data_VIsualIzatIon_Using_Plots

June 12, 2021

1 Data VIsualIzatIon Using Plots

[25]:
```
# Importance of data visualization
## - Converting the business data into interactive graphs for dynamic␣
 ↪interpretation to serve the business goals
## - Transforming data into visually appealing, interactive dashboards of␣
 ↪various data sources to serve the business with the insights
## - Creating more attractive and informative dashboards of various graphical␣
 ↪data representations
## - Making appropriate decisions by drilling into the data and finding the␣
 ↪insights
## - Figuring out the patterns, trends, and correlations in the data being␣
 ↪analyzed to determine where they must improve their operational processes␣
 ↪and thereby grow their business
## - Giving a fuller picture of the data under analysis
## - Organizing and presenting massive data intuitively to
##   present important findings from the data
## - Making better, quick, and informed decisions with data visualization
```

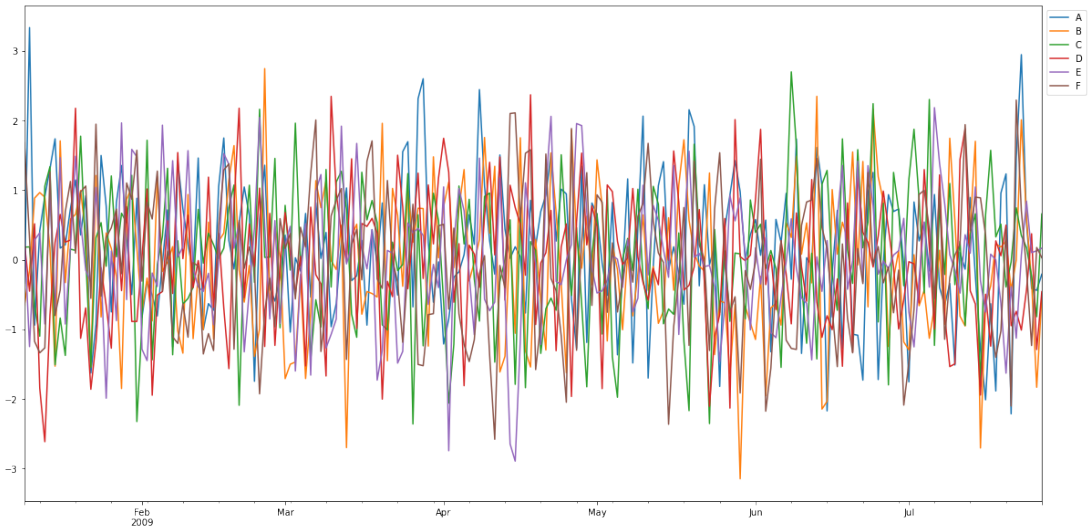
[26]:
```
# Why Do Modern Businesses Need Data Visualization?

## - Data visualization helps companies to analyze its differ- ent processes so␣
 ↪the management can focus on the areas for improvement to generate more␣
 ↪revenue and improve productivity.
## - It brings business intelligence to life.
## - It applies a creative approach to understanding the
##   hidden information within the business data.
## - It provides a better and faster way to identify patterns, trends, and␣
 ↪correlation in the data sets that would remain undetected with just text.
## - It identifies new business opportunities by predicting upcoming trends or␣
 ↪sales volumes and the revenue they will generate.
## - It supplies managers with information they need to make more effective␣
 ↪comparisons between data sets by plotting them on the same visualization.
## - It enables managers to understand the correlations between the operating␣
 ↪conditions and the business performance.
```

```
## - It helps businesses to discover the gray areas of the business and make␣
 ↪the right decisions for improvement.
## - Data visualization helps managers to understand custom- ers' behaviors and␣
 ↪interests and hence retains customers and market share.
```

1.1 Basic plotting

[7]:
```
import pandas as pd
import numpy as np
df = pd.DataFrame(np.random.randn(200,6),index= pd.date_range('1/9/2009',␣
 ↪periods=200), columns= list('ABCDEF'))
df.plot(figsize=(20, 10)).legend(bbox_to_anchor=(1, 1))
```

