Python: descriptive statistics II

Fourth tutorial session

Pie chart

Pie chart

Plot a pie-chart of rain season-wise for the year 1982 in Vidarbha region using the data SubDivisionWiseRainfall.csv

Picking specific data points

```
import pandas as pd
import matplotlib.pyplot as plt

Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')

x = Rainfall[(Rainfall['SUBDIVISION']=='VIDARBHA')
        & (Rainfall['YEAR']== ‘1982’)
        & (Rainfall['Parameter']=='Actual')]

print(x)
```

What went wrong?

```
(base) guru@BhaskarAngiras:~/.../Week4$ python3 PieChart1.py
Empty DataFrame
Columns: [SUBDIVISION, YEAR, Parameter, JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG,
SEP, OCT, NOV, DEC, ANNUAL, JF, MAM, JJAS, OND]
Index: []
```

```
import pandas as pd
import matplotlib.pyplot as plt
Rainfall =
pd.read_csv('SubDivisionWiseRainfall.csv')
print(Rainfall.dtypes)
```

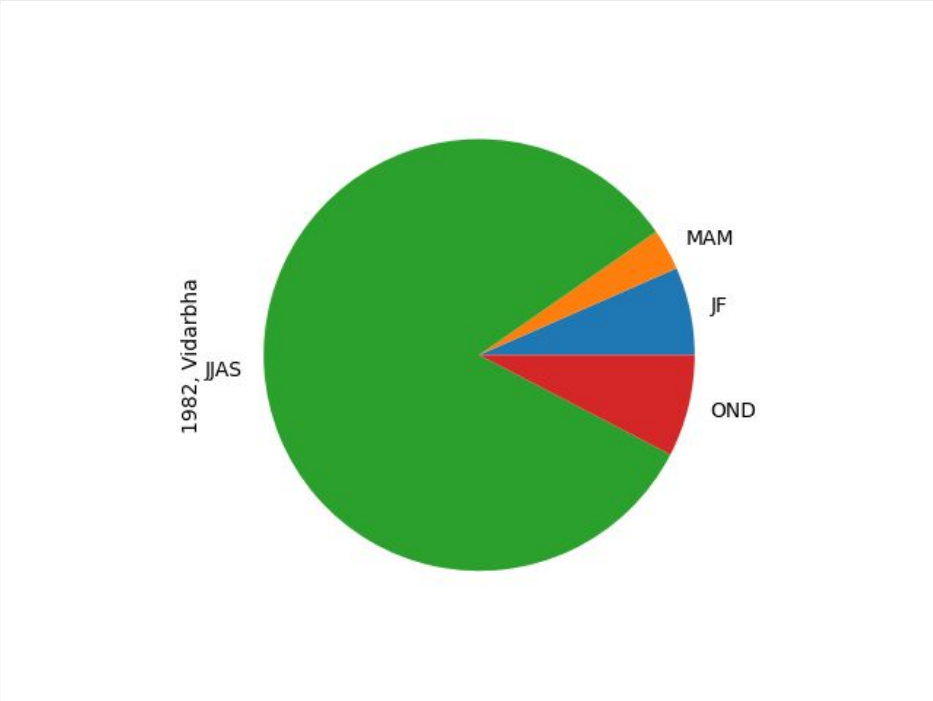
Wrong data type

```
(base) guru@BhaskarAngiras:~/.../Week4$ python3 PieChart1.py
SUBDIVISION     object
YEAR            object
Parameter       object
JAN            float64
FEB            float64
MAR            float64
APR            float64
MAY            float64
JUN            float64
JUL            float64
AUG            float64
SEP            float64
OCT            float64
NOV            float64
DEC            float64
ANNUAL         float64
JF             float64
MAM            float64
JJAS           float64
OND            float64
```

Piechart.py

```
import pandas as pd
import matplotlib.pyplot as plt
Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')
x = Rainfall[(Rainfall['SUBDIVISION']=='VIDARBHA')
       & (Rainfall['YEAR']== '1982')
       & (Rainfall['Parameter']=='Actual')]
print(x)
x[['JF','MAM','JJAS','OND']].iloc[0].plot(kind="pie",label="1982, Vidarbha")
plt.show()
```

Pie chart



Pie charts

Suppose we want to plot two pie-charts – of rain, season-wise, for the year 1982 in Vidarbha region and Marathwada using the data SubDivisionWiseRainfall.csv. How to do that?

