oakland-crime-statistics

May 4, 2020

[1]:
```
import matplotlib as mpl
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
import pandas as pd
import os
```

1 Data preparation

[2]:
```
filepath = r'C:\Users\chd\Desktop\data\oakland-crime-statistics-2011-to-2016'
```

[3]:
```
data = pd.read_csv(os.path.join(filepath,'records-for-2016.csv'))
```

[4]:
```
data.head
```

[4]:
```
<bound method NDFrame.head of        Agency              Create Time
Location Area Id Beat  \
0           OP  2016-01-01T00:00:57.000         ST&MARKET ST          P1  05X
1           OP  2016-01-01T00:01:25.000       AV&HAMILTON ST          P3  26Y
2           OP  2016-01-01T00:01:43.000       ST&CHESTNUT ST          P1  02X
3           OP  2016-01-01T00:01:48.000           WALLACE ST          P2  18Y
4           OP  2016-01-01T00:02:05.000              90TH AV          P3  34X
…        …                        …                      …     …  …
110823      OP  2016-07-31T23:45:50.000  WHITMORE ST&WOOD ST          P1  02Y
110824      OP  2016-07-31T23:50:54.000      WHITTLE 69TH AV          P3  26Y
110825      OP  2016-07-31T23:56:29.000    WHITTLE LOOMIS CT          P2  19X
110826      OP  2016-07-31T23:57:31.000       WYMAN LACEY AV          P3  29X
110827     NaN                      NaN                              NaN  NaN

         Priority Incident Type Id Incident Type Description     Event Number  \
0             2.0             415GS              415 GUNSHOTS  LOP160101000003
1             2.0             415GS              415 GUNSHOTS  LOP160101000005
2             2.0             415GS              415 GUNSHOTS  LOP160101000008
3             2.0             415GS              415 GUNSHOTS  LOP160101000007
4             2.0             415GS              415 GUNSHOTS  LOP160101000009
…           …               …                        …              …
```

```
110823       2.0            415GS             415 GUNSHOTS  LOP160731000892
110824       2.0             415N      DISTURBANCE-NEIGHBOR  LOP160731000893
110825       2.0              912         SUSPICIOUS PERSON  LOP160731000895
110826       2.0              415                415 FAMILY  LOP160731000897
110827       NaN              NaN                       NaN              NaN

                    Closed Time
0       2016-01-01T00:32:30.000
1       2016-01-01T00:48:23.000
2       2016-01-01T00:21:24.000
3       2016-01-01T01:15:03.000
4       2016-01-01T00:54:52.000
…                           …
110823  2016-07-31T23:58:03.000
110824  2016-08-01T00:08:00.000
110825  2016-08-01T01:33:31.000
110826  2016-08-01T00:16:16.000
110827                      NaN

[110828 rows x 10 columns]>
```

[5]:
```
attribute = data.columns
print(attribute)
```

```
Index(['Agency', 'Create Time', 'Location', 'Area Id', 'Beat', 'Priority',
       'Incident Type Id', 'Incident Type Description', 'Event Number',
       'Closed Time'],
      dtype='object')
```

2 data summary

[6]:
```
nominal = [attribute[i] for i in [0,1,2,3,4,6,7,8,9]]
print('  :',nominal)
numeric = [attribute[i] for i in [5]]
print('  :',numeric)
```