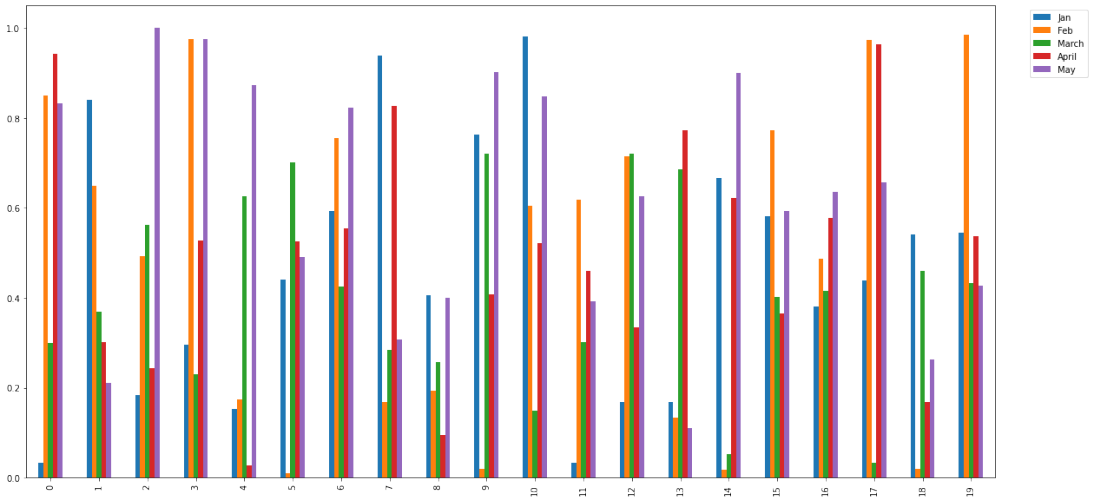
[7]: <matplotlib.legend.Legend at 0x7fd8ce7b52b0>



1.2 Direct Plotting

[8]:
```
import pandas as pd
import numpy as np
df = pd.DataFrame(np.random.rand(20,5), columns=['Jan','Feb','March','April',␣
 ↪'May'])
df.plot.bar(figsize=(20, 10)).legend(bbox_to_anchor=(1.1, 1))
```

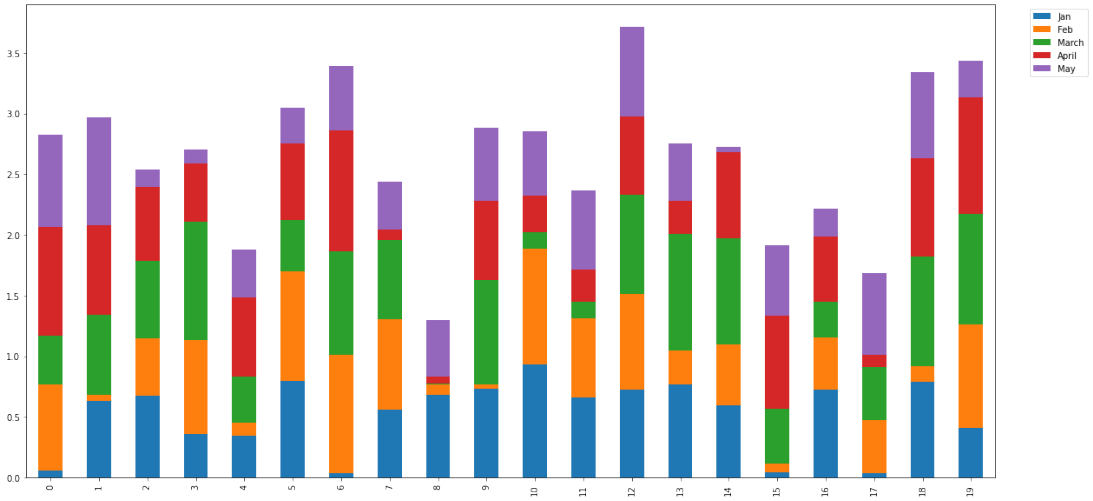
[8]: <matplotlib.legend.Legend at 0x7fd8d0c79c10>



