

# Heamnath

20104028

## Basic Analysis using Numpy and Pandas

### Importing libraries

In [1]:

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

### importing datasets

In [2]:

```
df=pd.read_csv("5_Instagram data.csv")
df
```

Out[2]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
0	3920	2586	1028	619	56	98	9	5	162	35	2
1	5394	2727	1838	1174	78	194	7	14	224	48	10
2	4021	2085	1188	0	533	41	11	1	131	62	12
3	4528	2700	621	932	73	172	10	7	213	23	8
4	2518	1704	255	279	37	96	5	4	123	8	0

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
...	...	...	...	...	...	...	...	...	...	...	...
<b>114</b>	13700	5185	3041	5352	77	573	2	38	373	73	80
...	...	...	...	...	...	...	...	...	...	...	...
<b>115</b>	5731	1923	1368	2266	65	135	4	1	148	20	18
...	...	...	...	...	...	...	...	...	...	...	...
<b>116</b>	4139	1133	1538	1367	33	36	0	1	92	34	10
...	...	...	...	...	...	...	...	...	...	...	...
<b>117</b>	32695	11815	3147	17414	170	1095	2	75	549	148	214
...	...	...	...	...	...	...	...	...	...	...	...
<b>118</b>	36919	13473	4176	16444	2547	653	5	26	443	611	228
...	...	...	...	...	...	...	...	...	...	...	...

119 rows × 13 columns

## To display first 10 rows

In [3]:

```
df.head(10)
```

Out[3]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
<b>0</b>	3920	2586	1028	619	56	98	9	5	162	35	2
i dæ	5394	2727	1838	1174	78	194	7	14	224	48	10

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
2	4021	2085	1188	0	533	41	11	1	131	62	12
3	4528	2700	621	932	73	172	10	7	213	23	8
4	2518	1704	255	279	37	96	5	4	123	8	0
5	3884	2046	1214	329	43	74	7	10	144	9	2
6	2621	1543	599	333	25	22	5	1	76	26	0
7	3541	2071	628	500	60	135	4	9	124	12	6
8	3749	2384	857	248	49	155	6	8	159	36	4
9	4115	2609	1104	178	46	122	6	3	191	31	6

To display last 5 rows

In [4]:

```
df.tail(5)
```

Out[4]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
114	13700	5185	3041	5352	77	573	2	38	373	73	80
115	5731	1923	1368	2266	65	135	4	1	148	20	18
116	4139	1133	1538	1367	33	36	0	1	92	34	10
117	32695	11815	3147	17414	170	1095	2	75	549	148	214
118	36919	13473	4176	16444	2547	653	5	26	443	611	228



## Statistical Summary

In [5]:

```
df.mean()
```

Out[5]:

Impressions	5703.991597
From Home	2475.789916
From Hashtags	1887.512605
From Explore	1078.100840
From Other	171.092437
Saves	153.310924
Comments	6.663866
Shares	9.361345
Likes	173.781513
Profile Visits	50.621849

```
Follows          20.756303
dtype: float64
```

In [6]: `df.median()`

```
Out[6]: Impressions      4289.0
From Home        2207.0
From Hashtags    1278.0
From Explore     326.0
From Other       74.0
Saves            109.0
Comments         6.0
Shares           6.0
Likes            151.0
Profile Visits   23.0
Follows          8.0
dtype: float64
```

In [7]: `df.mode()`

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
<b>0</b>	5394.0	1975.0	116	45.0	34.0	40.0	6.0	3.0	114.0	19.0	2.0
<b>1</b>	NaN	NaN	201	84.0	NaN	135.0	NaN	NaN	151.0	21.0	NaN
<b>2</b>	NaN	NaN	278	NaN	NaN	144.0	NaN	NaN	NaN	NaN	NaN
<b>3</b>	NaN	NaN	362	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>4</b>	NaN	NaN	411	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>5</b>	NaN	NaN	583	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>6</b>	NaN	NaN	655	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>7</b>	NaN	NaN	707	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>8</b>	NaN	NaN	771	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>9</b>	NaN	NaN	794	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>10</b>	NaN	NaN	1248	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>11</b>	NaN	NaN	1260	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>12</b>	NaN	NaN	1278	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>13</b>	NaN	NaN	1693	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
<b>14</b>	NaN	NaN	1938	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows	
15	NaN	NaN	2351	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
16	NaN	NaN	2975	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
17	NaN	NaN	3450	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
18	NaN	NaN	3551	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [8]: `df.sum()`