Generate the required data

```
import pandas as pd
import matplotlib.pyplot as plt
Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')
x = Rainfall[(Rainfall['SUBDIVISION']=='VIDARBHA')  & (Rainfall['YEAR']== '1982')
& (Rainfall['Parameter']=='Actual')]
y = Rainfall[(Rainfall['SUBDIVISION']=='MARATHWADA') & (Rainfall['YEAR']==
'1982') & (Rainfall['Parameter']=='Actual')]
print(x)
print(y)
```

What went wrong?

```
(base) guru@BhaskarAngiras:~/.../Week4$ python3 PieChart4.py
      SUBDIVISION  YEAR Parameter   JAN   FEB  ...  ANNUAL    JF   MAM   JJAS
OND
11310    VIDARBHA  1982    Actual  45.7  10.6  ...   856.1  56.3  26.5  708.0  6
5.3

[1 rows x 20 columns]
Empty DataFrame
Columns: [SUBDIVISION, YEAR, Parameter, JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG,
SEP, OCT, NOV, DEC, ANNUAL, JF, MAM, JJAS, OND]
Index: []
```

Marathwada is spelt as Matathwada!!

```
import pandas as pd
import matplotlib.pyplot as plt
Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')
x = Rainfall[(Rainfall['SUBDIVISION']=='VIDARBHA')  &
(Rainfall['YEAR']== '1982') & (Rainfall['Parameter']=='Actual')]
y = Rainfall[(Rainfall['SUBDIVISION'].contains.=='MARATHWADA') &
(Rainfall['YEAR']== '1982') & (Rainfall['Parameter']=='Actual')]
print(x)
print(y)
```

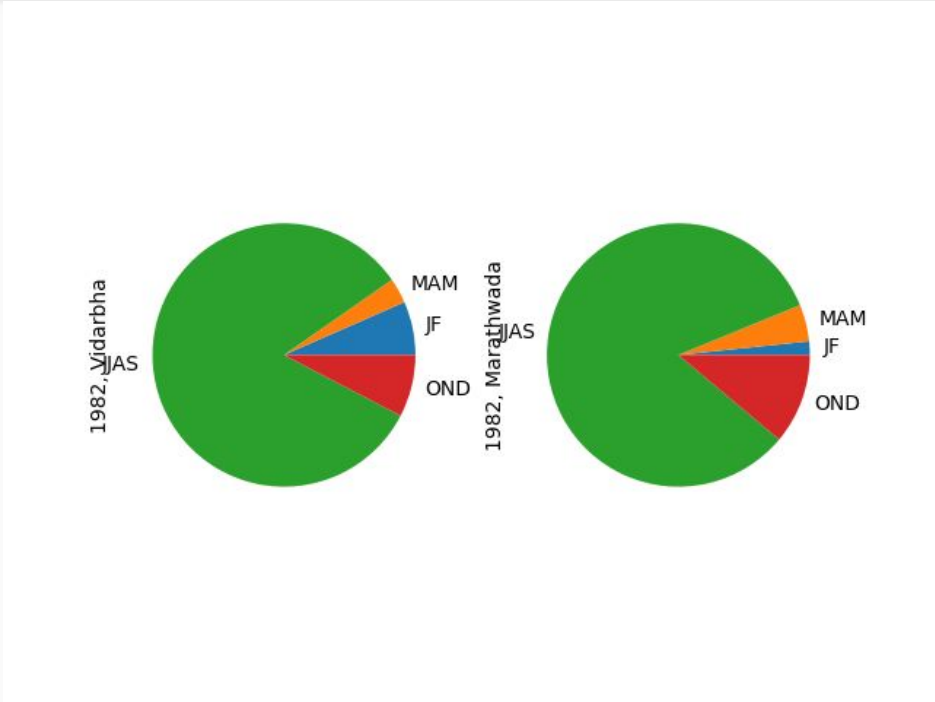
Subplots!

```
fig,axes = plt.subplots(nrows=1,ncols=2)

x[['JF','MAM','JJAS','OND']].iloc[0].plot(ax=axes[0],kind="pie",label="1982, Vidarbha")

y[['JF','MAM','JJAS','OND']].iloc[0].plot(ax=axes[1],kind="pie",label="1982, Marathwada")
```

Two pie-charts!