1.3 stacked bar plot

[10]:
```
import pandas as pd
df = pd.DataFrame(np.random.rand(20,5), columns=['Jan','Feb','March','April',␣
 ↪'May'])
df.plot.bar(stacked=True,figsize=(20, 10)).legend(bbox_to_anchor=(1.1, 1))
```

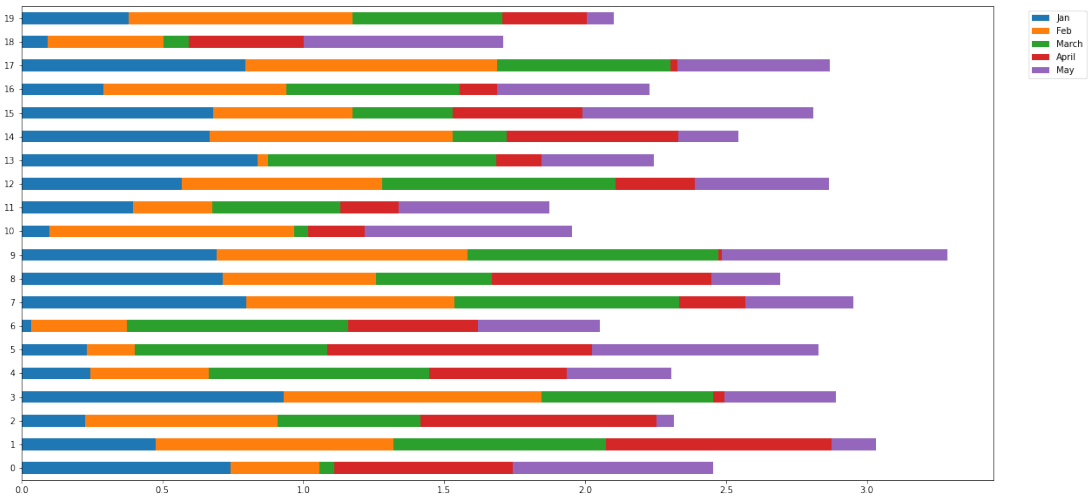
[10]: <matplotlib.legend.Legend at 0x7fd8d197f640>



1.4 Bar Plots

[11]:
```
import pandas as pd
df = pd.DataFrame(np.random.rand(20,5), columns=['Jan','Feb', 'March','April',␣
↪'May'])
df.plot.barh(stacked=True, figsize=(20, 10)).legend(bbox_to_anchor=(1.1, 1))
```

