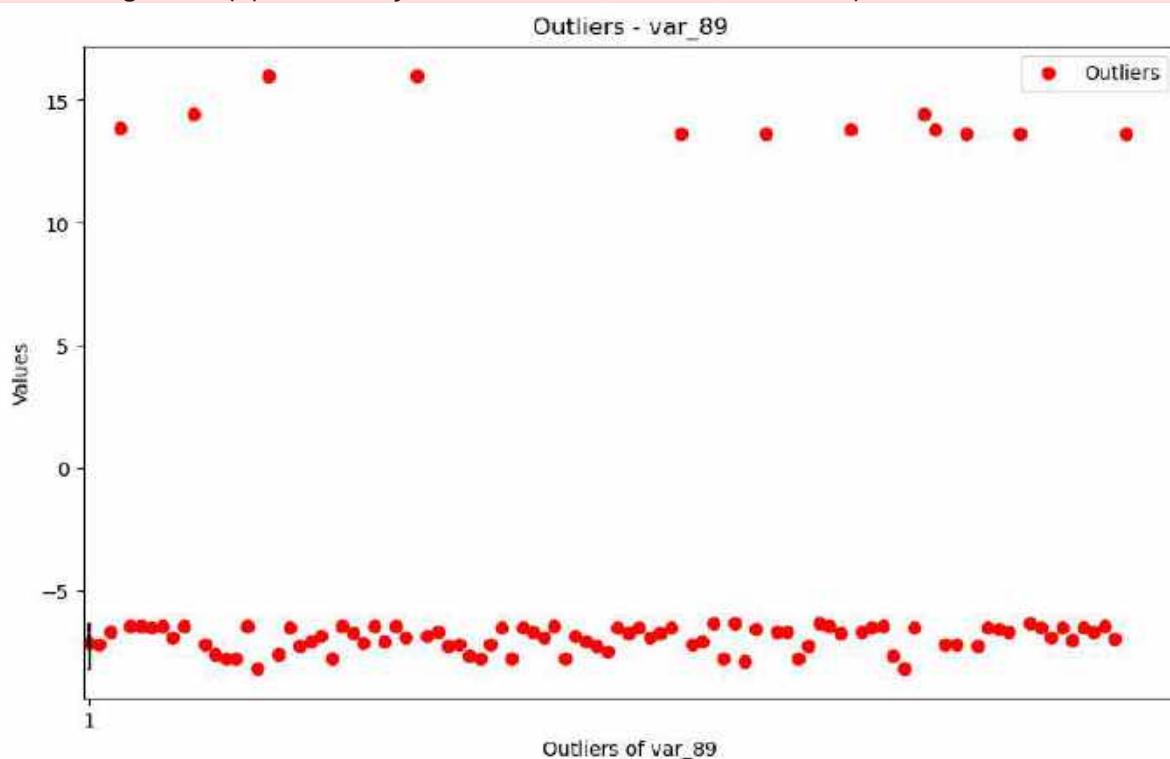
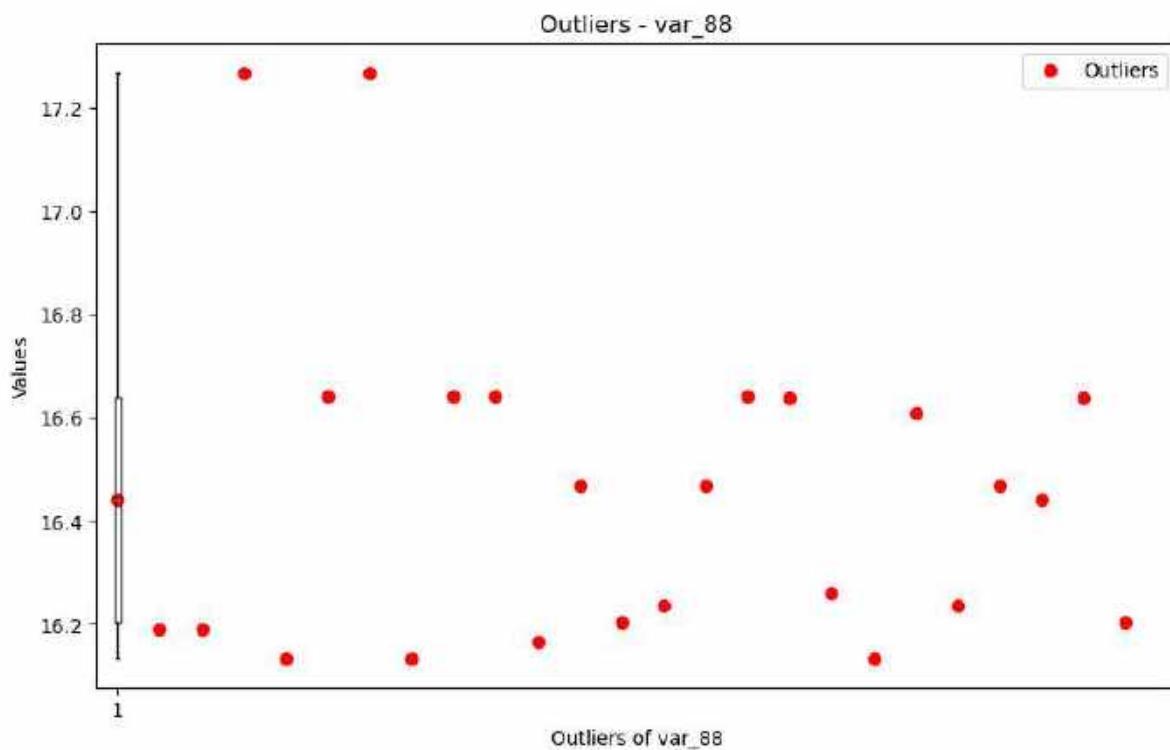
Total outliers in column var\_81: 174

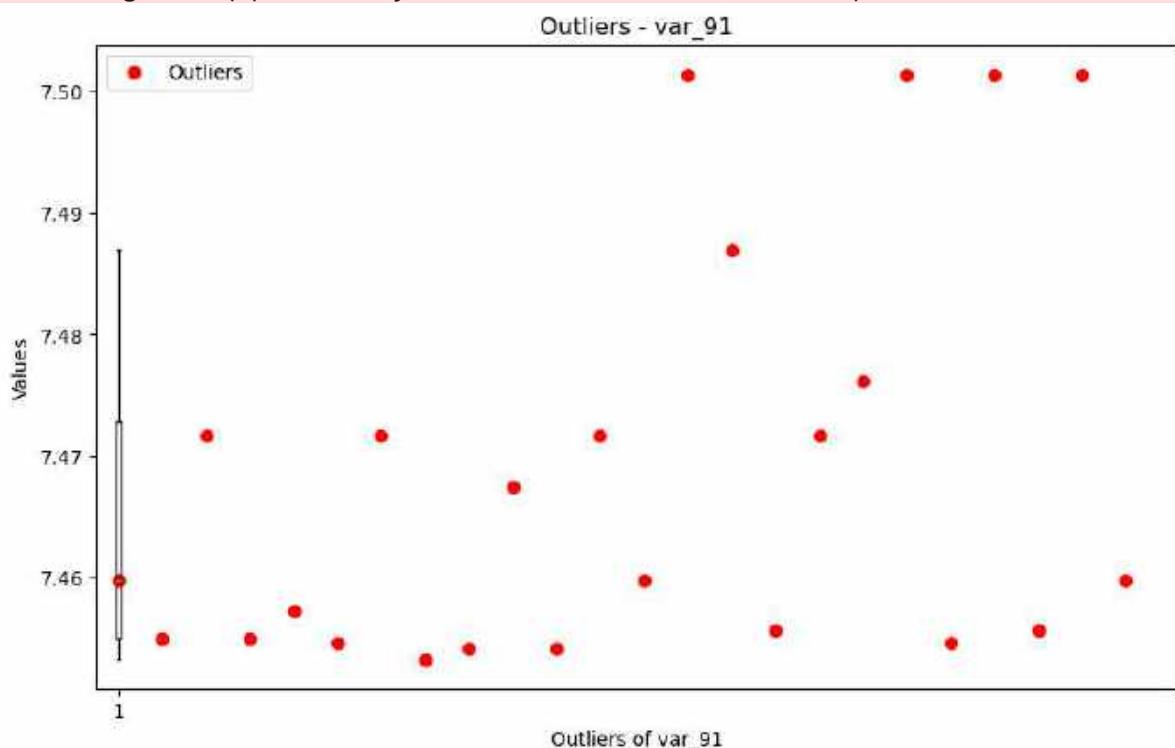
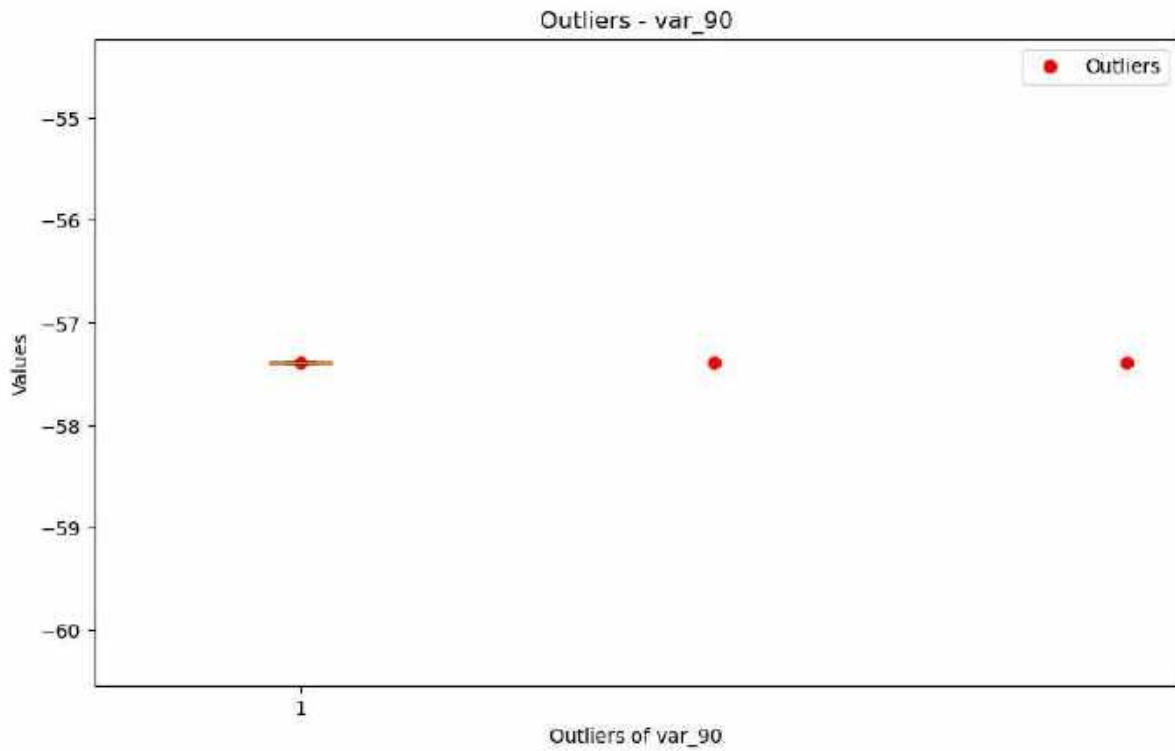
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

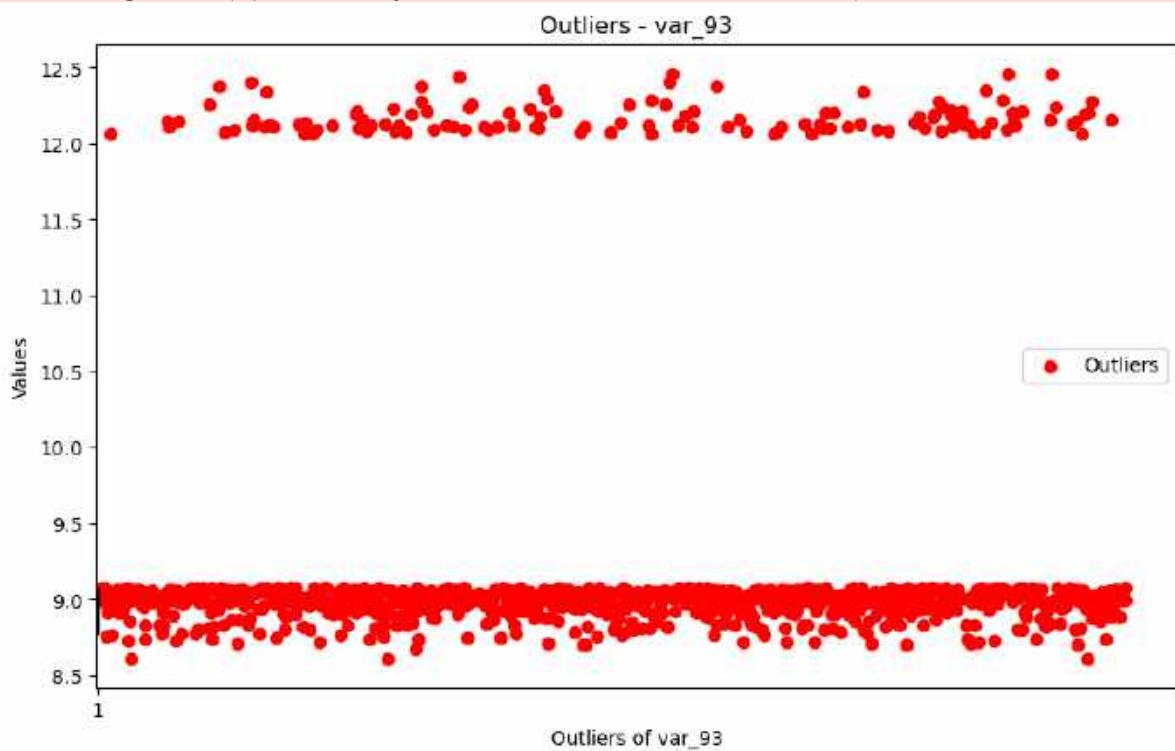
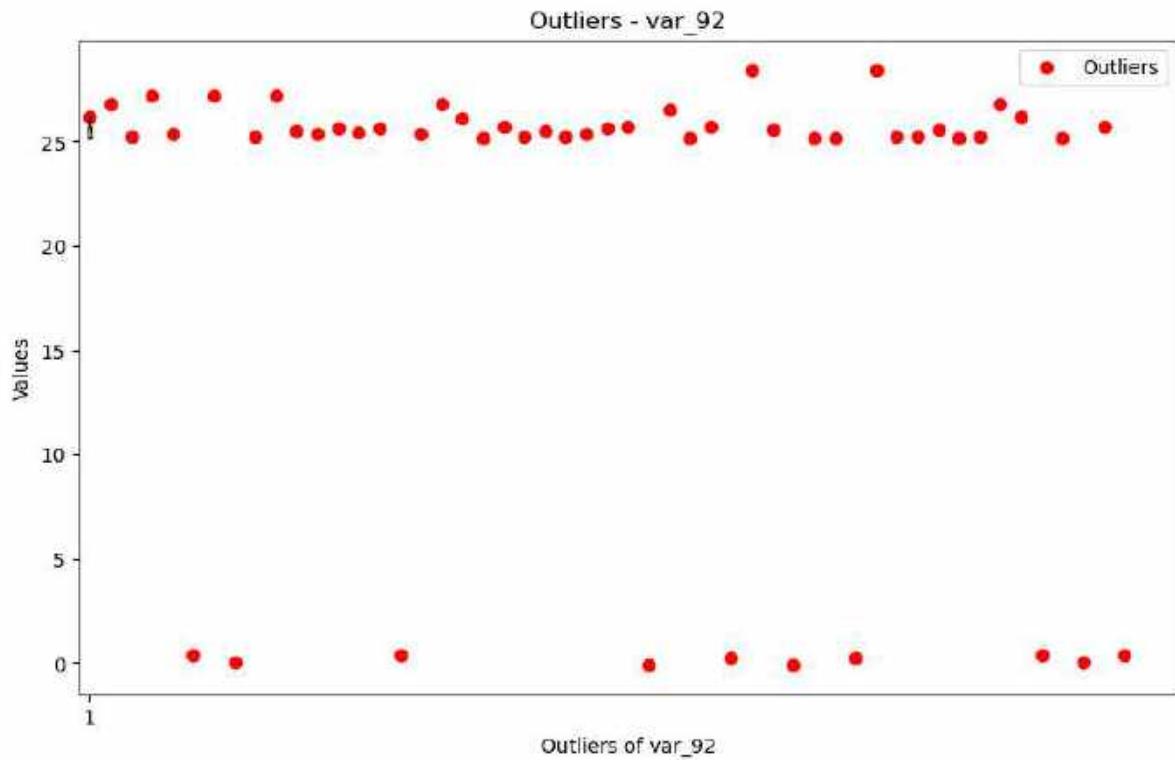


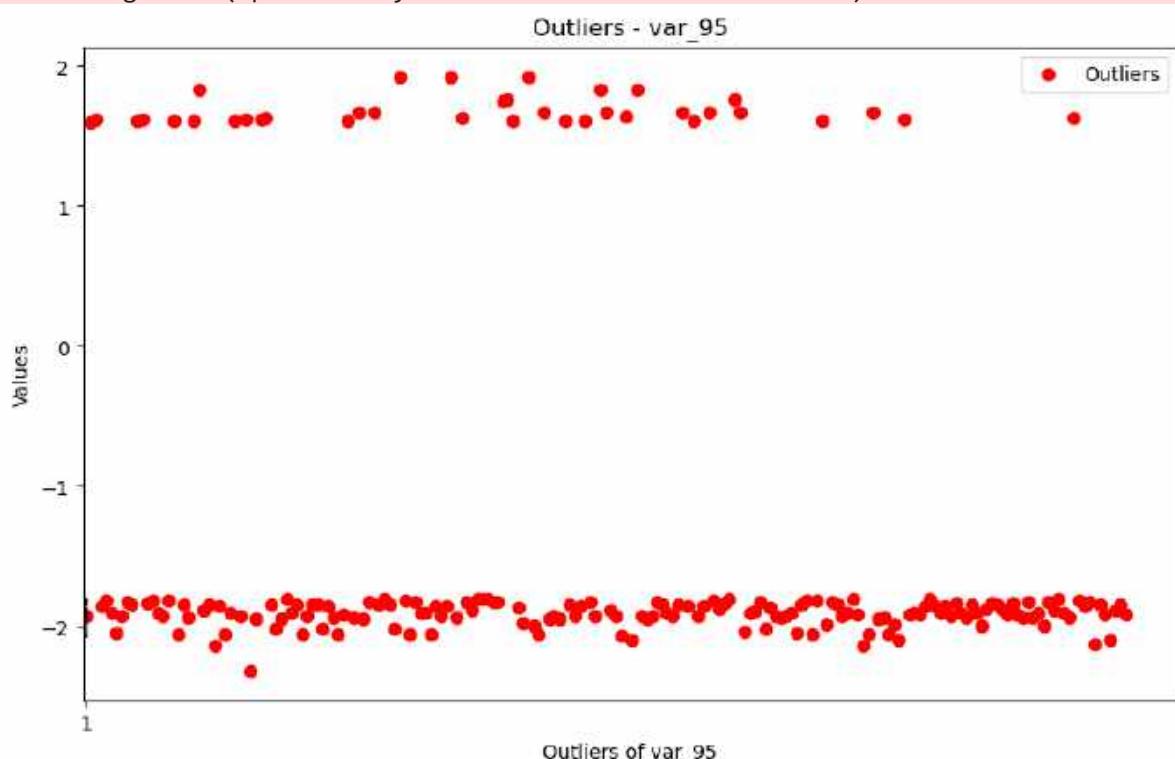
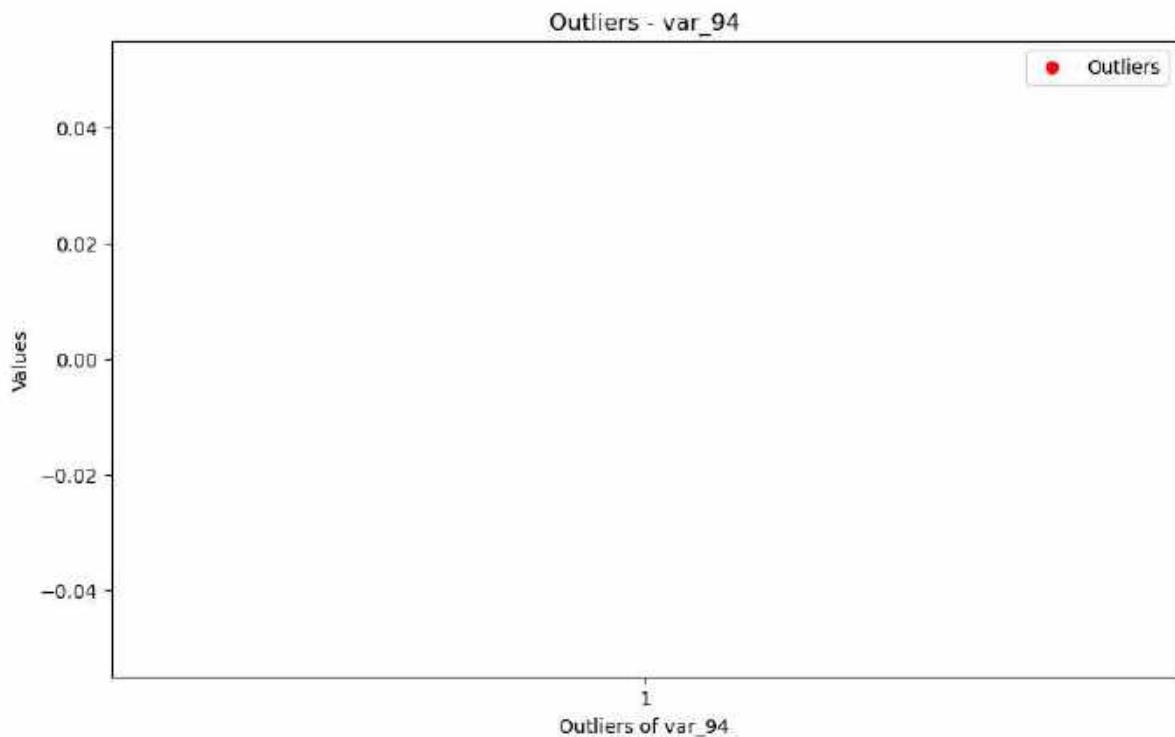