Out[8]:

Impressions	678775
From Home	294619
From Hashtags	224614
From Explore	128294
From Other	20360
Saves	18244
Comments	793
Shares	1114
Likes	20680
Profile Visits	6024
Follows	2470
Caption	Here are some of the most important data visua...
Hashtags	#finance◆#money◆#business◆#investing◆#investme...
dtype: object	

In [9]: `df.cumsum()`

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
0	3920	2586	1028	619	56	98	9	5	162	35	2
1	9314	5313	2866	1793	134	292	16	19	386	83	12
2	13335	7398	4054	1793	667	333	27	20	517	145	24
3	17863	10098	4675	2725	740	505	37	27	730	168	32

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
4	20381	11802	4930	3004	777	601	42	31	853	176	32
...	...	...	...	...	...	...	...	...	...	...	...
114	599291	266275	214385	90803	17545	16325	782	1011	19448	5211	2000
115	605022	268198	215753	93069	17610	16460	786	1012	19596	5231	2018
116	609161	269331	217291	94436	17643	16496	786	1013	19688	5265	2028
117	641856	281146	220438	111850	17813	17591	788	1088	20237	5413	2242
118	678775	294619	224614	128294	20360	18244	793	1114	20680	6024	2470

119 rows × 13 columns

In [10]:

df.count()

Out[10]:

Impressions	119
From Home	119
From Hashtags	119
From Explore	119
From Other	119
Saves	119

```
Comments      119
Shares        119
Likes         119
Profile Visits 119
Follows        119
Caption        119
Hashtags       119
dtype: int64
```

In [11]: `df.min()`

```
Out[11]: Impressions          1941
From Home           1133
From Hashtags       116
From Explore        0
From Other          9
Saves                22
Comments             0
Shares                0
Likes                 72
Profile Visits       4
Follows               0
Caption      170 Python Projects with Source Code solved an...
Hashtags      #career#job#jobs#jobsearch#education#busi...
dtype: object
```

In [12]: `df.max()`

```
Out[12]: Impressions          36919
From Home           13473
From Hashtags       11817
From Explore        17414
From Other          2547
Saves                1095
Comments              19
Shares                75
Likes                 549
Profile Visits       611
Follows               260
Caption      You must have seen the news divided into categ...
Hashtags      #timeseries#time#statistics#datascience#bi...
dtype: object
```

In [13]: `from numpy import cov`

In [14]: `cov(df['From Home'], df['From Hashtags'])`

```
Out[14]: array([[2218271.69277881,  498205.17639937],
   [ 498205.17639937, 3550818.04856858]])
```

In [15]: `from scipy.stats import pearsonr`  
`pearsonr(df['From Home'], df['From Hashtags'])`

```
Out[15]: (0.17751565433098784, 0.053434143091160374)
```

In [16]: `from scipy.stats import spearmanr`  
`spearmanr(df['From Home'], df['From Hashtags'])`

In [16]: SpearmanResult(correlation=0.11752786942921449, pvalue=0.203031655807403)

In [17]: df.describe()

Out[17]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Profile Visits	Follows
count	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000
mean	5703.991597	2475.789916	1887.512605	1078.100840	171.092437	153.310924	6.663866	1.0	1.0
std	4843.780105	1489.386348	1884.361443	2613.026132	289.431031	156.317731	3.544576	1.0	1.0
min	1941.000000	1133.000000	116.000000	0.000000	9.000000	22.000000	0.000000	1.0	1.0
25%	3467.000000	1945.000000	726.000000	157.500000	38.000000	65.000000	4.000000	1.0	1.0
50%	4289.000000	2207.000000	1278.000000	326.000000	74.000000	109.000000	6.000000	1.0	1.0
75%	6138.000000	2602.500000	2363.500000	689.500000	196.000000	169.000000	8.000000	1.0	1.0
max	36919.000000	13473.000000	11817.000000	17414.000000	2547.000000	1095.000000	19.000000	1.0	1.0

## To print no of rows and columns

In [18]: df.shape

Out[18]: (119, 13)

## To print total no of elements

In [19]: df.size

Out[19]: 1547

## To find the null value

In [20]: df.isna()

Out[20]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
0	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
...	...	...	...	...	...	...	...	...	...	...	...
114	False	False	False	False	False	False	False	False	False	False	False
115	False	False	False	False	False	False	False	False	False	False	False
116	False	False	False	False	False	False	False	False	False	False	False
117	False	False	False	False	False	False	False	False	False	False	False
118	False	False	False	False	False	False	False	False	False	False	False

119 rows × 13 columns

## To fill the missing value

In [21]:

```
df.fillna(value=0)
```

Out[21]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
0	3920	2586	1028	619	56	98	9	5	162	35	2
1	5394	2727	1838	1174	78	194	7	14	224	48	10
2	4021	2085	1188	0	533	41	11	1	131	62	12
3	4528	2700	621	932	73	172	10	7	213	23	8
4	2518	1704	255	279	37	96	5	4	123	8	0
...	...	...	...	...	...	...	...	...	...	...	...

