# shop-data-cleaning-and-analysis-2

October 4, 2024

[3]:
```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
af=pd.read_csv("/content/IOT-temp (1).csv")
af
```

[3]:
```
                                       id     room_id        noted_date  \
0      __export__.temp_log_196134_bd201015  Room Admin  08/12/2018 09:30
1      __export__.temp_log_196131_7bca51bc  Room Admin  08/12/2018 09:30
2      __export__.temp_log_196127_522915e3  Room Admin  08/12/2018 09:29
3      __export__.temp_log_196128_be0919cf  Room Admin  08/12/2018 09:29
4      __export__.temp_log_196126_d30b72fb  Room Admin  08/12/2018 09:29
…                                      …         …               …
97601   __export__.temp_log_91076_7fbd08ca  Room Admin  28/07/2018 07:07
97602  __export__.temp_log_147733_62c03f31  Room Admin  28/07/2018 07:07
97603  __export__.temp_log_100386_84093a68  Room Admin  28/07/2018 07:06
97604  __export__.temp_log_123297_4d8e690b  Room Admin  28/07/2018 07:06
97605  __export__.temp_log_133741_32958703  Room Admin  28/07/2018 07:06

       temp out/in
0      29.0     In
1      29.0     In
2      41.0    Out
3      41.0    Out
4      31.0     In
…       …      …
97601  31.0     In
97602  31.0     In
97603  31.0     In
97604  31.0     In
97605  31.0     In

[97606 rows x 5 columns]
```

[4]:
```
af.shape
```

[4]:
```
(97606, 5)
```

[5]:
```
af.rename(columns={'out/in':'status'},inplace=True)
af
```

[5]:

| | id | room_id | noted_date | temp | status |
|---|---|---|---|---|---|
| 0 | __export__.temp_log_196134_bd201015 | Room Admin | 08/12/2018 09:30 | 29.0 | In |
| 1 | __export__.temp_log_196131_7bca51bc | Room Admin | 08/12/2018 09:30 | 29.0 | In |
| 2 | __export__.temp_log_196127_522915e3 | Room Admin | 08/12/2018 09:29 | 41.0 | Out |
| 3 | __export__.temp_log_196128_be0919cf | Room Admin | 08/12/2018 09:29 | 41.0 | Out |
| 4 | __export__.temp_log_196126_d30b72fb | Room Admin | 08/12/2018 09:29 | 31.0 | In |
| … | … | … | … | … | … |
| 97601 | __export__.temp_log_91076_7fbd08ca | Room Admin | 28/07/2018 07:07 | 31.0 | In |
| 97602 | __export__.temp_log_147733_62c03f31 | Room Admin | 28/07/2018 07:07 | 31.0 | In |
| 97603 | __export__.temp_log_100386_84093a68 | Room Admin | 28/07/2018 07:06 | 31.0 | In |
| 97604 | __export__.temp_log_123297_4d8e690b | Room Admin | 28/07/2018 07:06 | 31.0 | In |
| 97605 | __export__.temp_log_133741_32958703 | Room Admin | 28/07/2018 07:06 | 31.0 | In |

[97606 rows x 5 columns]

[6]:
```
af.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 97606 entries, 0 to 97605
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   id          97606 non-null  object
 1   room_id     97606 non-null  object
 2   noted_date  97603 non-null  object
 3   temp        97601 non-null  float64
 4   status      97601 non-null  object
dtypes: float64(1), object(4)
memory usage: 3.7+ MB
```

[7]:
```
af.describe()
```

[7]:
```
                temp
count  97601.000000
mean      35.054948
std        5.719685
min        2.000000
25%       30.000000
50%       35.000000
75%       40.000000
max      135.000000
```

[9]:
```
af.isnull().sum()
```

[9]:
```
id            0
room_id       0
noted_date    3
temp          5
status        0
dtype: int64
```

[8]:
```
af.status.fillna(0,inplace=True)
```

```
<ipython-input-8-2ba7654b6ce6>:1: FutureWarning: A value is trying to be set on
a copy of a DataFrame or Series through chained assignment using an inplace
method.
The behavior will change in pandas 3.0. This inplace method will never work
because the intermediate object on which we are setting values always behaves as
a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using
'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value)
instead, to perform the operation inplace on the original object.


  af.status.fillna(0,inplace=True)
```

[10]:
```
af.columns
```

[10]:
```
Index(['id', 'room_id', 'noted_date', 'temp', 'status'], dtype='object')
```