Bar plots and stacked bar plots

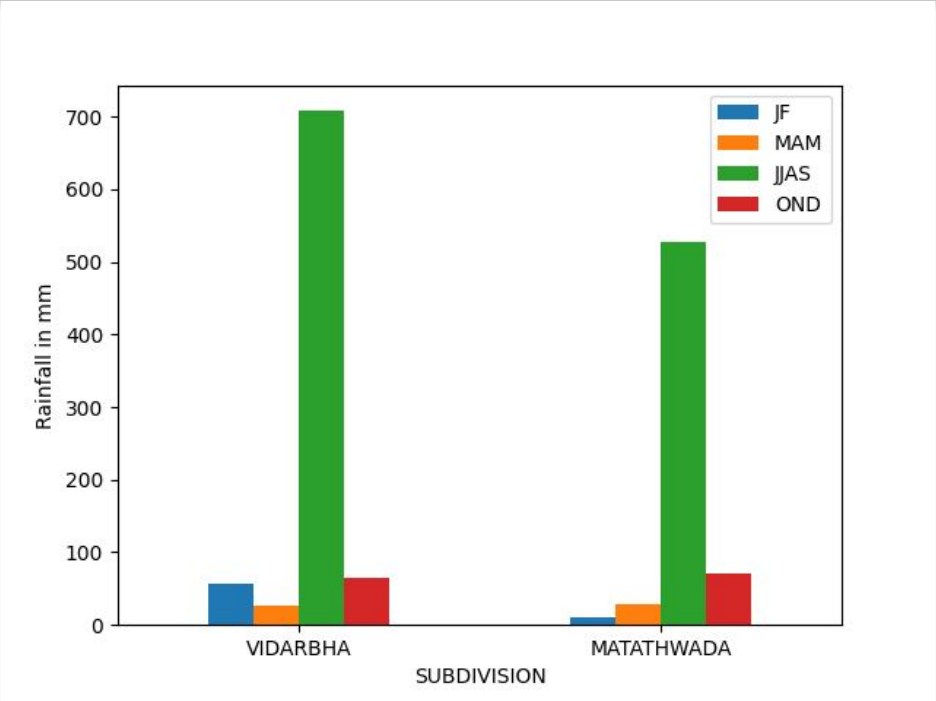
Concatenate data

```
import pandas as pd
import matplotlib.pyplot as plt
Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')
x = Rainfall[(Rainfall['SUBDIVISION']=='VIDARBHA')
        & (Rainfall['YEAR']== '1982')
        & (Rainfall['Parameter']=='Actual')]
y = Rainfall[(Rainfall['SUBDIVISION'].str.contains('WADA'))
        & (Rainfall['YEAR']== '1982')
        & (Rainfall['Parameter']=='Actual')]
z = pd.concat([x,y])
print(z)
```

Bar plot

```
z[['SUBDIVISION','JF','MAM','JJAS','OND']].plot(x='SUBDIVISI
ON', kind="bar")
plt.xticks(rotation=0)
plt.ylabel("Rainfall in mm")
plt.show()
```

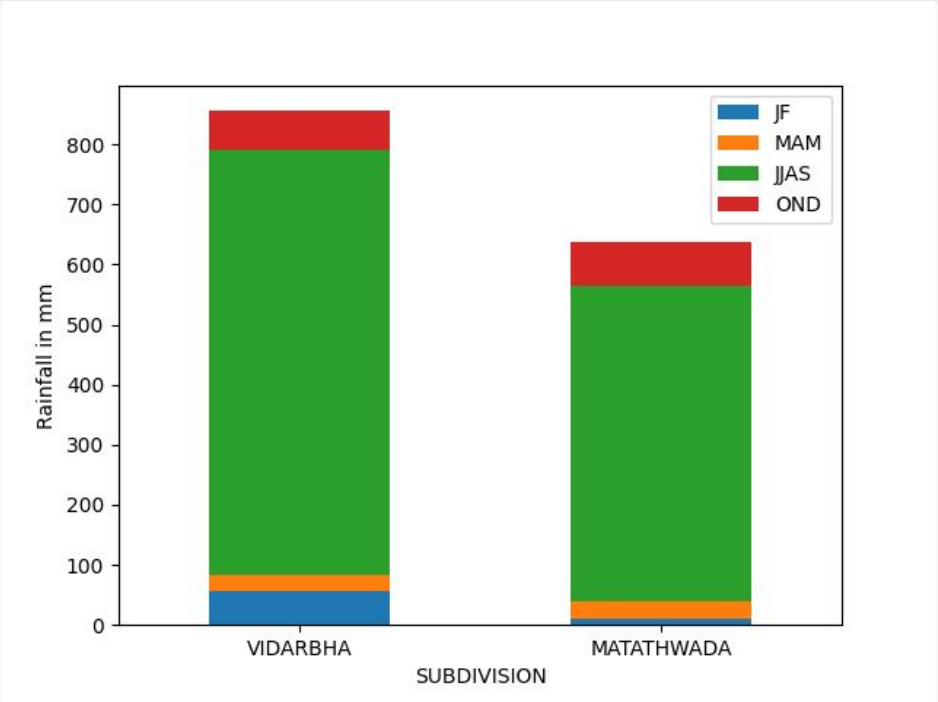
Bar plot


JF
MAM
JJAS
OND
Rainfall in mm
0
100
200
300
400
500
600
700
VIDARBHA
MATATHWADA
SUBDIVISION