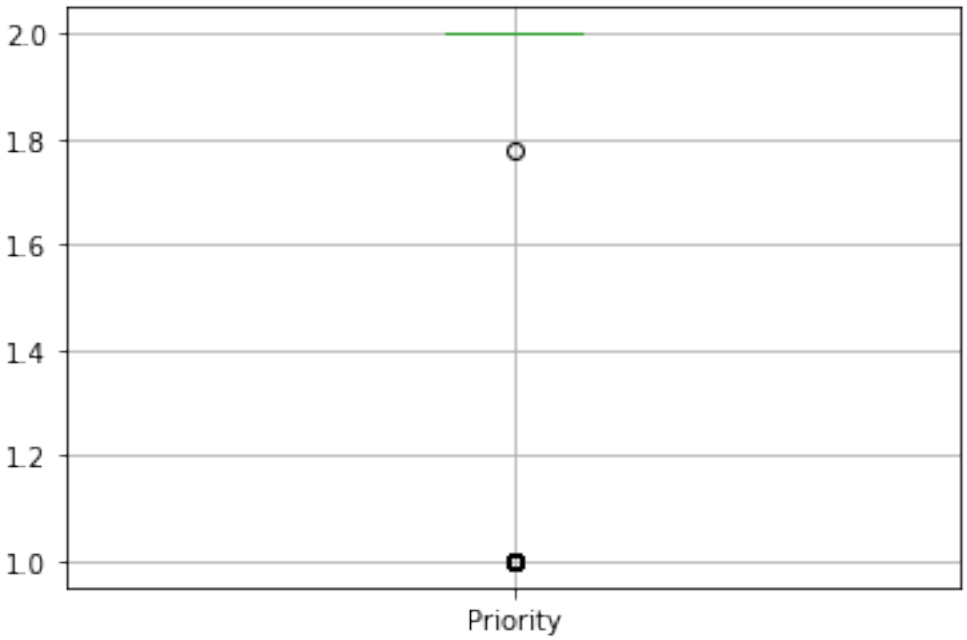
```
  : ['Agency', 'Create Time', 'Location', 'Area Id', 'Beat', 'Incident Type
Id', 'Incident Type Description', 'Event Number', 'Closed Time']
  : ['Priority']
```

[7]:
```
for a  in numeric:
    n = data[a].shape[0]-1
    split = [int(i*n) for i in [0,0.25,0.5,0.75,1]]
    data[a] = data[a].fillna(data[a].mean())
    num = [data[a].sort_values().iloc[i]  for i in split]
    print(a+'   :', num)
```

```
Priority    : [1.0, 2.0, 2.0, 2.0, 2.0]
```

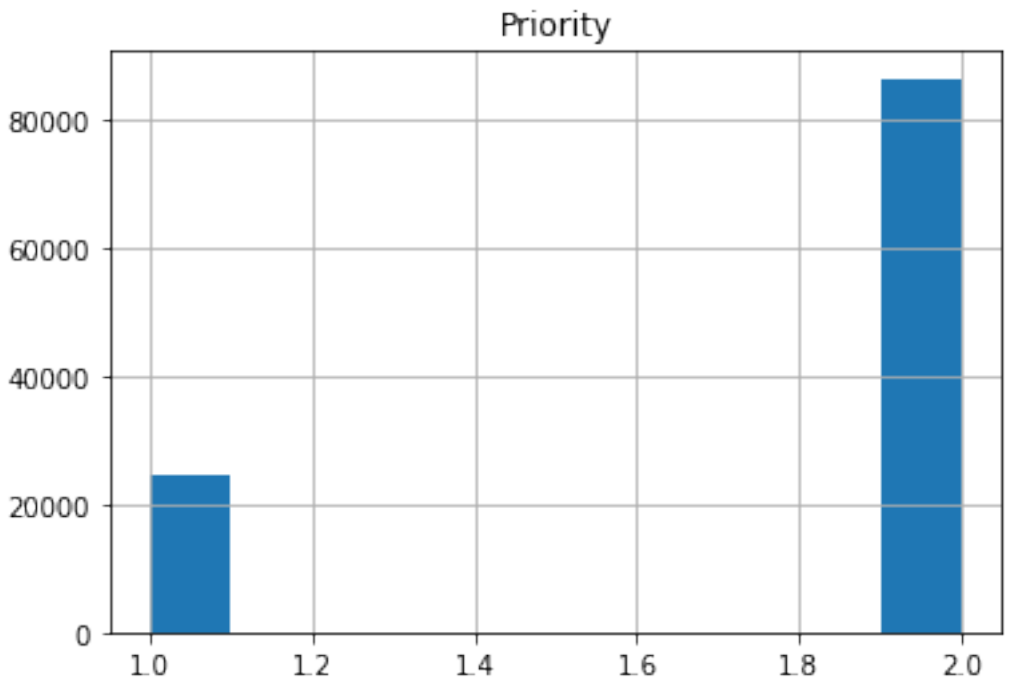
[9]:
```
data[numeric].boxplot()  #
```

[9]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x18085726308>
```



[10]:
```
data[numeric].hist() #
```

[10]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000001808571C548>]],
      dtype=object)
```