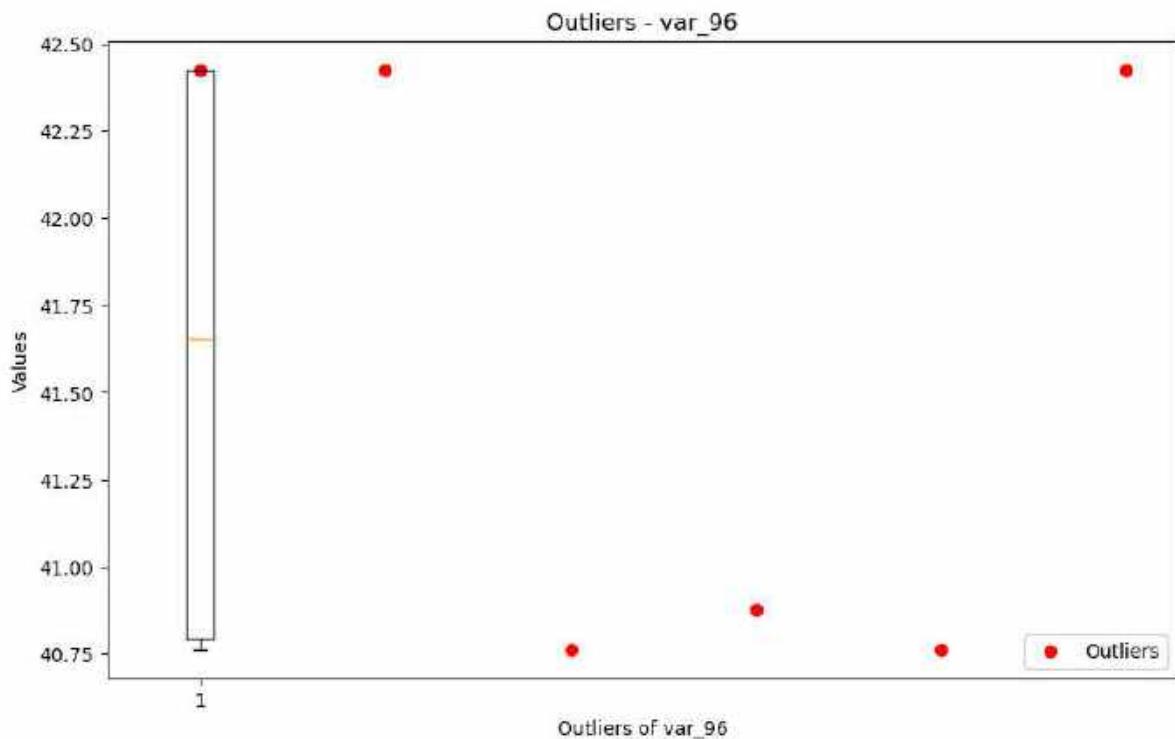




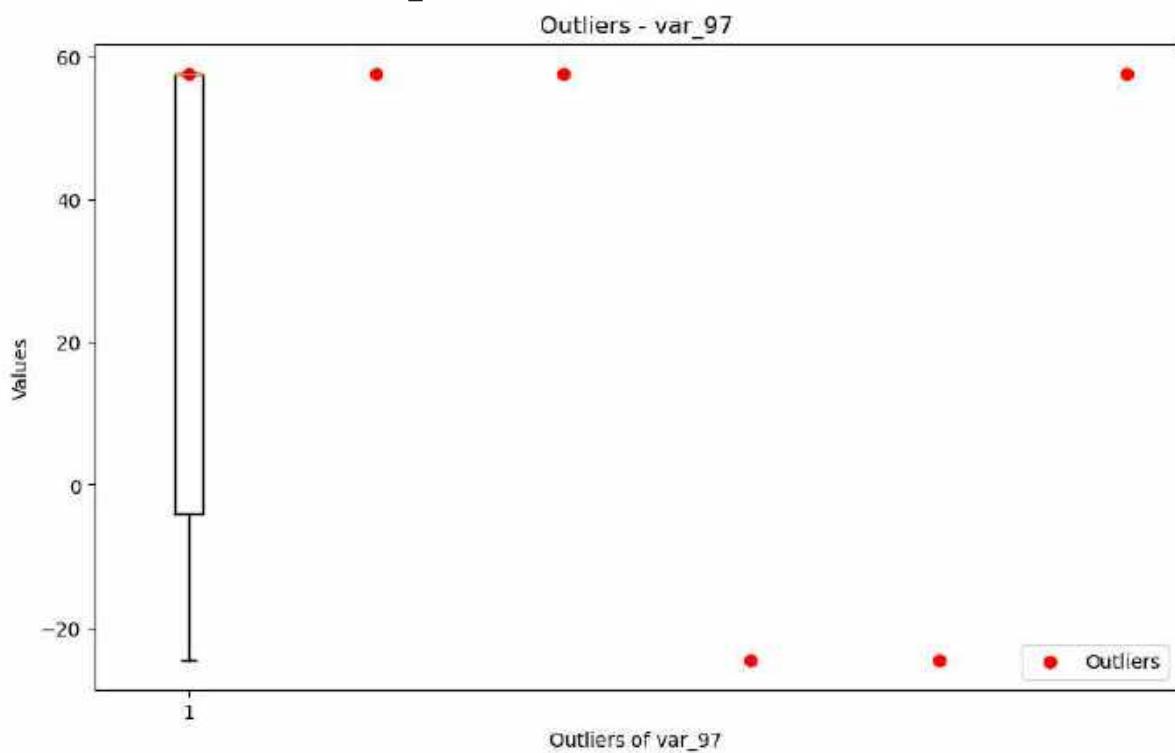






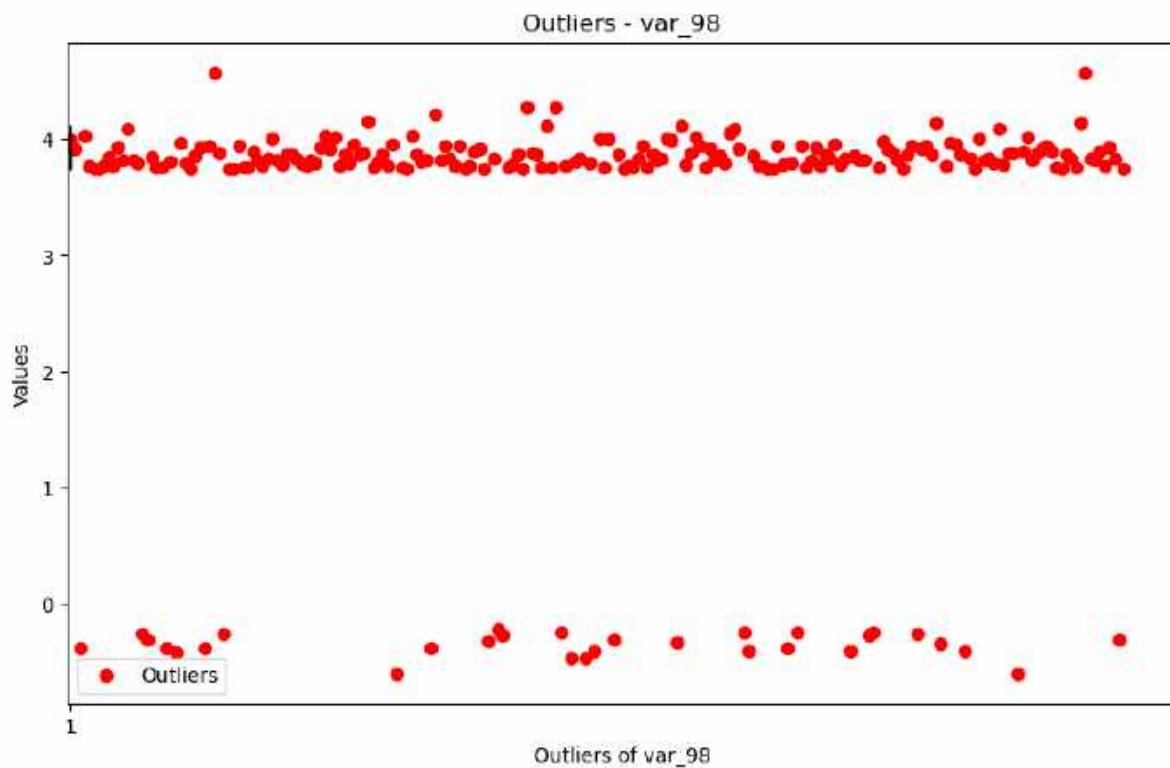


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_96: 6
```



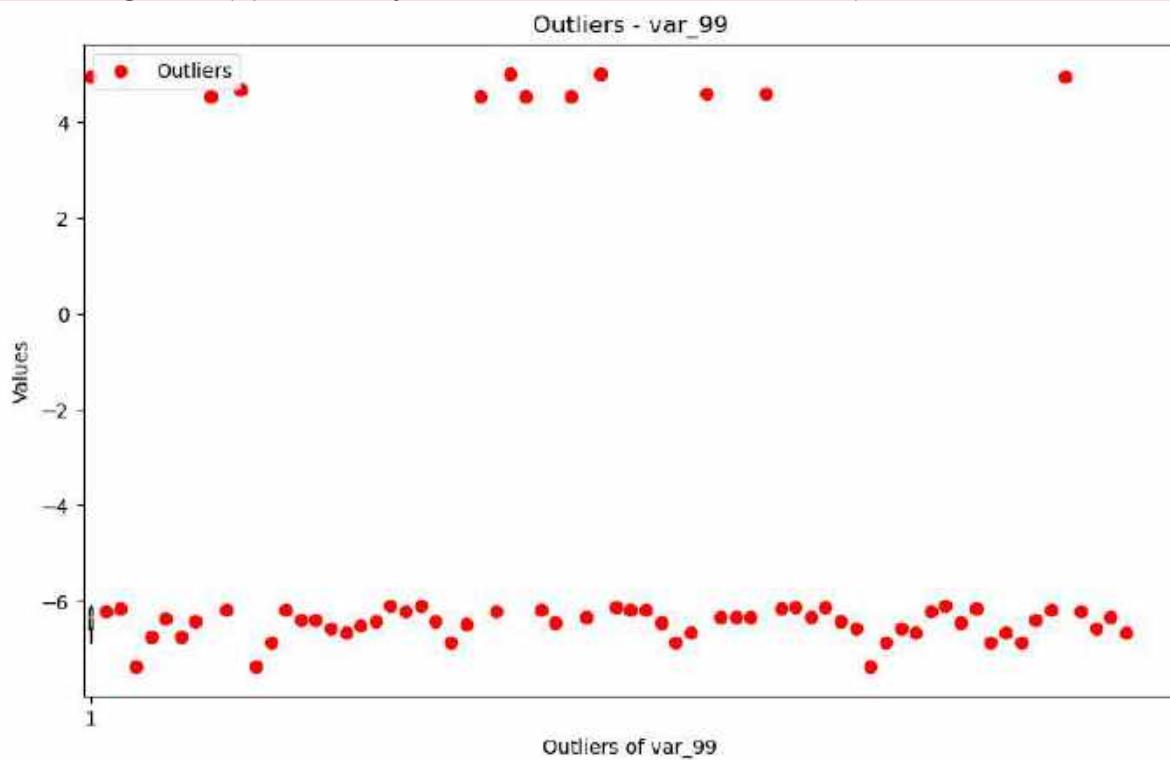
```
Total outliers in column var_97: 6
```

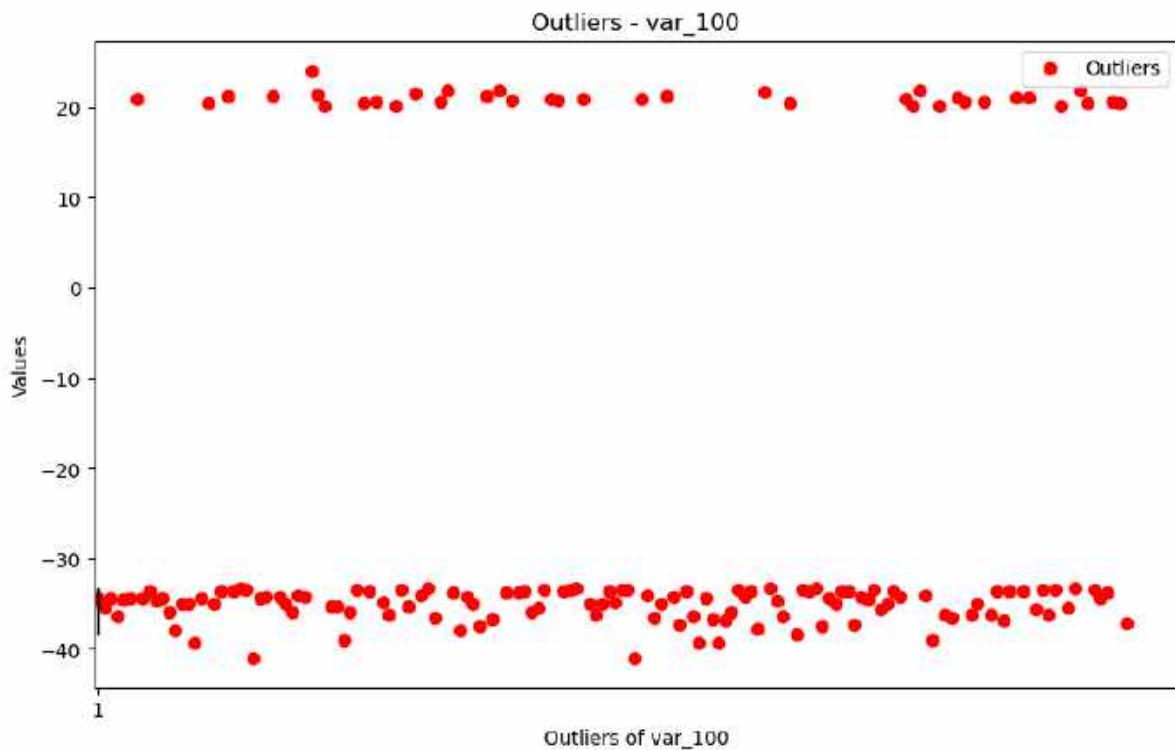
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_98: 220

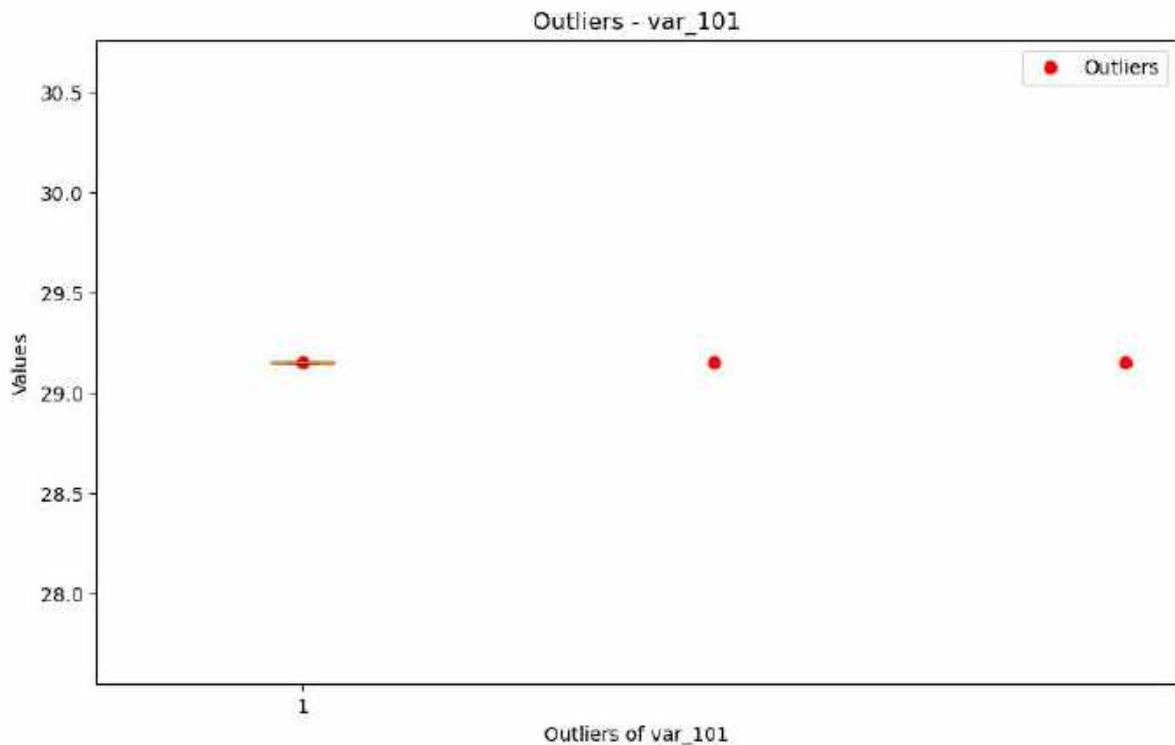
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





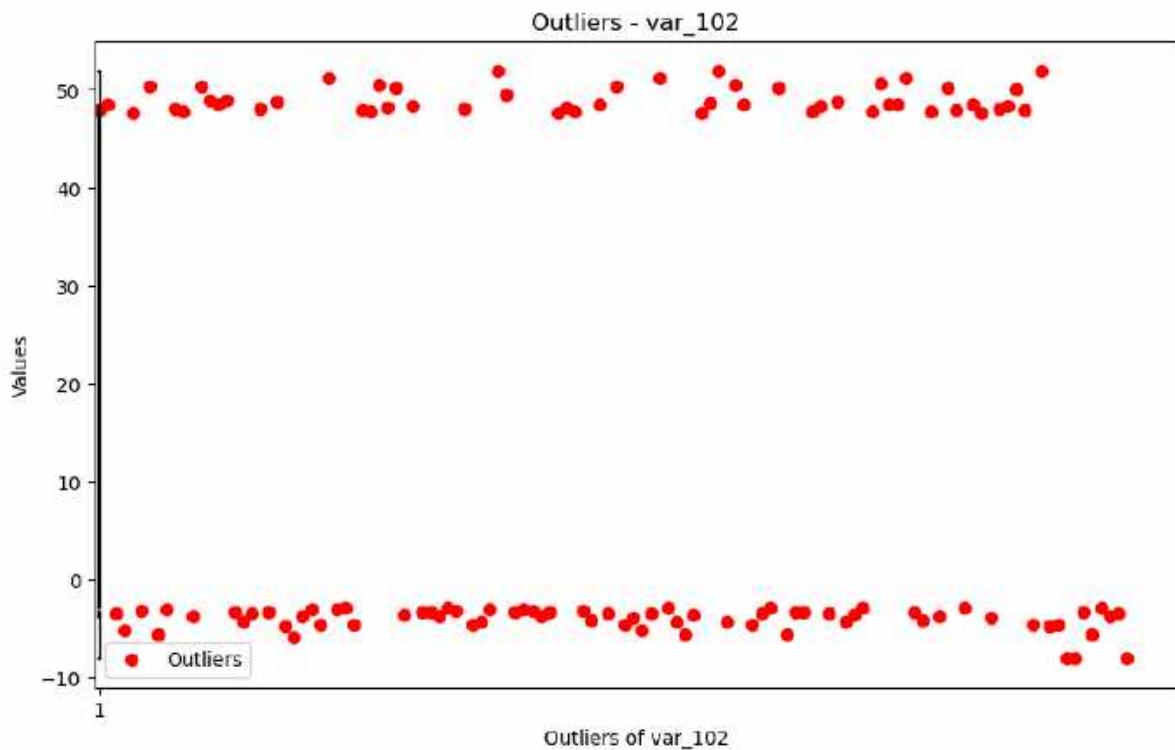
Total outliers in column var\_100: 160

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



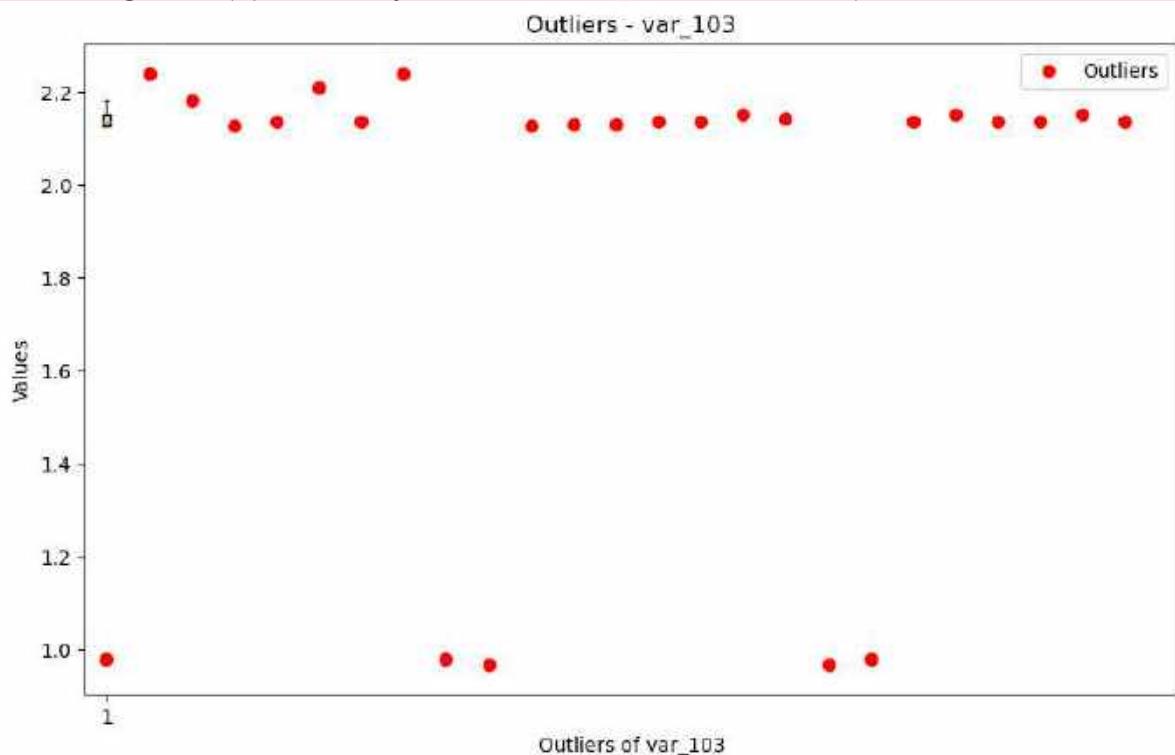
Total outliers in column var\_101: 3

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



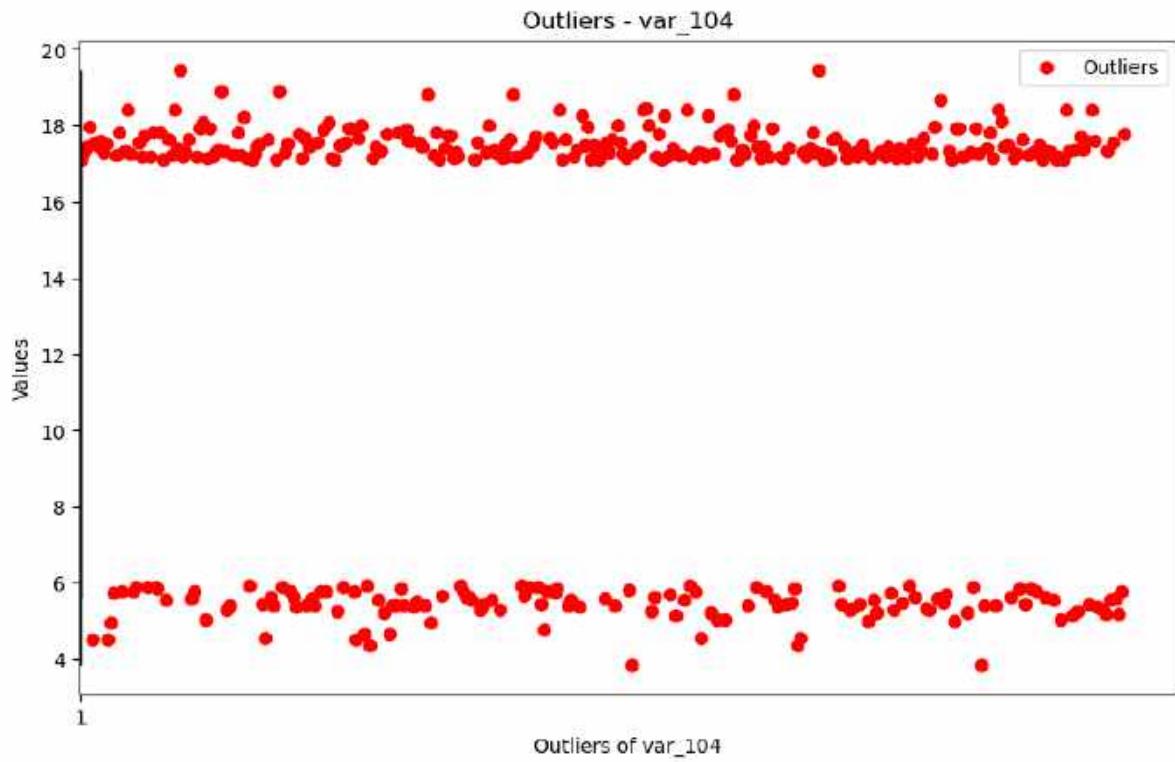
Total outliers in column var\_102: 122

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



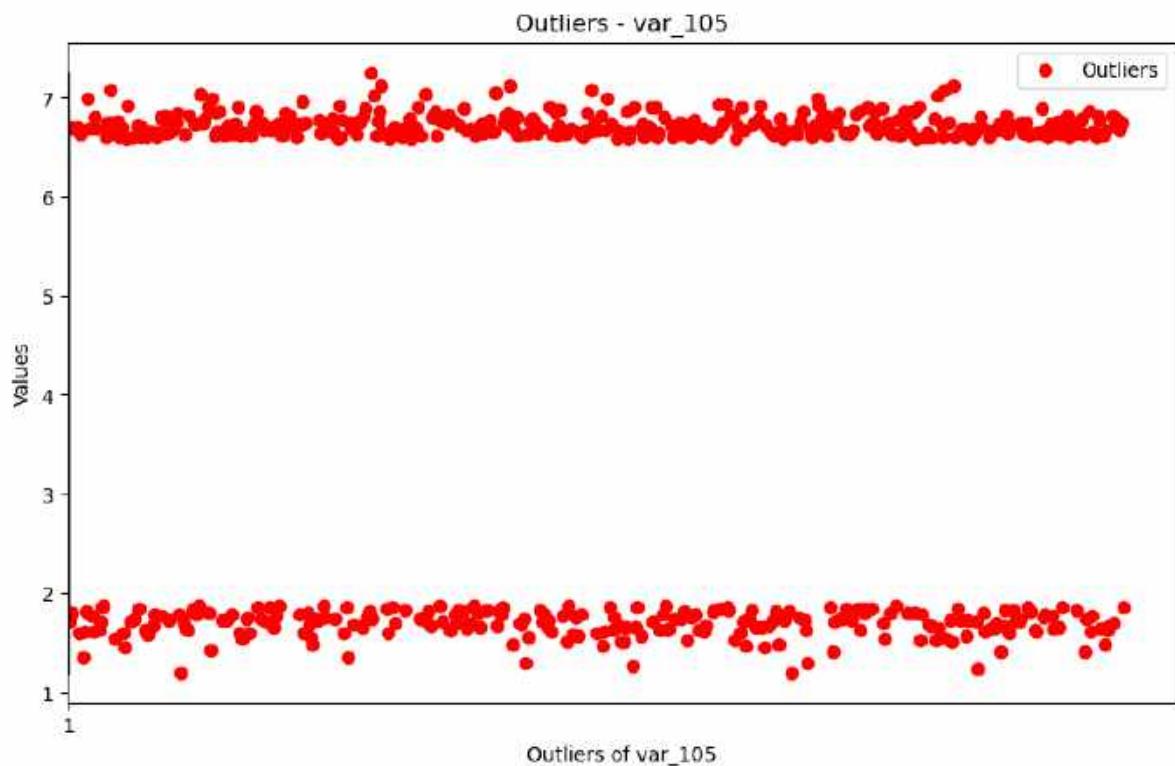
Total outliers in column var\_103: 25

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



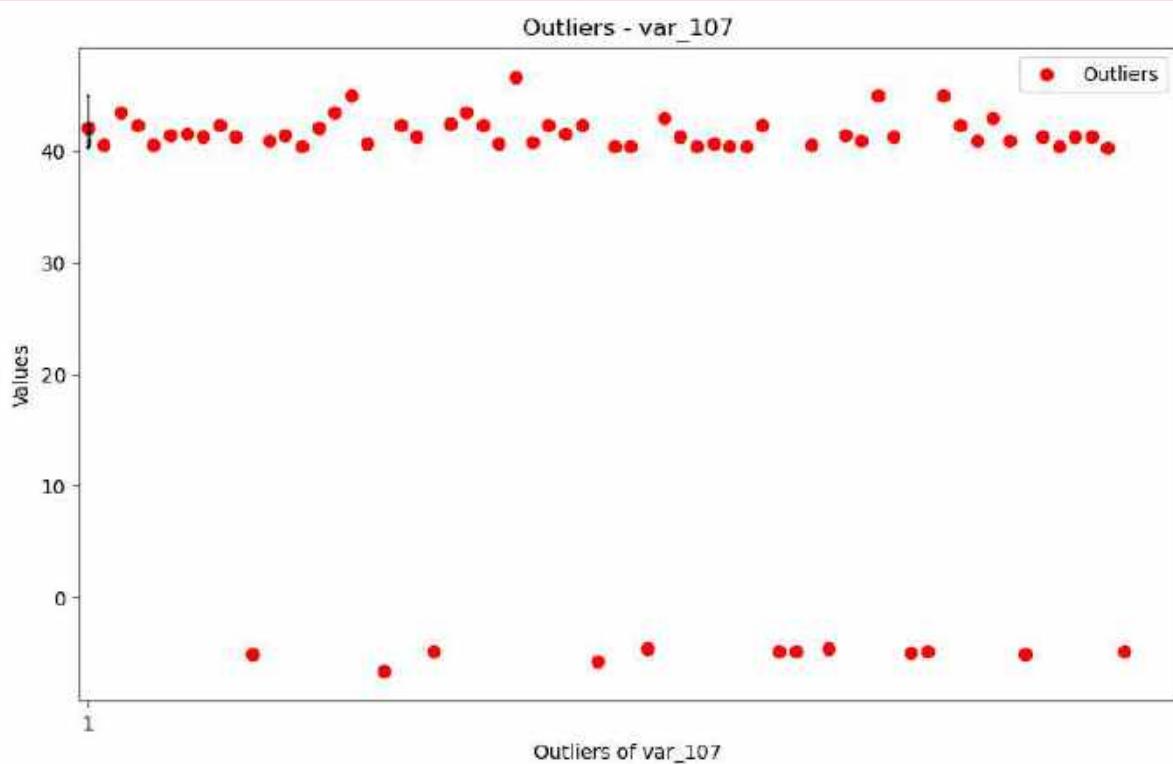
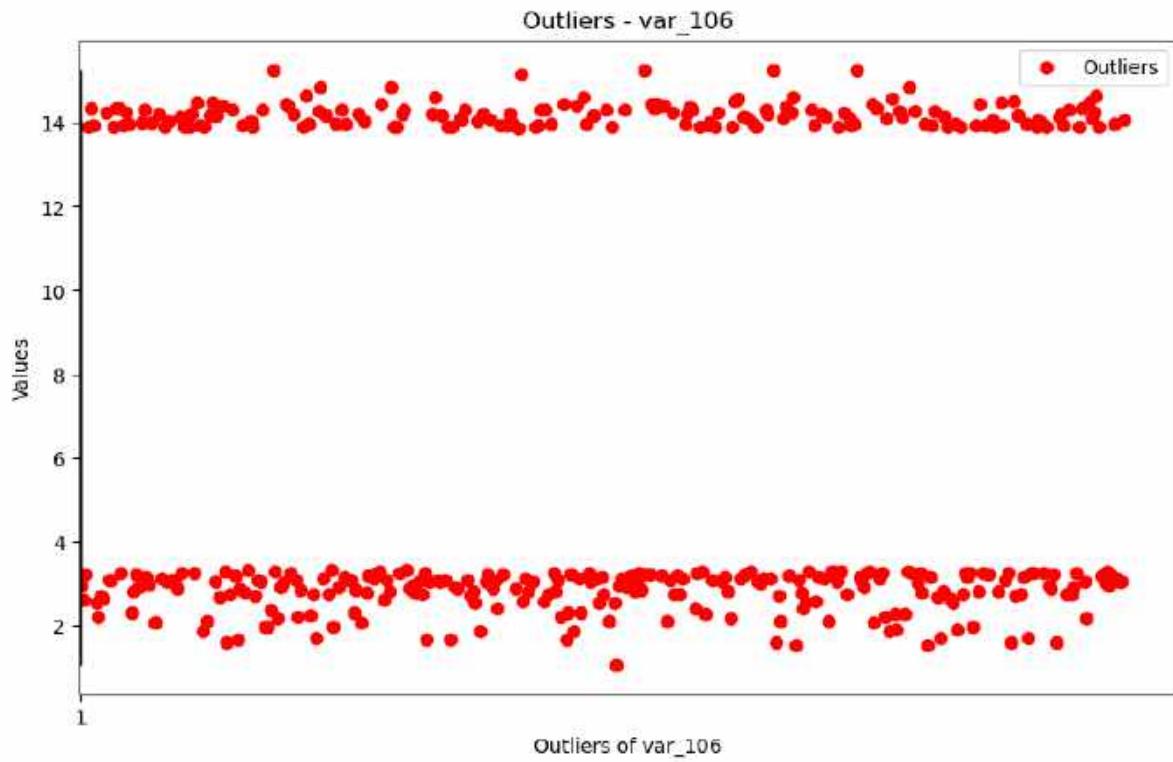
Total outliers in column var\_104: 380

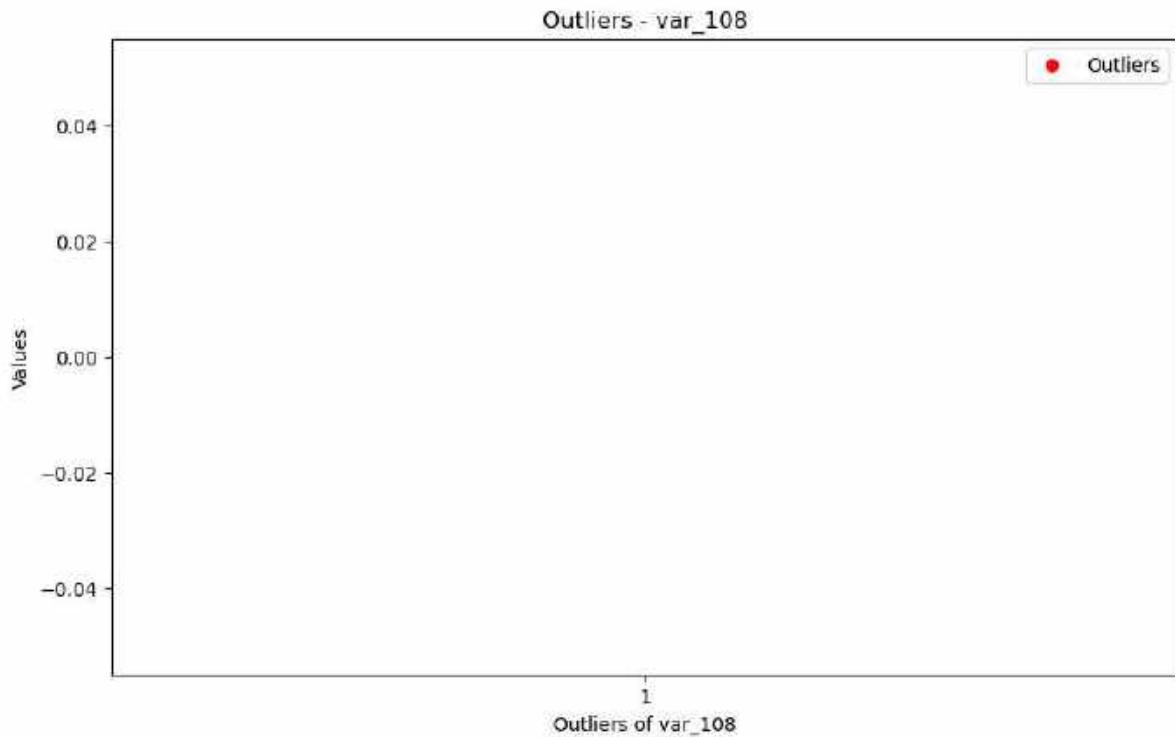
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_105: 602

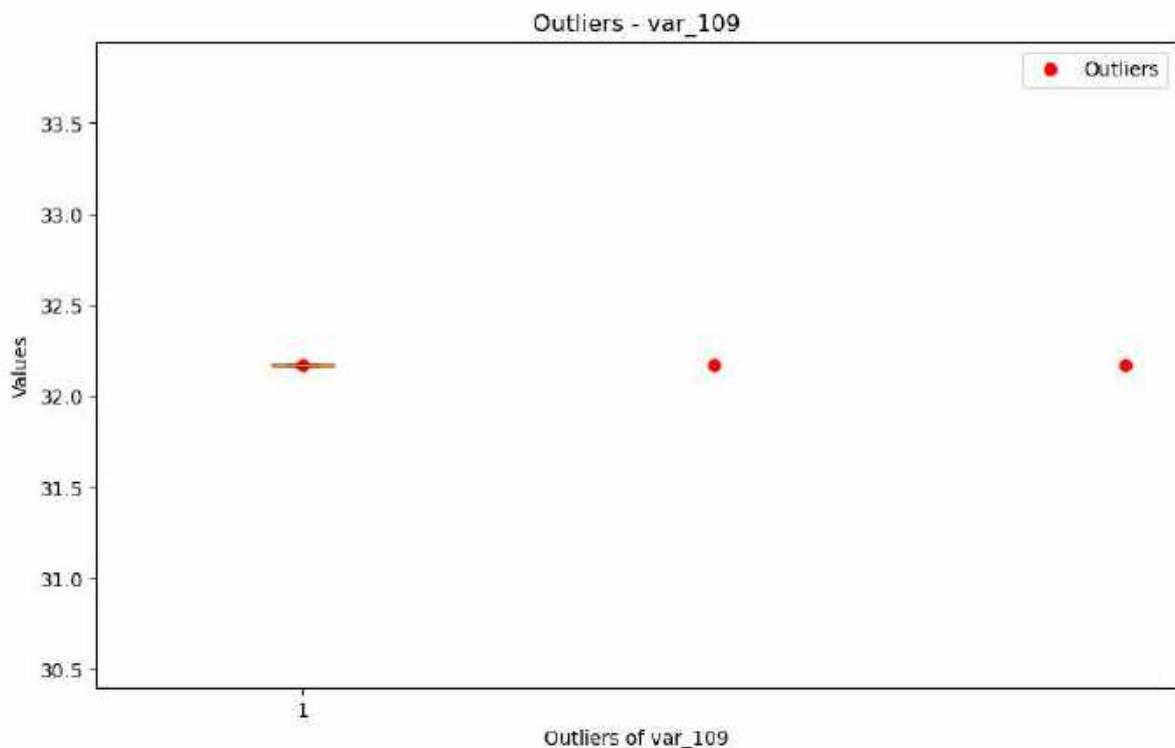
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





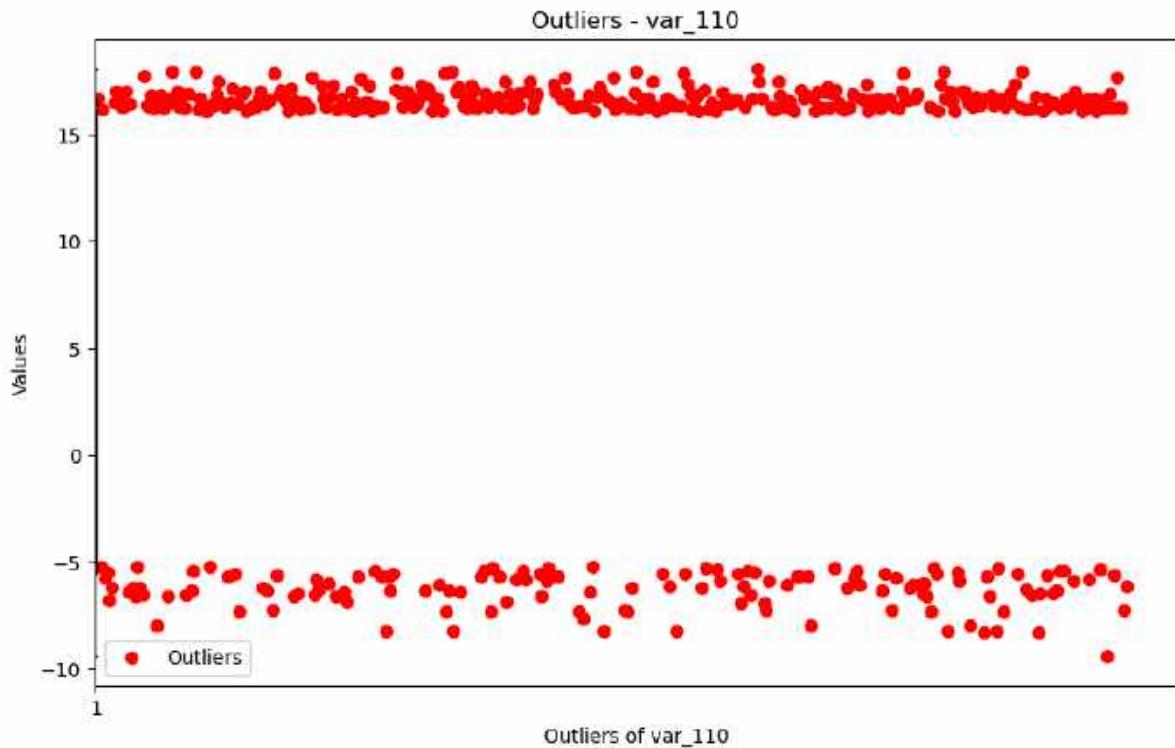
```
Total outliers in column var_108: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

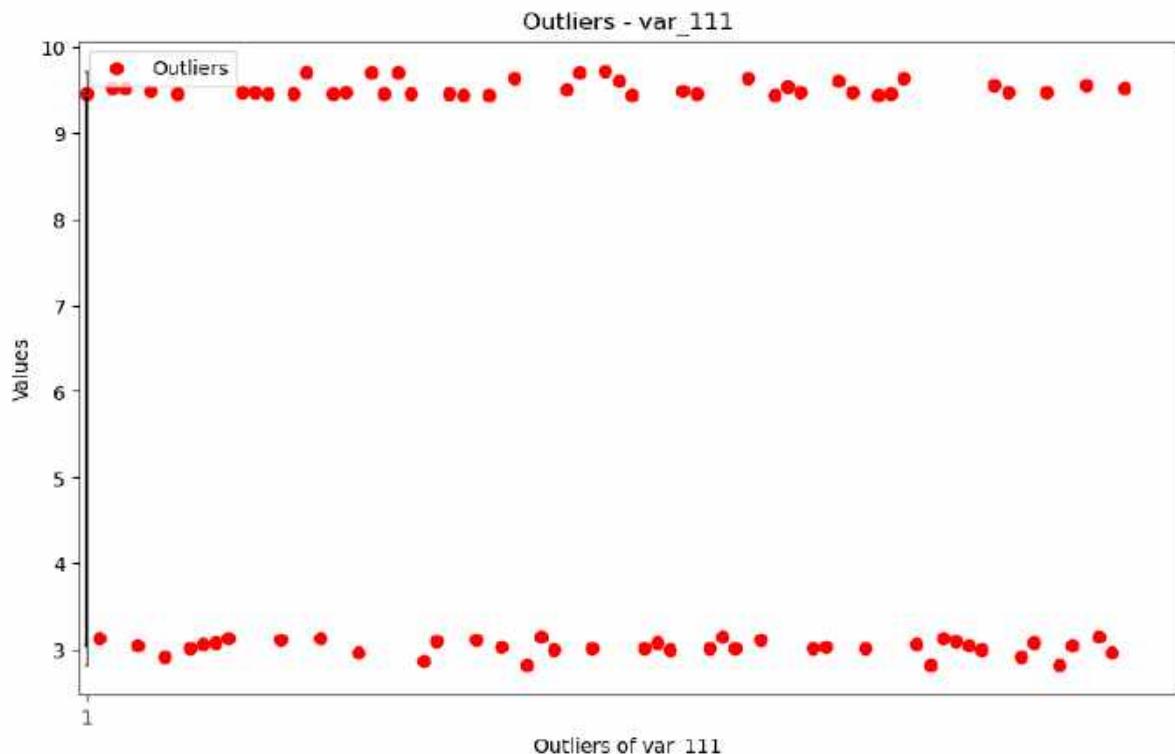


```
Total outliers in column var_109: 3
```

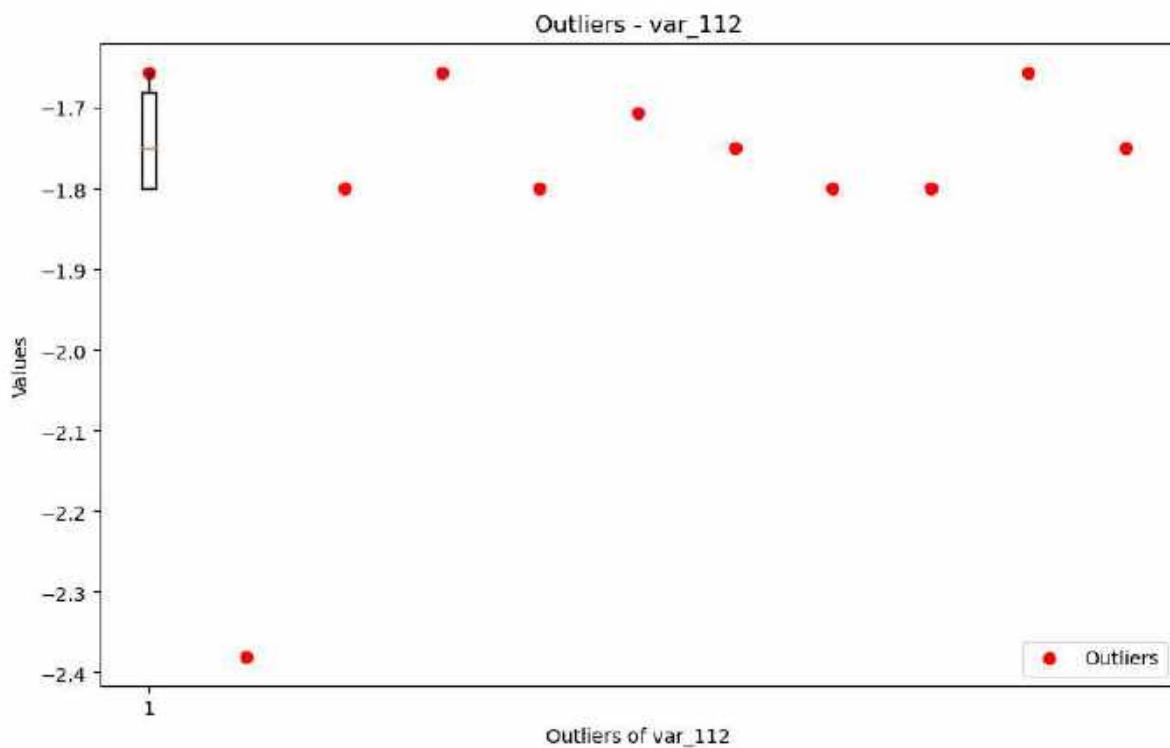
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_110: 445
```

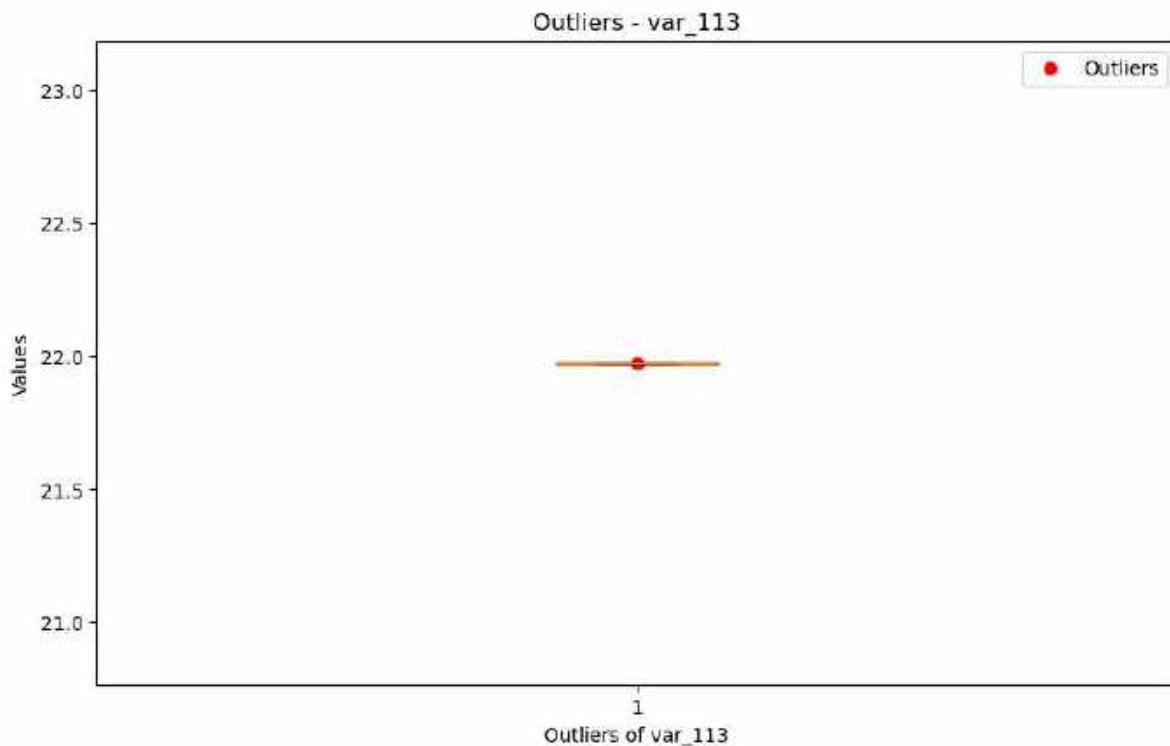


```
Total outliers in column var_111: 81  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



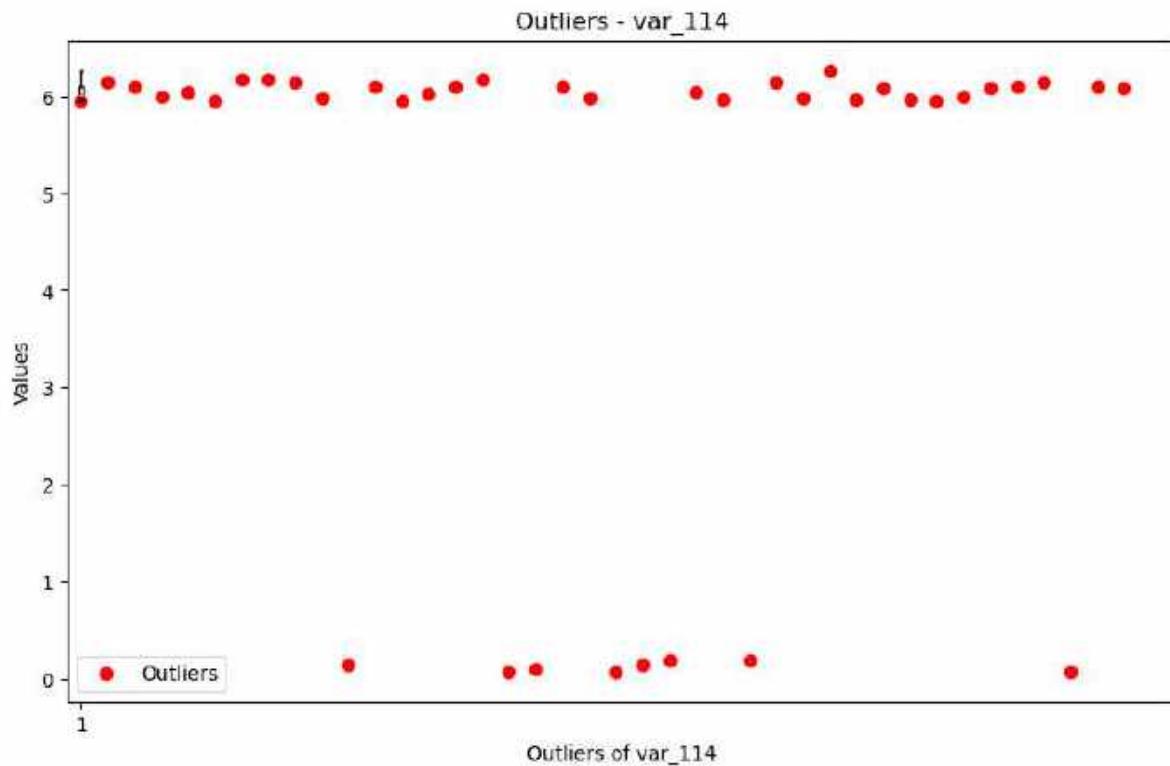
Total outliers in column var\_112: 11

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



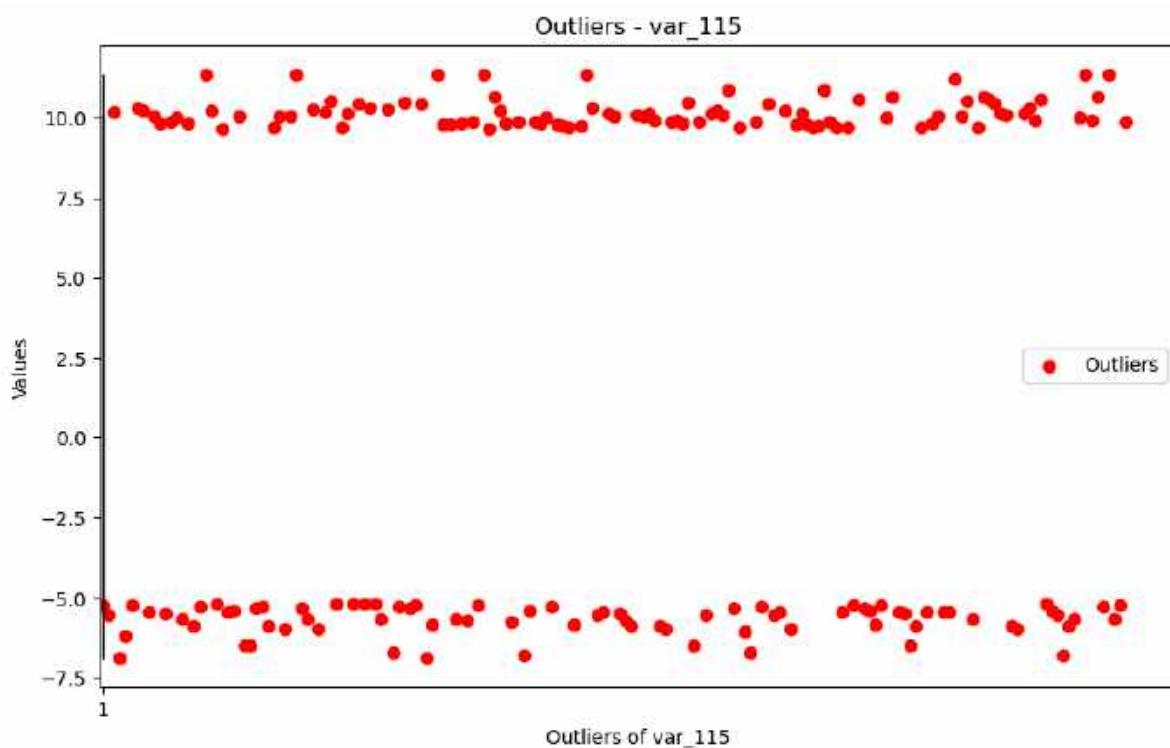
Total outliers in column var\_113: 1

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_114: 40

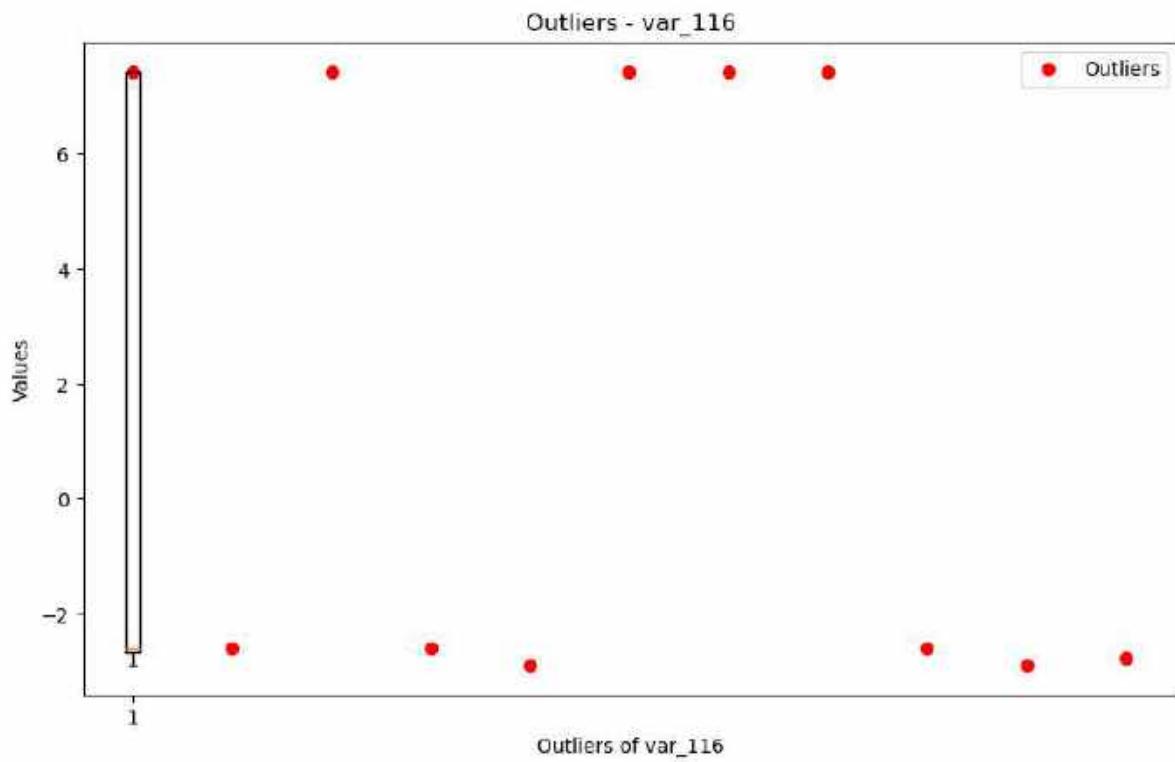
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.
```

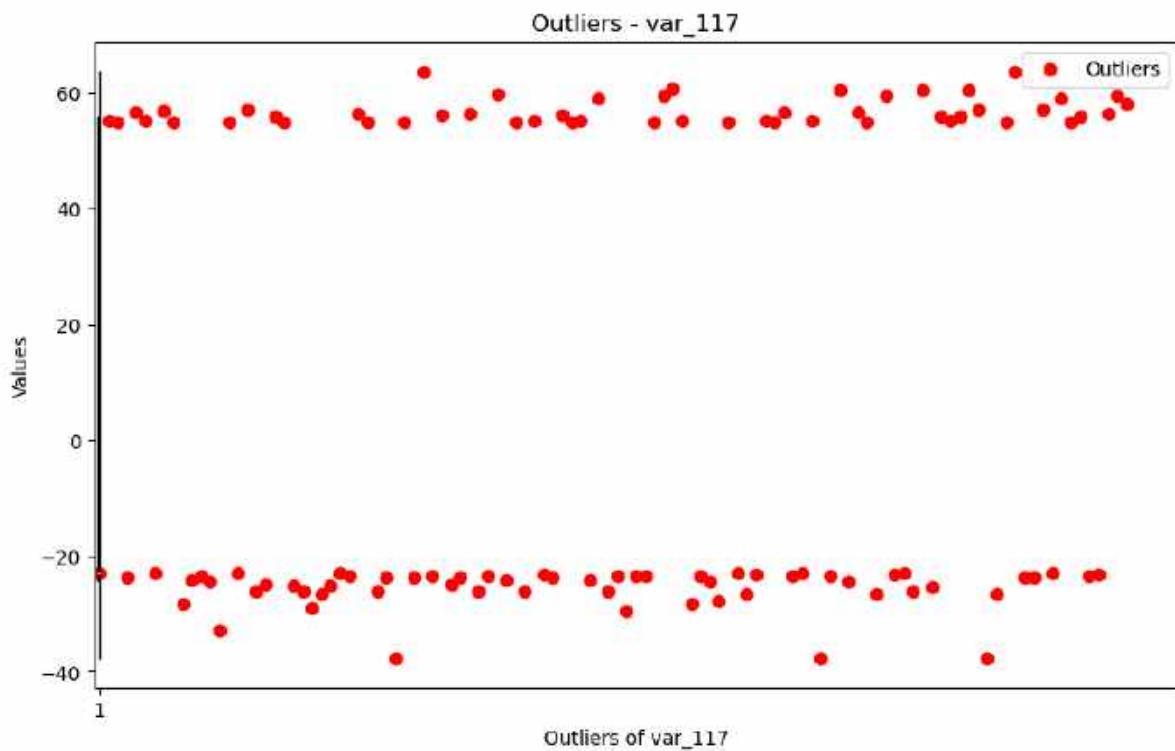
```
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_115: 181