Stacked bar plot

```
z[['SUBDIVISION','JF','MAM','JJAS','OND']].plot(x='SUBDIVISI
ON', kind="bar", stacked=True)
plt.xticks(rotation=0)
plt.ylabel("Rainfall in mm")
plt.show()
```

Stacked bar plot

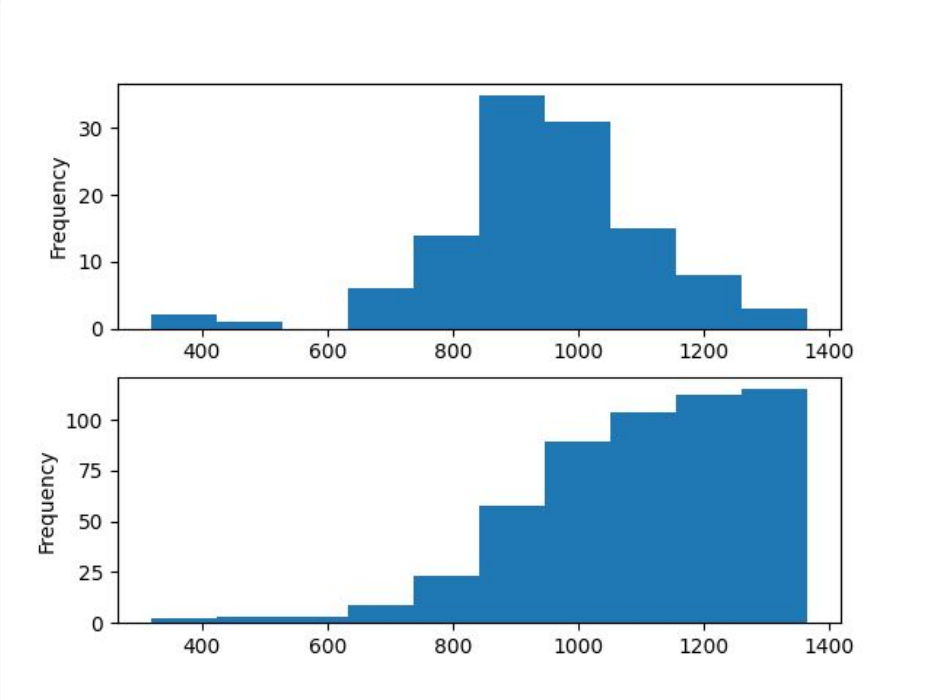


Ogives or cumulative histograms

Ogives and histograms

```
import pandas as pd
import matplotlib.pyplot as plt
Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')
x = Rainfall[(Rainfall['SUBDIVISION']=='TAMIL NADU') &
(Rainfall['Parameter']=='Actual')]
print(x)
fig,axes = plt.subplots(nrows=2,ncols=1)
x['ANNUAL'].plot(ax=axes[0],kind="hist")
x['ANNUAL'].plot(ax=axes[1],kind="hist",cumulative=True)
plt.show()
```