[11]:
```
af.room_id.fillna(method="ffill",inplace=True)
```

```
<ipython-input-11-dbfb34e7e7ef>:1: FutureWarning: A value is trying to be set on
a copy of a DataFrame or Series through chained assignment using an inplace
method.
The behavior will change in pandas 3.0. This inplace method will never work
because the intermediate object on which we are setting values always behaves as
```

```
a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using
'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value)
instead, to perform the operation inplace on the original object.


  af.room_id.fillna(method="ffill",inplace=True)
<ipython-input-11-dbfb34e7e7ef>:1: FutureWarning: Series.fillna with 'method' is
deprecated and will raise in a future version. Use obj.ffill() or obj.bfill()
instead.
  af.room_id.fillna(method="ffill",inplace=True)
```

[12]:
```
af.noted_date.fillna(method="ffill",inplace=True)
```

```
<ipython-input-12-38386371f40e>:1: FutureWarning: A value is trying to be set on
a copy of a DataFrame or Series through chained assignment using an inplace
method.
The behavior will change in pandas 3.0. This inplace method will never work
because the intermediate object on which we are setting values always behaves as
a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using
'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value)
instead, to perform the operation inplace on the original object.


  af.noted_date.fillna(method="ffill",inplace=True)
<ipython-input-12-38386371f40e>:1: FutureWarning: Series.fillna with 'method' is
deprecated and will raise in a future version. Use obj.ffill() or obj.bfill()
instead.
  af.noted_date.fillna(method="ffill",inplace=True)
```

[13]:
```
af.isnull().sum()
```

[13]:
```
id           0
room_id      0
noted_date   0
temp         5
status       0
dtype: int64
```

[14]:
```
af
```

[14]:

| | id | room_id | noted_date | \ |
|---|---|---|---|---|
| 0 | __export__.temp_log_196134_bd201015 | Room Admin | 08/12/2018 09:30 | |
| 1 | __export__.temp_log_196131_7bca51bc | Room Admin | 08/12/2018 09:30 | |
| 2 | __export__.temp_log_196127_522915e3 | Room Admin | 08/12/2018 09:29 | |

```
3      __export__.temp_log_196128_be0919cf  Room Admin  08/12/2018 09:29
4      __export__.temp_log_196126_d30b72fb  Room Admin  08/12/2018 09:29
…                                      …         …              …
97601   __export__.temp_log_91076_7fbd08ca  Room Admin  28/07/2018 07:07
97602  __export__.temp_log_147733_62c03f31  Room Admin  28/07/2018 07:07
97603  __export__.temp_log_100386_84093a68  Room Admin  28/07/2018 07:06
97604  __export__.temp_log_123297_4d8e690b  Room Admin  28/07/2018 07:06
97605  __export__.temp_log_133741_32958703  Room Admin  28/07/2018 07:06

       temp status
0      29.0     In
1      29.0     In
2      41.0    Out
3      41.0    Out
4      31.0     In
…      …     …
97601  31.0     In
97602  31.0     In
97603  31.0     In
97604  31.0     In
97605  31.0     In

[97606 rows x 5 columns]
```

[15]:
```
af.status.unique()
```

[15]:
```
array(['In', 'Out', 0], dtype=object)
```

[16]:
```
af.head(10)
```

[16]:
```
                                    id     room_id        noted_date  temp  \
0  __export__.temp_log_196134_bd201015  Room Admin  08/12/2018 09:30  29.0
1  __export__.temp_log_196131_7bca51bc  Room Admin  08/12/2018 09:30  29.0
2  __export__.temp_log_196127_522915e3  Room Admin  08/12/2018 09:29  41.0
3  __export__.temp_log_196128_be0919cf  Room Admin  08/12/2018 09:29  41.0
4  __export__.temp_log_196126_d30b72fb  Room Admin  08/12/2018 09:29  31.0
5  __export__.temp_log_196125_b0fa0b41  Room Admin  08/12/2018 09:29  31.0
6  __export__.temp_log_196121_01544d45  Room Admin  08/12/2018 09:28  29.0
7  __export__.temp_log_196122_f8b80a9f  Room Admin  08/12/2018 09:28  29.0
8  __export__.temp_log_196111_6b7a0848  Room Admin  08/12/2018 09:26  29.0
9  __export__.temp_log_196112_e134aebd  Room Admin  08/12/2018 09:26  29.0

  status
0     In
1     In
2    Out
3    Out
```

```
4     In
5     In
6     In
7     In
8     In
9     In
```