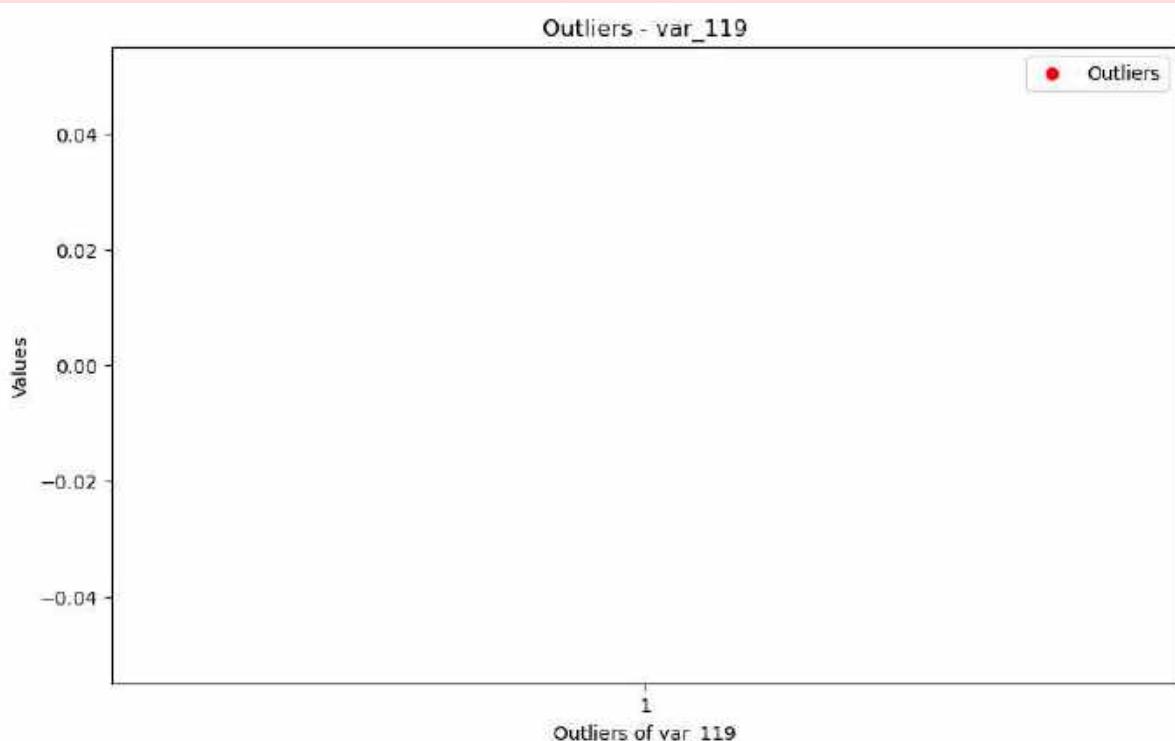
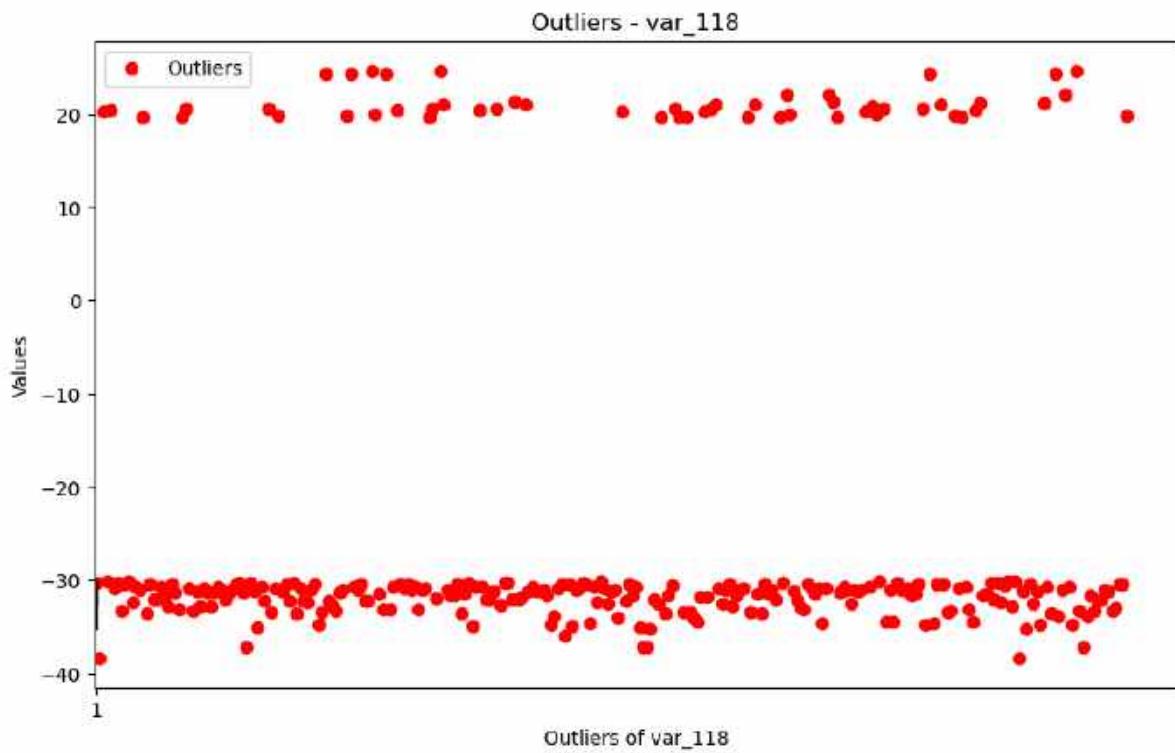
Total outliers in column var\_116: 11

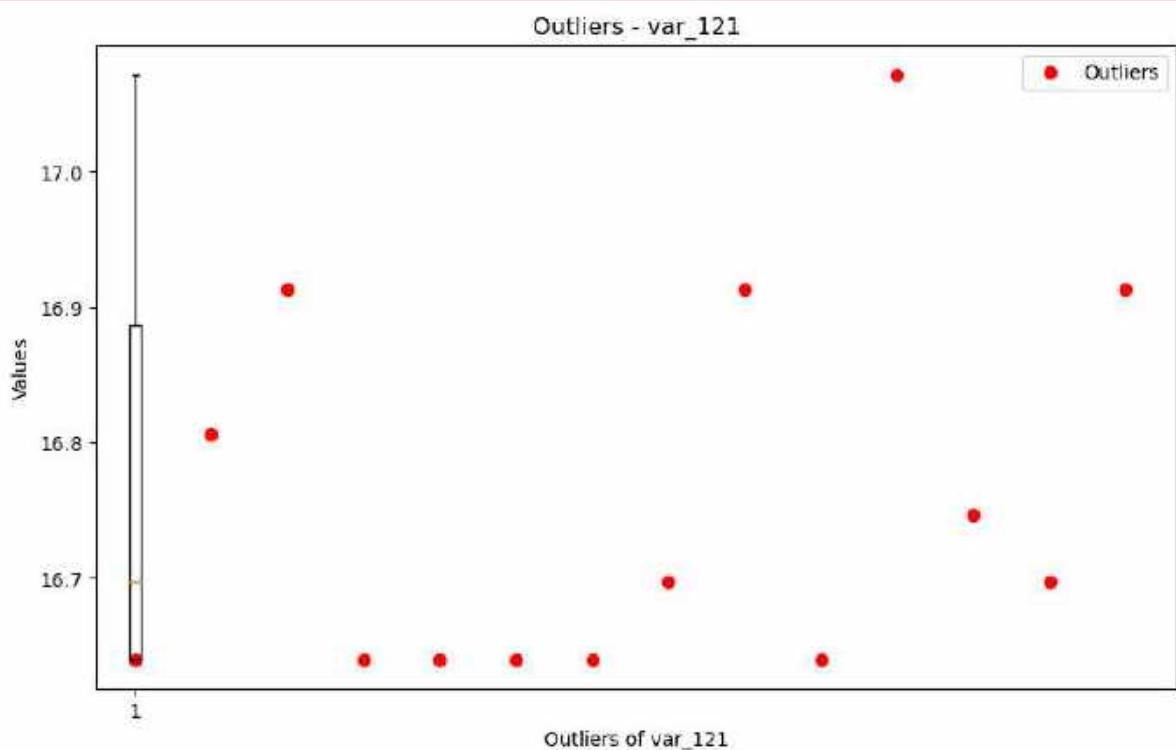
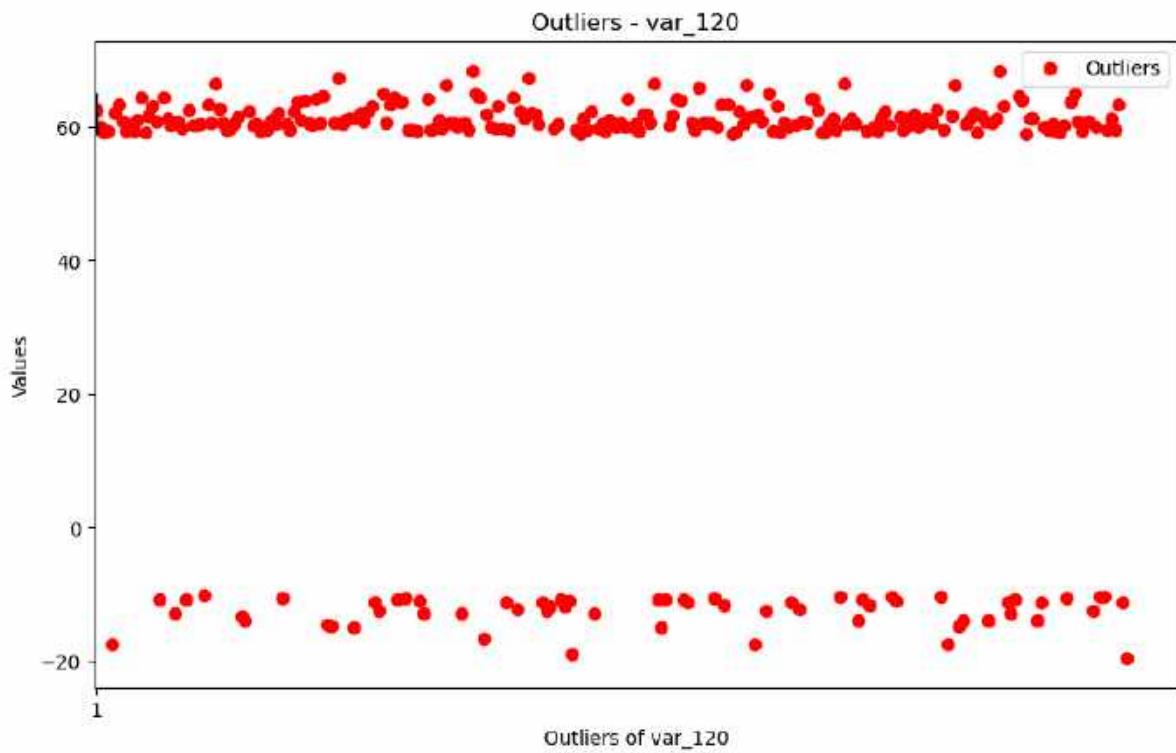
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

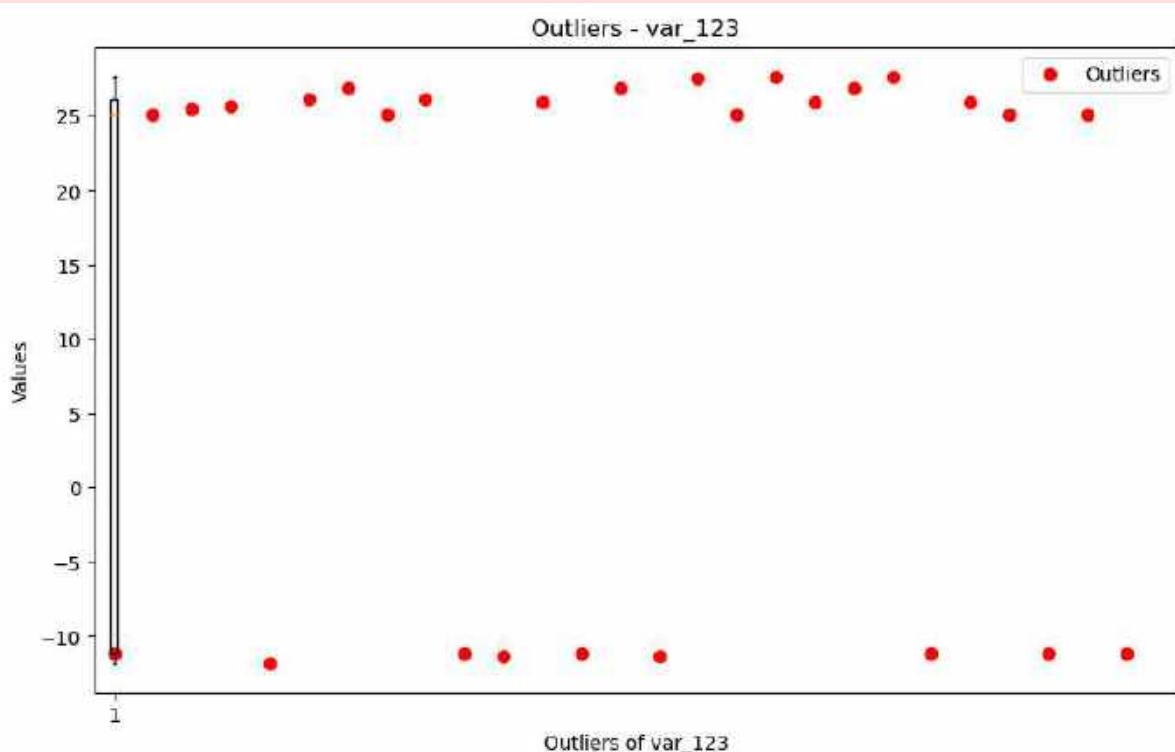
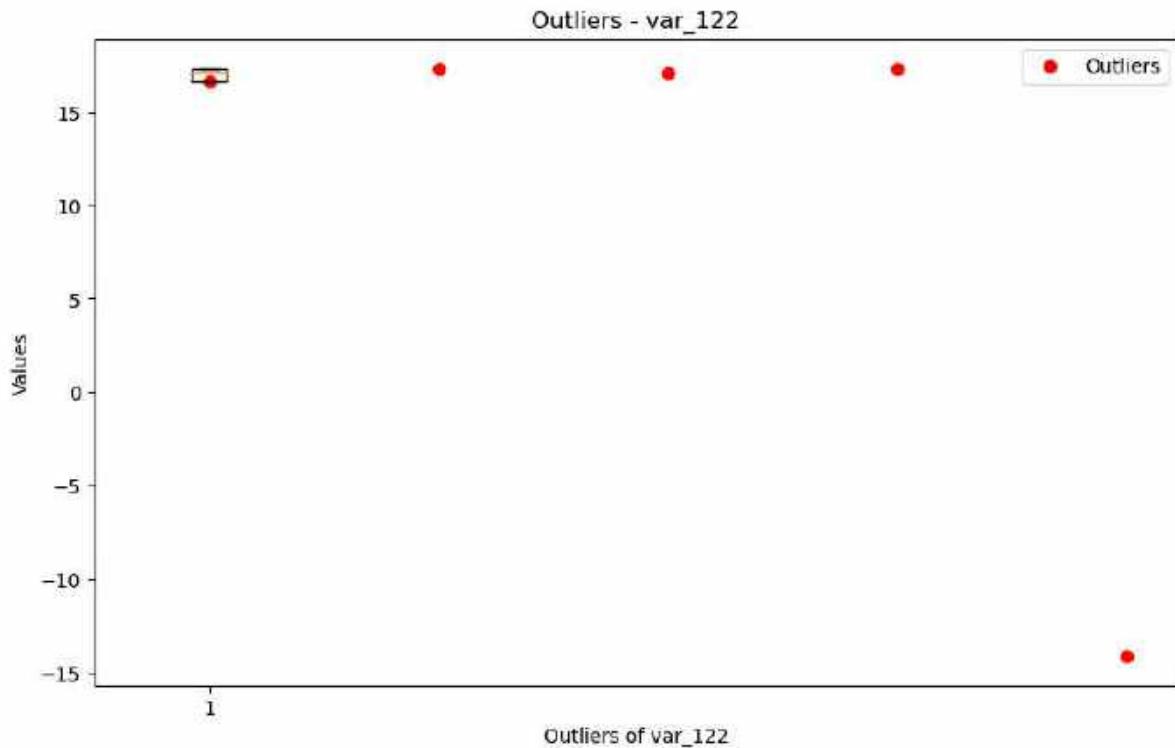


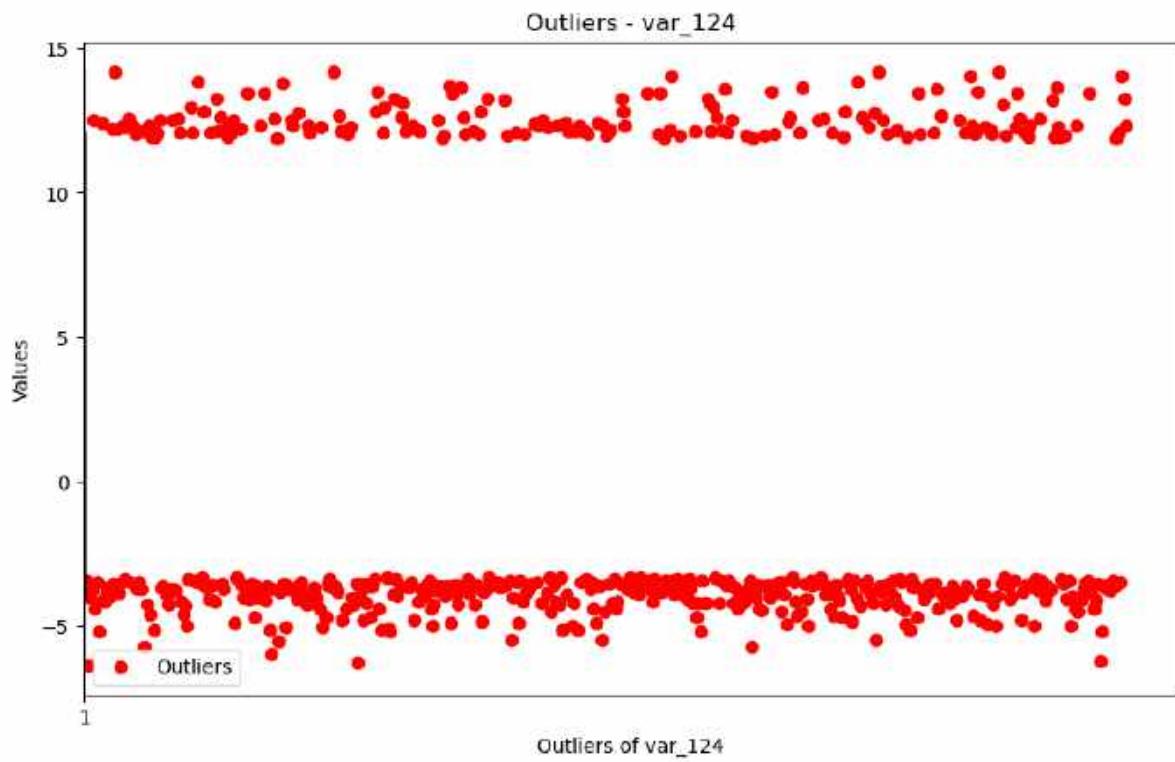
Total outliers in column var\_117: 112

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



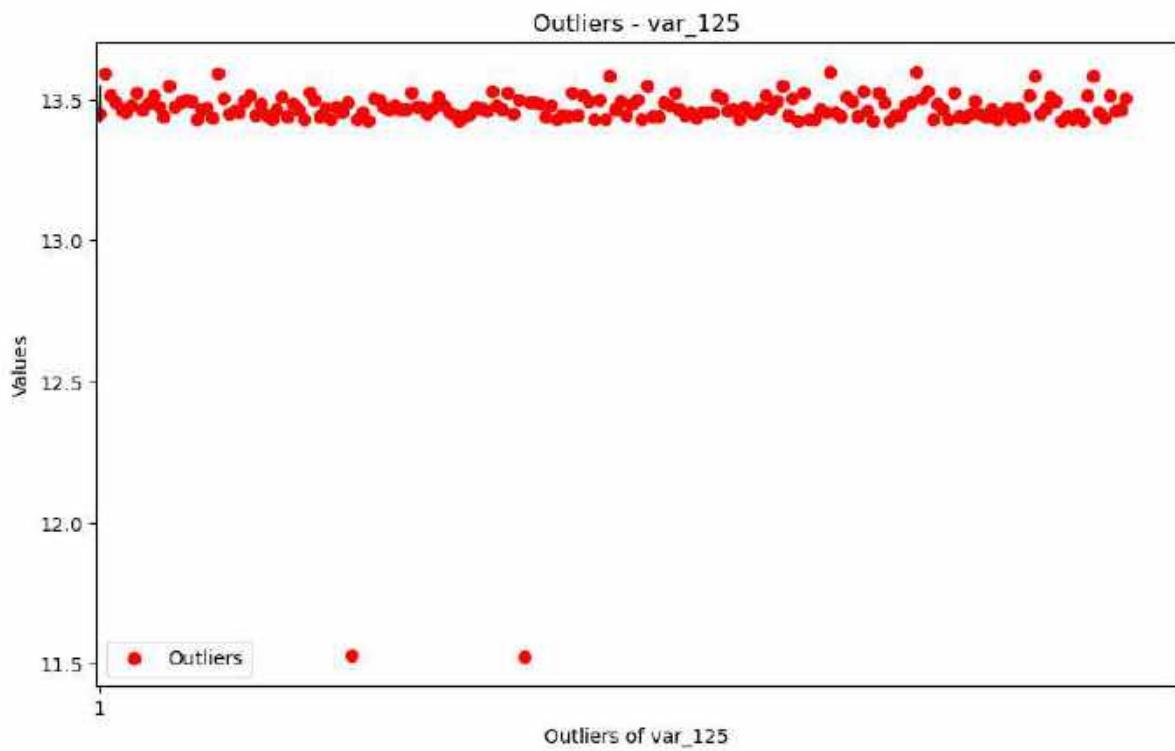






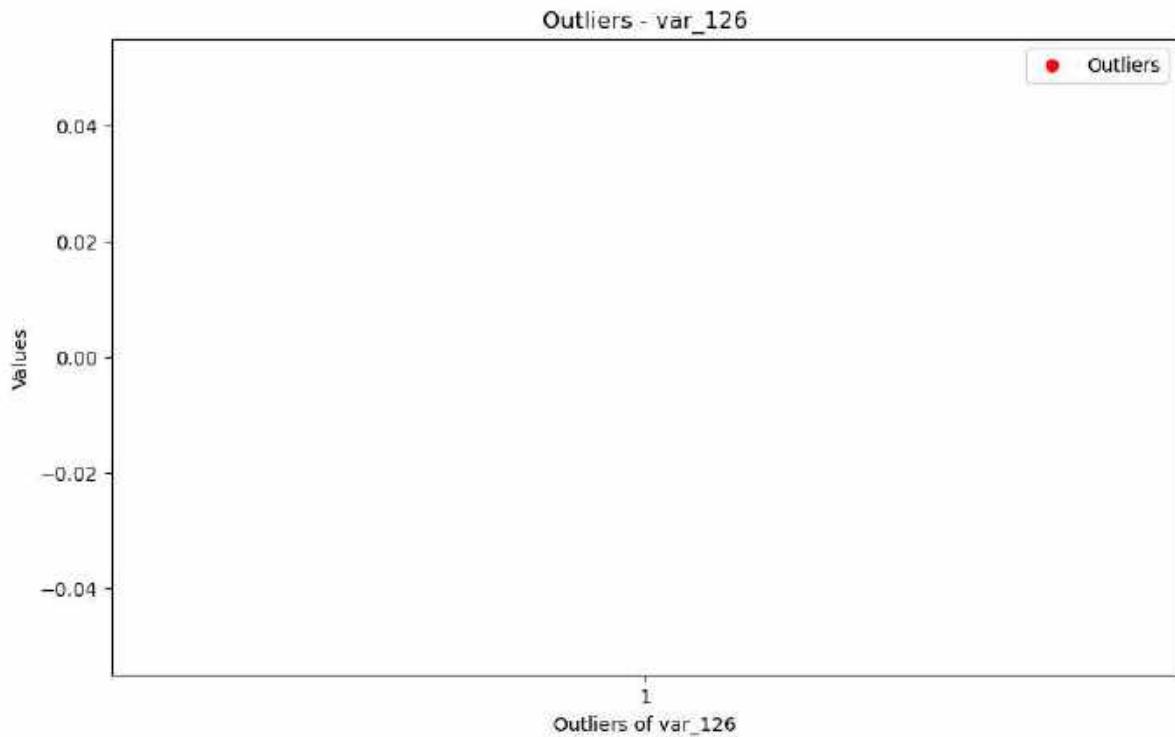
Total outliers in column var\_124: 569

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



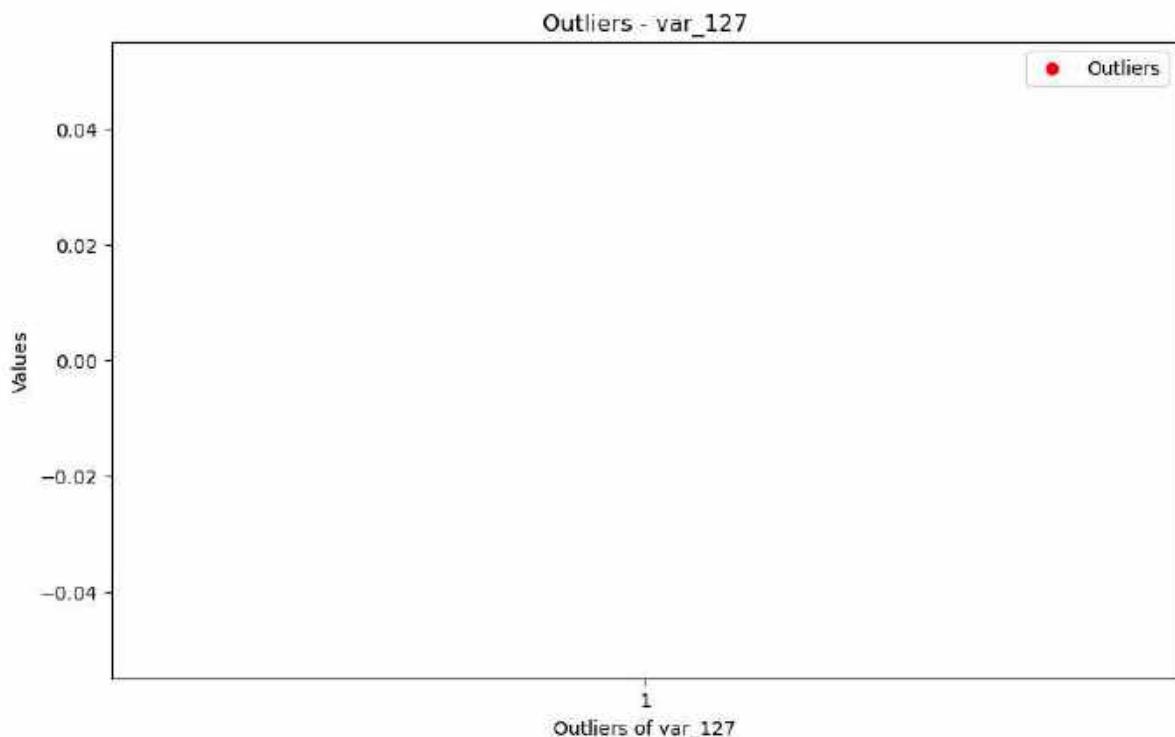
Total outliers in column var\_125: 192

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



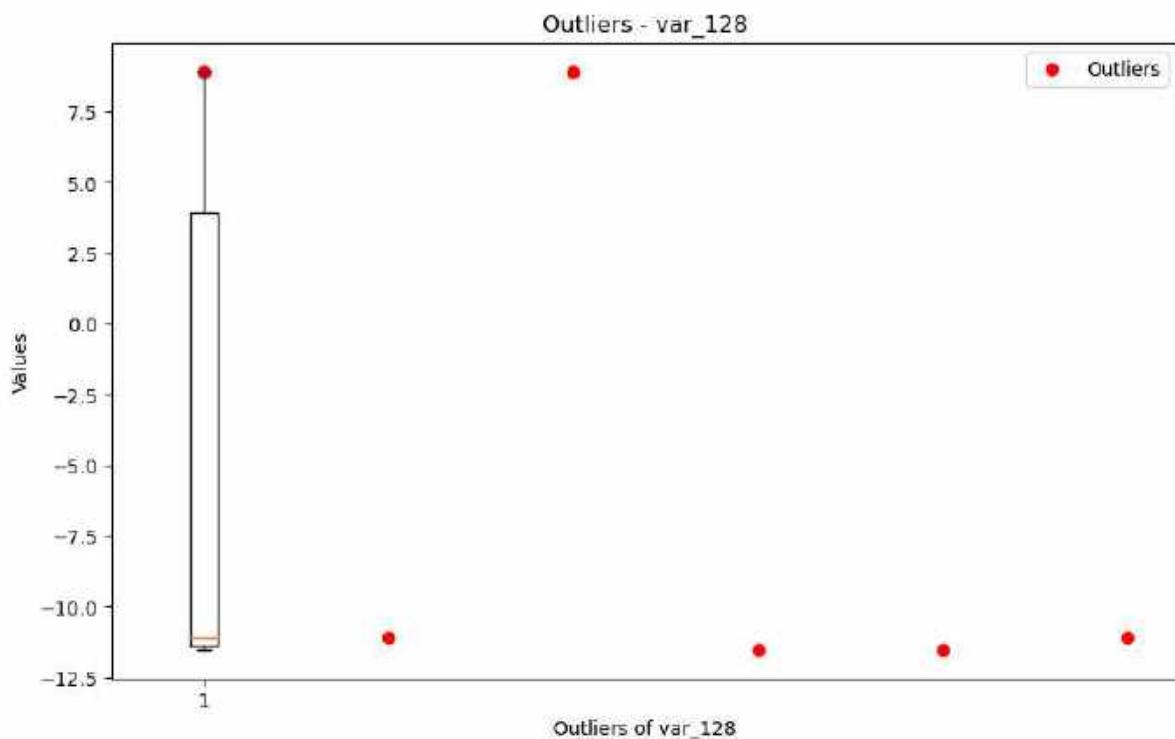
```
Total outliers in column var_126: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

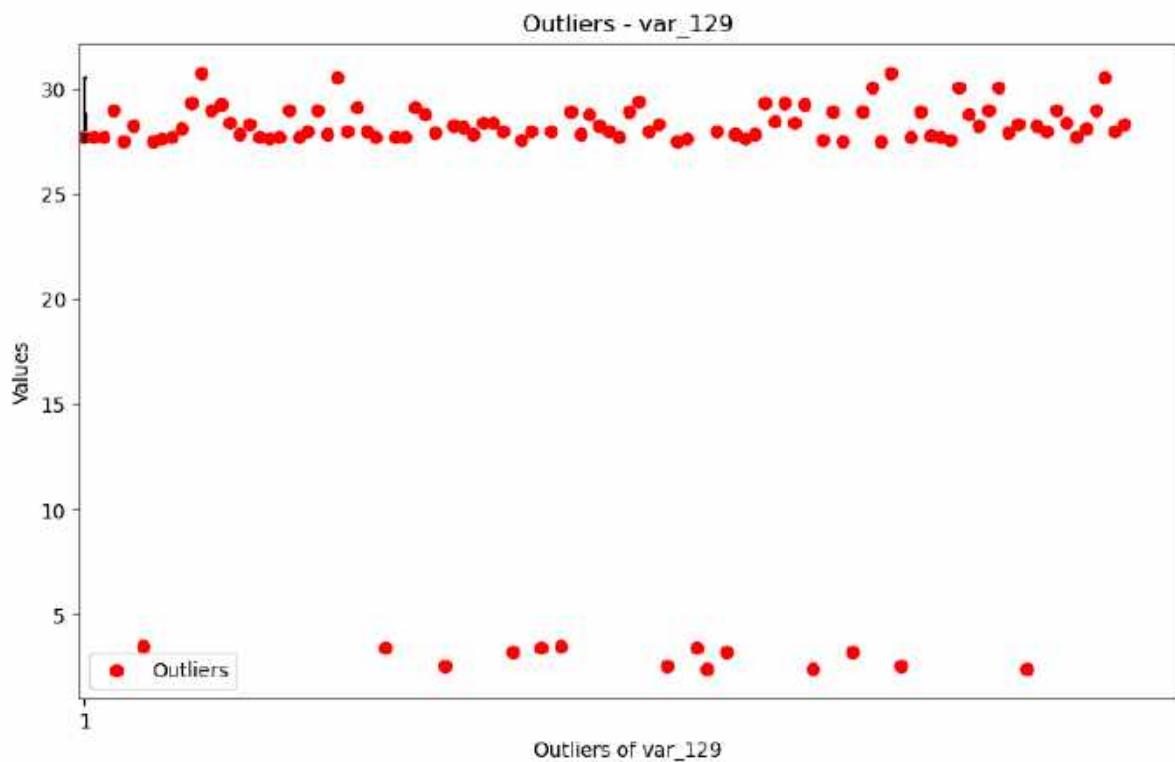


```
Total outliers in column var_127: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

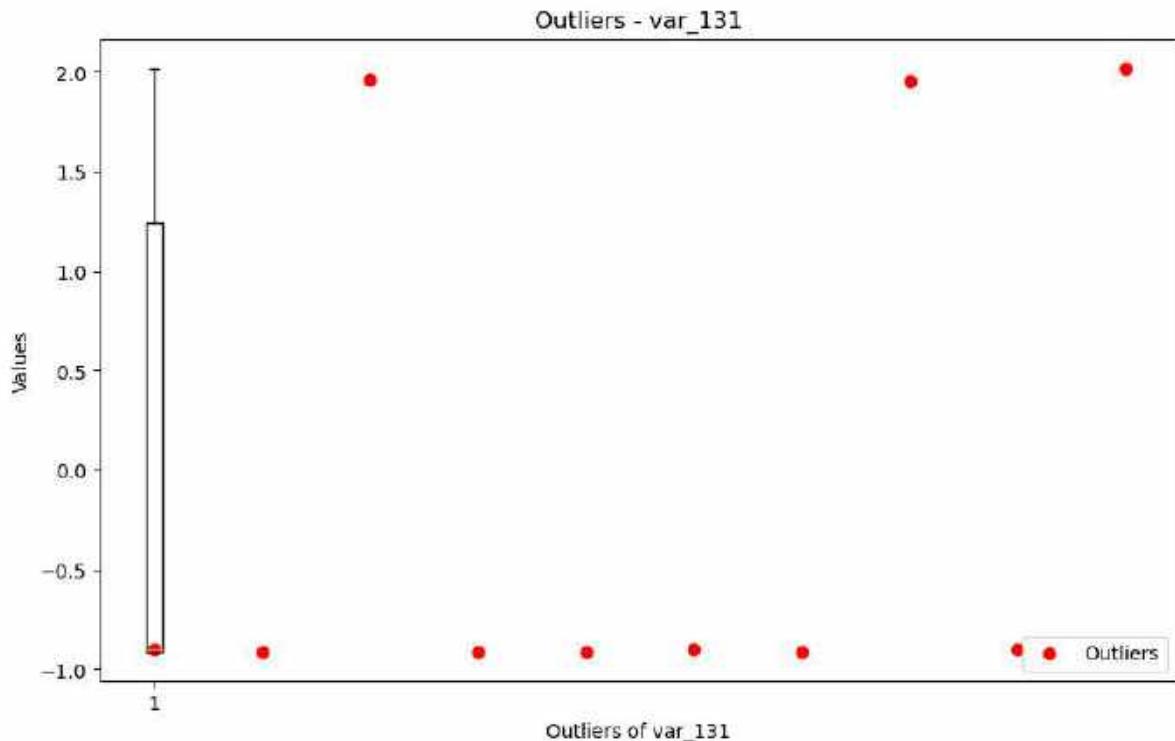
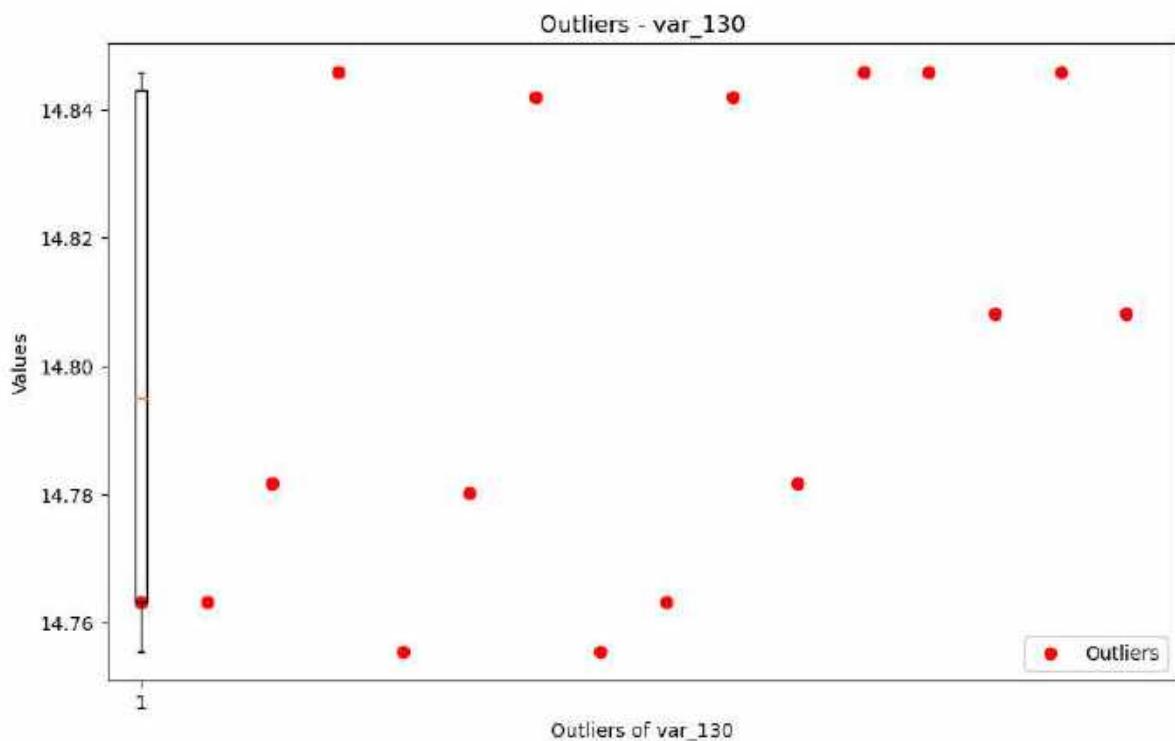