Histogram and ogive

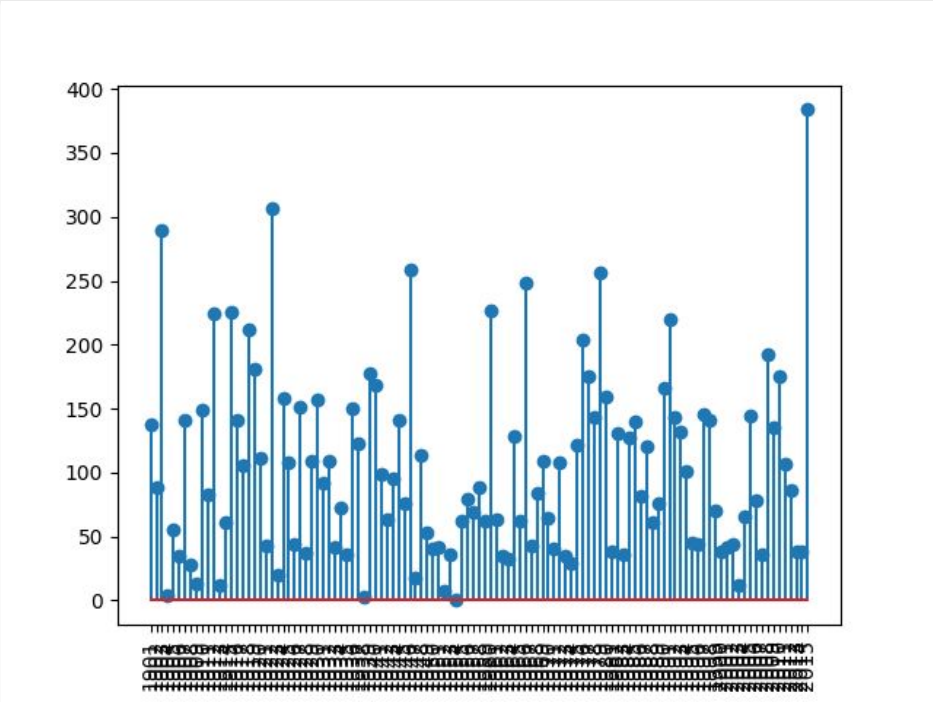


Stem-and-leaf plots

Stem-and-leaf plot

```
import pandas as pd
import matplotlib.pyplot as plt
Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')
x = Rainfall[(Rainfall['SUBDIVISION']=='RAYALSEEMA')
        & (Rainfall['Parameter']=='Actual')]
print(x)
plt.stem(x['YEAR'],x['NOV'])
plt.xticks(rotation=90)
plt.show()
```

Stem-and-leaf plot

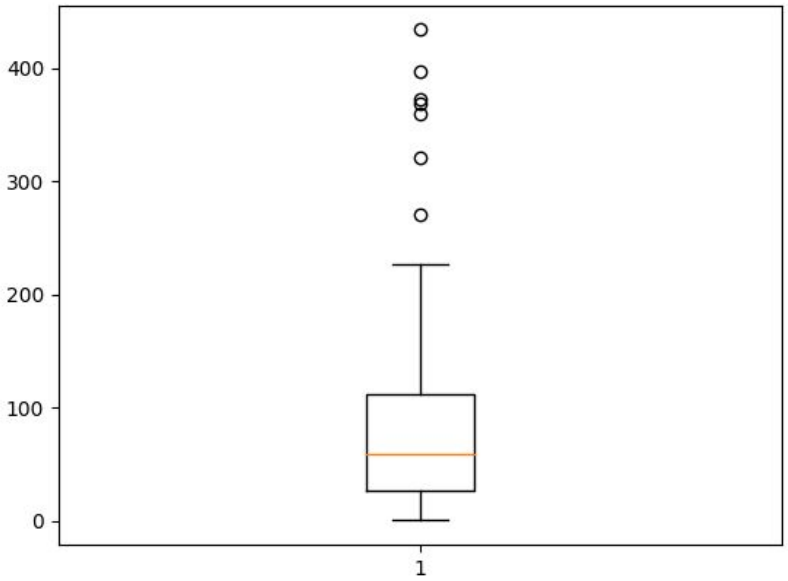


Box plots

Box plot

```
import pandas as pd
import matplotlib.pyplot as plt
Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')
x = Rainfall[(Rainfall['SUBDIVISION']=='PUNJAB')
        & (Rainfall['Parameter']=='Actual')]
print(x)
plt.boxplot(x['SEP'],vert=0,notch=True)
plt.show()
```

Box plot



How to read?

- Median
- Interquartile range (Q3-Q1)
- 1.5 times interquartile range: whiskers
- Outliers

Python: numerical description

Numpy and pandas: statistics commands

Numpy: mean(), median(), ptp() (for calculating range), percentile(), quantile(), std(), var() …

Pandas: pandas.dataframe.describe()

```
import pandas as pd
Rainfall = pd.read_csv('SubDivisionWiseRainfall.csv')
x = Rainfall[(Rainfall['SUBDIVISION']=='BIHAR') & (Rainfall['Parameter']=='Actual')]
print(x['MAR'].describe())
```

Pandas: descriptive statistics

```
(base) guru@BhaskarAngiras:~/.../Week4$ python3 DescriptiveStat.py
count    115.000000
mean      10.124348
std       11.695340
min        0.000000
25%        1.800000
50%        6.500000
75%       12.850000
max       65.500000
Name: MAR, dtype: float64
```

Thank you!!

ALL THE BEST!