[17]:
```
af.tail(10)
```

[17]:
```
                                        id     room_id        noted_date  \
97596  __export__.temp_log_111718_05b8d88d  Room Admin  28/07/2018 07:07
97597   __export__.temp_log_96677_28b0568f  Room Admin  28/07/2018 07:07
97598  __export__.temp_log_110269_898976fc  Room Admin  28/07/2018 07:07
97599  __export__.temp_log_124155_da310d7b  Room Admin  28/07/2018 07:07
97600  __export__.temp_log_121414_9004c748  Room Admin  28/07/2018 07:07
97601   __export__.temp_log_91076_7fbd08ca  Room Admin  28/07/2018 07:07
97602  __export__.temp_log_147733_62c03f31  Room Admin  28/07/2018 07:07
97603  __export__.temp_log_100386_84093a68  Room Admin  28/07/2018 07:06
97604  __export__.temp_log_123297_4d8e690b  Room Admin  28/07/2018 07:06
97605  __export__.temp_log_133741_32958703  Room Admin  28/07/2018 07:06

       temp status
97596  32.0    Out
97597  31.0     In
97598  31.0     In
97599  32.0    Out
97600  31.0     In
97601  31.0     In
97602  31.0     In
97603  31.0     In
97604  31.0     In
97605  31.0     In
```

[18]:
```
tp=af.temp.median()
tp
```

[18]:
```
35.0
```

[19]:
```
af.fillna(tp,inplace=True)
```

[20]:
```
af.nunique()
```

[20]:
```
id           97605
room_id          1
noted_date   27920
temp            37
status           3
```

```
dtype: int64
```

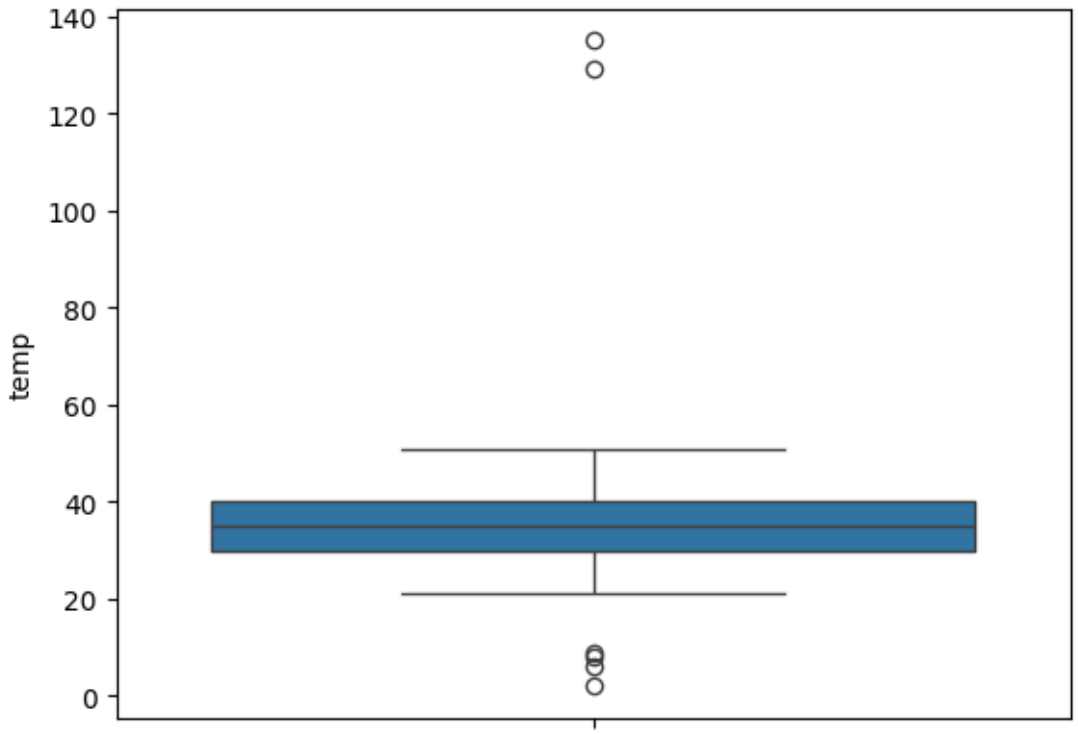
[21]:
```
af.dtypes
```

[21]:
```
id             object
room_id        object
noted_date     object
temp          float64
status         object
dtype: object
```