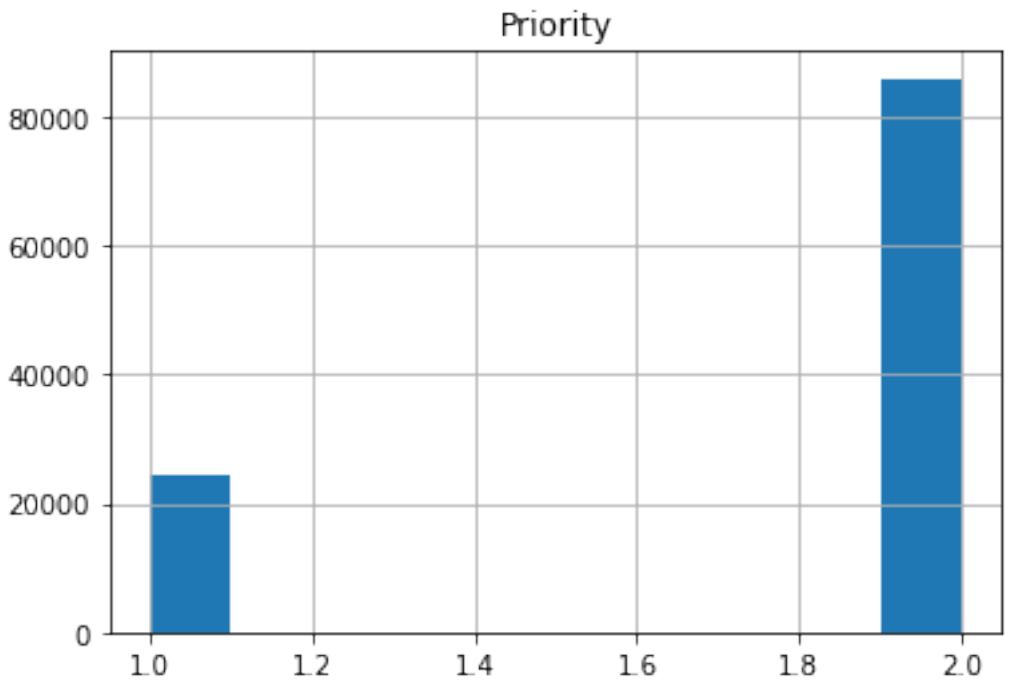
3 Incomplete (Missing) Data

1 Ignore the tuple

[12]:
```
data = pd.read_csv(os.path.join(filepath,'records-for-2016.csv'))
d1 = data.dropna()                          #
```