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits	Follows
114	13700	5185	3041	5352	77	573	2	38	373	73	80
115	5731	1923	1368	2266	65	135	4	1	148	20	18
116	4139	1133	1538	1367	33	36	0	1	92	34	10
117	32695	11815	3147	17414	170	1095	2	75	549	148	214
118	36919	13473	4176	16444	2547	653	5	26	443	611	228

119 rows × 13 columns

## Print column names

In [22]:

```
df.columns
```

```
Out[22]: Index(['Impressions', 'From Home', 'From Hashtags', 'From Explore',
       'From Other', 'Saves', 'Comments', 'Shares', 'Likes', 'Profile Visits',
       'Follows', 'Caption', 'Hashtags'],
      dtype='object')
```

## To print particular column names

In [23]:

```
data=df[['From Home', 'From Hashtags']]  
data
```

Out[23]:

	From Home	From Hashtags
--	-----------	---------------

0	2586	1028
---	------	------

1 From Home From Hashtags

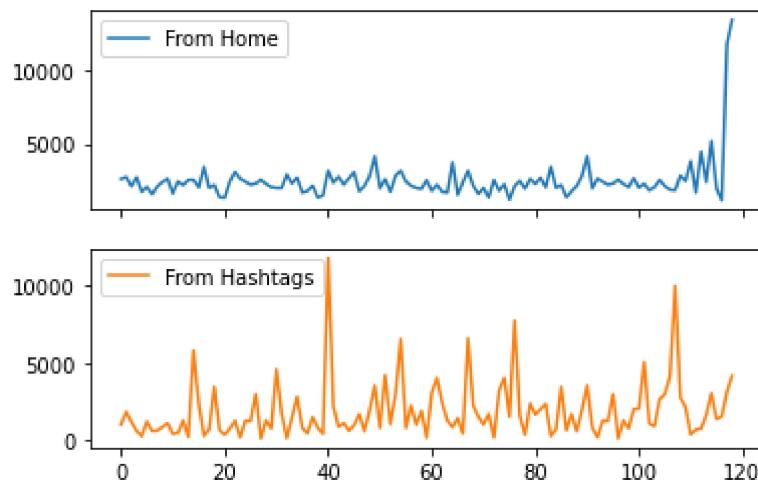
1	From Home	From Hashtags
2	2085	1188
3	2700	621
4	1704	255
...	...	...
114	5185	3041
115	1923	1368
116	1133	1538
117	11815	3147
118	13473	4176

119 rows × 2 columns

## Line chart with subplots

In [24]: `data.plot.line(subplots=True)`

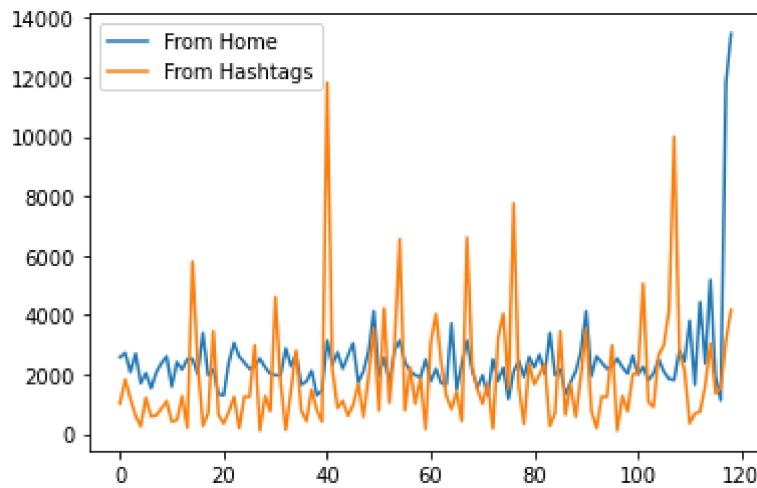
Out[24]: `array([<AxesSubplot: >, <AxesSubplot: >], dtype=object)`