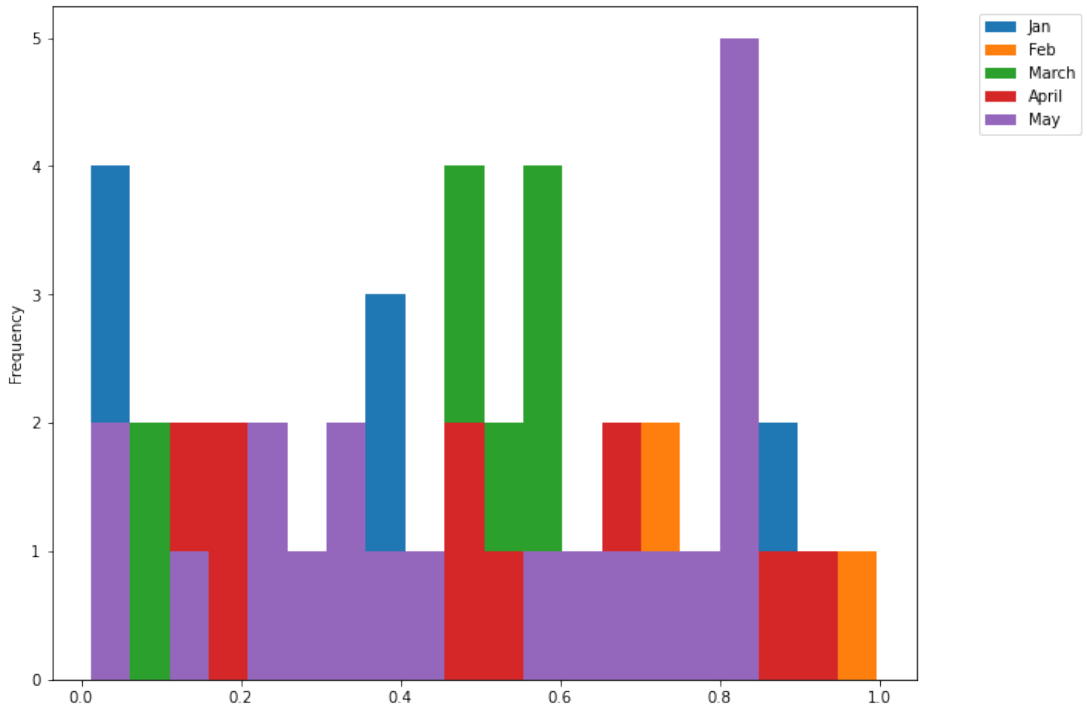
[11]: <matplotlib.legend.Legend at 0x7fd8d19d1370>



1.5 Bar's bins Attribute

[19]:
```
import pandas as pd
df = pd.DataFrame(np.random.rand(20,5), columns=['Jan','Feb','March','April',␣
↪'May'])
df.plot.hist(bins= 20, figsize=(10,8)).legend(bbox_to_anchor=(1.2, 1))
```

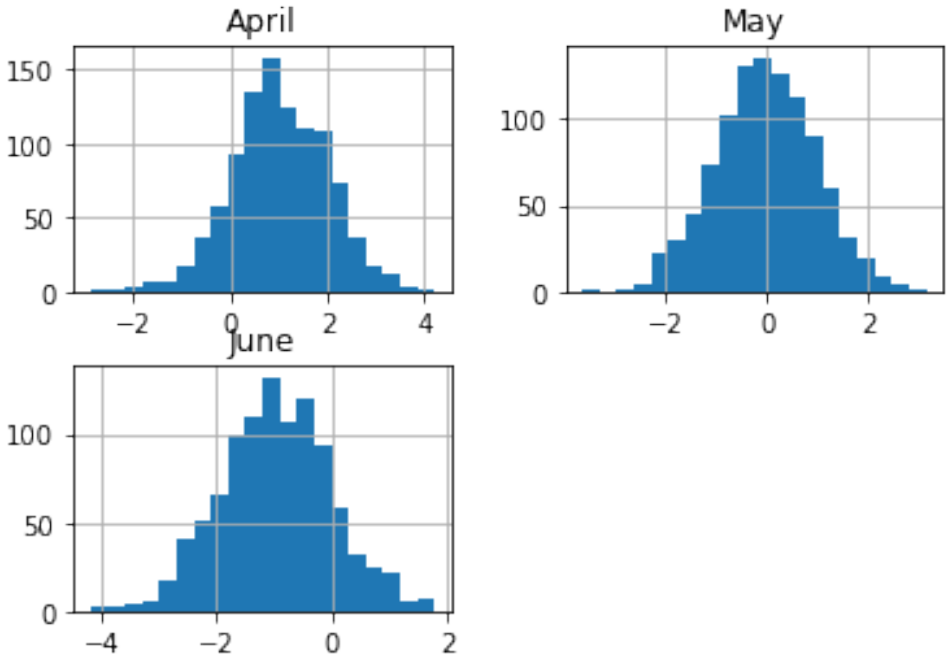
[19]: <matplotlib.legend.Legend at 0x7fd8d2630e20>