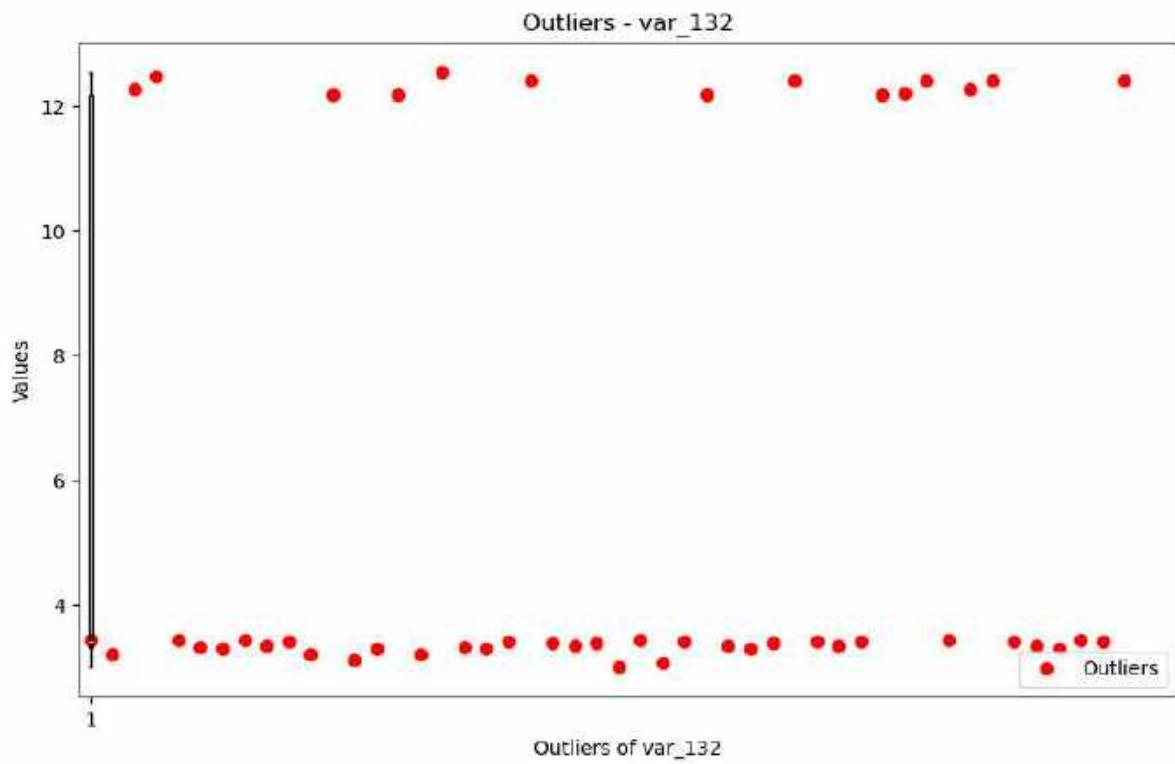
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_128: 6
```



```
Total outliers in column var_129: 108
```

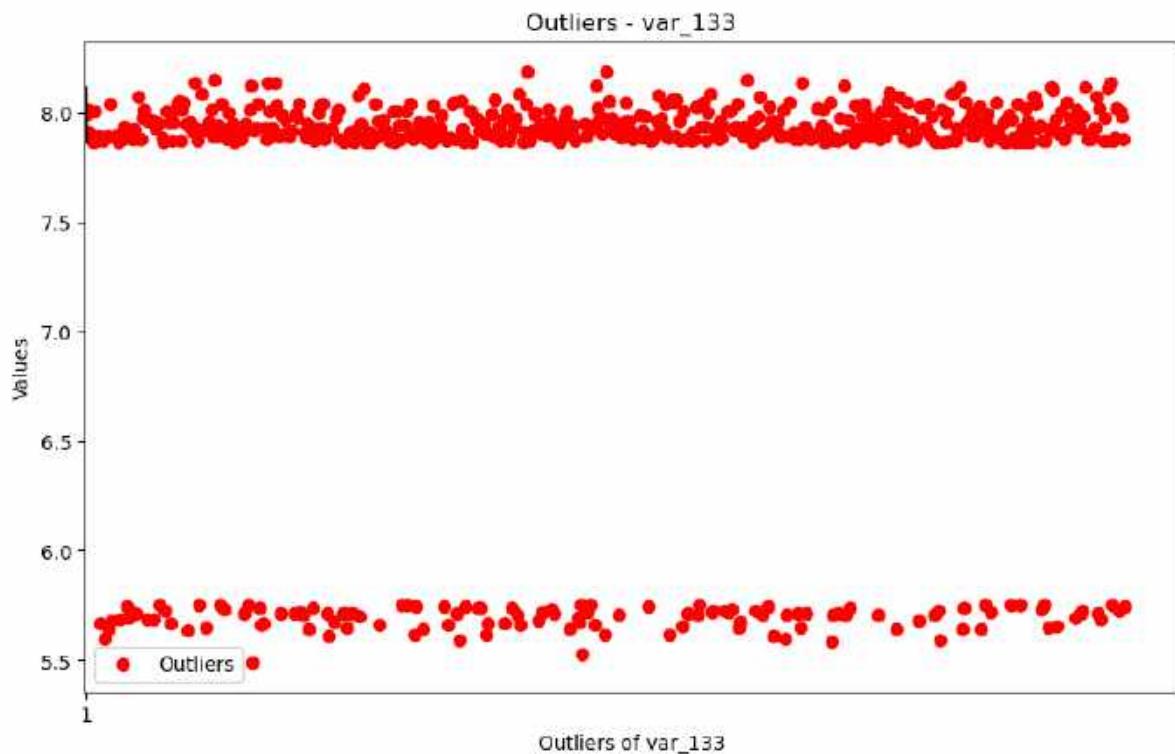
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





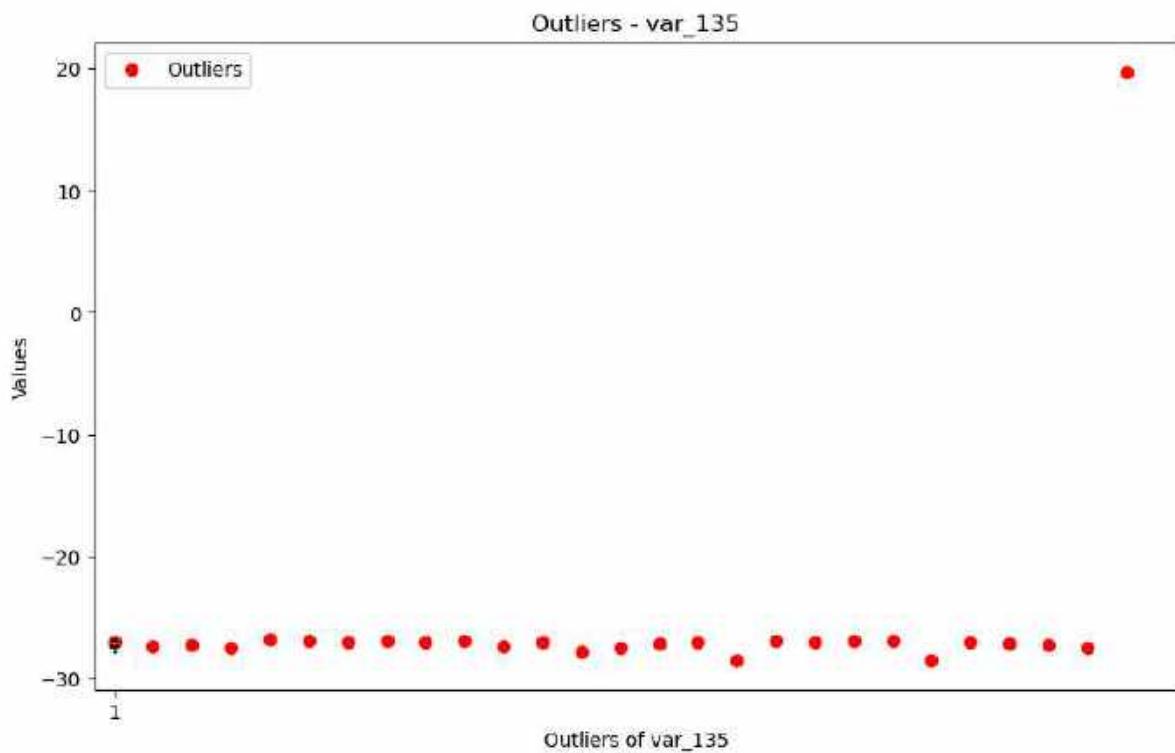
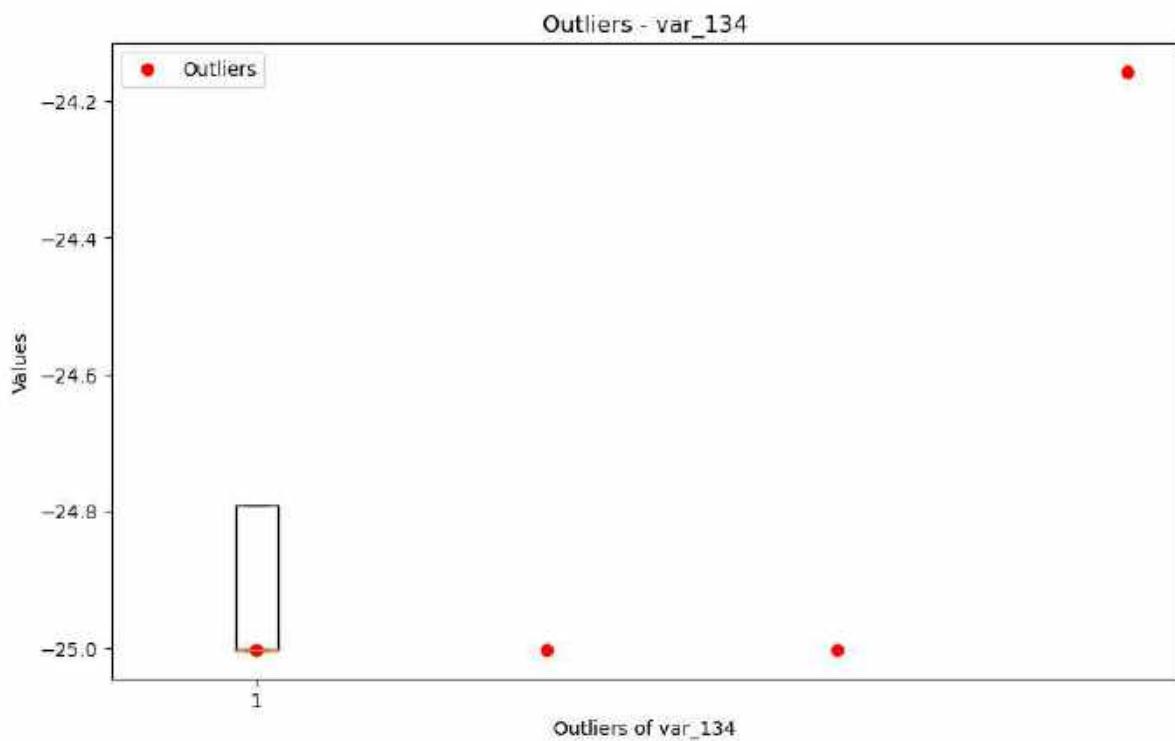
Total outliers in column var\_132: 48

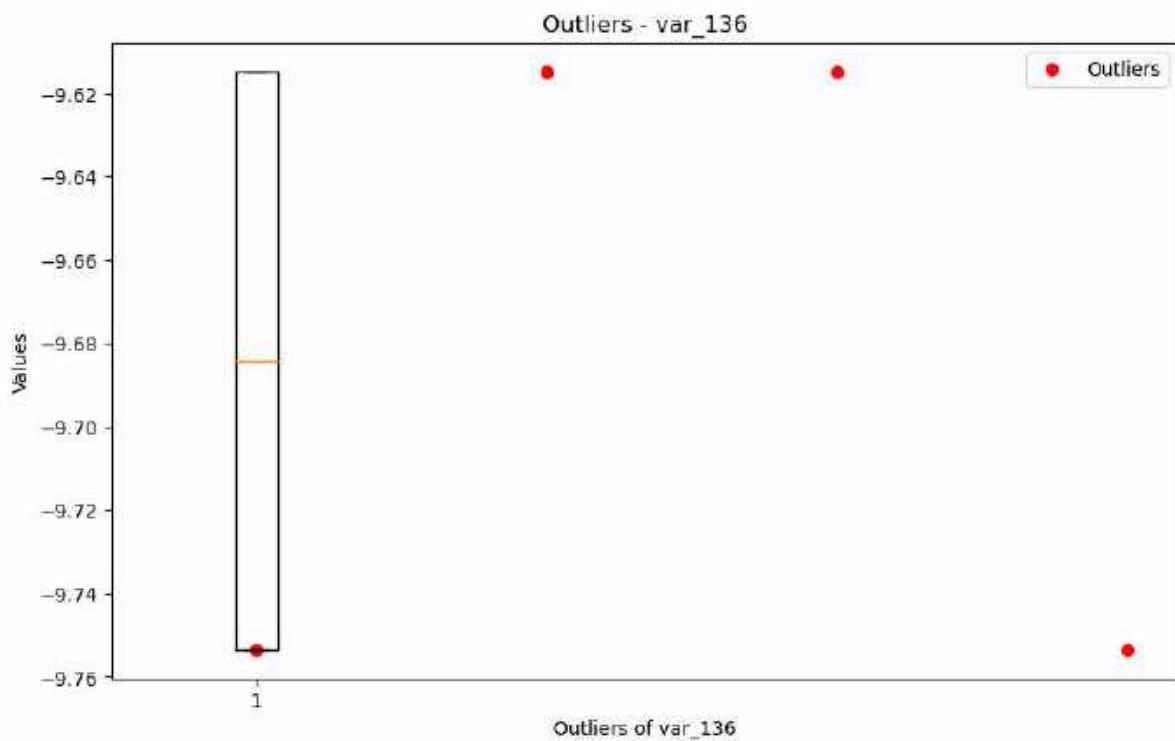
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

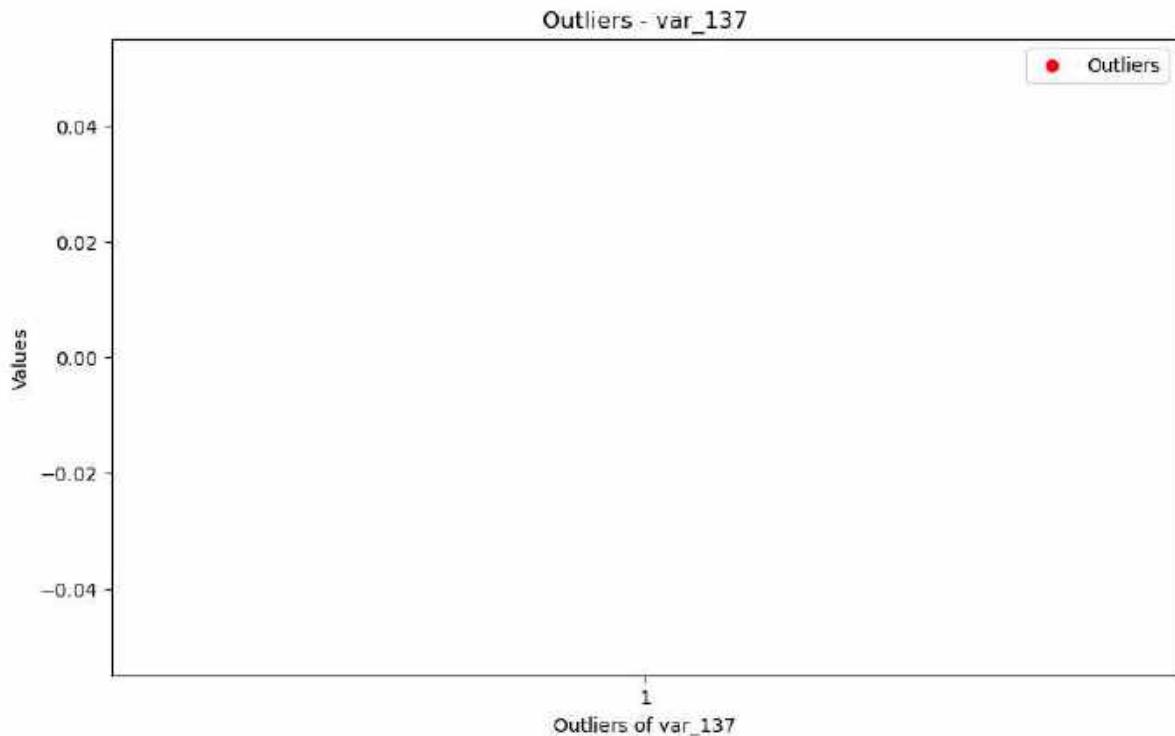
Total outliers in column var\_133: 606





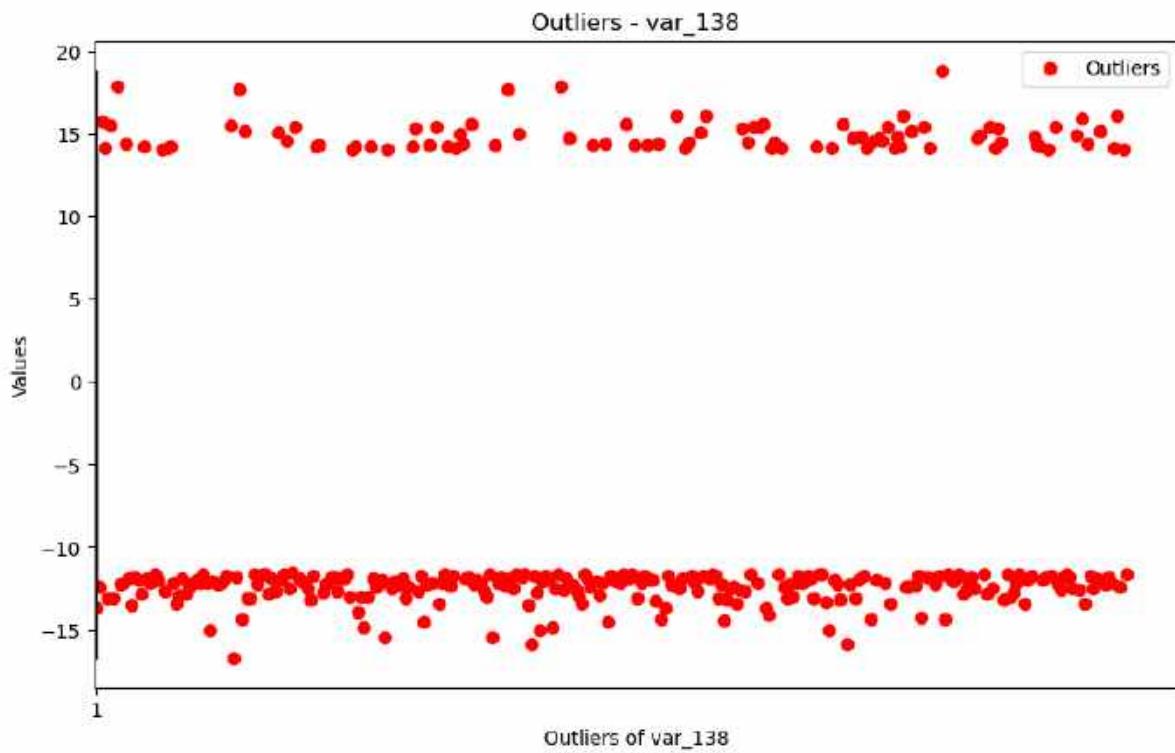
```
Total outliers in column var_136: 4
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



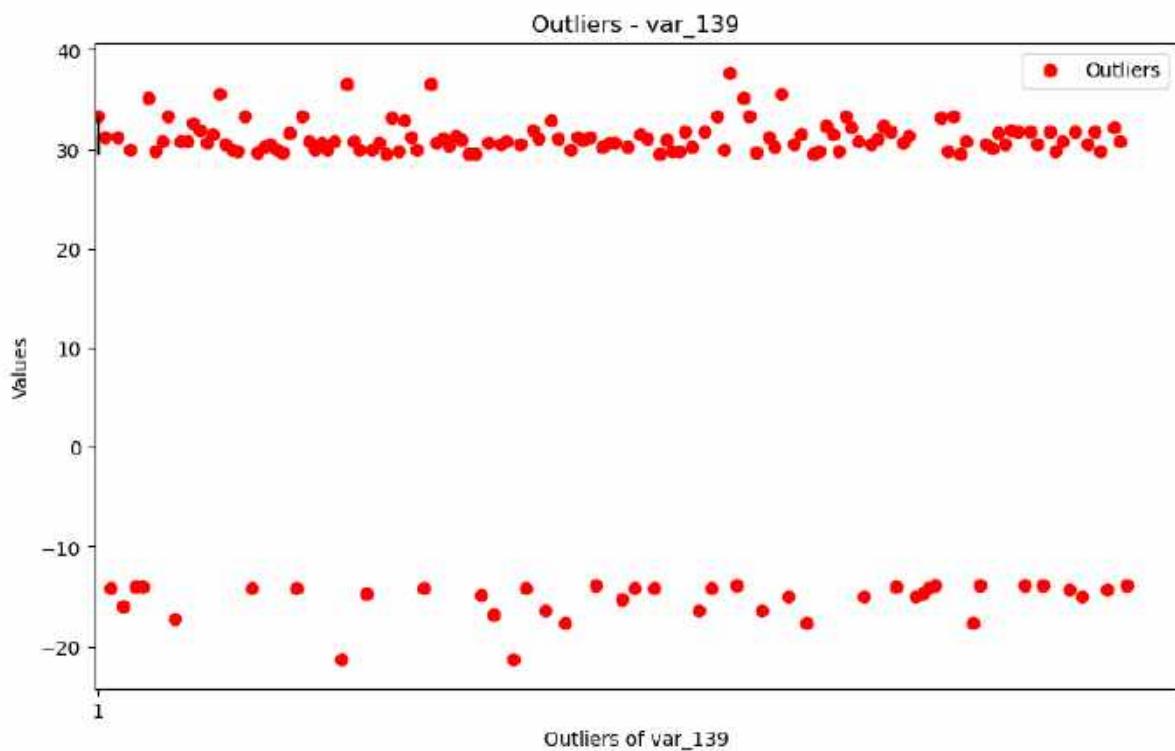
```
Total outliers in column var_137: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



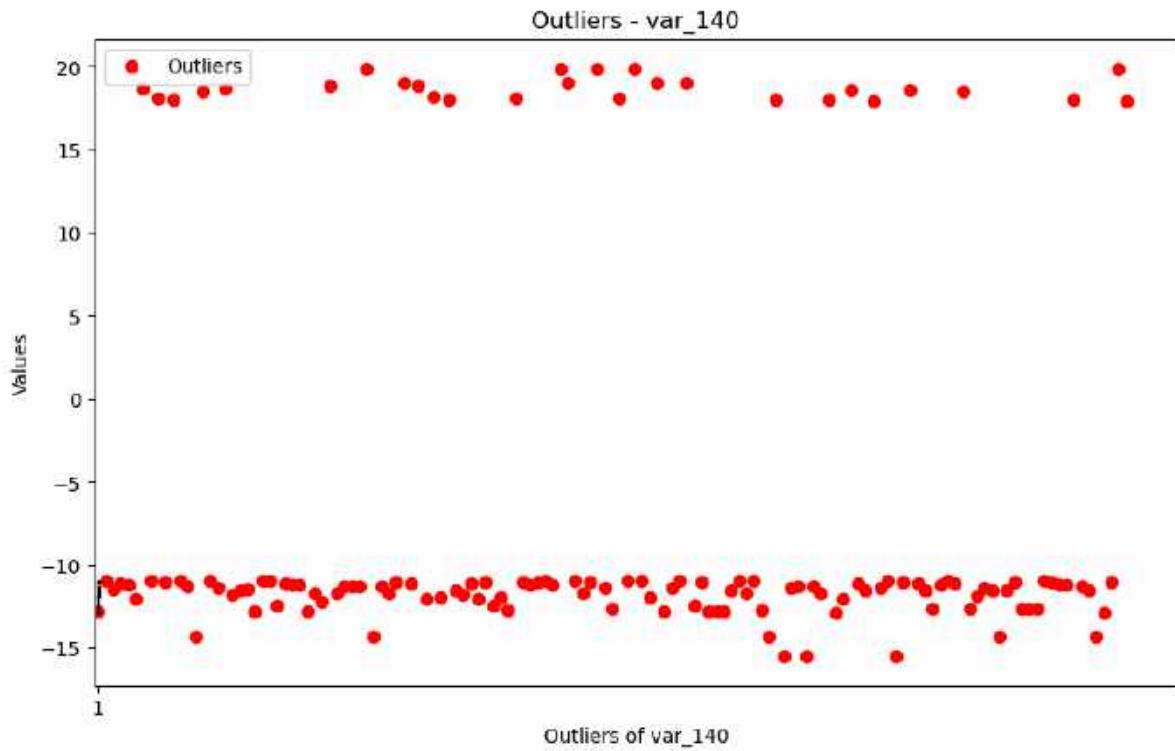
Total outliers in column var\_138: 347

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



Total outliers in column var\_139: 162

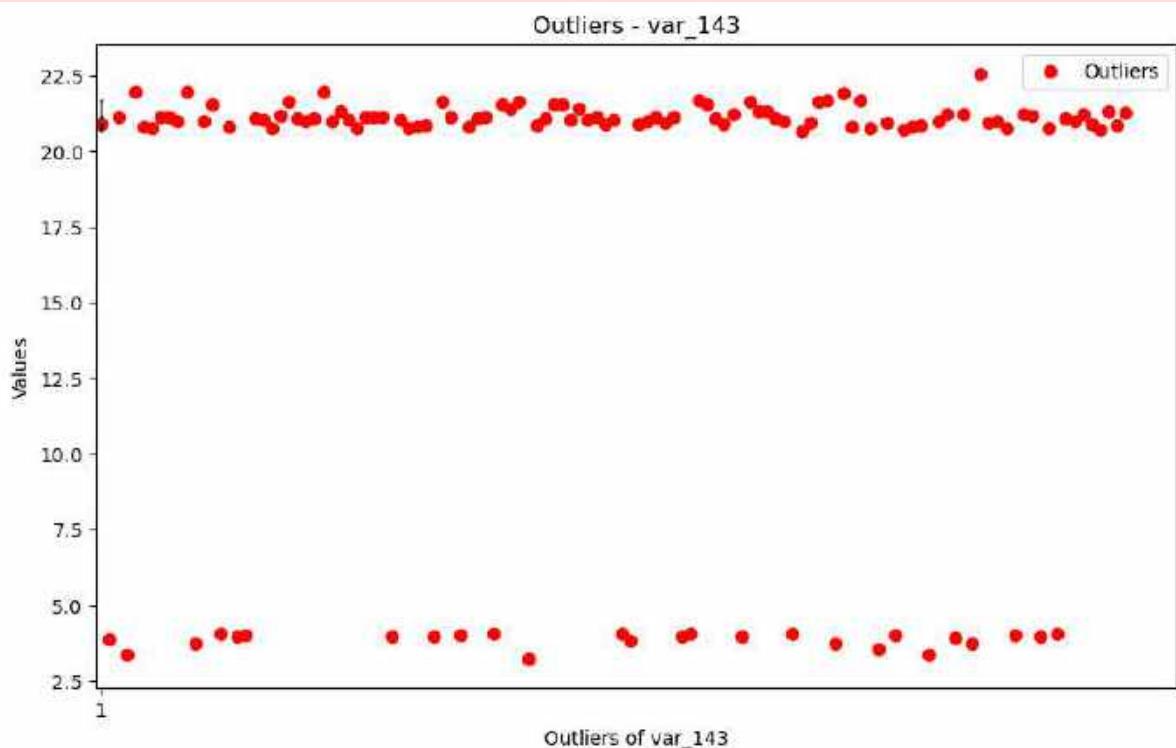
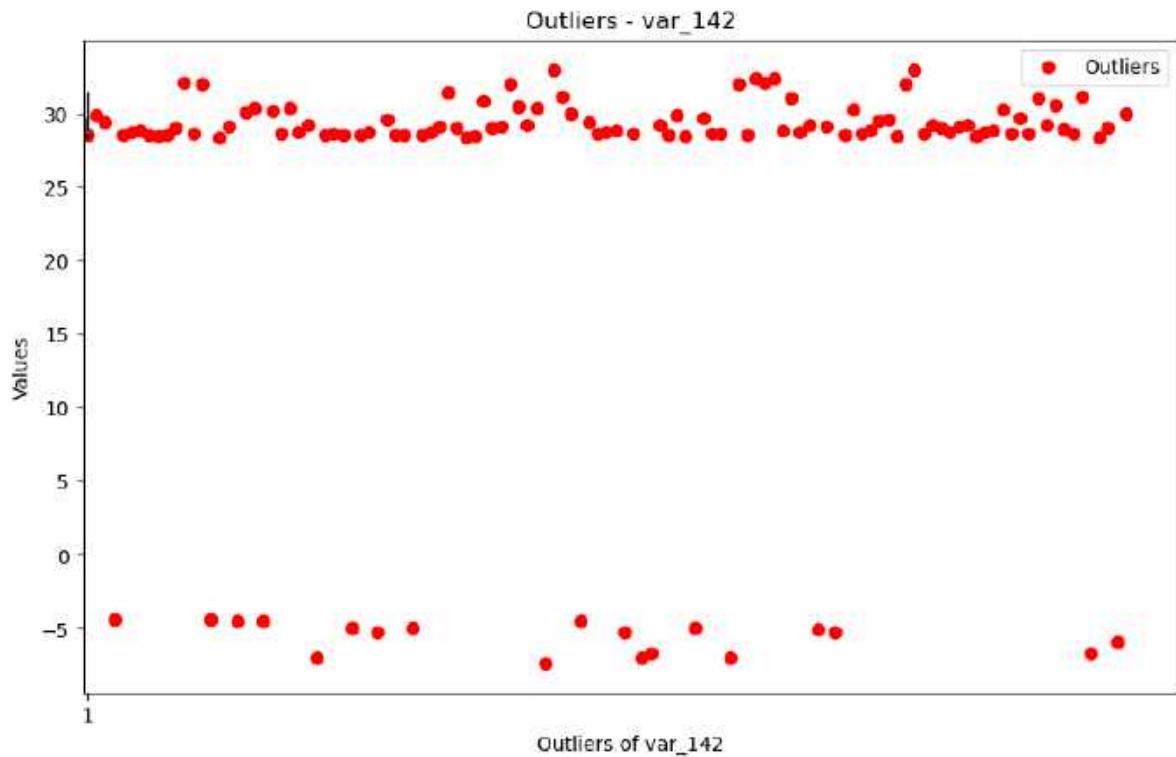
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

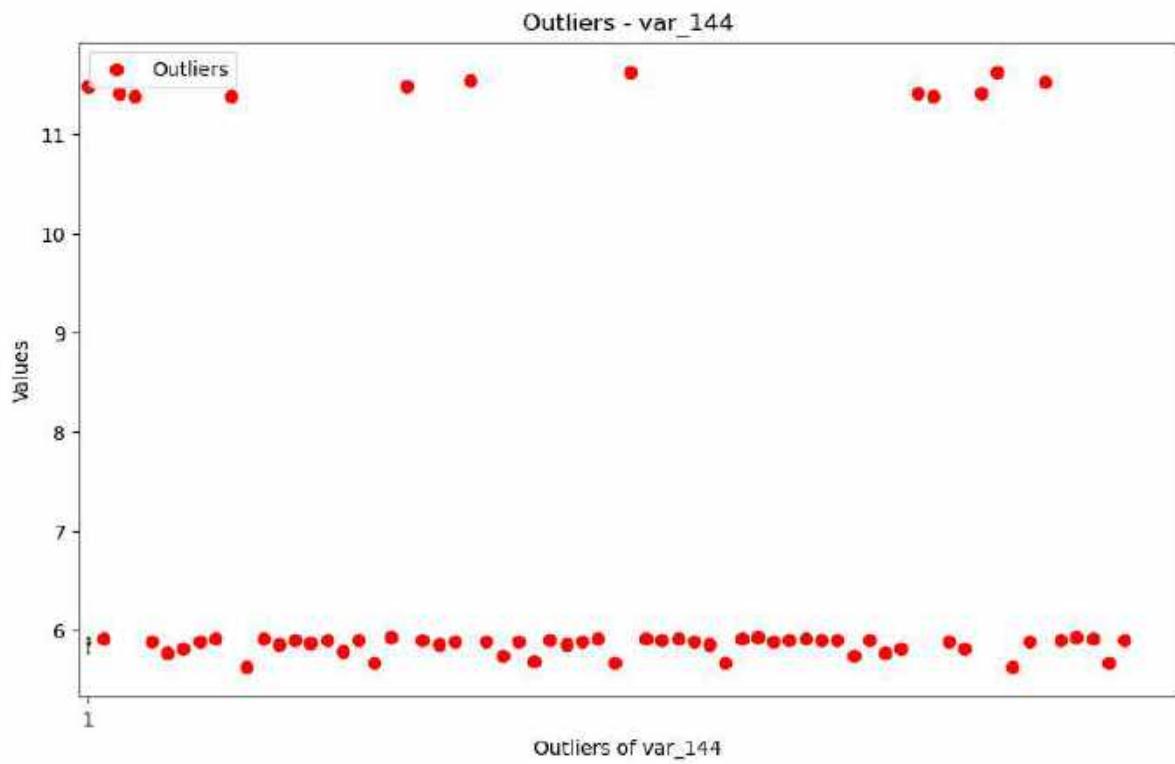


```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_140: 139
```



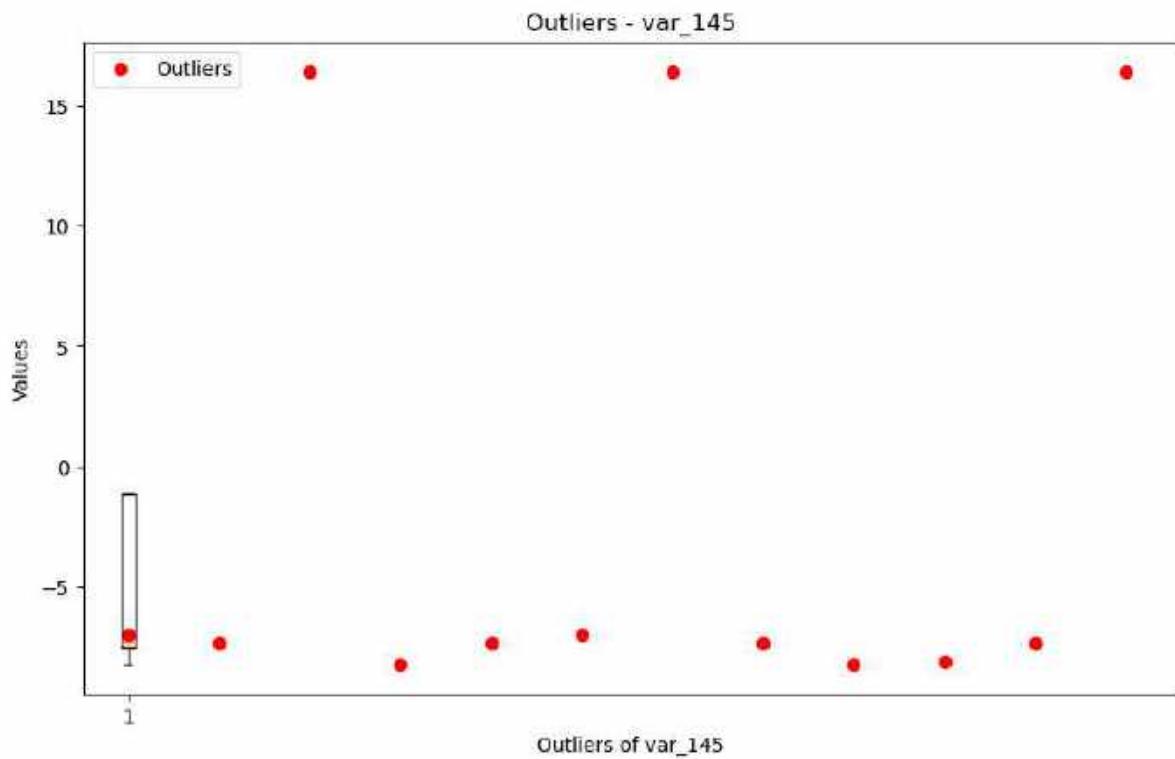
```
Total outliers in column var_141: 0  
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