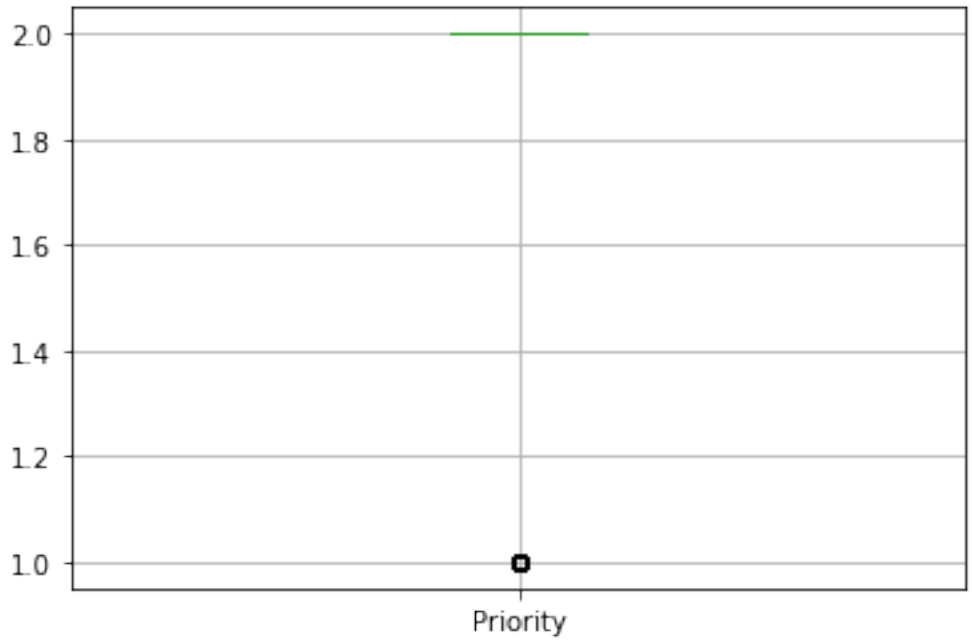
[13]:
```
d1[numeric].hist()
```

[13]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x0000018083E65C48>]],
      dtype=object)
```



[14]:
```
d1[numeric].boxplot()
```

[14]: <matplotlib.axes._subplots.AxesSubplot at 0x18082ac8508>

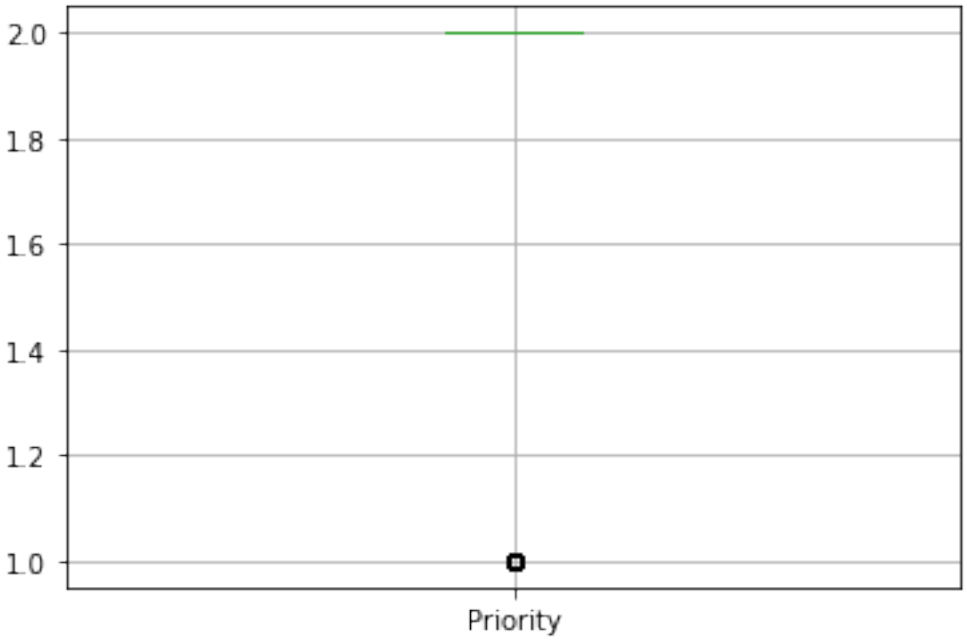


2 Replace with the most frequent data