1.6 Multiple Histograms per Column

[20]:
```
import pandas as pd
import numpy as np
df=pd.DataFrame({'April':np.random.randn(1000)+1,'May':np.random.␣
↪randn(1000),'June': np.random.randn(1000) - 1}, columns=['April', 'May',␣
↪'June'])
df.hist(bins=20)
```

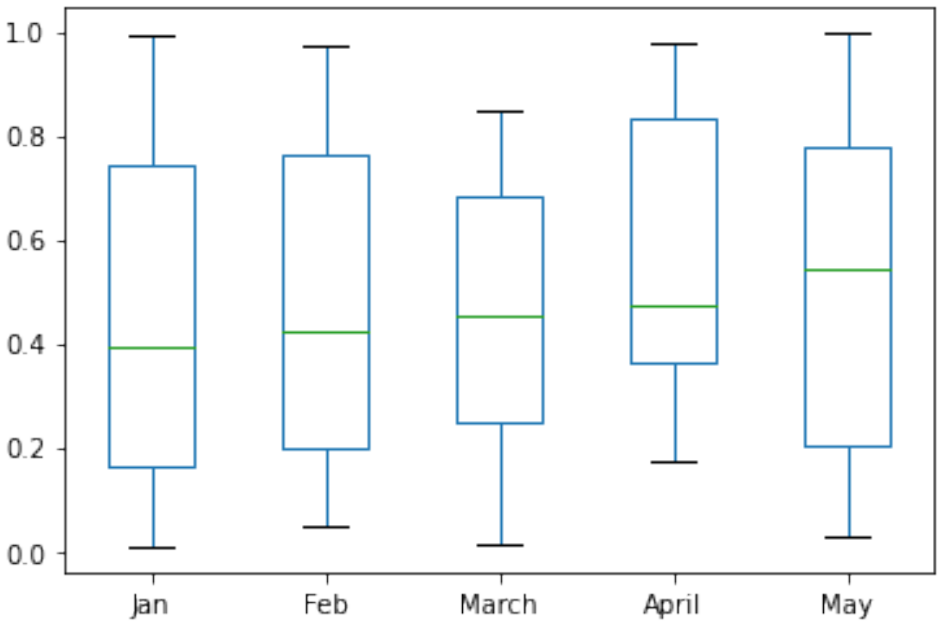
[20]:
```
array([[<AxesSubplot:title={'center':'April'}>,
        <AxesSubplot:title={'center':'May'}>],
       [<AxesSubplot:title={'center':'June'}>, <AxesSubplot:>]],
      dtype=object)
```



1.7 Box Plot

[21]:
```
import pandas as pd
import numpy as np
df = pd.DataFrame(np.random.rand(20,5),columns=['Jan','Feb','March','April',␣
↪'May'])
df.plot.box()
```