[22]:
```
sns.boxplot(data=af['temp'])
```

```
/usr/local/lib/python3.10/dist-packages/seaborn/categorical.py:640:
FutureWarning: SeriesGroupBy.grouper is deprecated and will be removed in a
future version of pandas.
  positions = grouped.grouper.result_index.to_numpy(dtype=float)
```

[22]:
```
<Axes: ylabel='temp'>
```




[ ]:

[23]:
```
q1=af.temp.quantile(0.25)
q3=af.temp.quantile(0.75)
iqr=q3-q1
lower_bound=q1-1.5*iqr
upper_bound=q3+1.5*iqr
print(lower_bound)
print(upper_bound)
```

```
15.0
55.0
```

[24]:
```
outliers=[x for x in af['temp'] if x<lower_bound or x>upper_bound]
outliers
```

[24]:
```
[8.0, 2.0, 129.0, 135.0, 6.0, 9.0]
```

[25]:
```
af=af[((af['temp']>=lower_bound) & (af['temp']<=upper_bound))]
af
```

[25]:

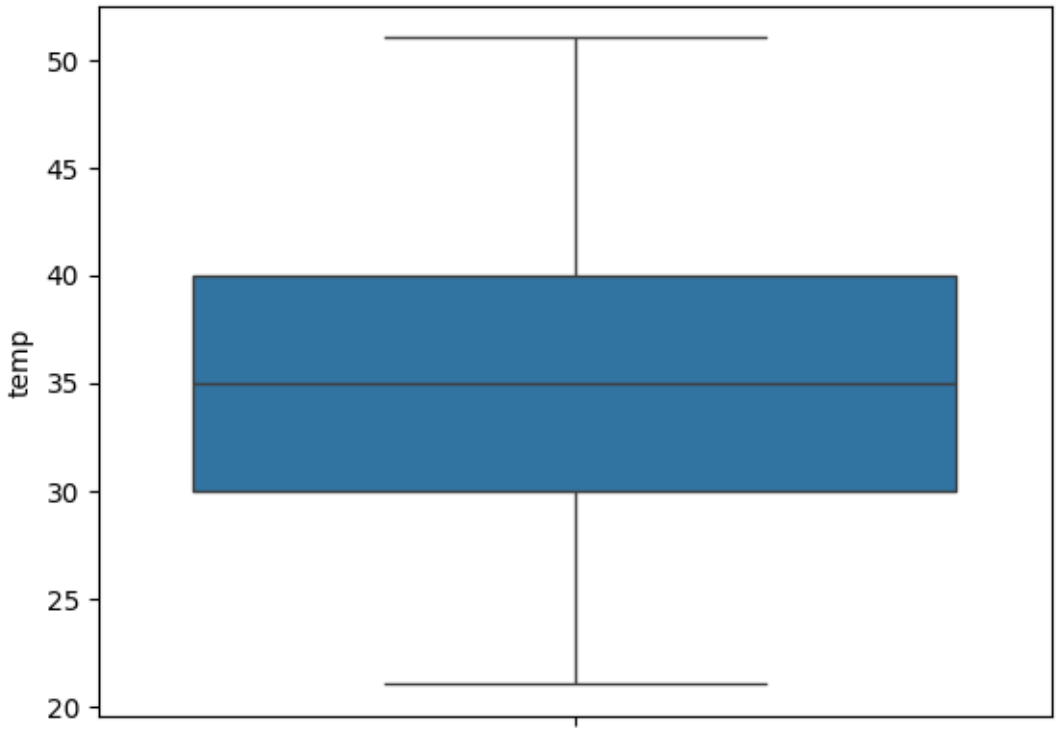
| | id | room_id | noted_date | temp | status |
|---|---|---|---|---|---|
| 0 | __export__.temp_log_196134_bd201015 | Room Admin | 08/12/2018 09:30 | 29.0 | In |
| 1 | __export__.temp_log_196131_7bca51bc | Room Admin | 08/12/2018 09:30 | 29.0 | In |
| 2 | __export__.temp_log_196127_522915e3 | Room Admin | 08/12/2018 09:29 | 41.0 | Out |
| 3 | __export__.temp_log_196128_be0919cf | Room Admin | 08/12/2018 09:29 | 41.0 | Out |
| 4 | __export__.temp_log_196126_d30b72fb | Room Admin | 08/12/2018 09:29 | 31.0 | In |
| … | … | … | … | … | … |
| 97601 | __export__.temp_log_91076_7fbd08ca | Room Admin | 28/07/2018 07:07 | 31.0 | In |
| 97602 | __export__.temp_log_147733_62c03f31 | Room Admin | 28/07/2018 07:07 | 31.0 | In |
| 97603 | __export__.temp_log_100386_84093a68 | Room Admin | 28/07/2018 07:06 | 31.0 | In |
| 97604 | __export__.temp_log_123297_4d8e690b | Room Admin | 28/07/2018 07:06 | 31.0 | In |
| 97605 | __export__.temp_log_133741_32958703 | Room Admin | 28/07/2018 07:06 | 31.0 | In |