## Line chart

In [25]: `data.plot.line()`

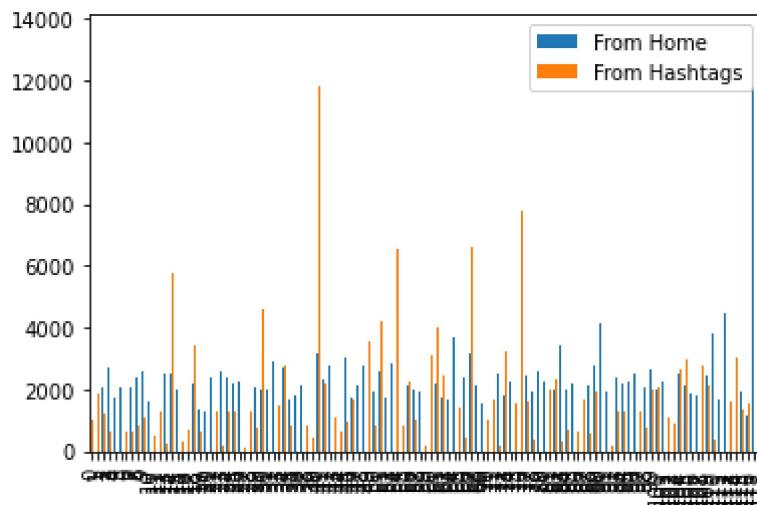
Out[25]: `<AxesSubplot: >`



## Bar chart

In [26]: `data.plot.bar()`

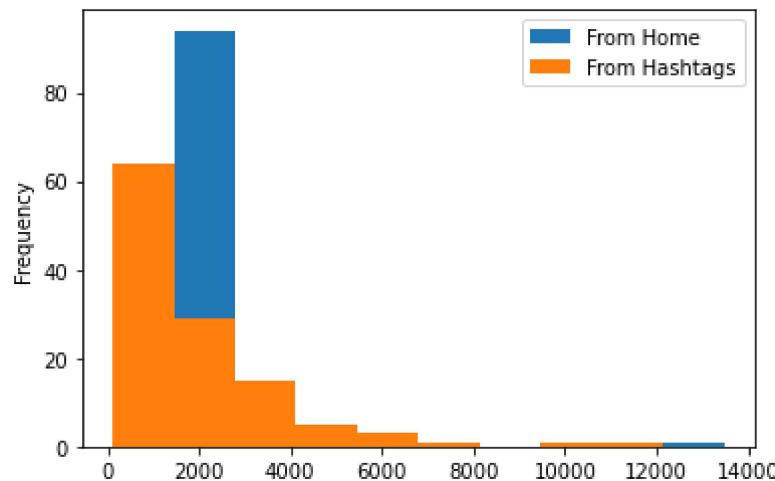
Out[26]: <AxesSubplot:>



## Histogram

In [27]: `data.plot.hist()`

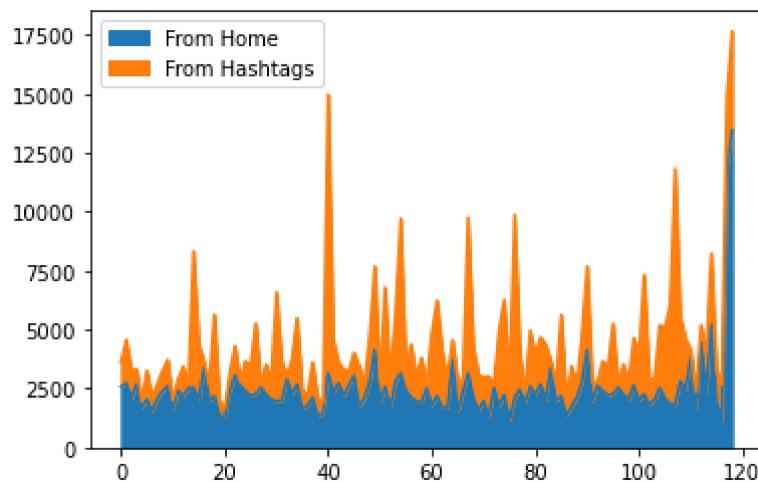
Out[27]: <AxesSubplot:ylabel='Frequency'>