[17]:
```
d2 = pd.read_csv(os.path.join(filepath,'records-for-2016.csv'))
for i in range(1,9):
    d2[attribute[i]] = d2[attribute[i]].fillna(value= d2[attribute[i]].
 ↪value_counts().index[0])  #
```

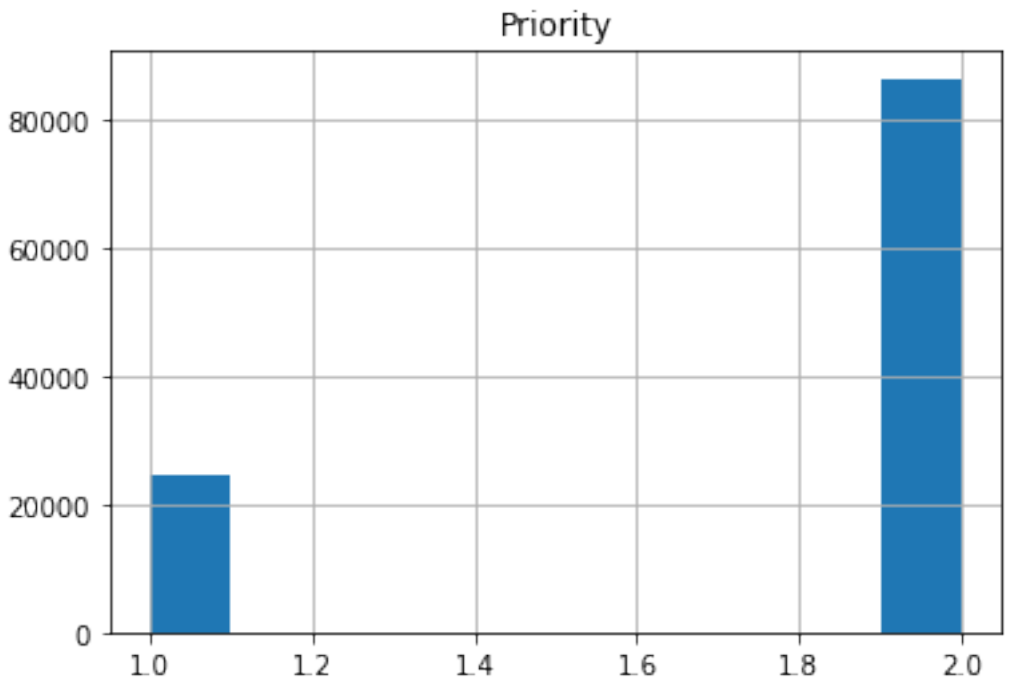
[18]:
```
d2[numeric].boxplot()
```

[18]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x18082b2bfc8>
```