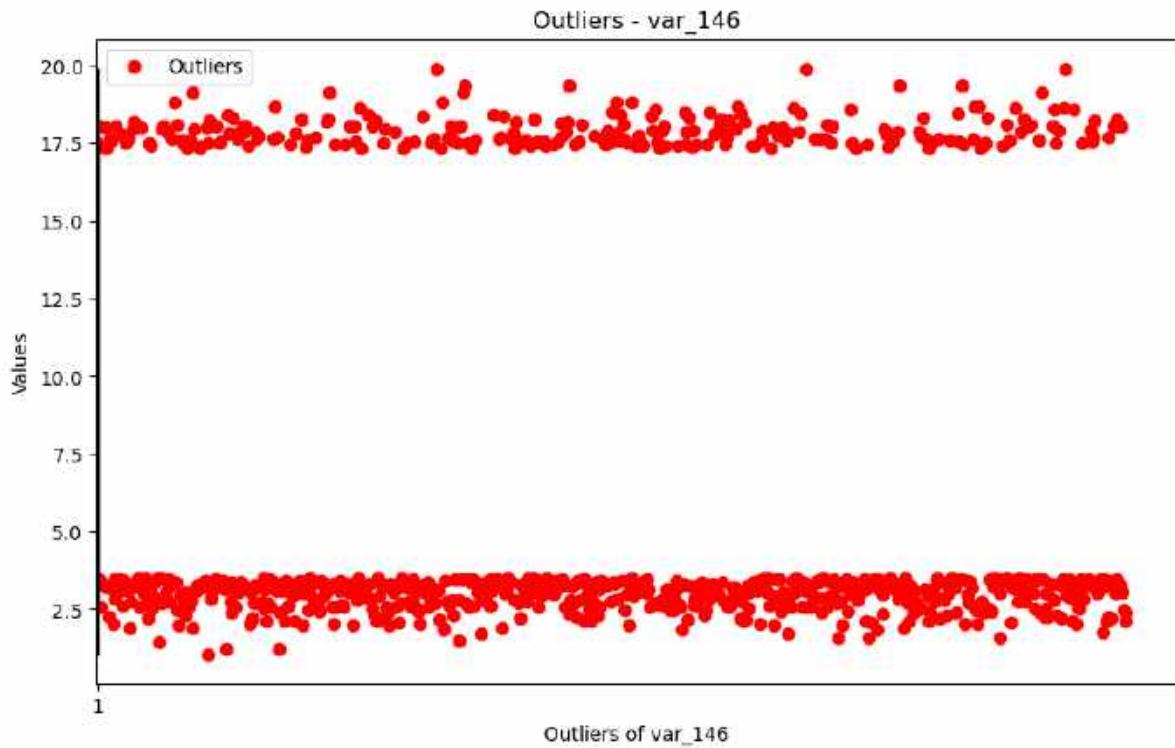
Total outliers in column var\_144: 66

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

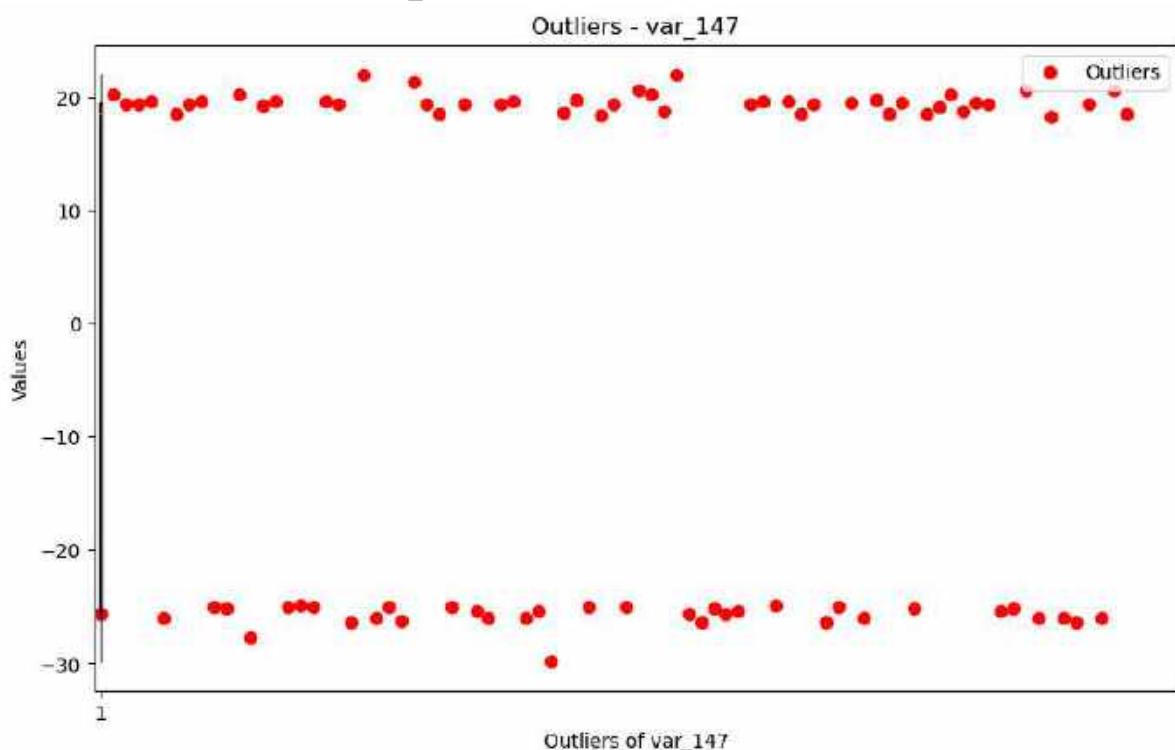


Total outliers in column var\_145: 12

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

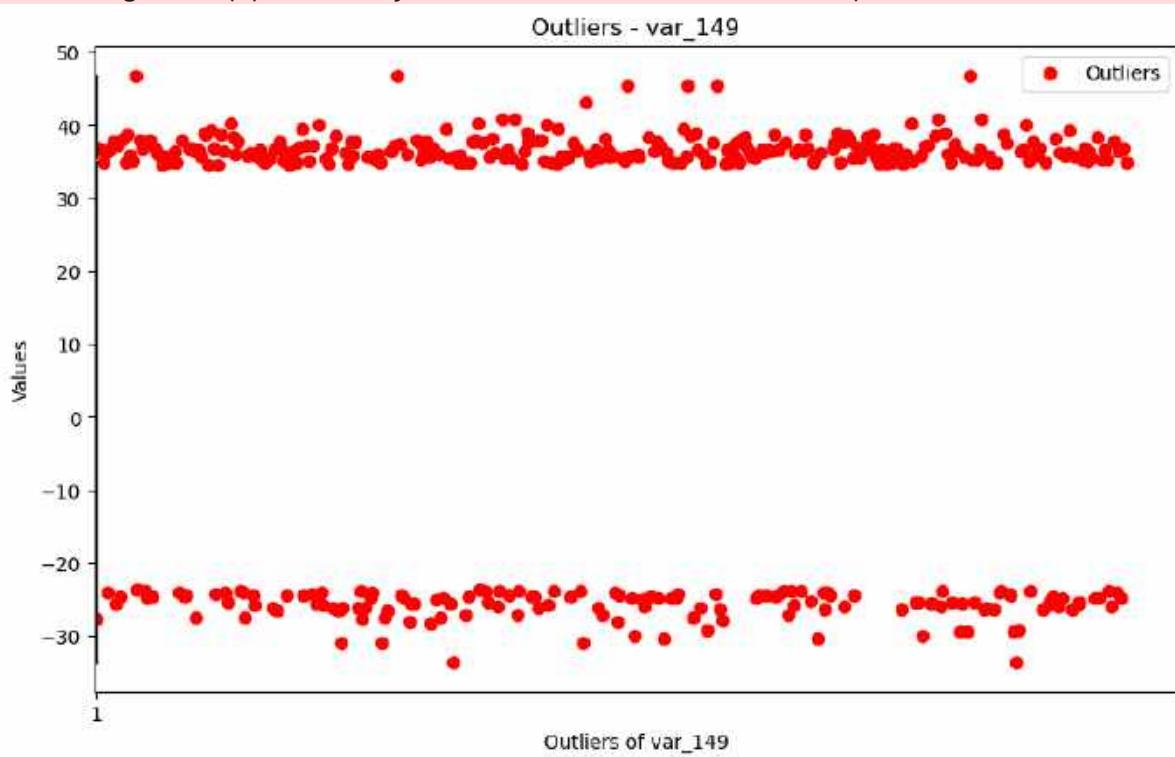
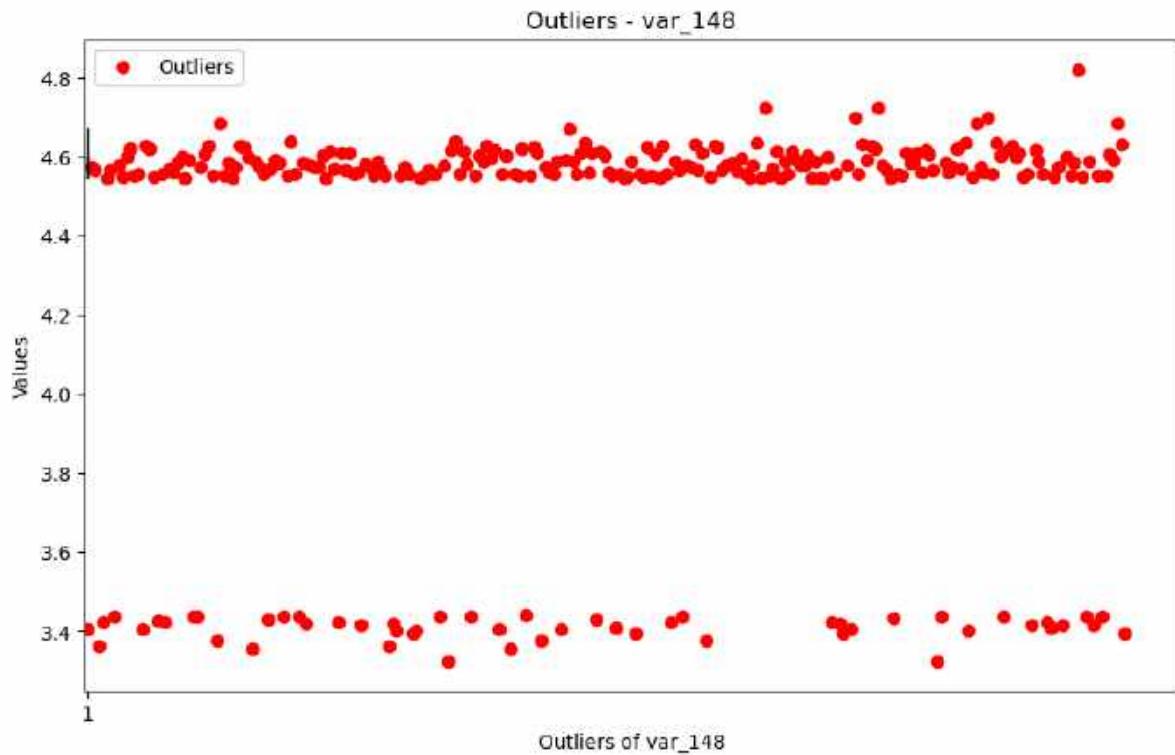


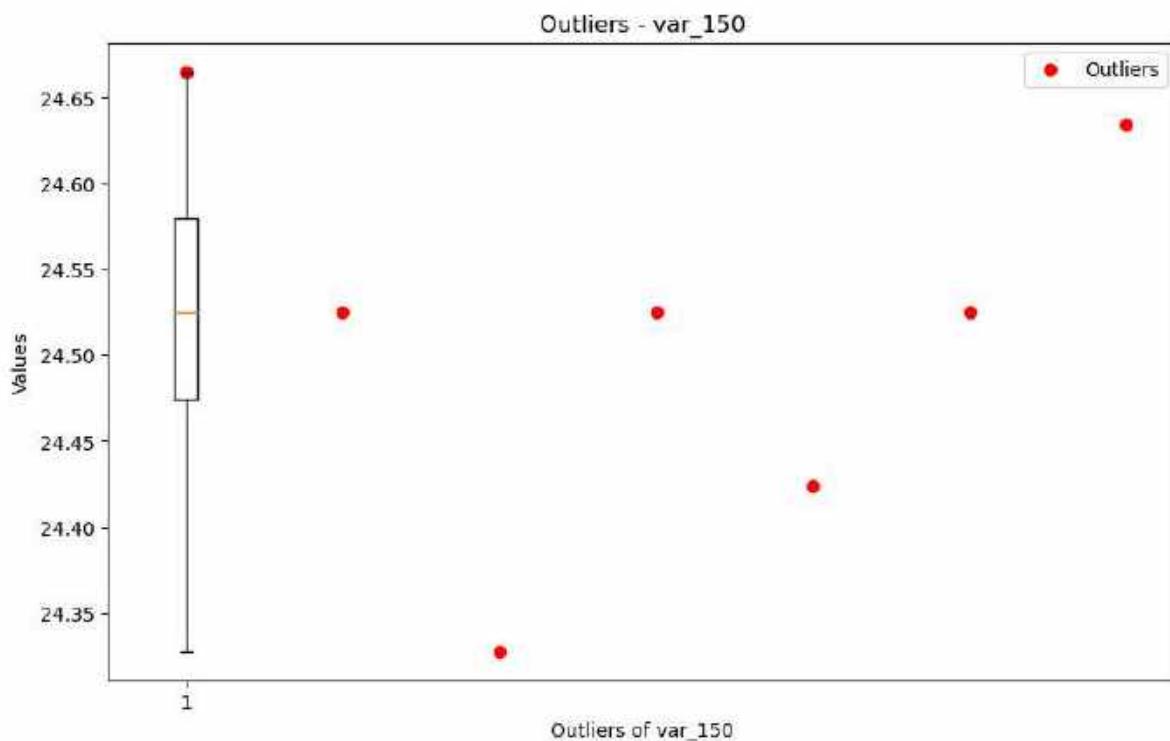
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_146: 870
```