[97600 rows x 5 columns]

[26]:
```
sns.boxplot(data=af['temp'])
```

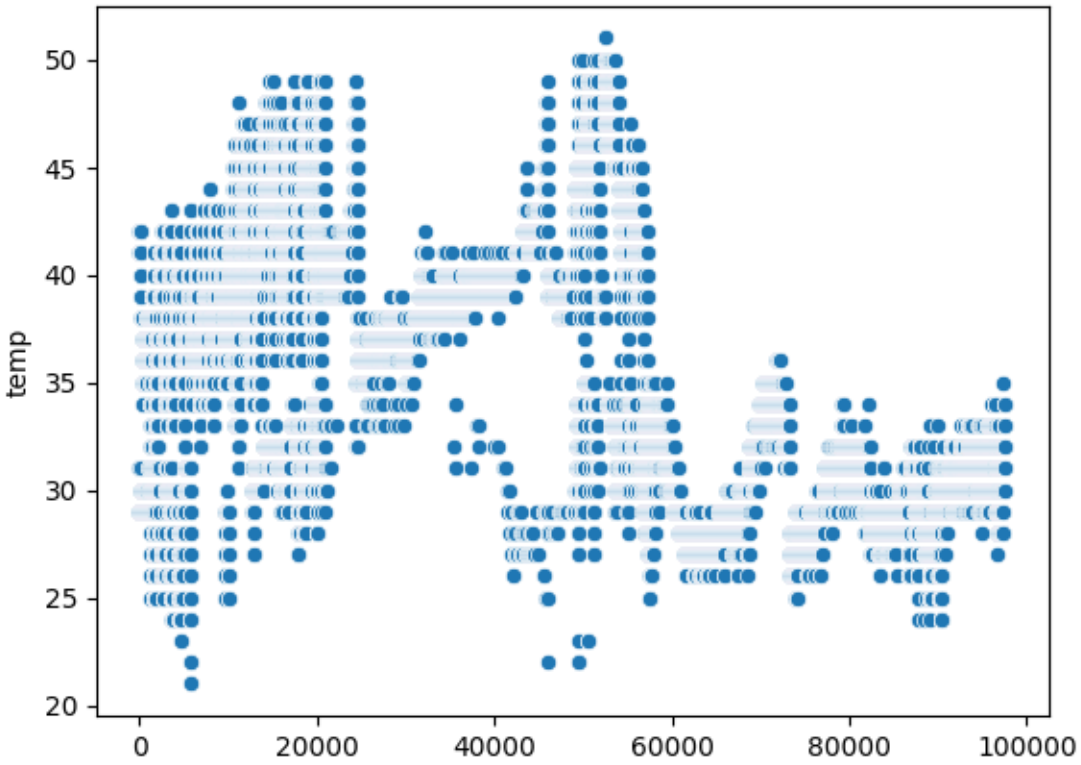
```
/usr/local/lib/python3.10/dist-packages/seaborn/categorical.py:640:
FutureWarning: SeriesGroupBy.grouper is deprecated and will be removed in a
future version of pandas.
  positions = grouped.grouper.result_index.to_numpy(dtype=float)
```

[26]: <Axes: ylabel='temp'>




[27]:
```
sns.scatterplot(data=af['temp'])
```

[27]: <Axes: ylabel='temp'>




[28]:
```
af['temp'].value_counts()
```

[28]:
```
temp
39.0    10203
28.0     8831
29.0     7918
40.0     7798
31.0     7236
30.0     6613
37.0     5721
32.0     5408
27.0     4631
41.0     4354
36.0     3963
38.0     3866
42.0     3447
33.0     3437
34.0     2613
43.0     2004
44.0     1774
35.0     1586
45.0     1508
```

```
46.0     1201
47.0     1044
48.0      971
26.0      699
49.0      401
25.0      224
24.0       66
50.0       55
22.0       19
23.0        5
21.0        2
51.0        2
Name: count, dtype: int64
```