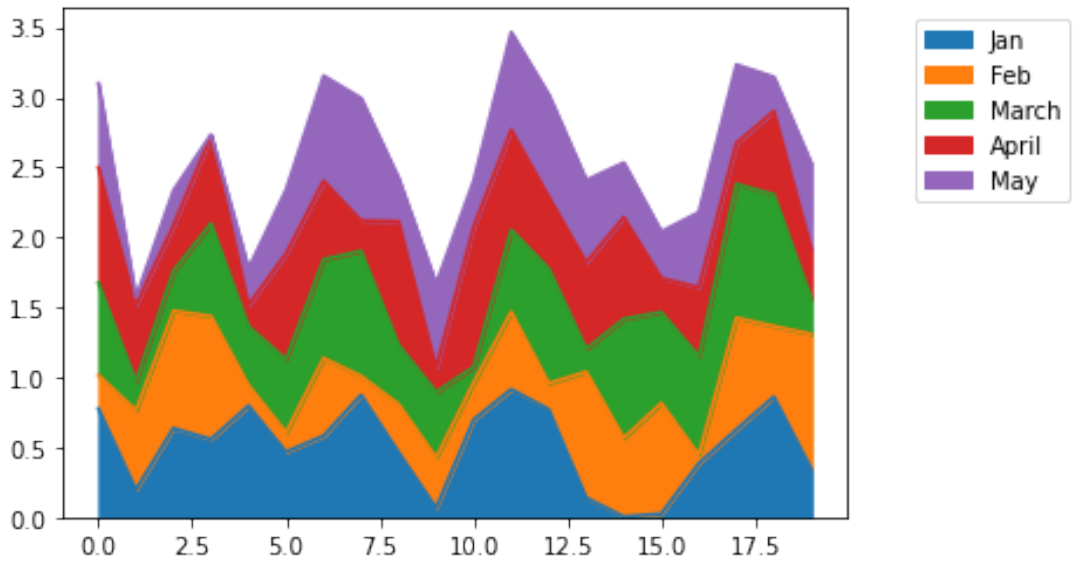
[21]: <AxesSubplot:>



1.8 Area Plot

[22]:
```
import pandas as pd
import numpy as np
df = pd.DataFrame(np.random.rand(20,5),columns= ['Jan','Feb','March','April',␣
↪'May'])
df.plot.area(figsize=(6, 4)).legend(bbox_to_anchor=(1.3, 1))
```

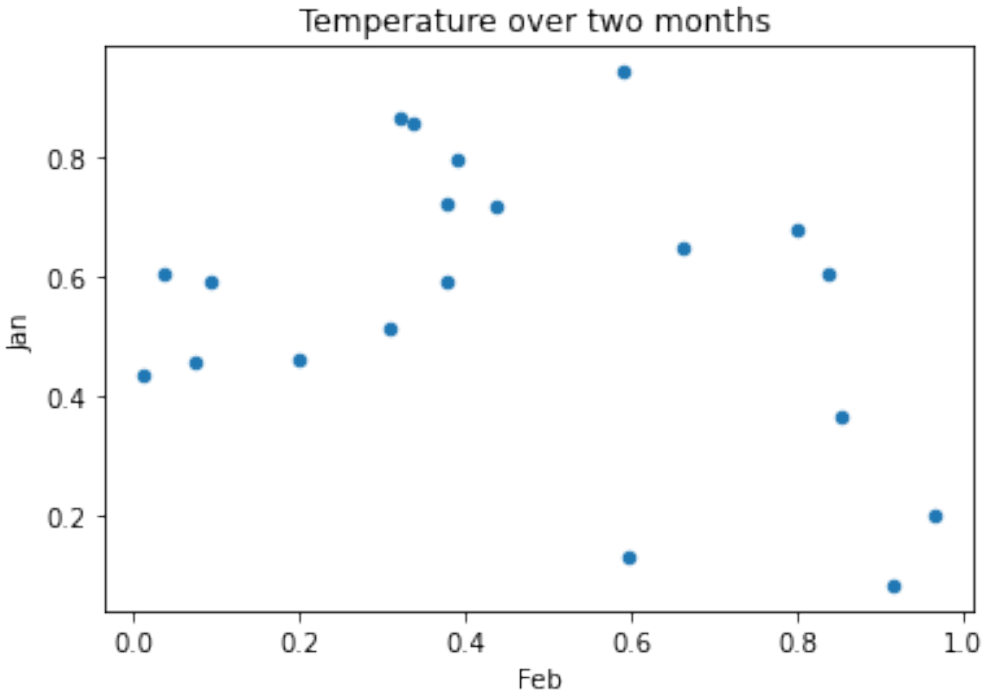
[22]: <matplotlib.legend.Legend at 0x7fd8d214a850>



1.9 Scatter Plot

[23]:
```
import pandas as pd
import numpy as np
df = pd.DataFrame(np.random.rand(20,5),columns= ['Jan','Feb','March','April',␣
↪'May'])
df.plot.scatter(x='Feb', y='Jan', title='Temperature over two months ')
```

[23]:
```
<AxesSubplot:title={'center':'Temperature over two months '}, xlabel='Feb',
ylabel='Jan'>
```


Temperature over two months
Jan
Feb
0.0
0.2
0.4
0.6
0.8
1.0
0.2
0.4
0.6
0.8