```
Total outliers in column var_147: 83
```

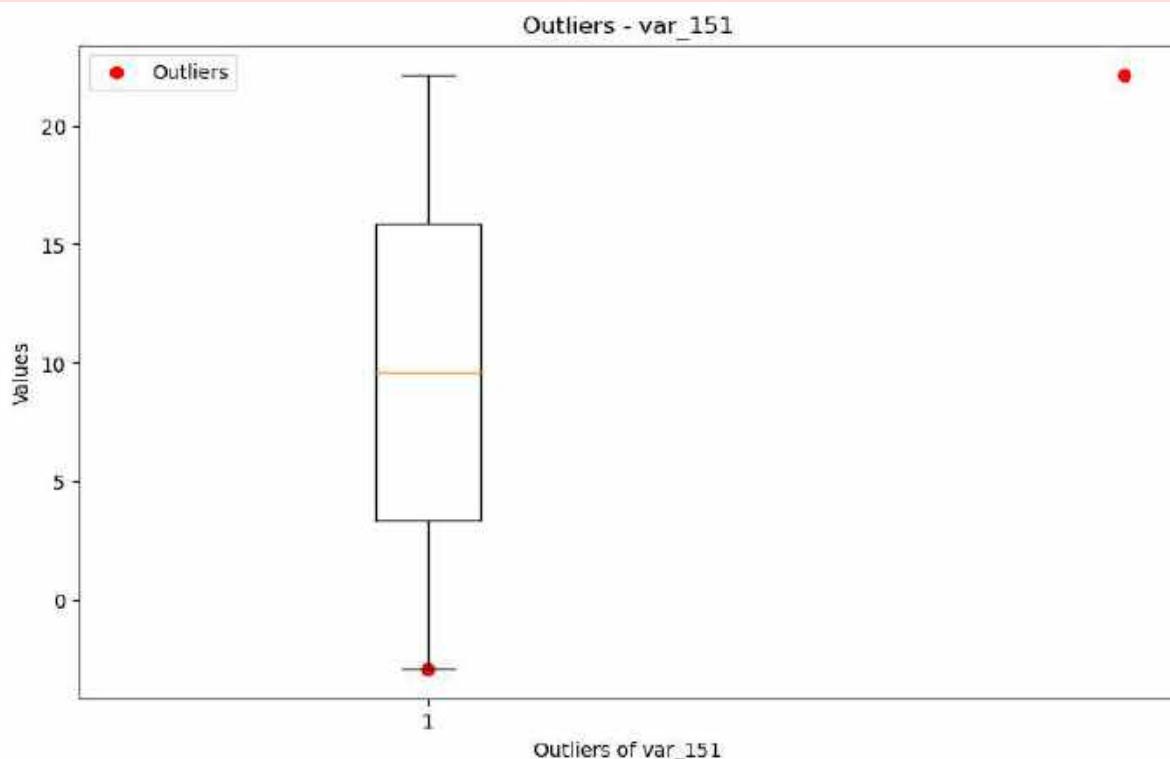
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```





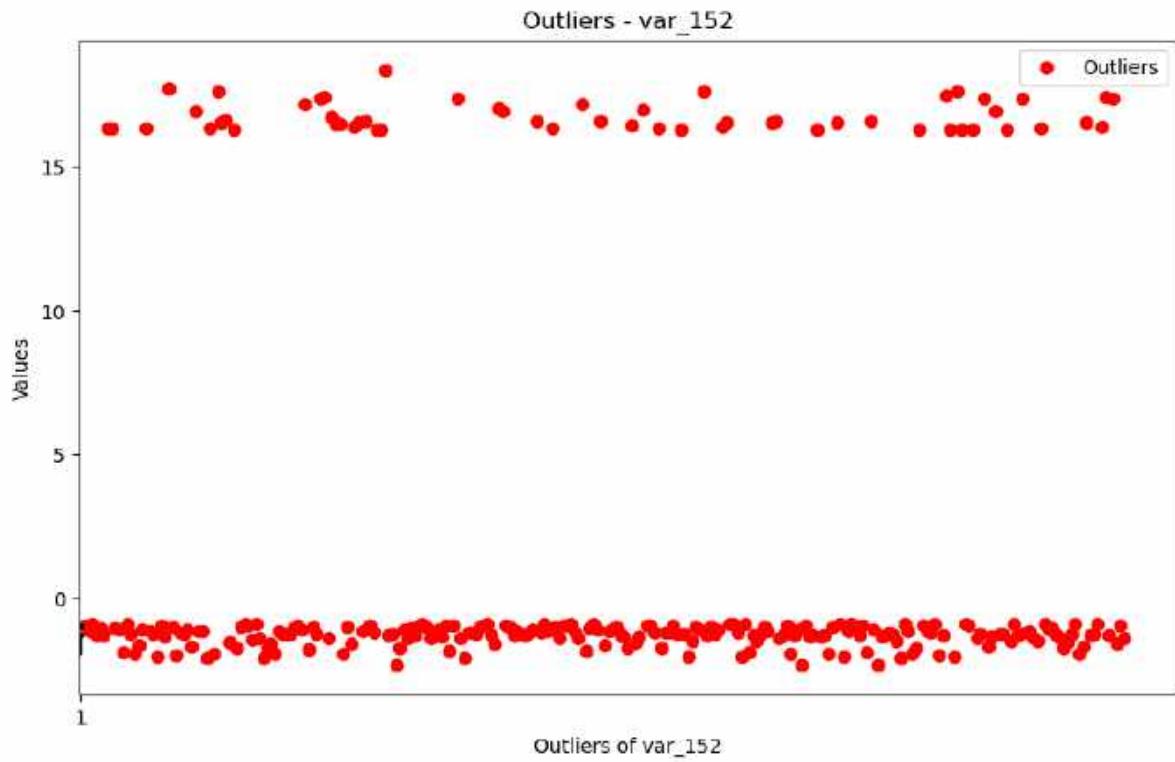
Total outliers in column var\_150: 7

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



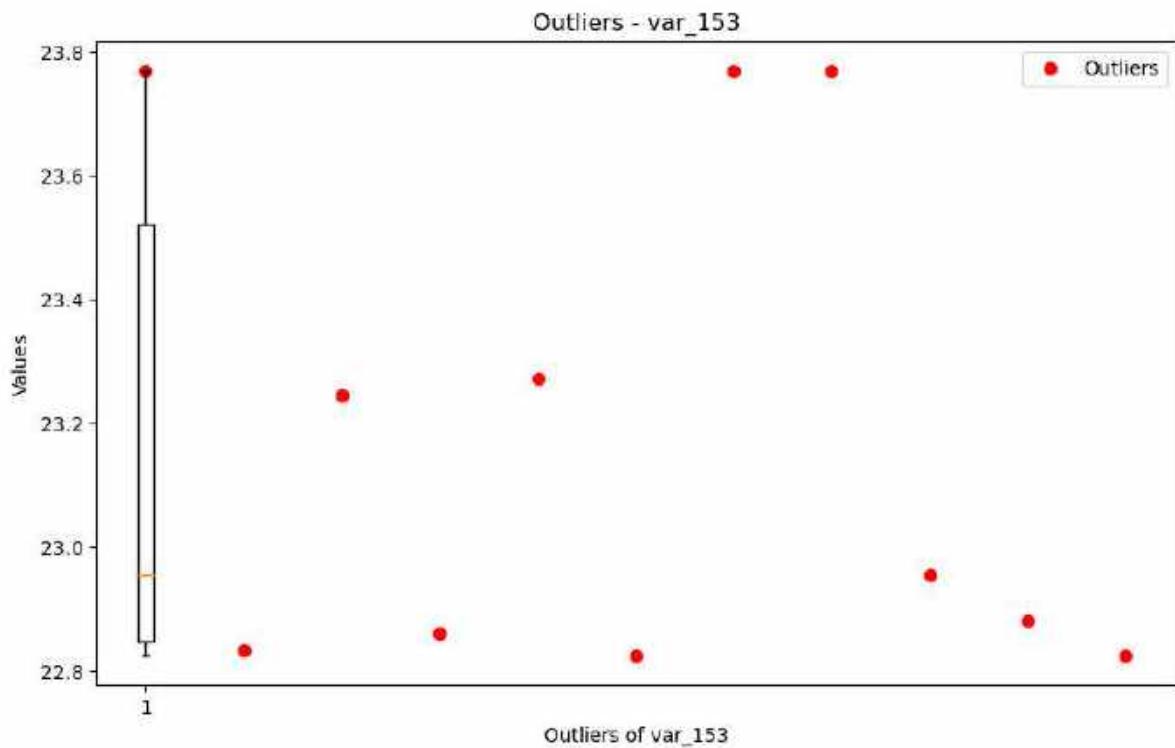
Total outliers in column var\_151: 2

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



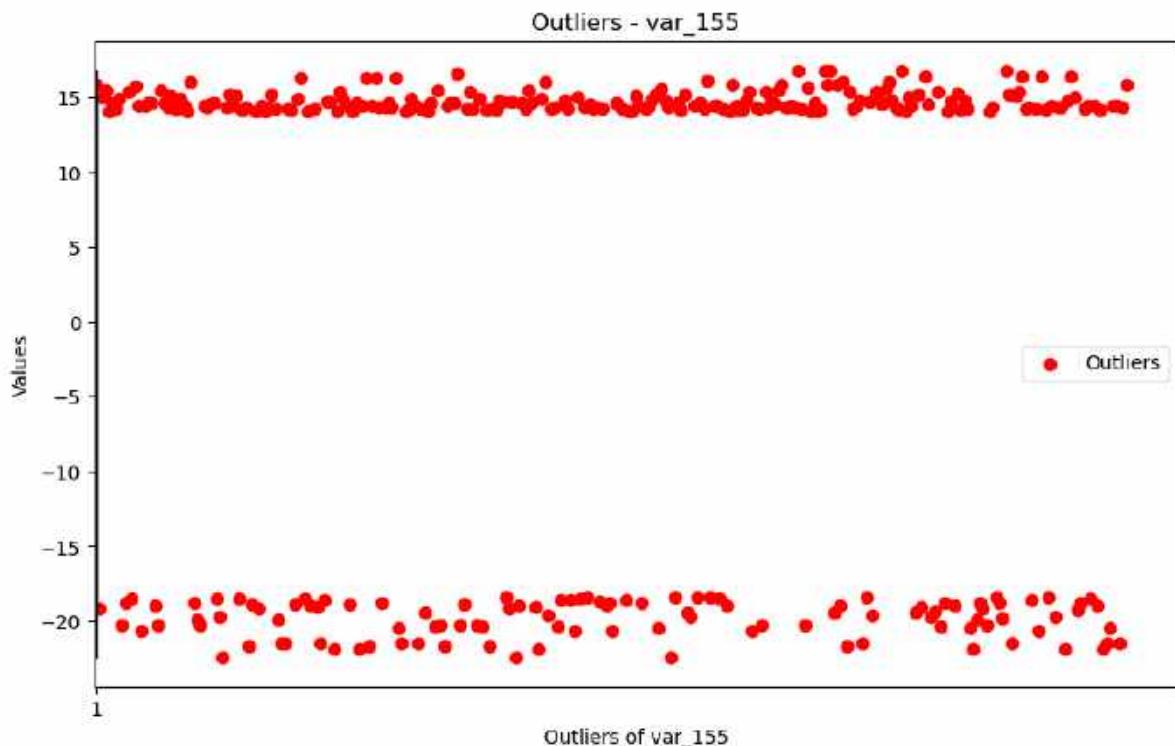
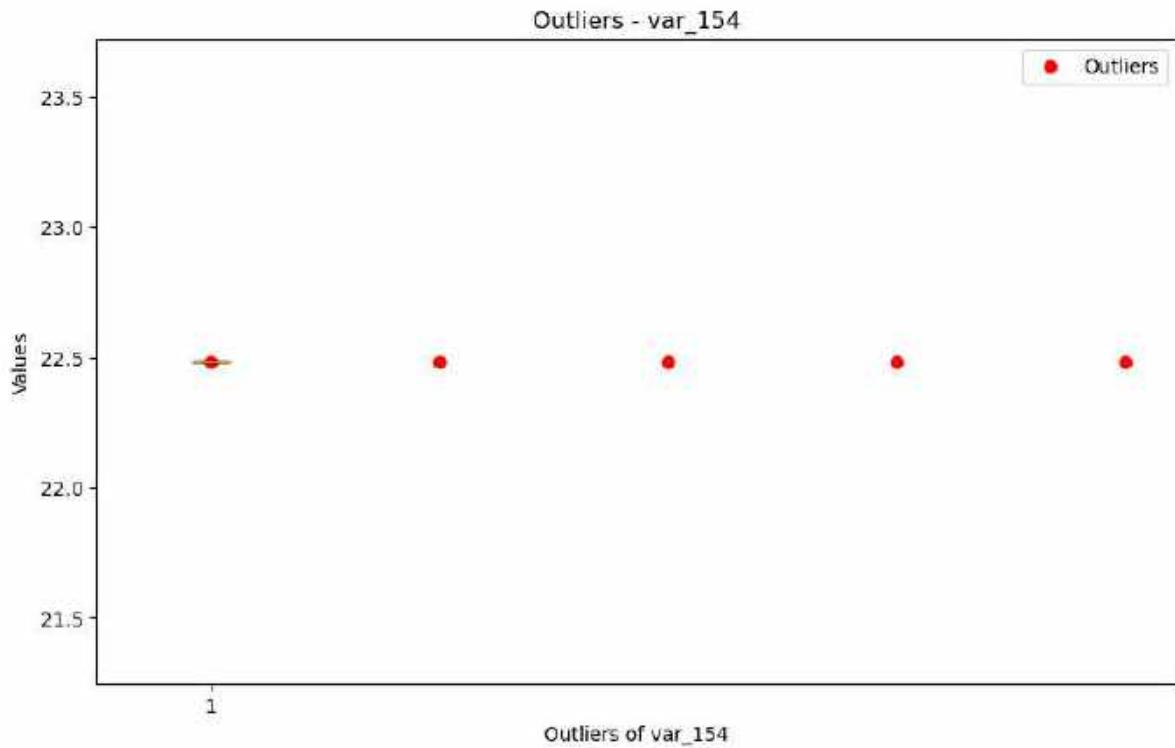
Total outliers in column var\_152: 276

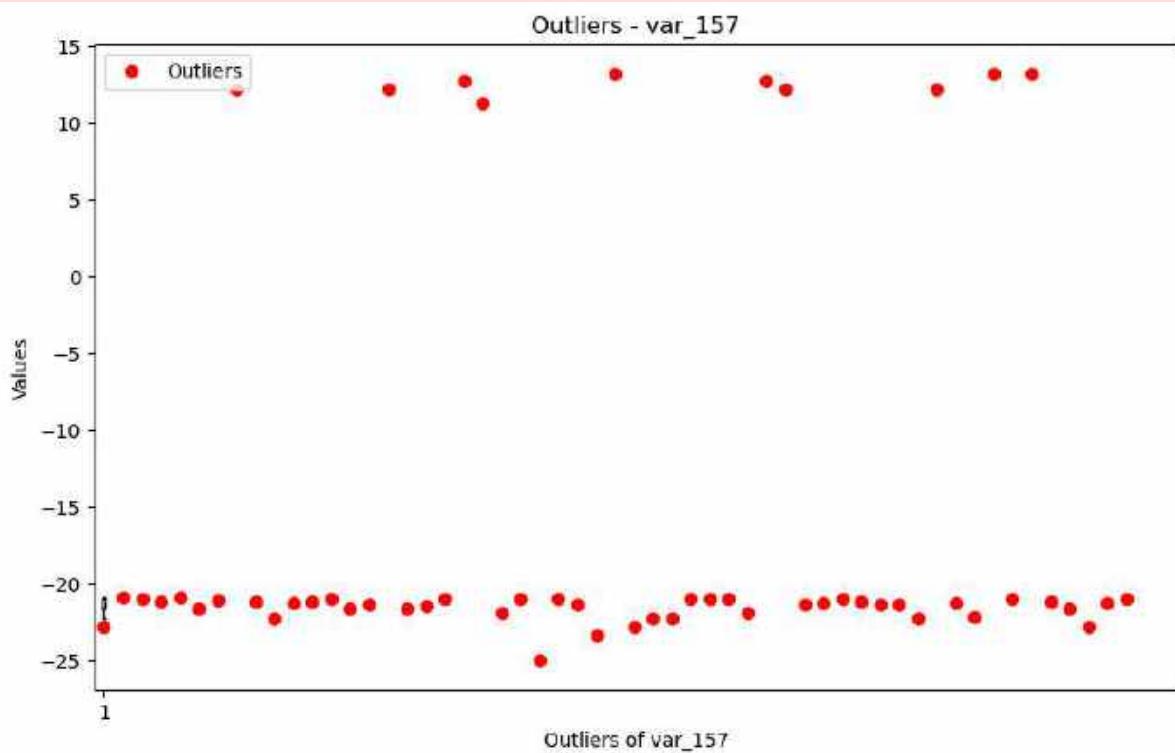
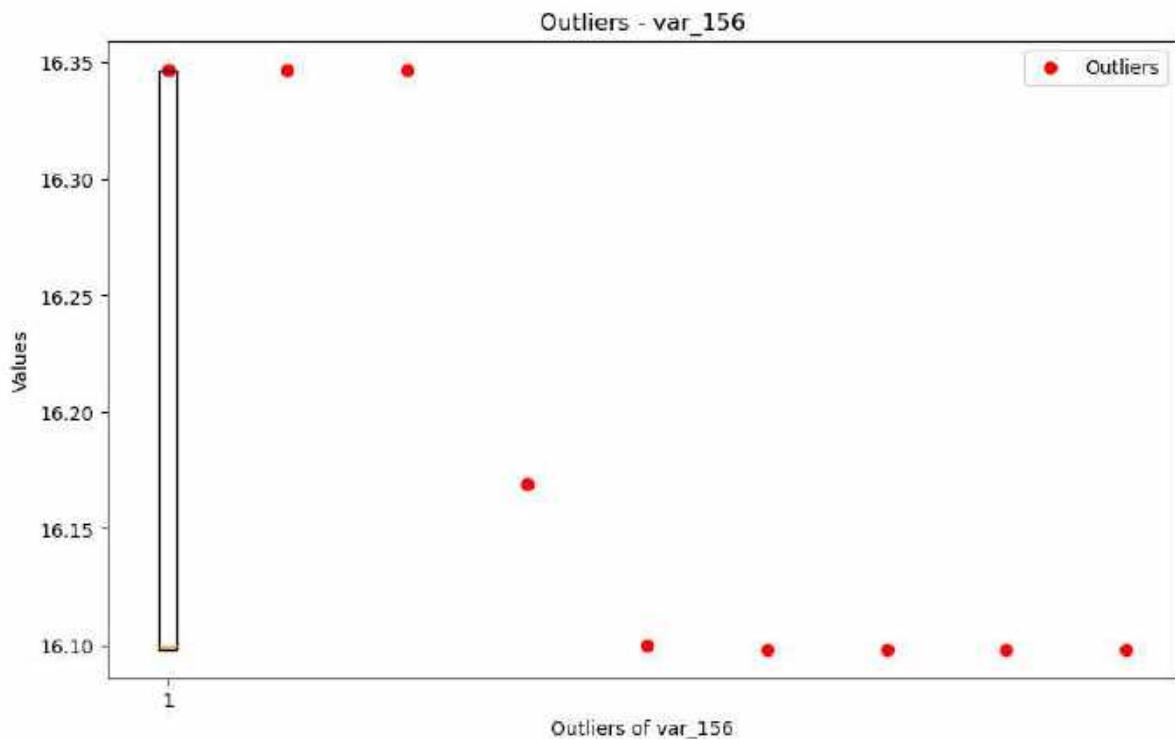
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

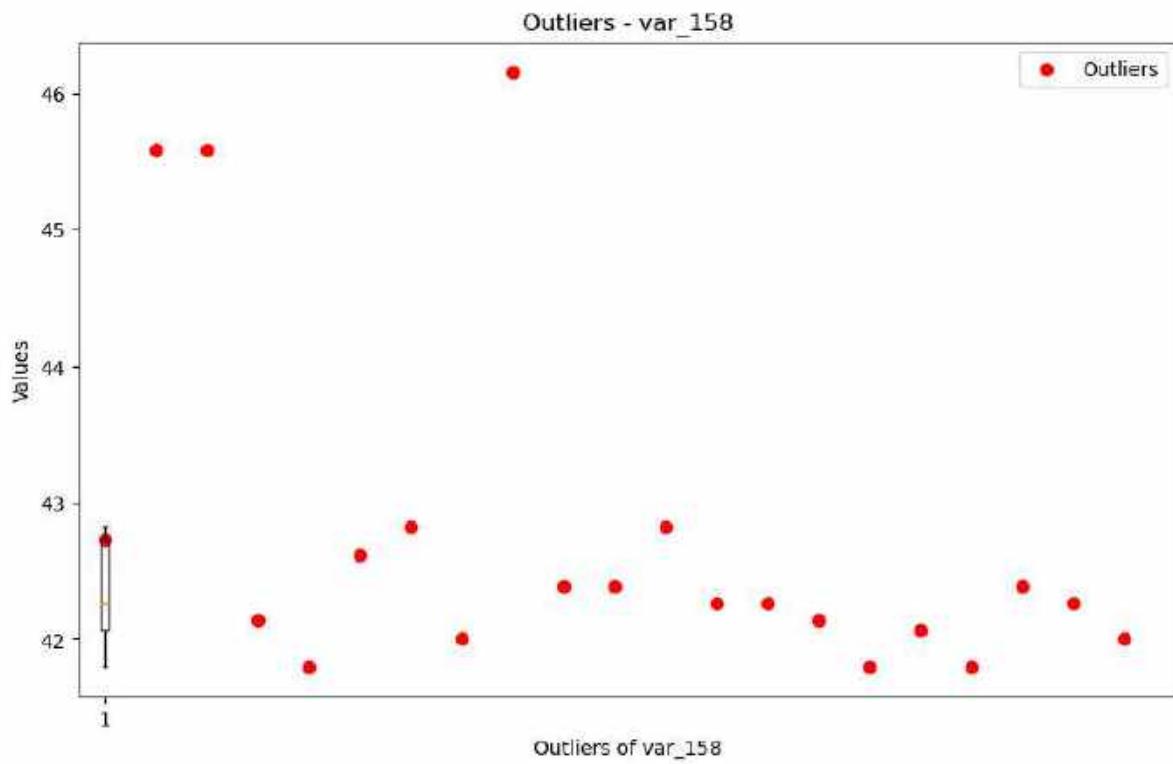


Total outliers in column var\_153: 11

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

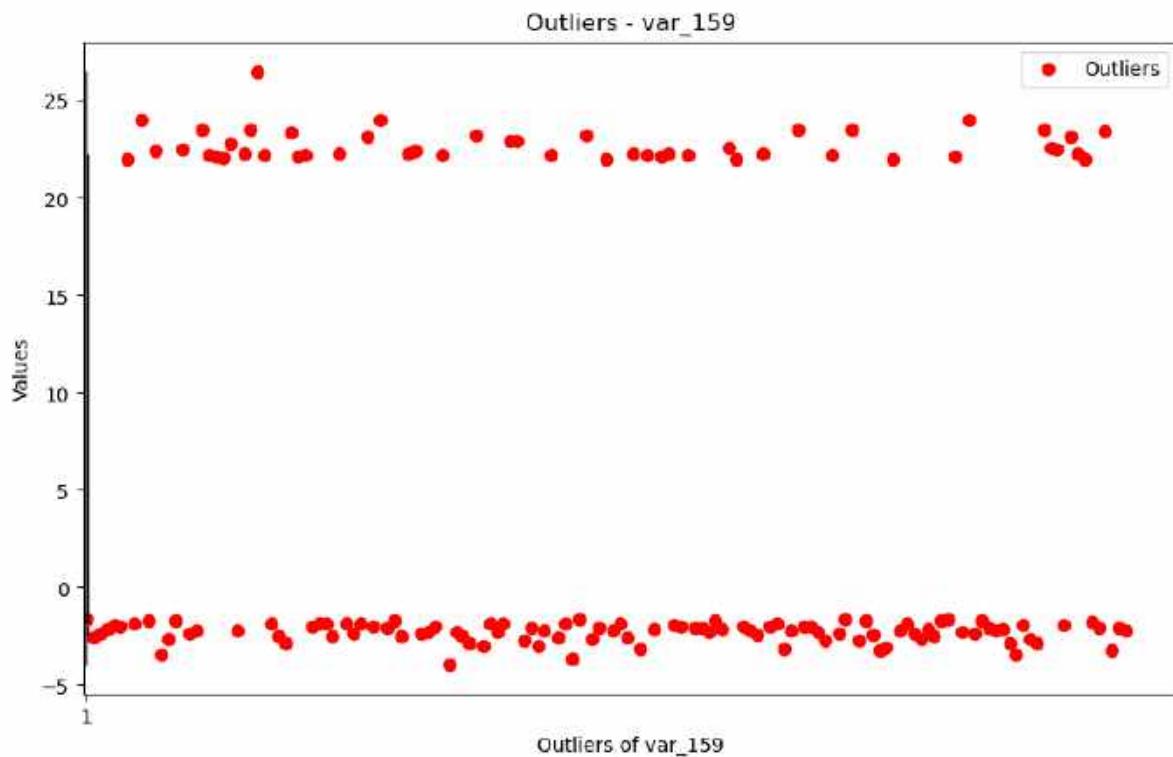






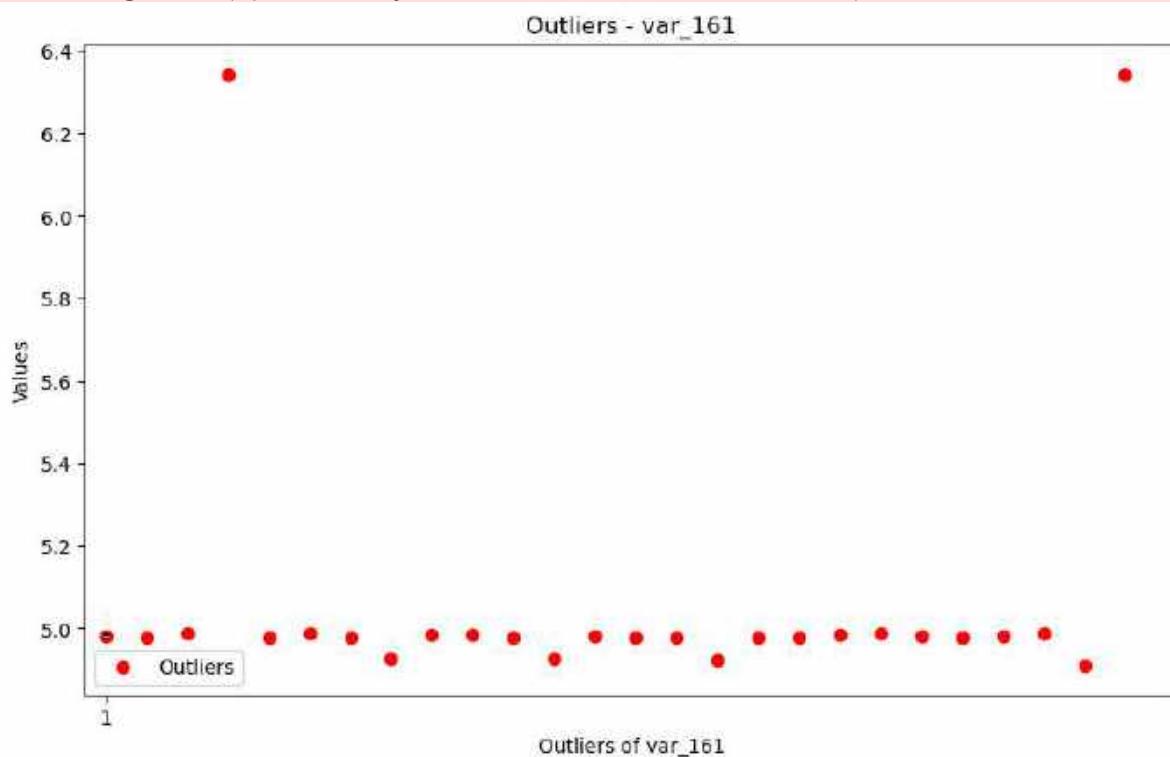
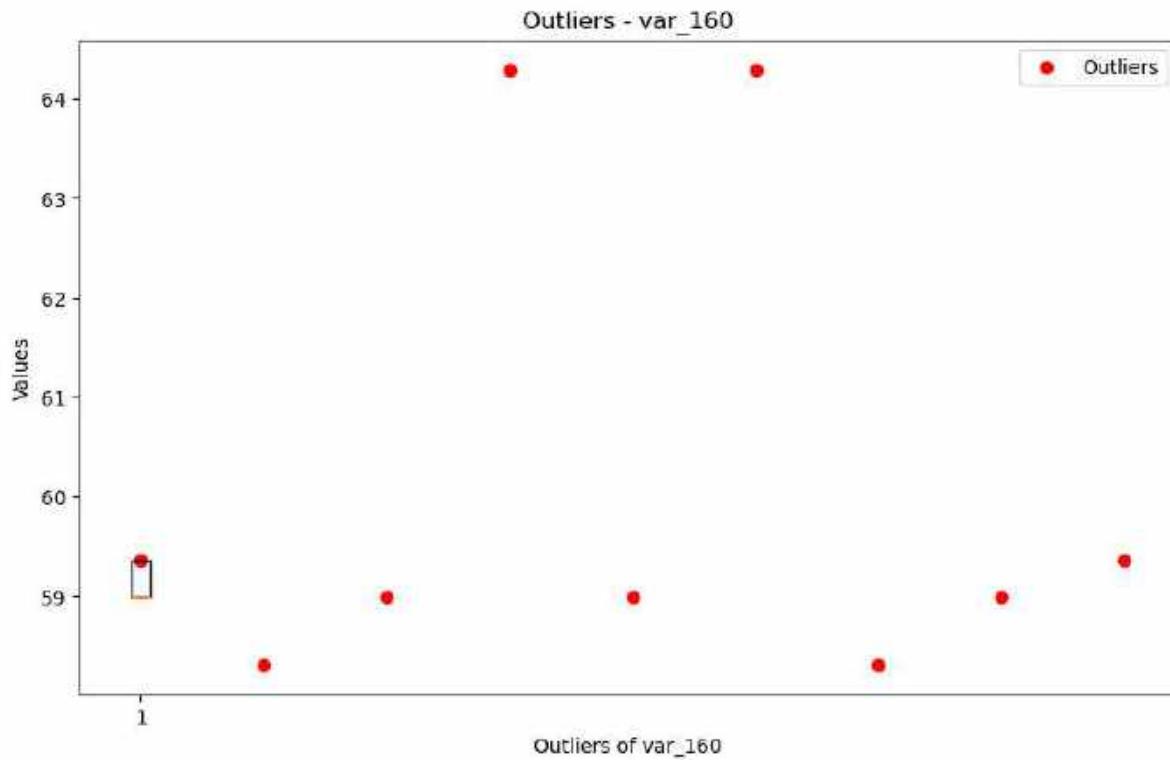
Total outliers in column var\_158: 21

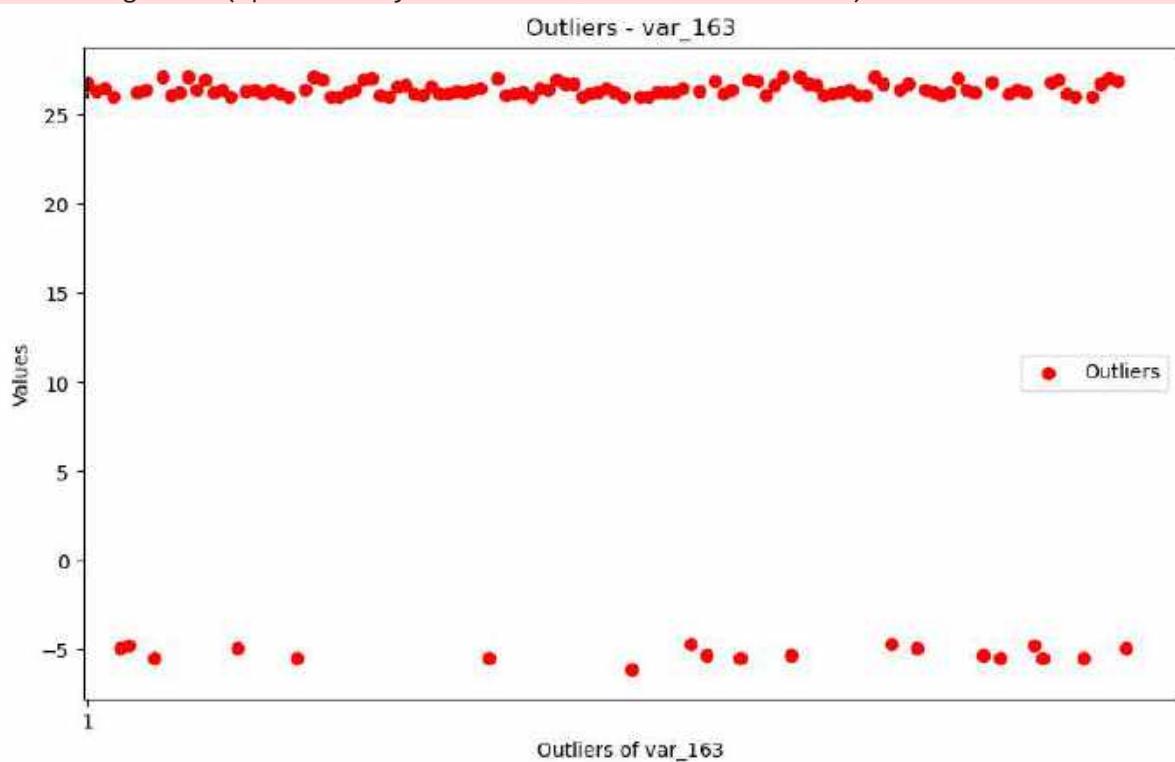
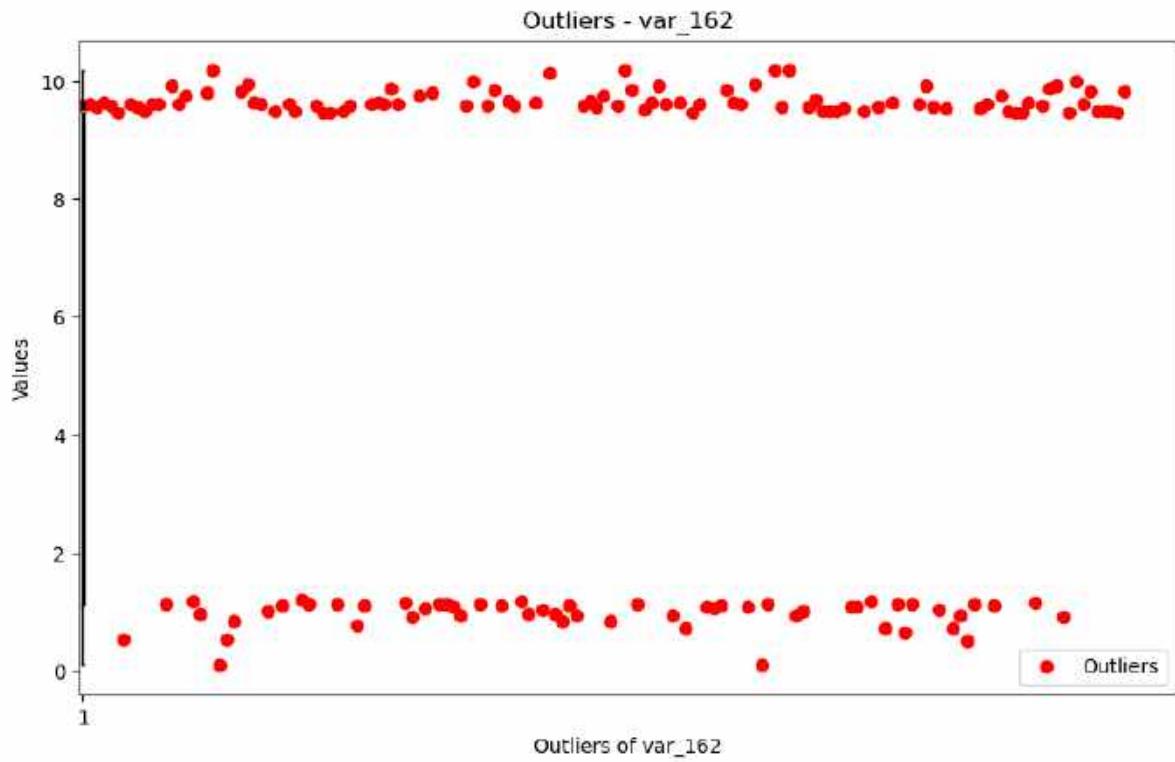
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

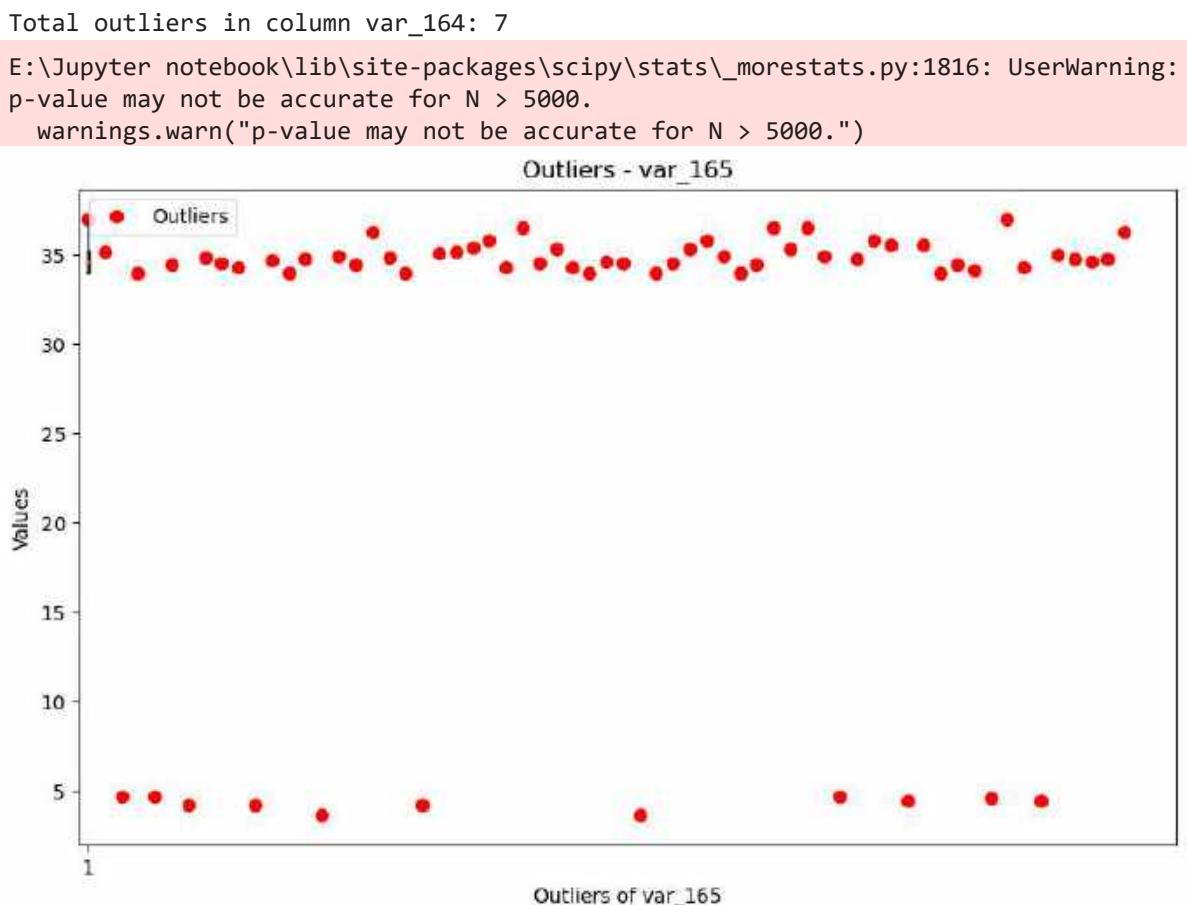
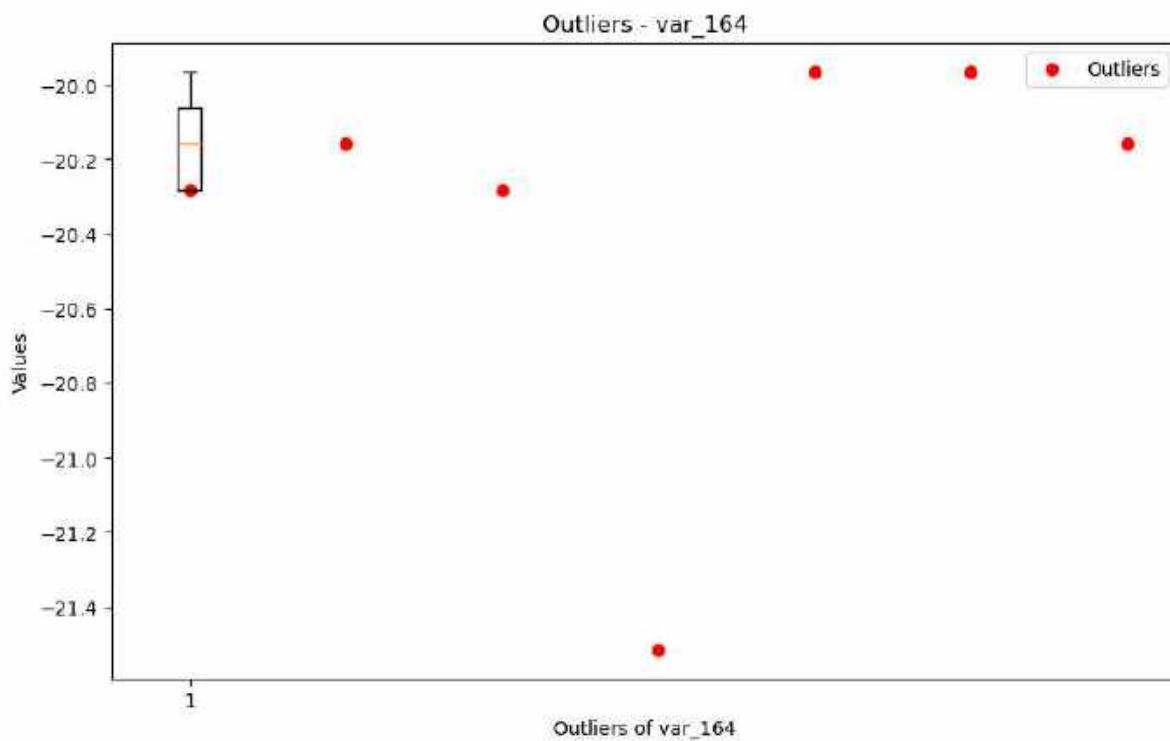


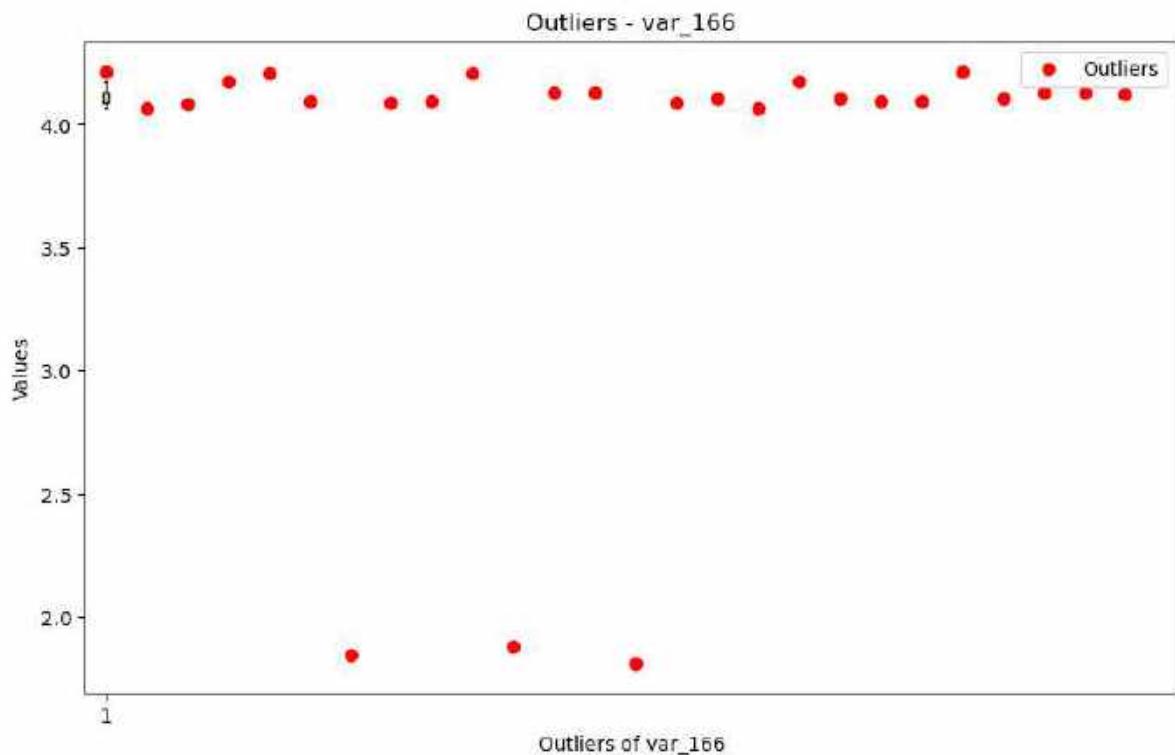
Total outliers in column var\_159: 153

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



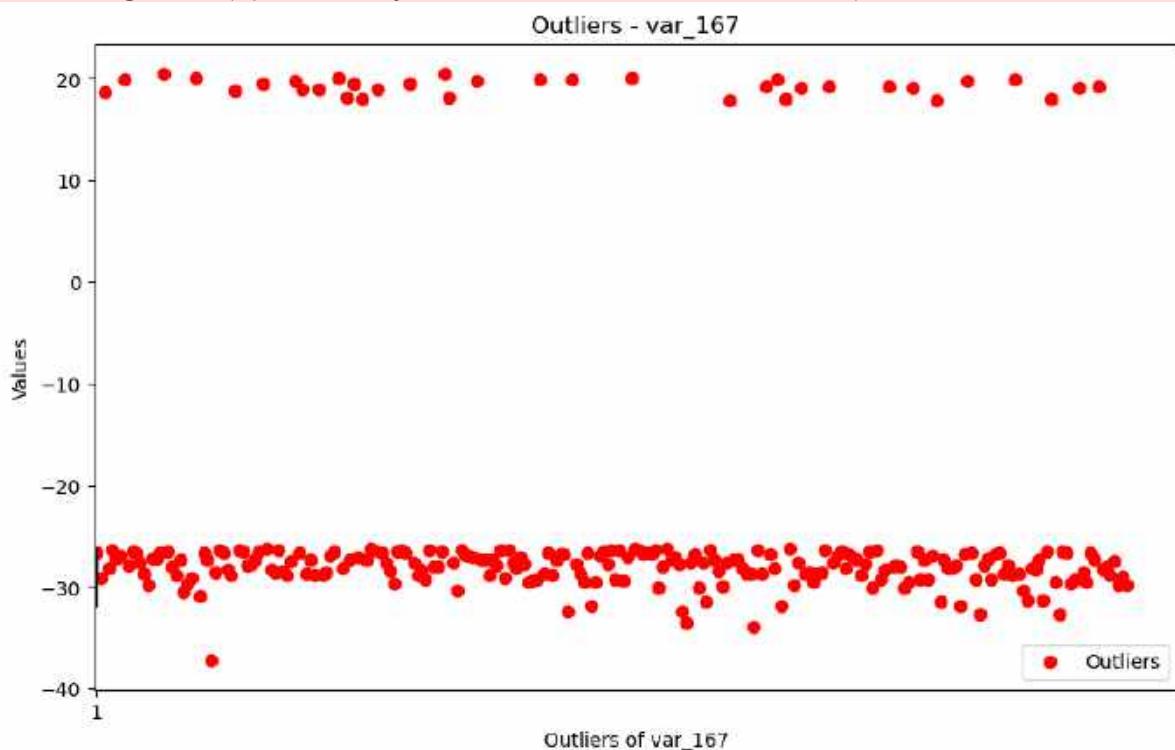






Total outliers in column var\_166: 26

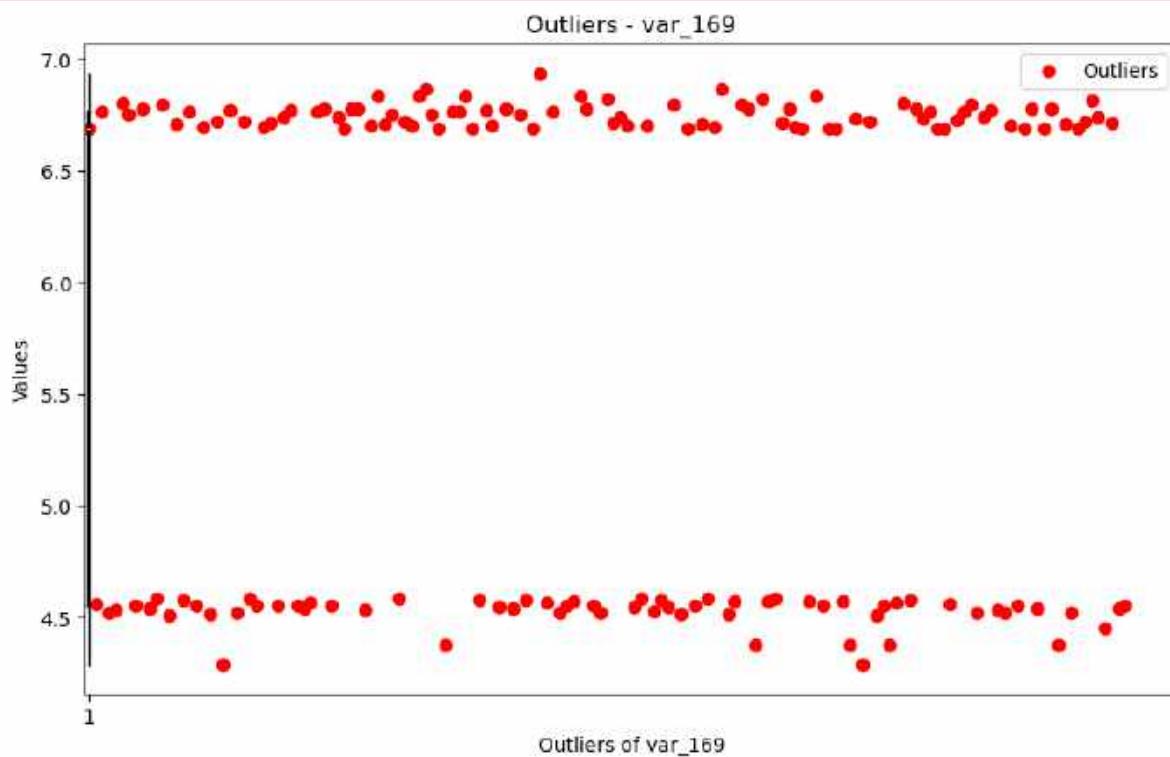
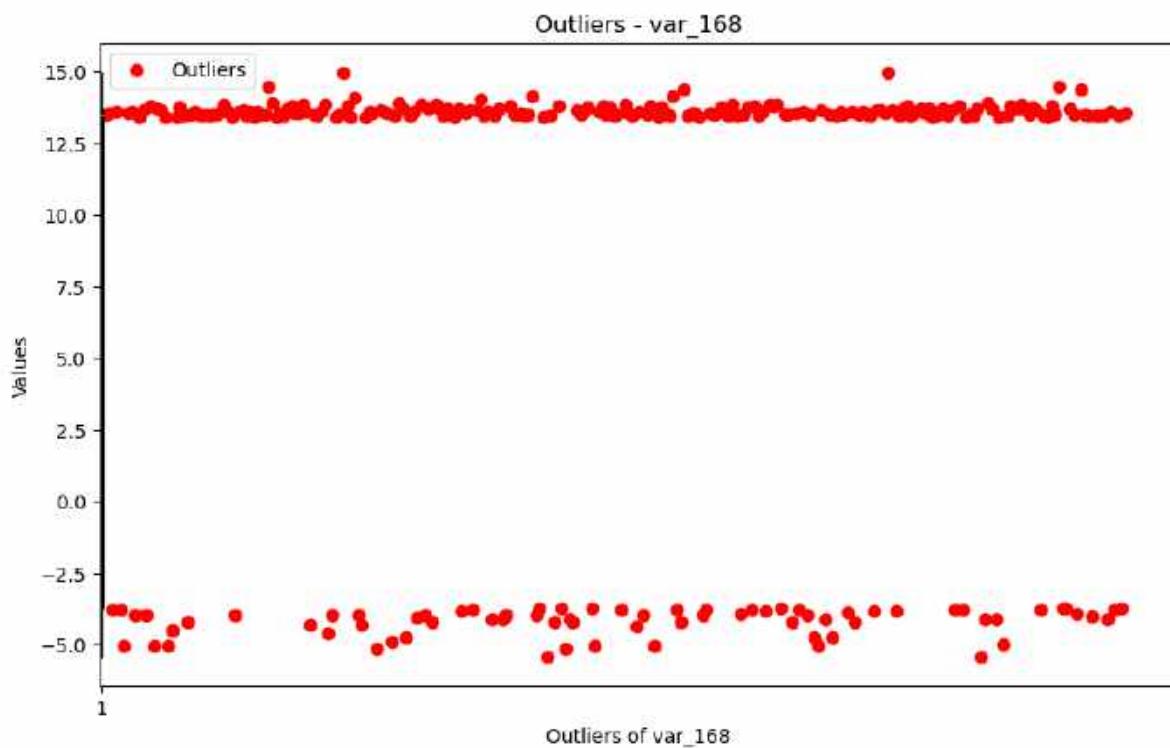
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

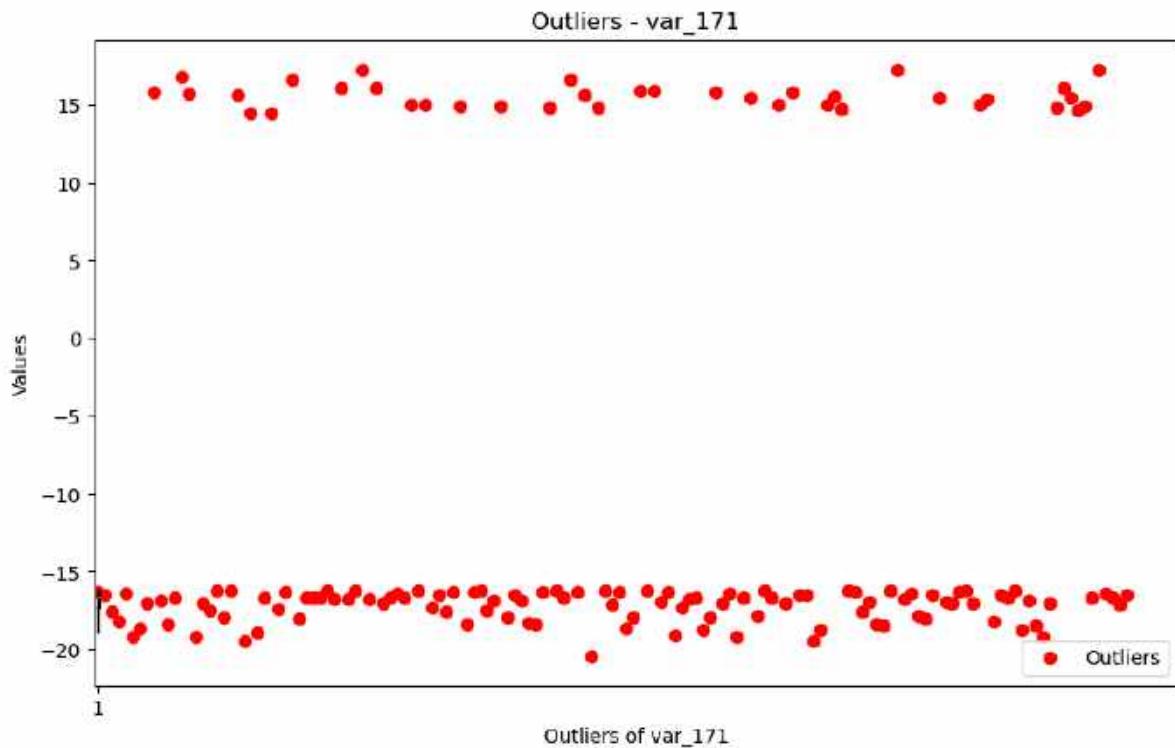
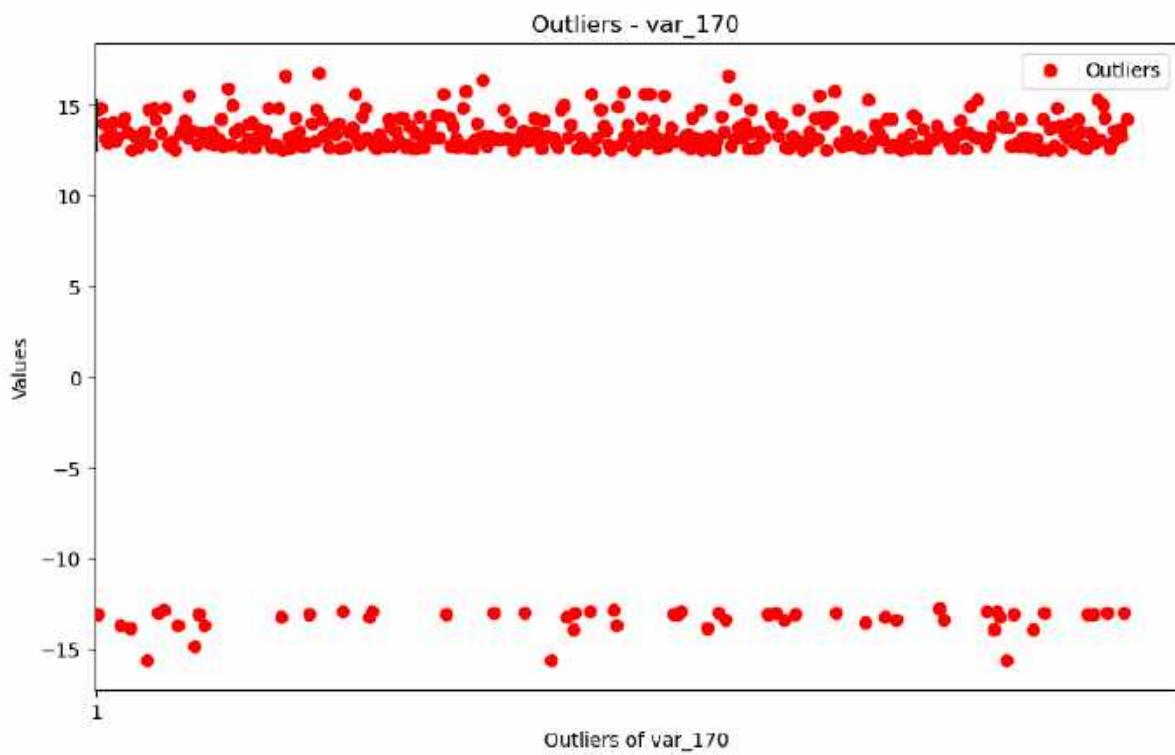


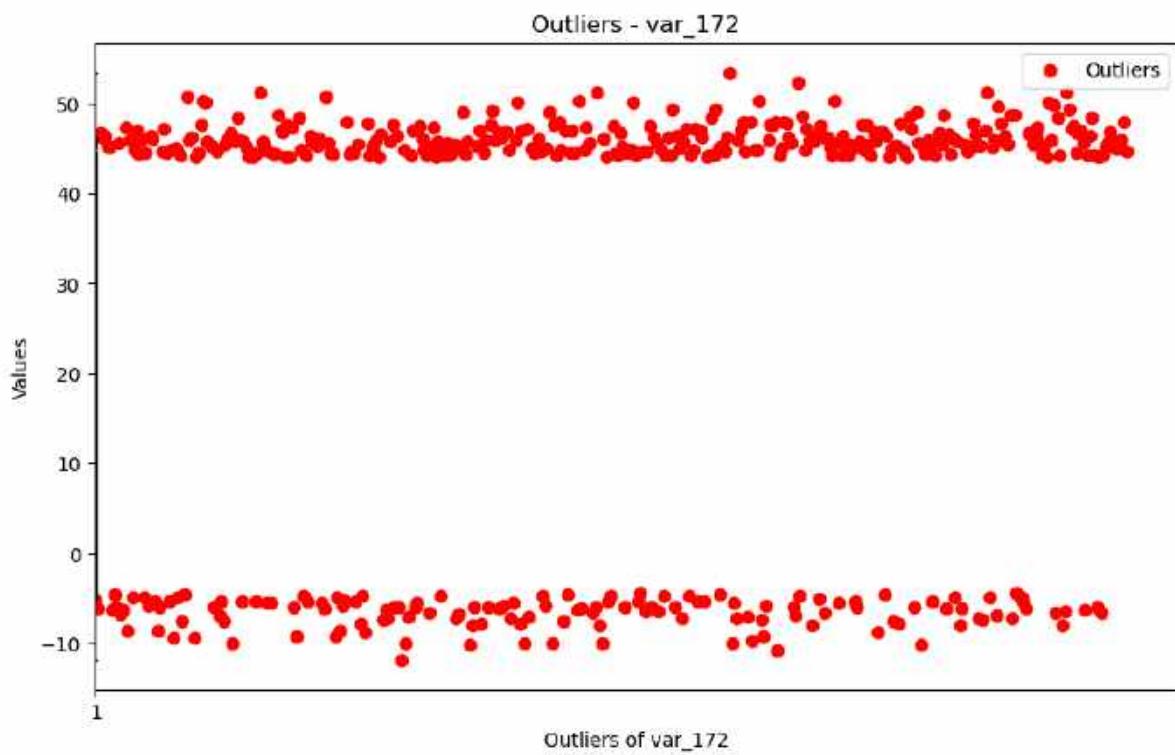
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.
```

```
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_167: 261

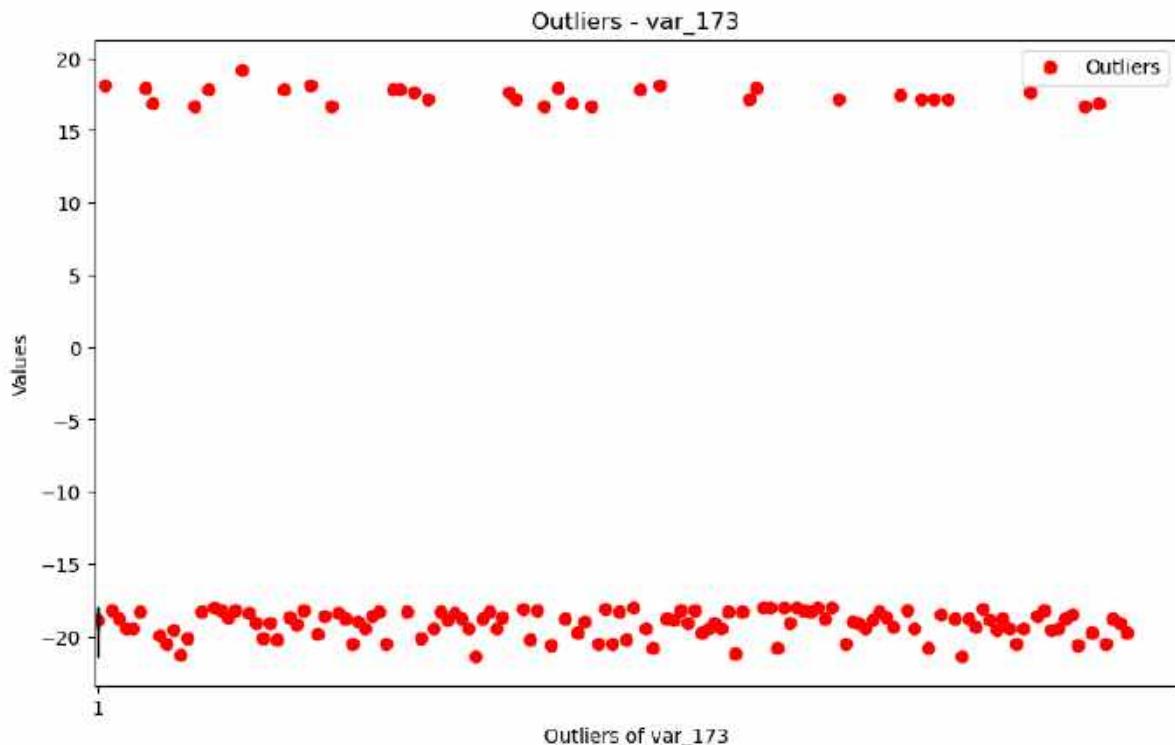






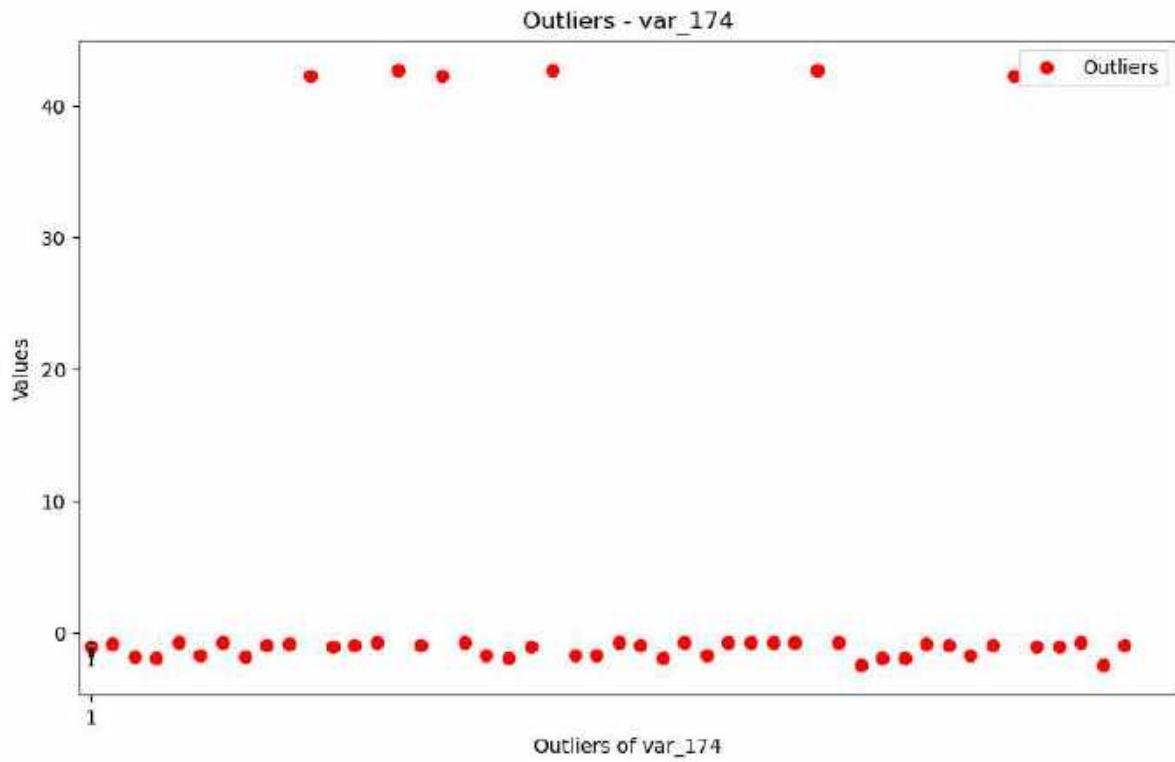
Total outliers in column var\_172: 453

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



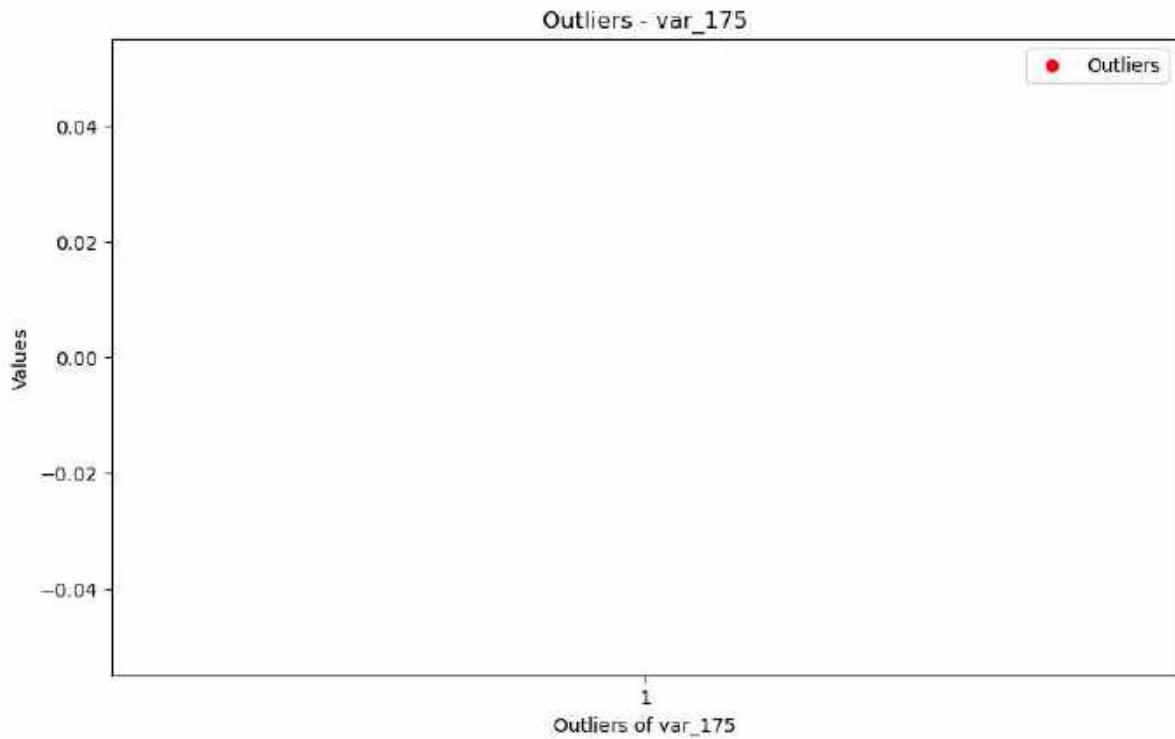
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_173: 151



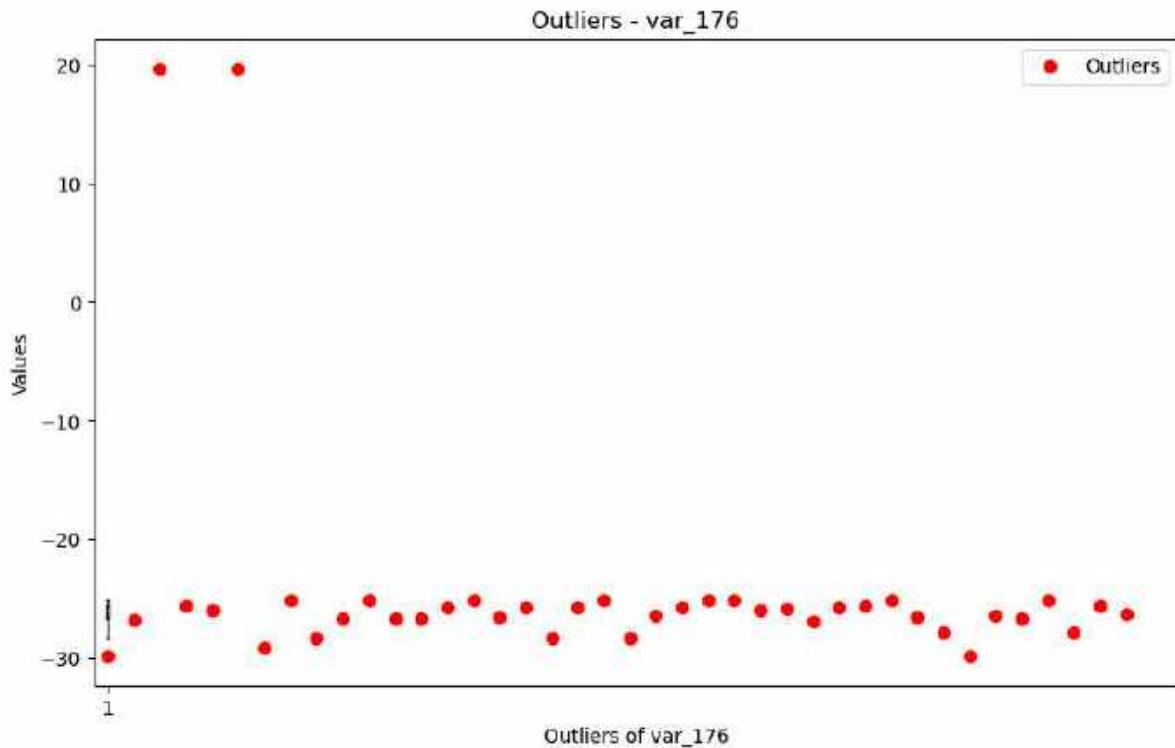
```
Total outliers in column var_174: 48
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
Total outliers in column var_175: 0
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



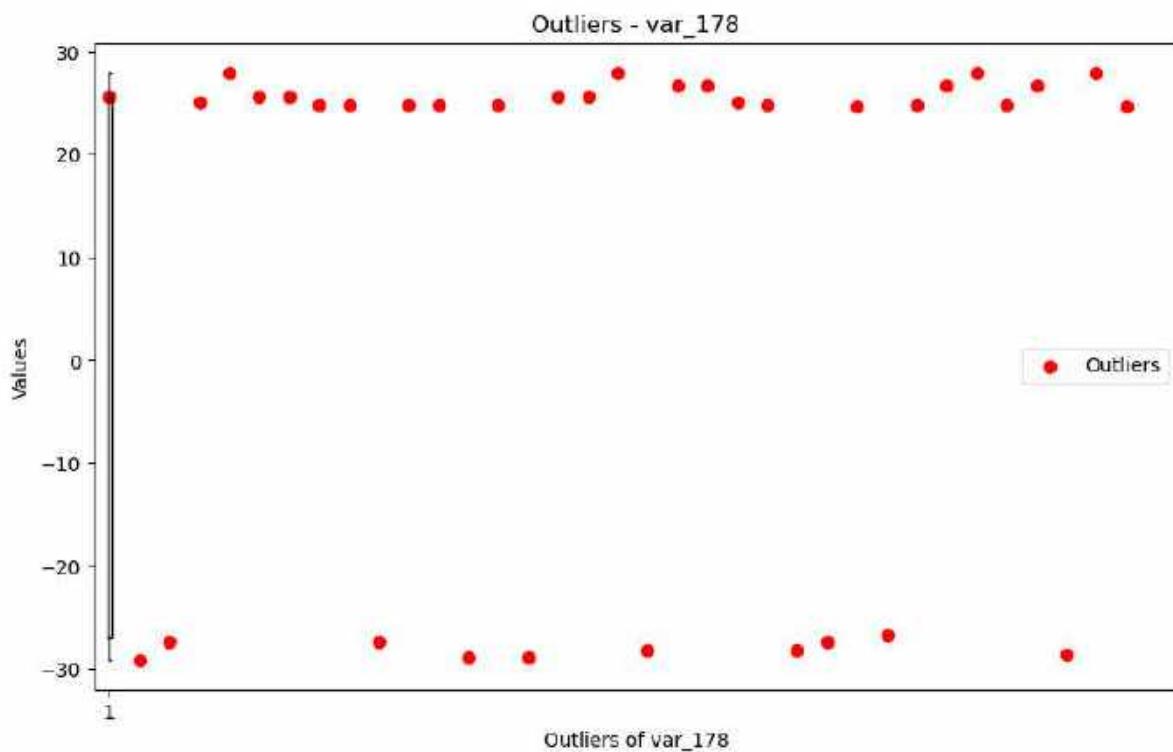
Total outliers in column var\_176: 40

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

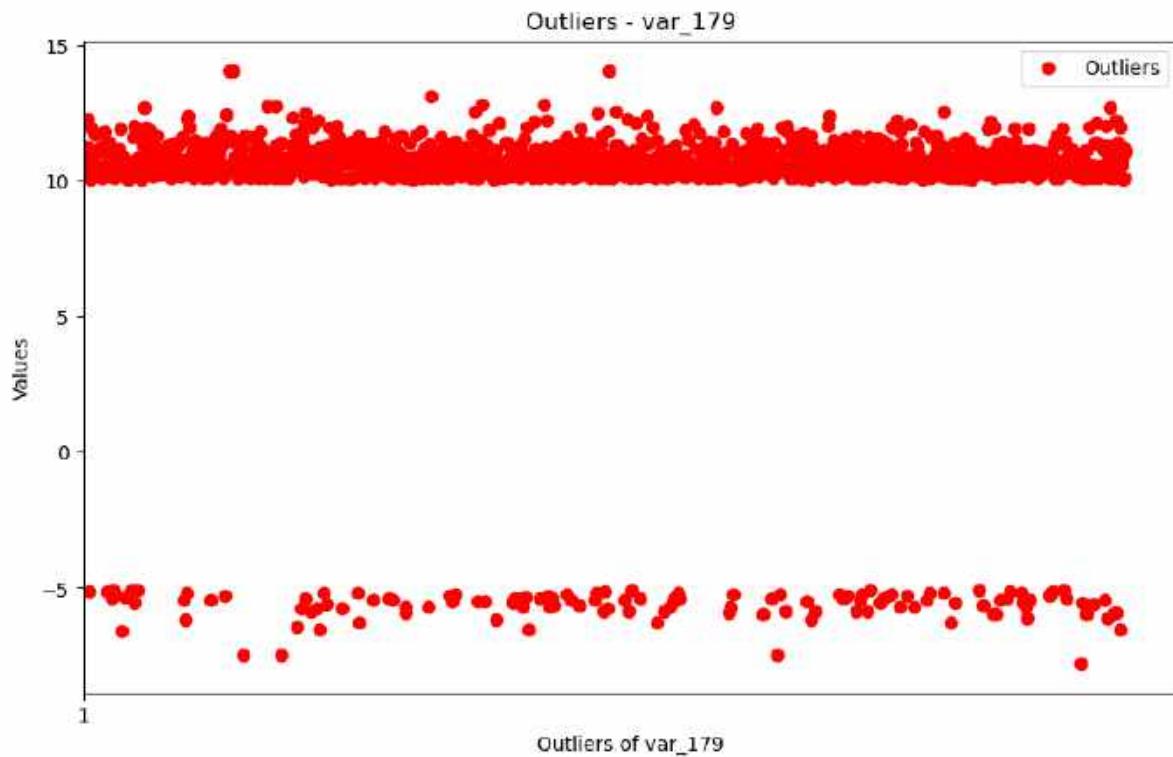


Total outliers in column var\_177: 0

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

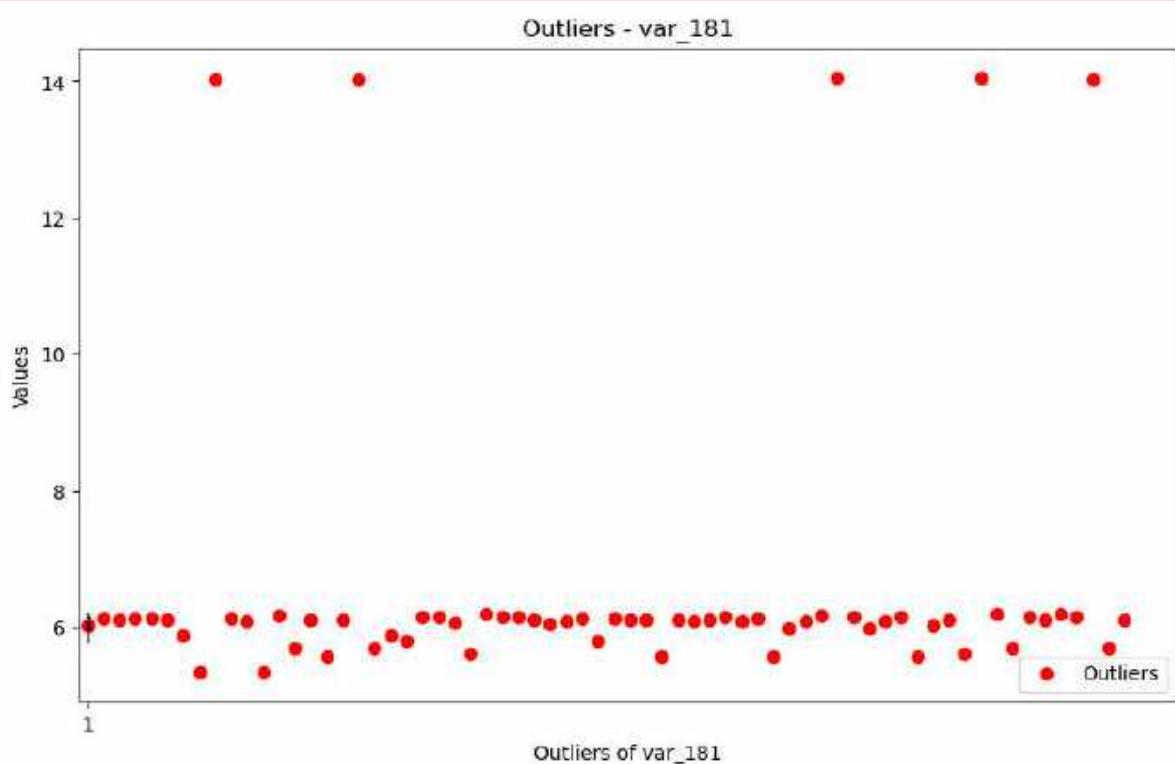
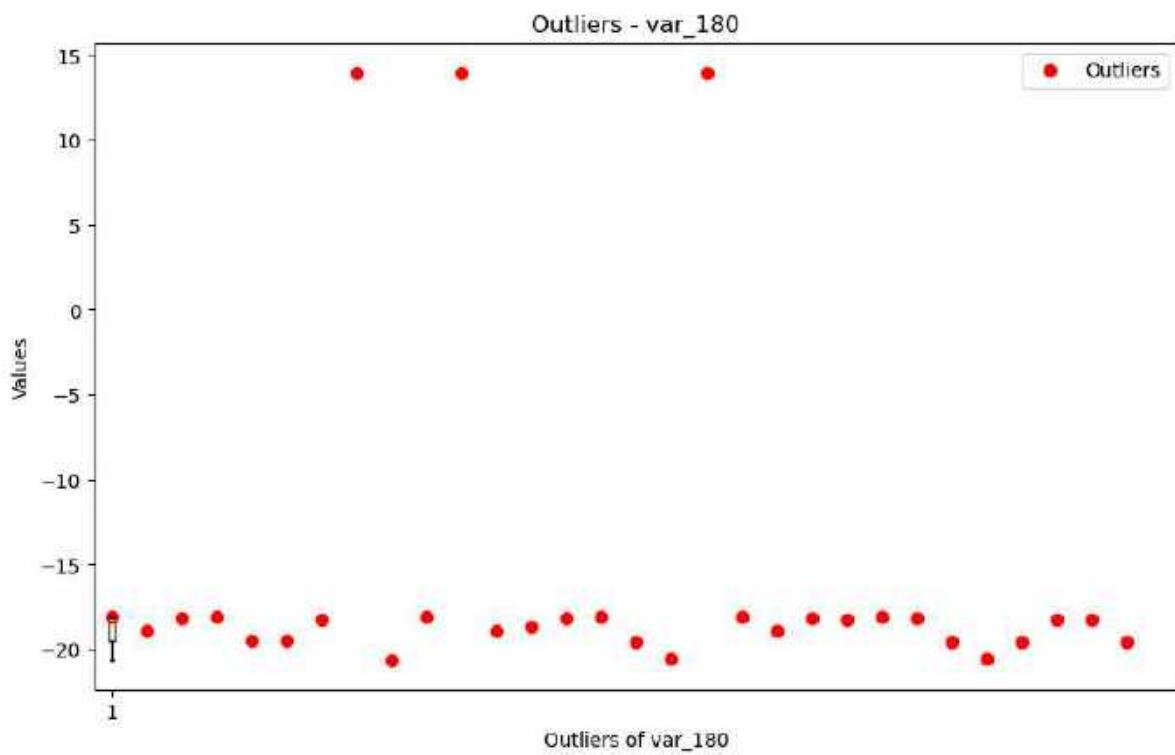


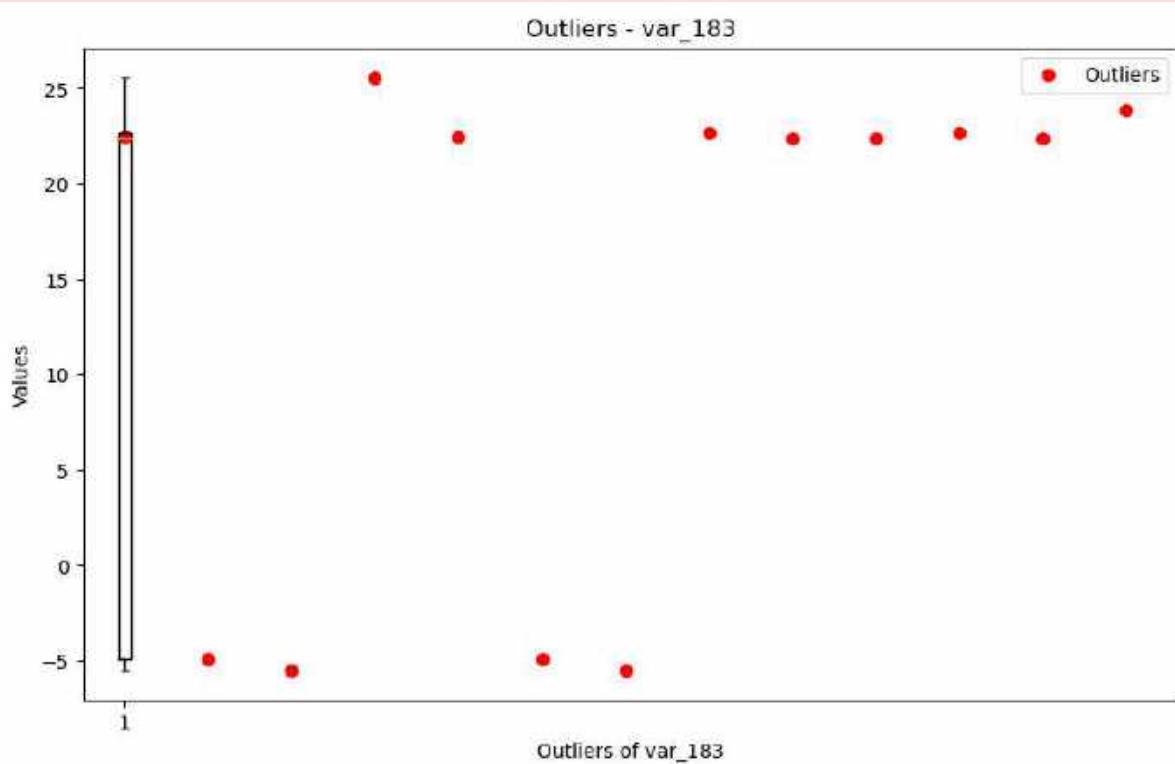
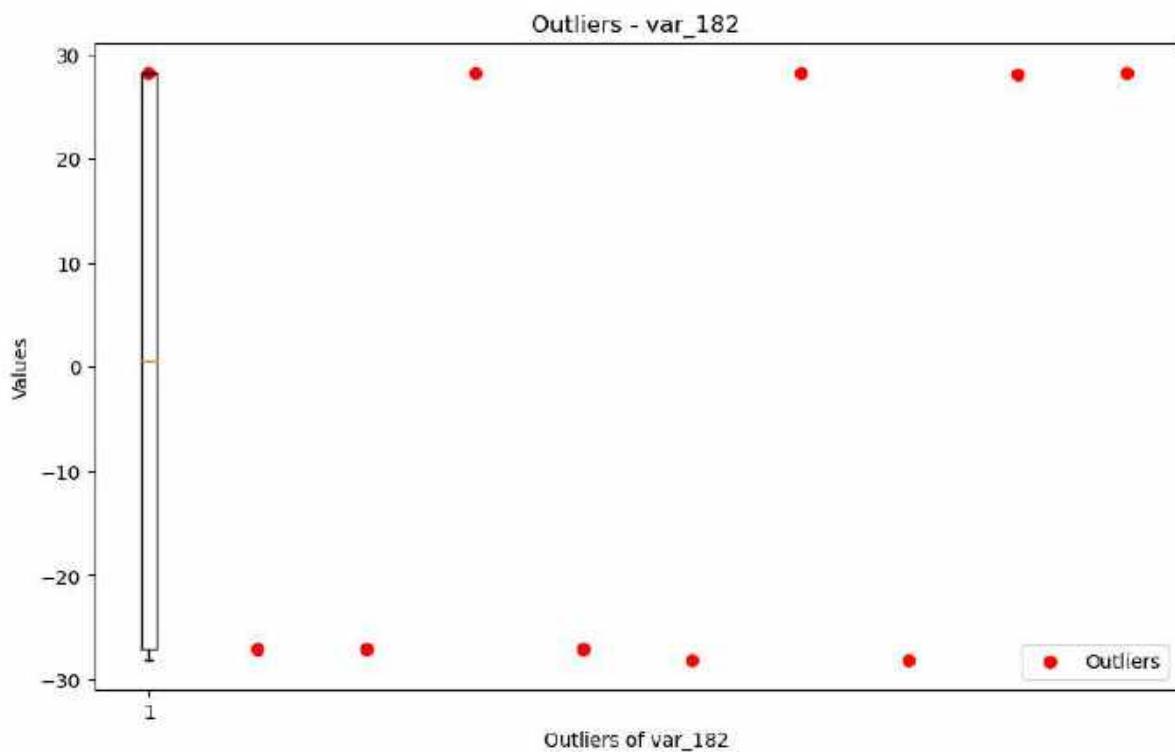
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")  
Total outliers in column var_178: 35
```

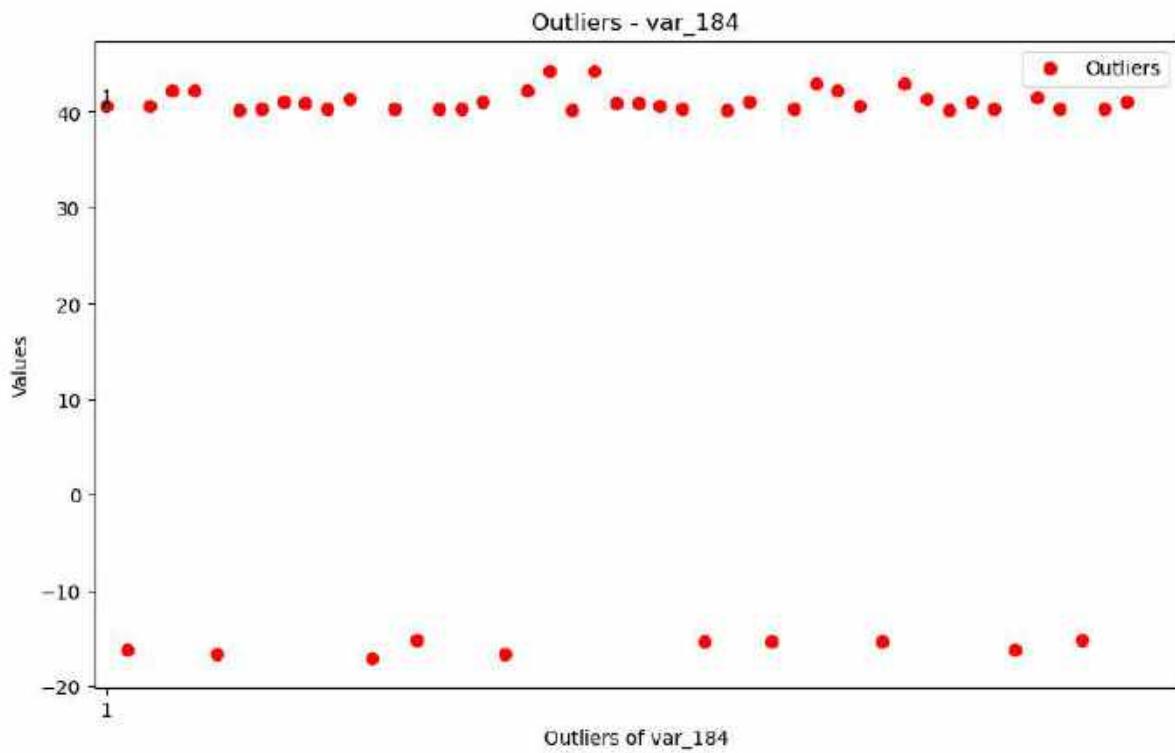


```
Total outliers in column var_179: 1524
```

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

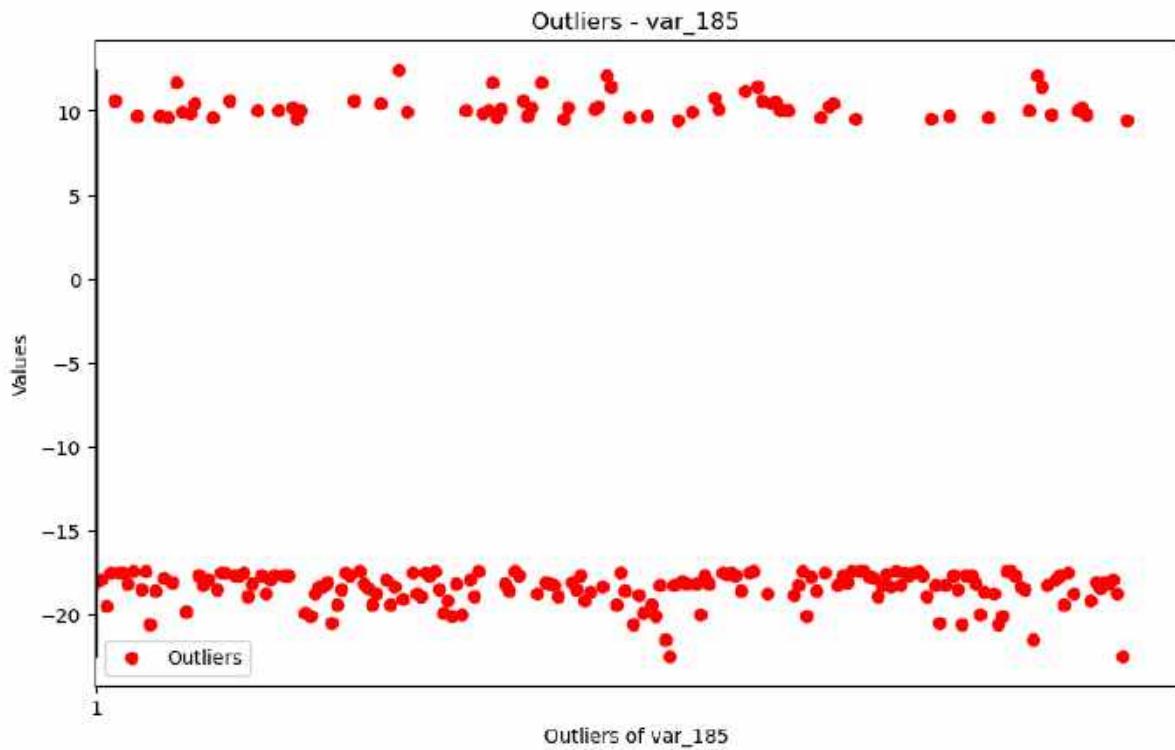






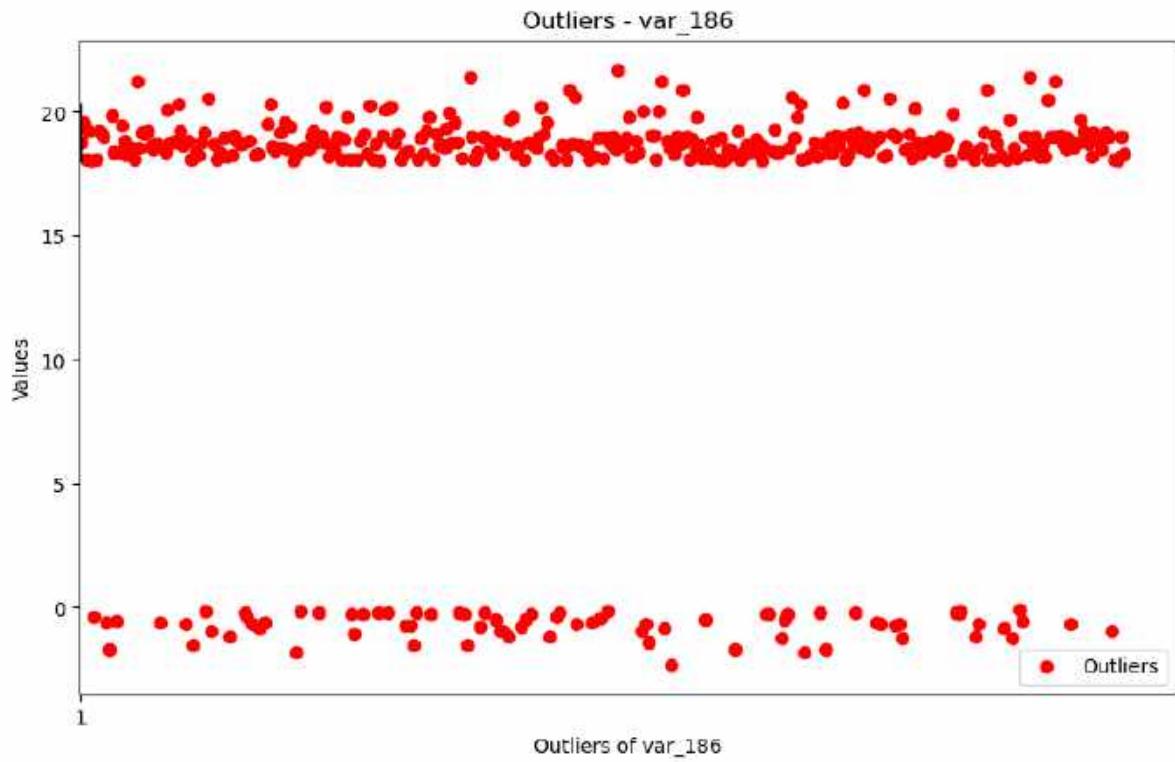
Total outliers in column var\_184: 47

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



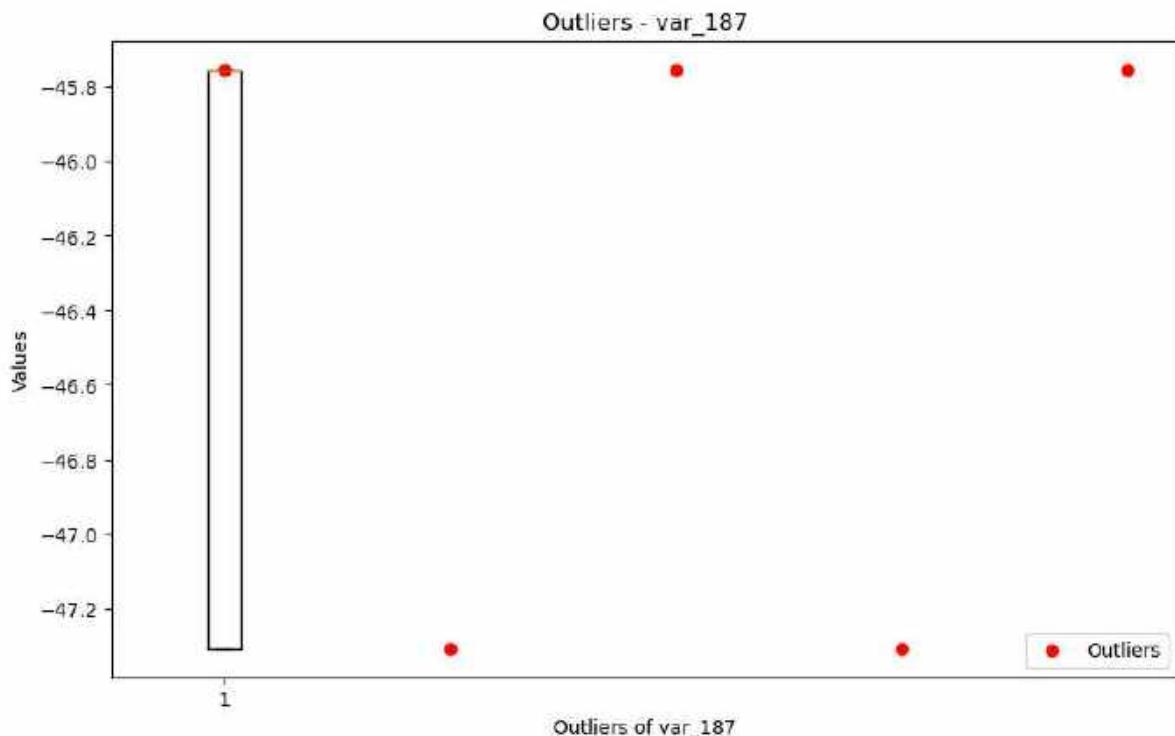
Total outliers in column var\_185: 233

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



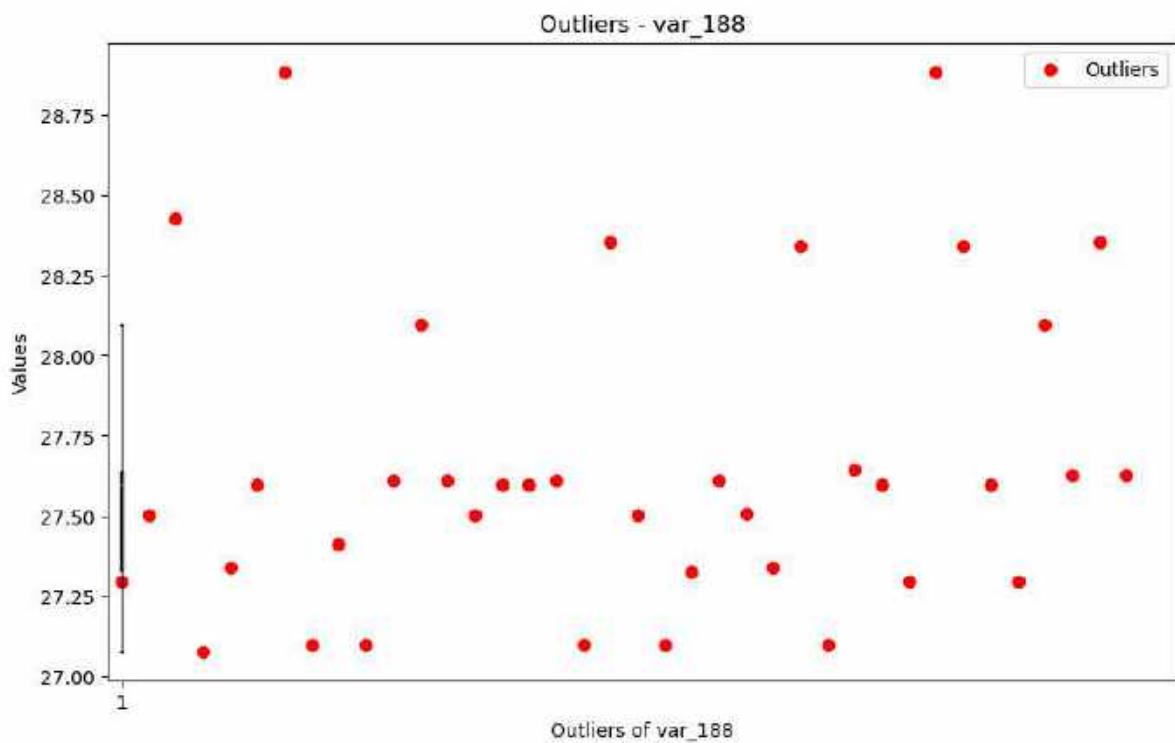
Total outliers in column var\_186: 409

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



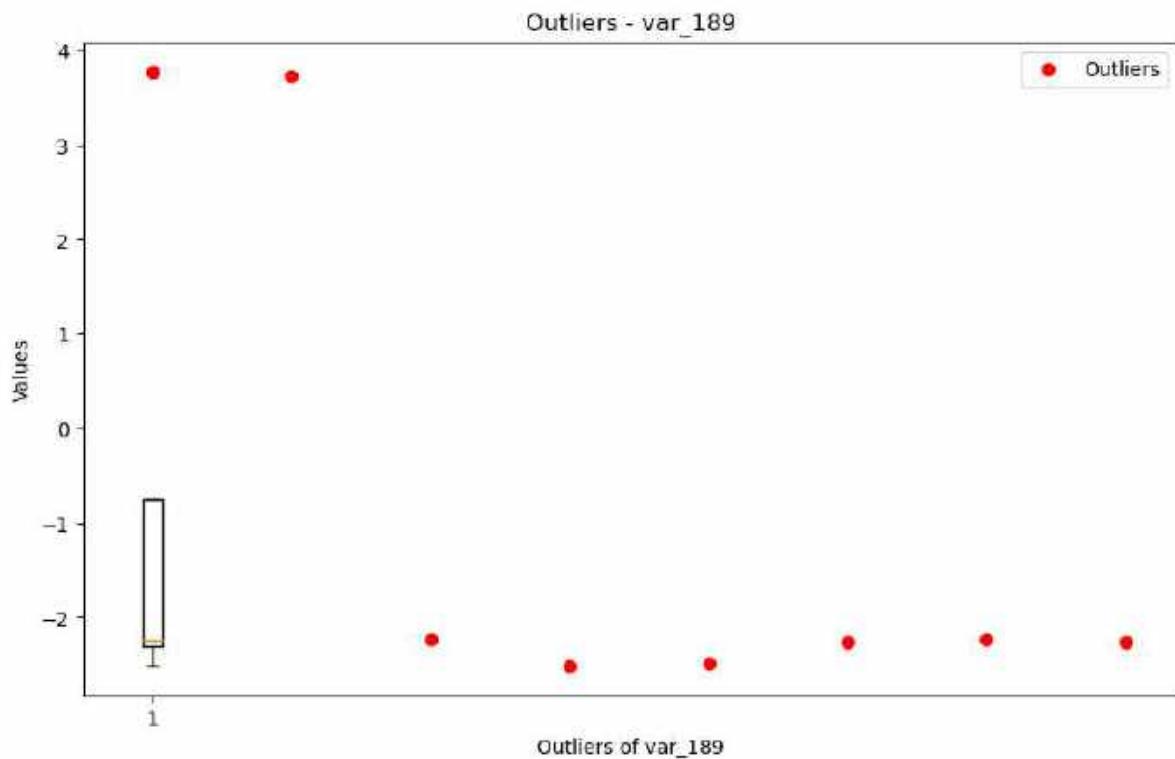
```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```

Total outliers in column var\_187: 5



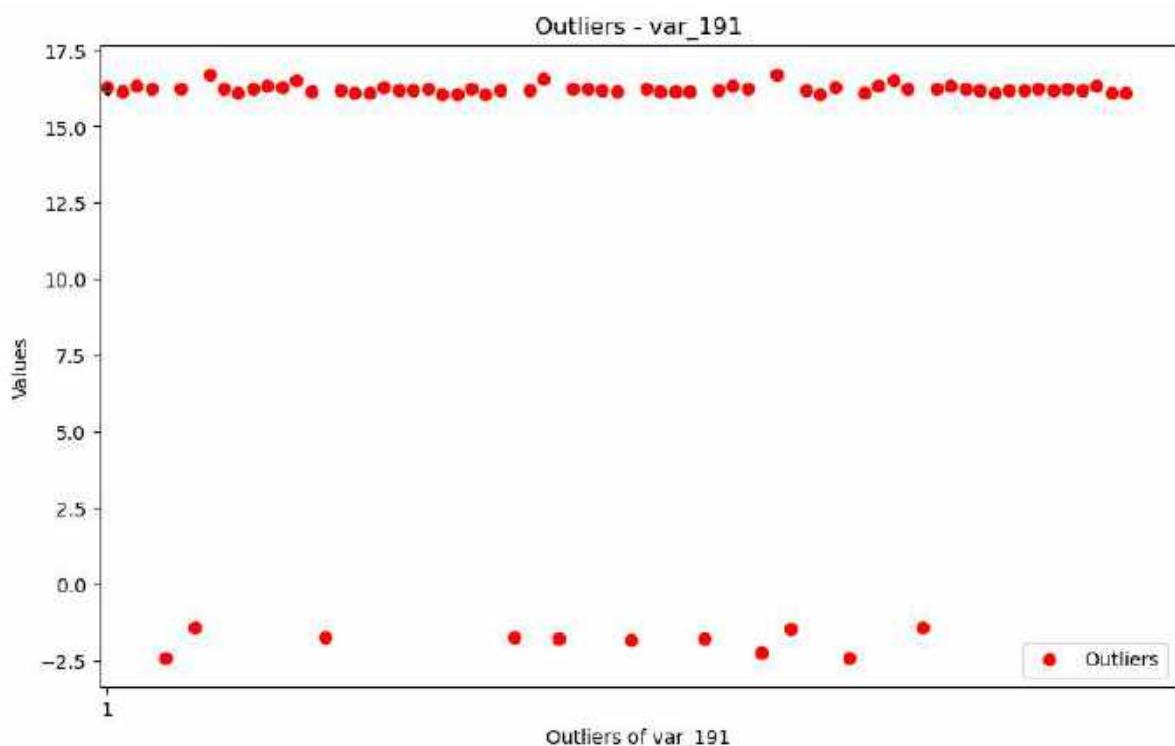
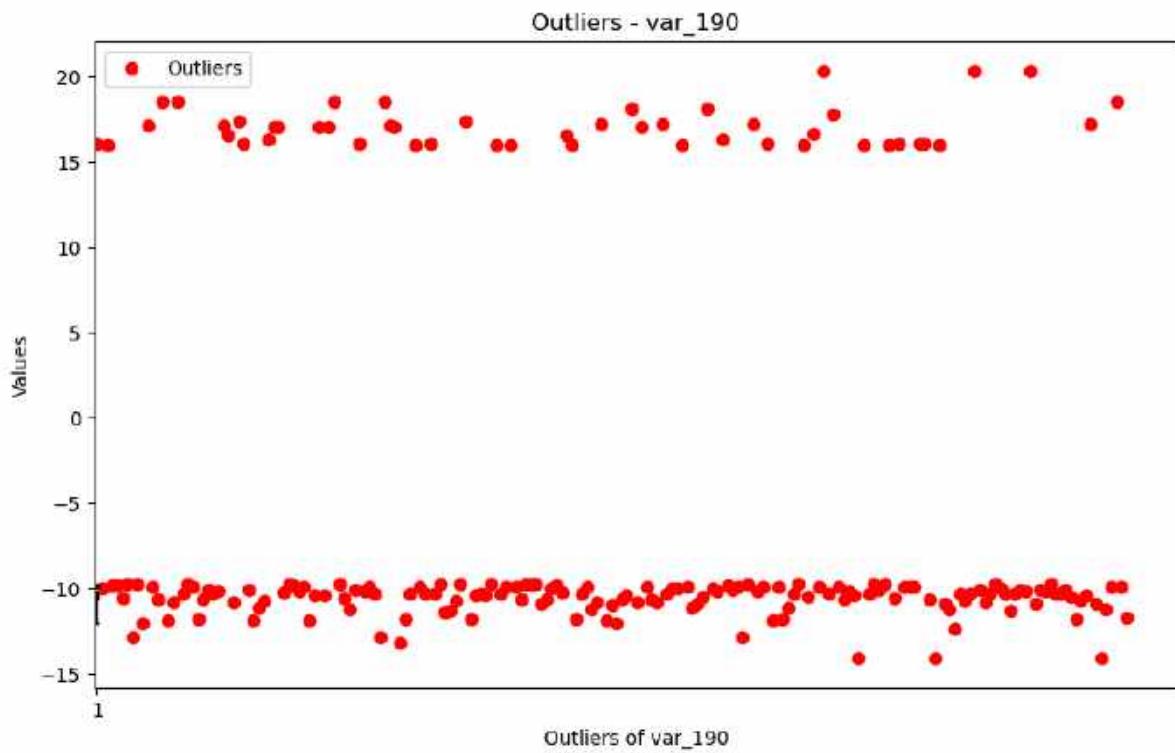
Total outliers in column var\_188: 38

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:
p-value may not be accurate for N > 5000.
warnings.warn("p-value may not be accurate for N > 5000.")
```

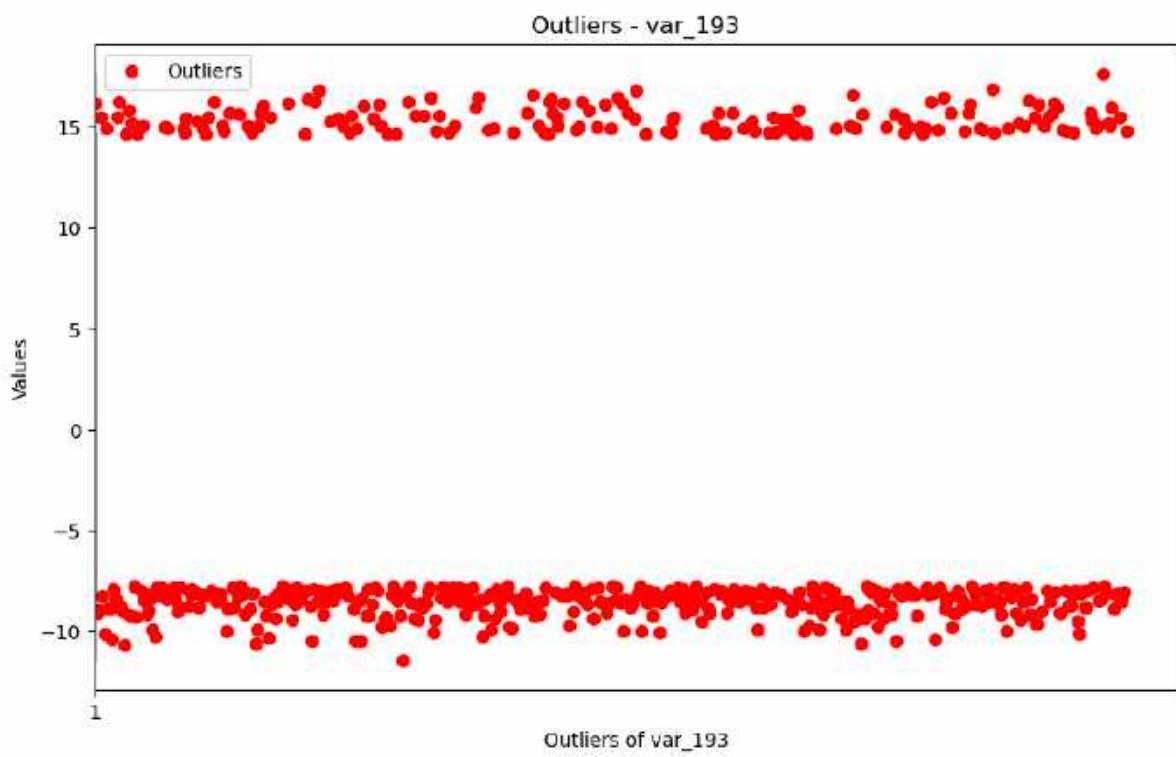
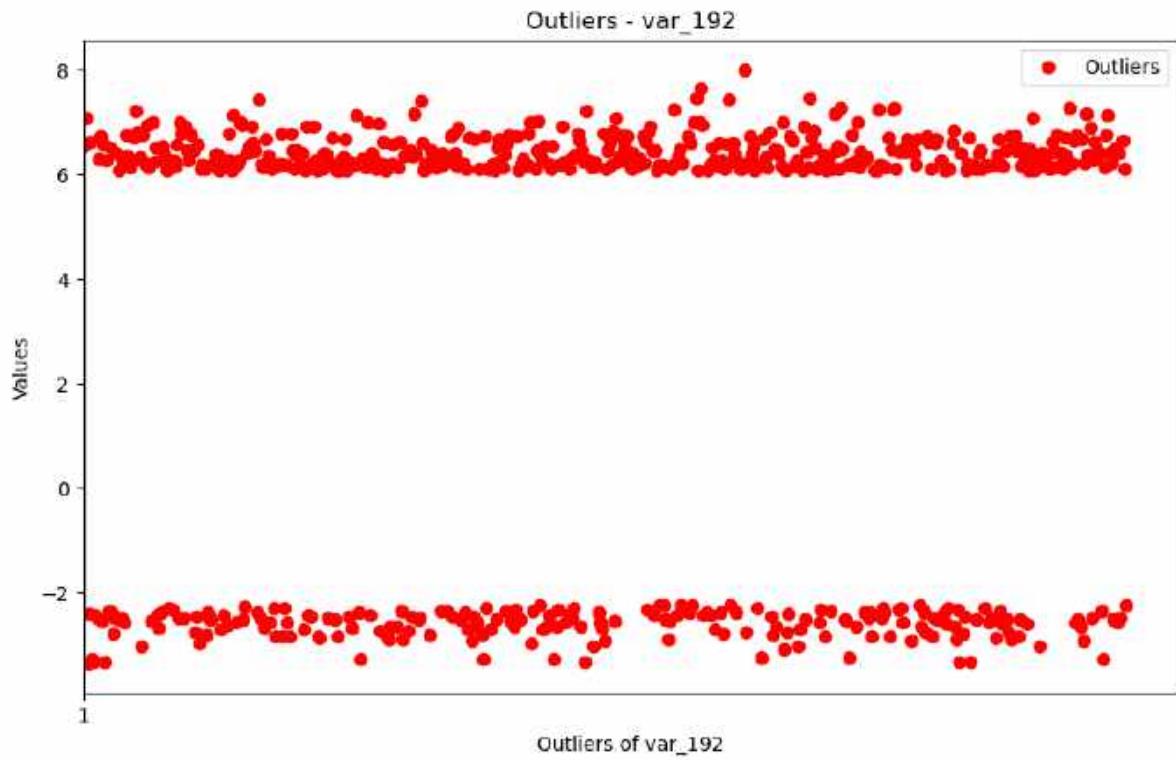


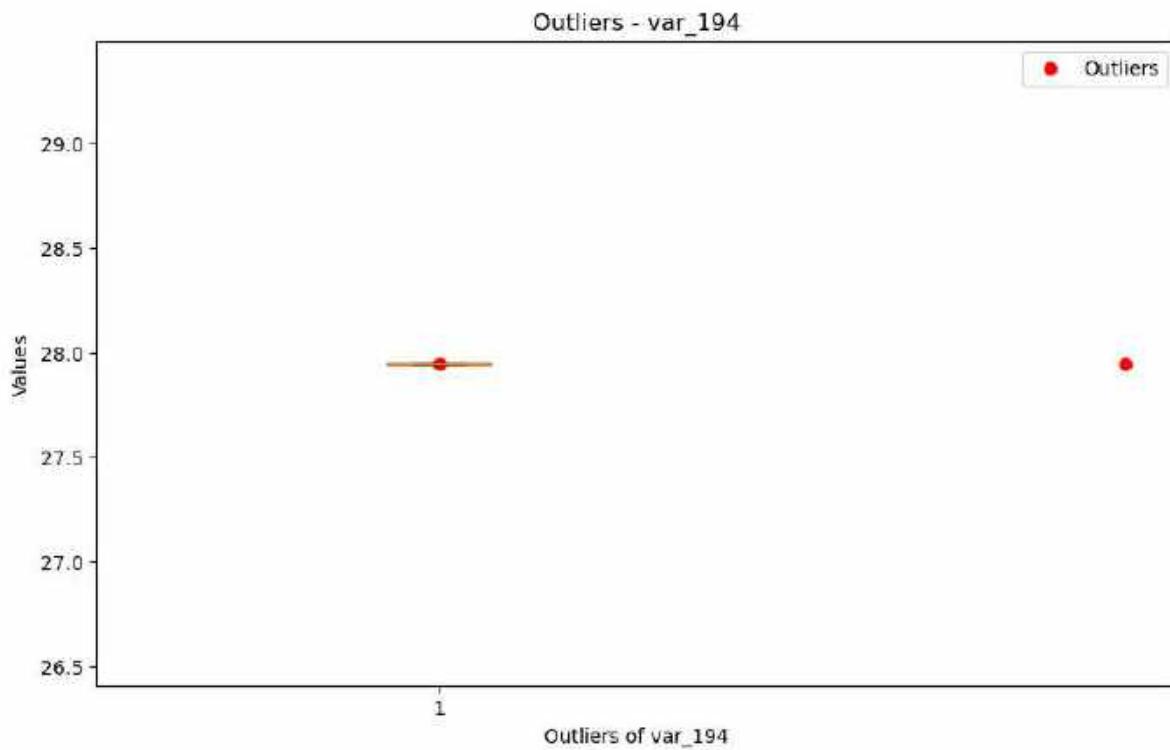
Total outliers in column var\_189: 8

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:
p-value may not be accurate for N > 5000.
warnings.warn("p-value may not be accurate for N > 5000.")
```



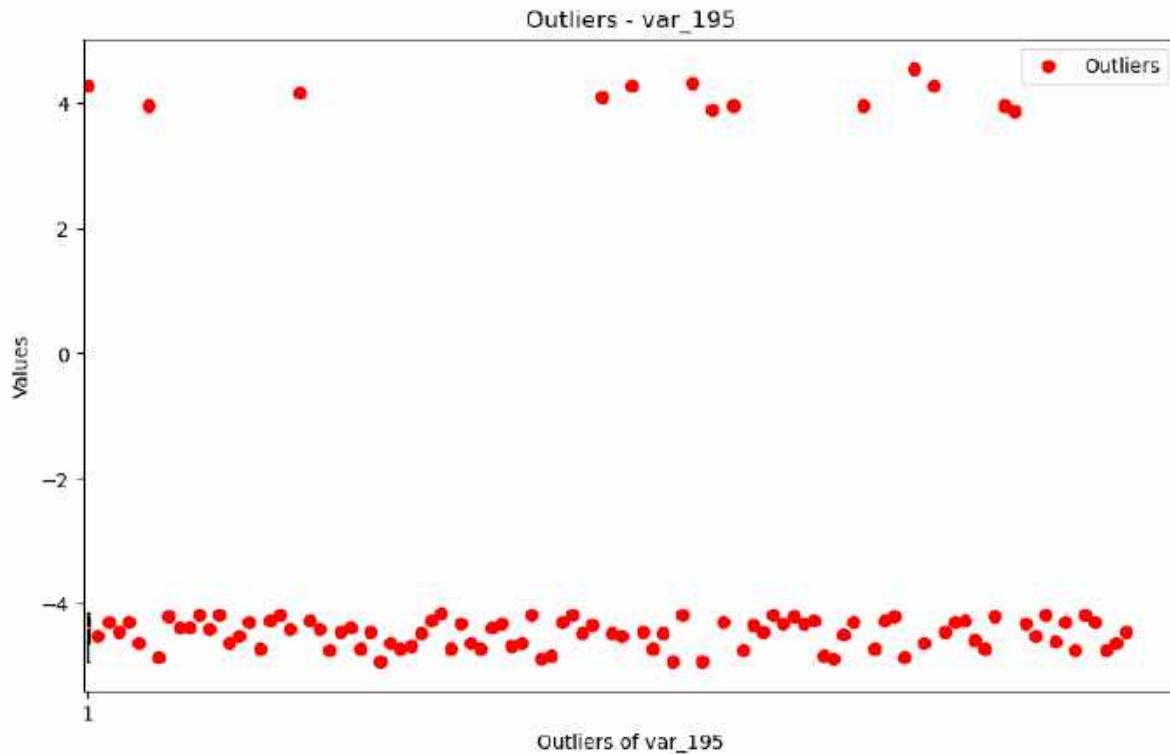
Total outliers in column var\_191: 71





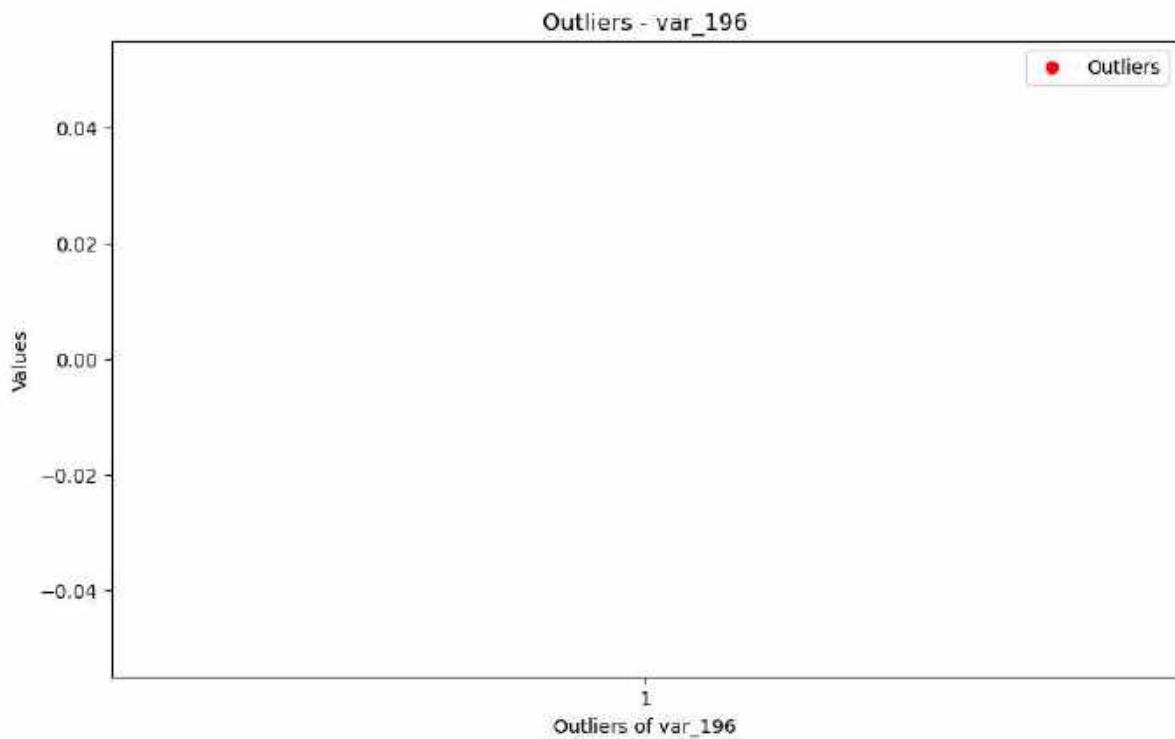
Total outliers in column var\_194: 2

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



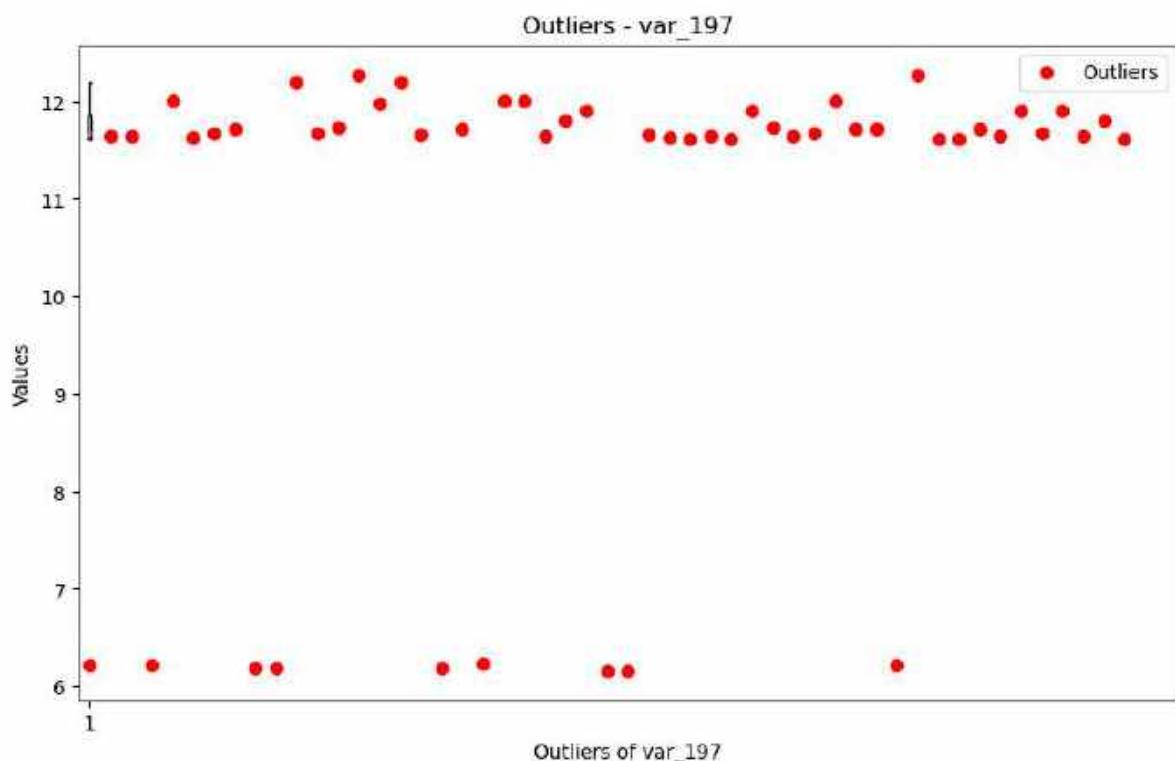
Total outliers in column var\_195: 104

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



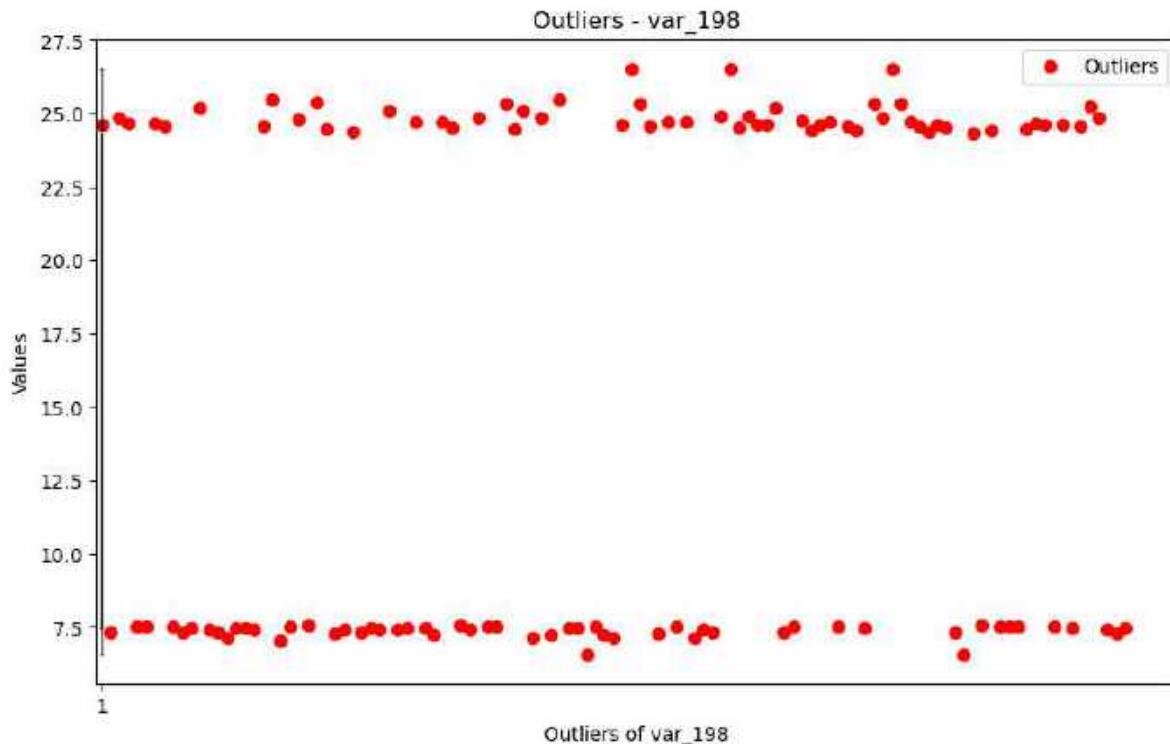
Total outliers in column var\_196: 0

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



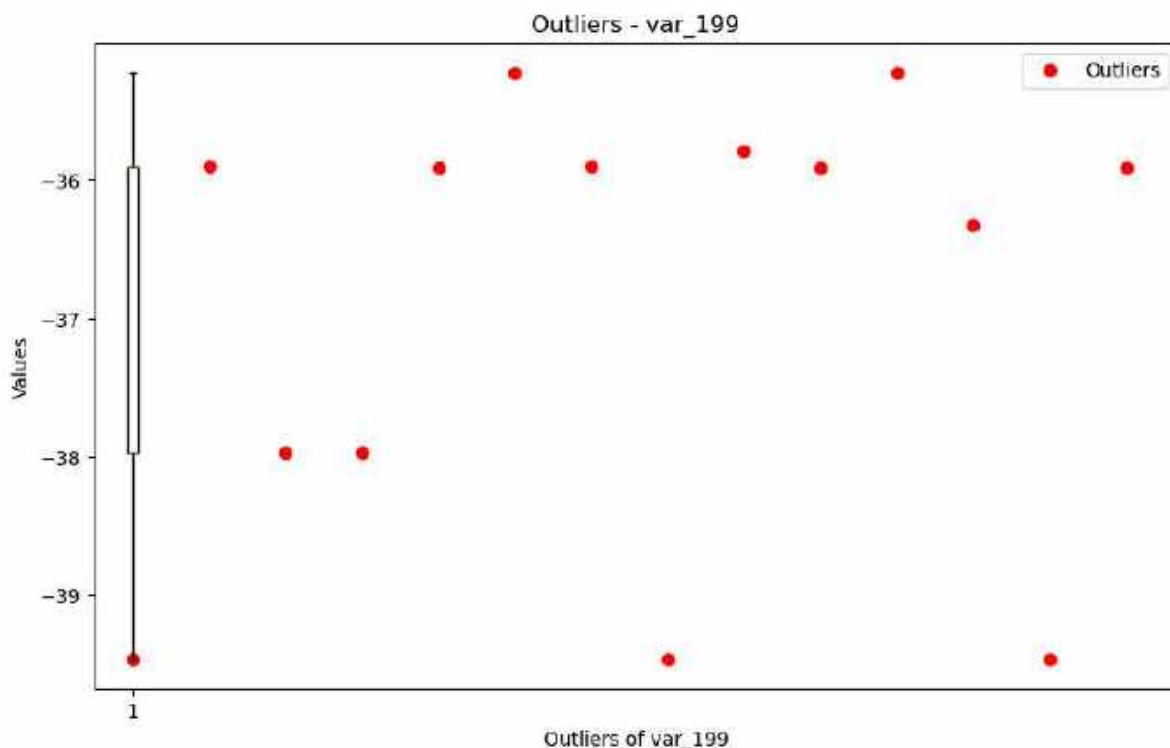
Total outliers in column var\_197: 51

```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:  
p-value may not be accurate for N > 5000.  
warnings.warn("p-value may not be accurate for N > 5000.")
```



```
E:\Jupyter notebook\lib\site-packages\scipy\stats\_morestats.py:1816: UserWarning:
p-value may not be accurate for N > 5000.
warnings.warn("p-value may not be accurate for N > 5000.")

Total outliers in column var_198: 115
```



```
Total outliers in column var_199: 14
```

## Transformation (Standardization, Normalization, encoding categorical to numerical)

```
In [39]: from sklearn.preprocessing import StandardScaler, MinMaxScaler

# numeric columns for transformation
columns_to_analyze = data_training_set.select_dtypes(include=np.number).columns
```

```
# Standardization
scaler = StandardScaler()
data_training_set[columns_to_analyze] = scaler.fit_transform(data_training_set[columns_to_analyze])

# Normalization
minmax_scaler = MinMaxScaler()
data_training_set[columns_to_analyze] = minmax_scaler.fit_transform(data_training_set[columns_to_analyze])
```

- Define your research questions/objectives
  - Perform Hypothesis testing
  - Interpret the results findings in the context of your research question or objective. Draw conclusions and make recommendations based on your analysis.
  - Communicate your results: Present your insights and conclusions in a clear and concise manner, using visualizations and descriptive statistics. Tailor your communication to your audience, whether it be technical or non-technical.

## Research question: Can we predict whether customers will make a particular transaction in the future?

In [40]:

```
train_data = data_training_set

# Splitting the training set into features (X) and target variable (y)
X = train_data.drop(['ID_code', 'target'], axis=1)
y = train_data['target']

alpha = 0.05

# Performing the t-test and calculating the p-value
t_statistic, p_value = stats.ttest_ind(X, y)

print("P-value:", p_value[0])
if p_value[0] < alpha:
    print('Reject the null hypothesis. There is a significant relationship between')
else:
    print('Fail to reject the null hypothesis. There is no significant relationship')
```

P-value: 0.0  
Reject the null hypothesis. There is a significant relationship between the features and the target variable.

- For predicting future transactions, my first choice would be the logistic regression model because of its interpretability and ability to handle binary classification tasks effectively in this context.
- To validate the results, I will assess the training score, validation score, and ROC AUC score. These metrics will provide insights into the model's performance, generalization ability, and its capacity to distinguish between classes.

# logistic regression model

```
In [41]: from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, roc_auc_score
from sklearn.preprocessing import StandardScaler

# Loading the training and test sets
train_data = data_training_set
test_data = data_test_set

# Splitting the training set into features (X) and target variable (y)
X = train_data.drop(['ID_code', 'target'], axis=1)
y = train_data['target']

# Splitting the training set into a training subset and a validation subset
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state=42)

# Scale the features
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_val_scaled = scaler.transform(X_val)

# Training a logistic regression model with increased max_iter
model = LogisticRegression(max_iter=1000, solver='liblinear')
model.fit(X_train_scaled, y_train)

# Making predictions on the validation set
val_predictions = model.predict(X_val_scaled)

# Calculating the accuracy and ROC AUC score of the model on the validation set
accuracy = accuracy_score(y_val, val_predictions)
roc_auc = roc_auc_score(y_val, val_predictions)
print('Validation Accuracy:', accuracy)
print('ROC AUC Score:', roc_auc)

# Making predictions on the test set
X_test = test_data.drop('ID_code', axis=1)
X_test_scaled = scaler.transform(X_test)
test_predictions = model.predict(X_test_scaled)
```

Validation Accuracy: 0.9131  
 ROC AUC Score: 0.6278981597630602

```
In [42]: from sklearn.metrics import accuracy_score

# Training model on the training set
model.fit(X_train, y_train)

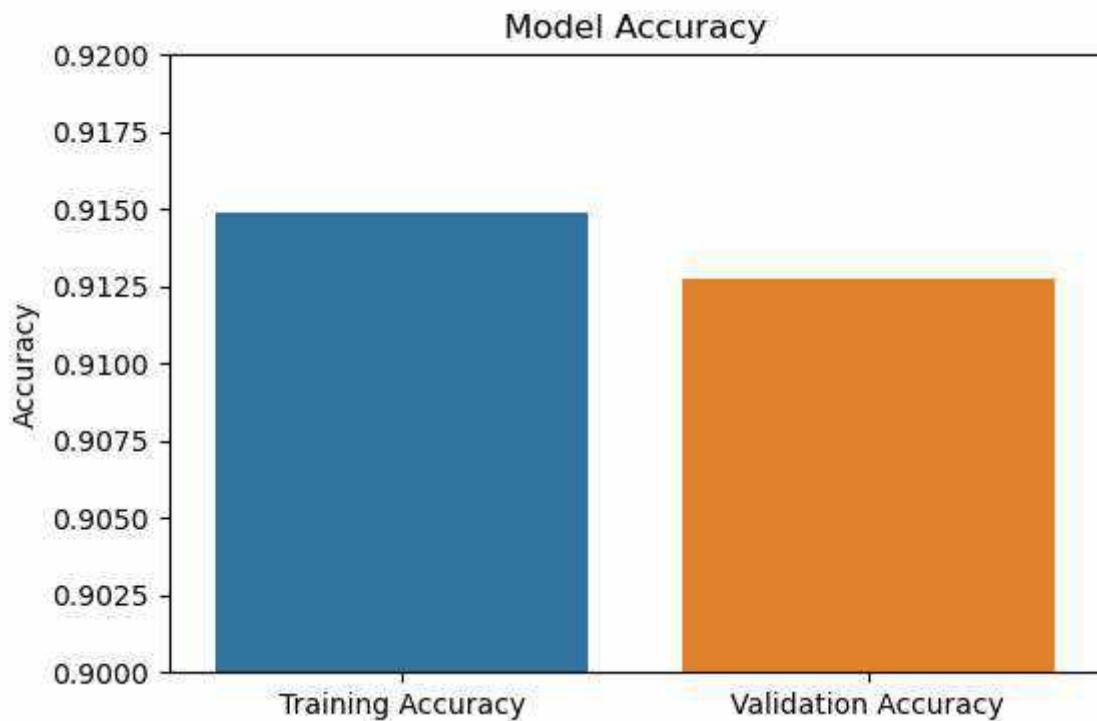
# Making predictions on the training set and calculating accuracy
train_predictions = model.predict(X_train)
train_accuracy = accuracy_score(y_train, train_predictions)

# Comparing the training accuracy and validation accuracy
print('Training Accuracy:', train_accuracy)
print('Validation Accuracy:', accuracy)
```

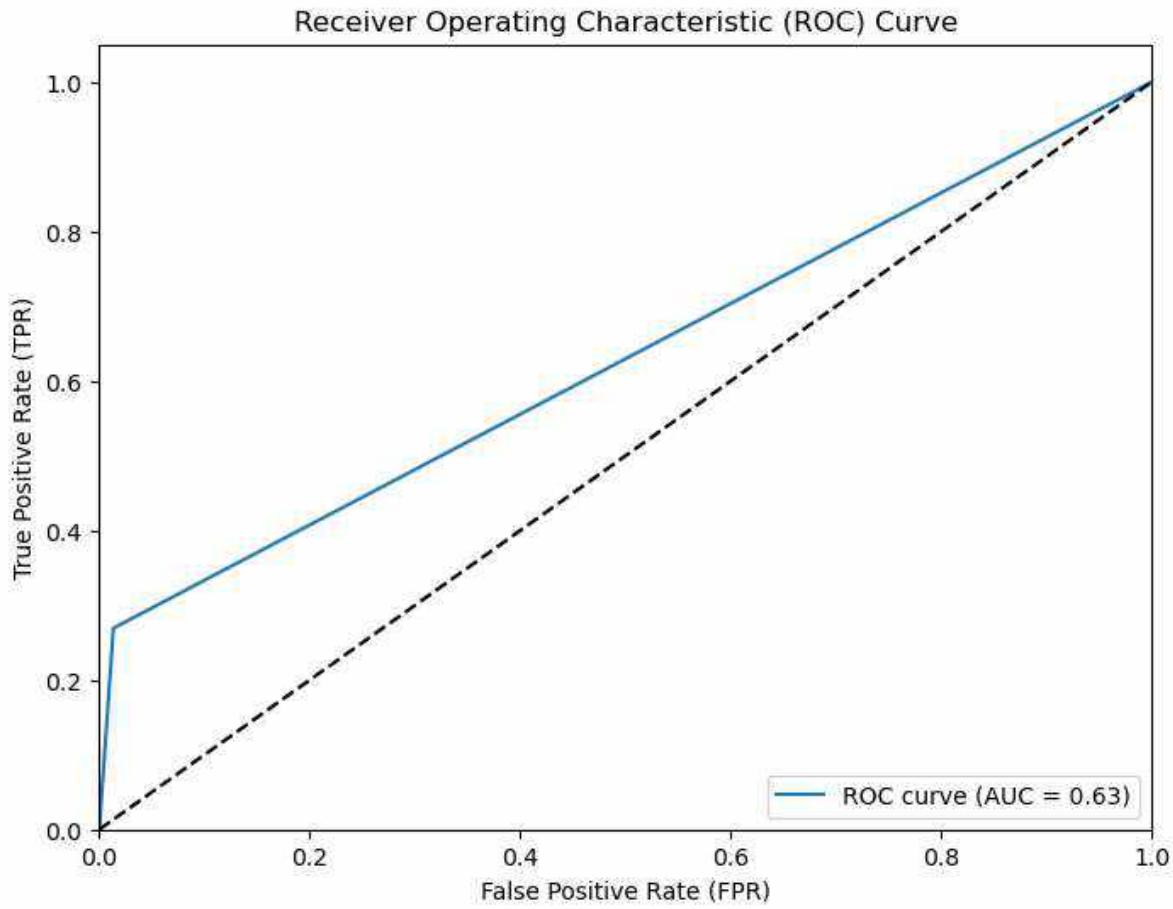
Training Accuracy: 0.9149125  
 Validation Accuracy: 0.9131

```
In [43]: # bar plot to visualize the training and validation accuracies
plt.figure(figsize=(6, 4))
```

```
sns.barplot(x=['Training Accuracy', 'Validation Accuracy'], y=[0.91490625, 0.91272])  
plt.ylim(0.9, 0.92)  
plt.ylabel('Accuracy')  
plt.title('Model Accuracy')  
plt.show()
```



```
In [44]: from sklearn.metrics import roc_curve, auc  
  
# the false positive rate (FPR) and true positive rate (TPR)  
fpr, tpr, thresholds = roc_curve(y_val, val_predictions)  
  
# the AUC score  
roc_auc = auc(fpr, tpr)  
  
# Plotting the ROC curve  
plt.figure(figsize=(8, 6))  
plt.plot(fpr, tpr, label='ROC curve (AUC = %0.2f)' % roc_auc)  
plt.plot([0, 1], [0, 1], 'k--')  
plt.xlim([0.0, 1.0])  
plt.ylim([0.0, 1.05])  
plt.xlabel('False Positive Rate (FPR)')  
plt.ylabel('True Positive Rate (TPR)')  
plt.title('Receiver Operating Characteristic (ROC) Curve')  
plt.legend(loc='lower right')  
plt.show()  
  
print(f'AUC Score: {roc_auc}')
```



AUC Score: 0.6278981597630602

- Trained a logistic regression model on the training set and evaluated its performance on the validation set. Accuracy on validation: 91.31%.
- ROC AUC score: 0.6279. Indicates model's ability to distinguish classes. Score suggests moderate discrimination.
- Hypothesis test p-value: 0.0 (<0.05 significance). Null hypothesis rejected, significant relationship between features and target variable.
- Logistic regression model performs reasonably well on validation set, indicating significant feature-target relationship.
- Relatively low ROC AUC score suggests moderate discrimination ability (the model's ability to distinguish between positive and negative samples is moderate or not very strong), room for improvement.
- Summary: Model achieves 91.31% accuracy, features have significant relationship with target variable.
- ROC AUC score of 0.6279 indicates the logistic regression model's moderate ability to distinguish classes, and considering this, I would like to explore using the LightGBM model for potentially improved performance.

## LightGBM model

In [45]:

```

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
import lightgbm as lgb
from sklearn.metrics import roc_auc_score, accuracy_score

# Scale the input features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Splitting the data into training and validation sets
X_train, X_val, y_train, y_val = train_test_split(X_scaled, y, test_size=0.2, random_state=42)

# Creating a LightGBM dataset
train_data = lgb.Dataset(X_train, label=y_train)

# Set hyperparameters for LightGBM
params = {
    'objective': 'binary',
    'metric': 'auc',
    'boosting_type': 'gbdt',
    'num_leaves': 31,
    'learning_rate': 0.05,
    'feature_fraction': 0.9,
    'bagging_fraction': 0.8,
    'bagging_freq': 5,
    'verbose': 0
}

# Training the LightGBM model
model = lgb.train(params, train_data, num_boost_round=100)

# Evaluating the model on the validation set
y_pred = model.predict(X_val)
roc_auc = roc_auc_score(y_val, y_pred)
print("ROC AUC Score:", roc_auc)

# Converting probabilities to binary predictions
y_pred_binary = (y_pred > 0.5).astype(int)

# Calculating the accuracy
val_accuracy = accuracy_score(y_val, y_pred_binary)
print('Validation Accuracy:', val_accuracy)

# Scale the test data
X_test = scaler.transform(data_test_set.drop('ID_code', axis=1))

# Make predictions on the test set
predictions = model.predict(X_test)

# Save the predictions to a CSV file
submission = pd.DataFrame({'ID_code': data_test_set['ID_code'], 'target': predictions})
submission.to_csv('predictions.csv', index=False)

```

[LightGBM] [Warning] Auto-choosing col-wise multi-threading, the overhead of testing was 0.152311 seconds.

You can set `force\_col\_wise=true` to remove the overhead.

ROC AUC Score: 0.8487351312598571

Validation Accuracy: 0.89915

The LightGBM model achieves a remarkable ROC AUC score of 0.8487, indicating its strong ability to discriminate between classes. Moreover, with a validation accuracy of 0.8992, the model demonstrates its excellent performance in accurately predicting the target variable.

These results validate the effectiveness of the LightGBM model in achieving our main goal of accurate classification.