## Area chart

In [28]: `data.plot.area()`

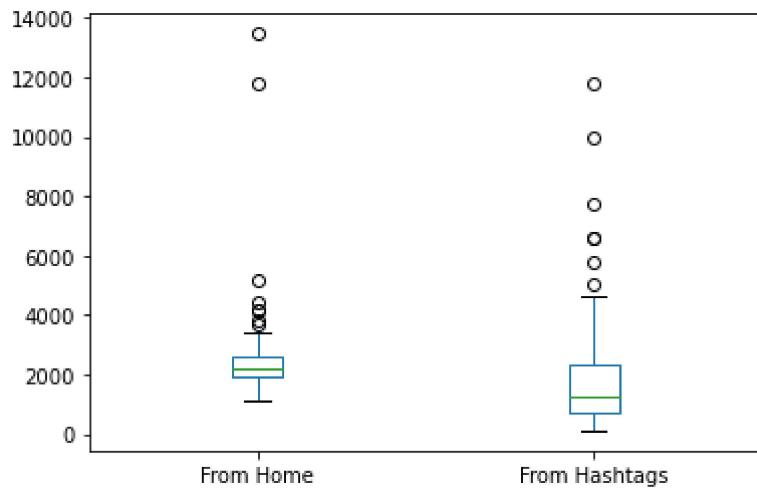
Out[28]: <AxesSubplot:>



## Box chart

In [29]: `data.plot.box()`

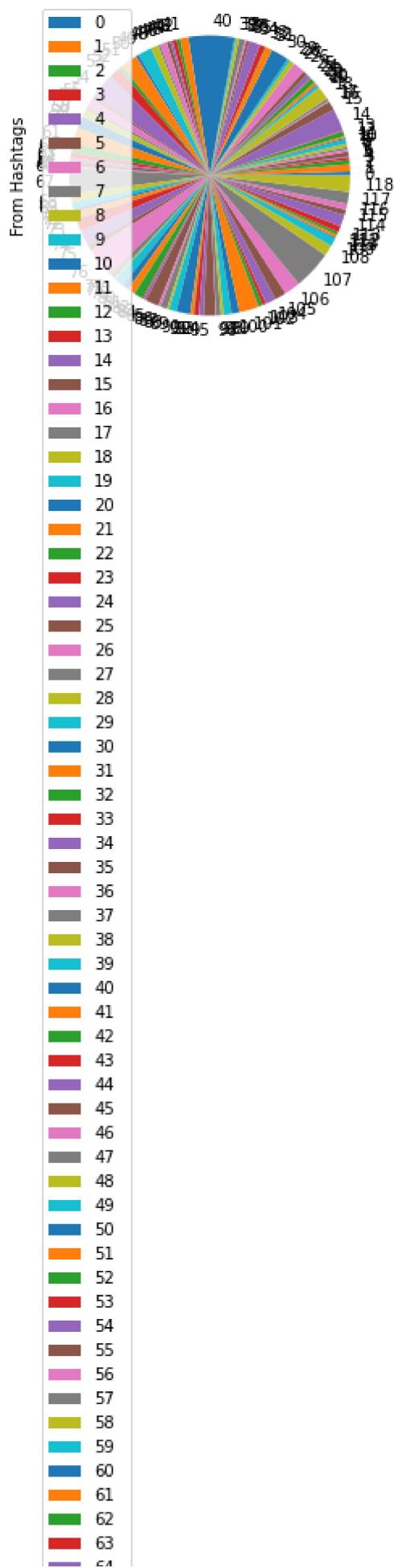
Out[29]: <AxesSubplot:>

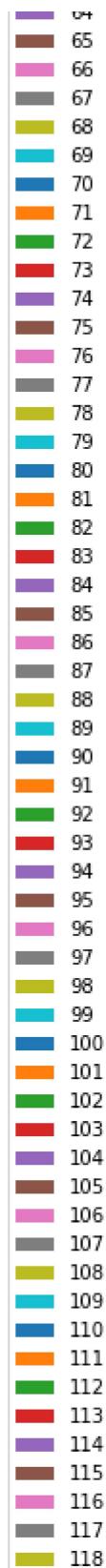


## Pie chart

In [30]: `data.plot.pie(y='From Hashtags')`

Out[30]: <AxesSubplot:ylabel='From Hashtags'>





## Scatter chart

```
In [31]: data.plot.scatter(x='From Home',y='From Hashtags')
```

Out[31]: &lt;AxesSubplot:xlabel='From Home', ylabel='From Hashtags'&gt;