[19]:
```
d2[numeric].hist()
```

[19]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000001808B87AF88>]],
      dtype=object)
```



3 Replace with related attribute

There are only two numeric attributes so this substitution does not exist

4 Replace with similar data

[18]:
```
d4 = pd.read_csv(os.path.join(filepath,'records-for-2016.csv'))    #
```

123

[19]:
```
def sim(x):
    maxsim = 0
    idx = -1
    for i in range(50):
        tmp = 0
        flag = 1
        for j in range(11):
            if x.iloc[j] == d4.iloc[i,j]:
                tmp+=1
        if tmp>maxsim:
            idx = d4.iloc[i]
            maxsim = tmp
        if maxsim>=3:
            break

    for i in range(11):
```

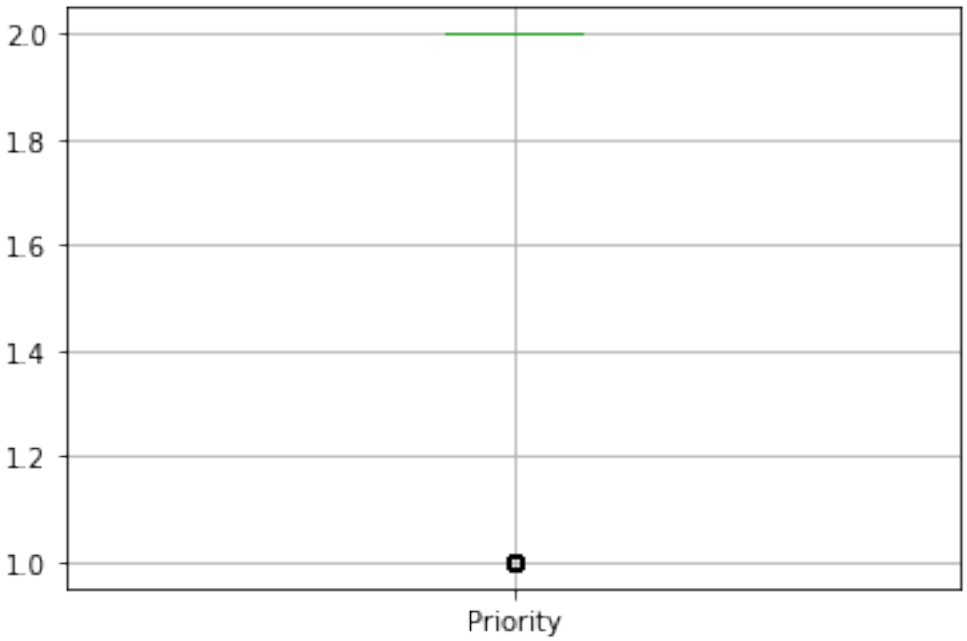
```
        if pd.isna(x.iloc[j]):
            x.iloc[i] = idx.iloc[i]
    return x
```

[28]:
```
for i in tqdm.tqdm(range(d4.shape[0])):
    x = d4.iloc[i]
    if x.isnull().any():
        simx = sim(x)
        d4.iloc[i] = simx
```

```
100%|
   | 10/10 [00:00<00:00, 435.50it/s]
```

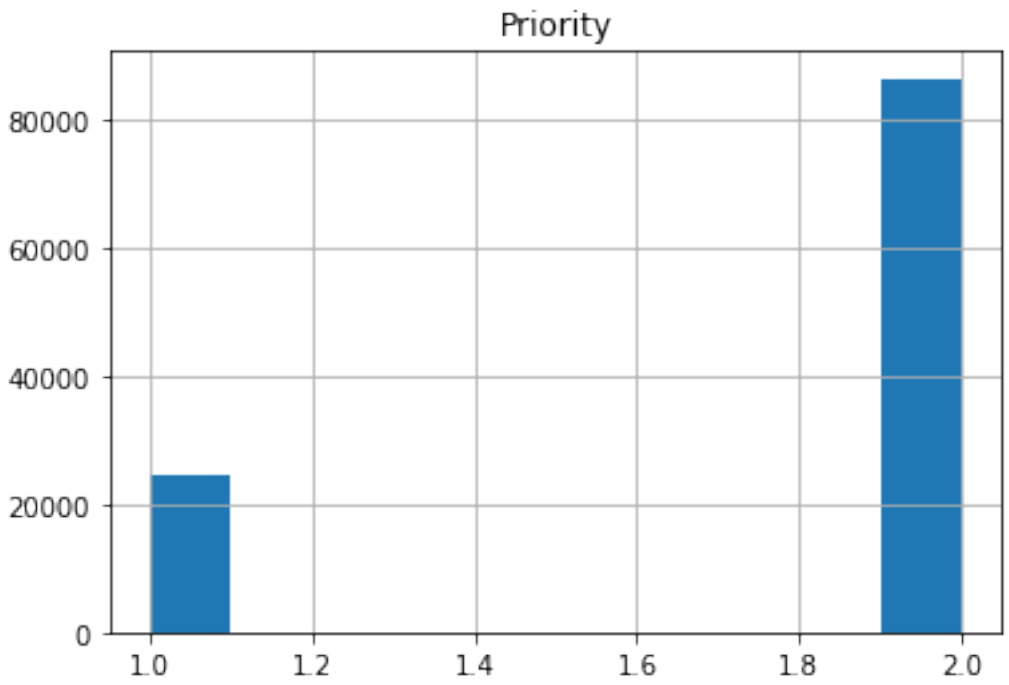
[20]:
```
d4[numeric].boxplot()
```

[20]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2bec37270c8>
```



[21]:
```
d4[numeric].hist()
```

[21]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000002BEBFBA7988>]],
      dtype=object)
```


Priority
80000
60000
40000
20000
0
1.0
1.2
1.4
1.6
1.8
2.0

[ ]:

[ ]: