



第 11 章

数据可视化案例

11.1 商业图表绘制示例

商业图表一般以国外的《华尔街日报》《商业周刊》《经济学人》等经典期刊的图表作为案例与代表。近两年，国内的网易数读、TD 财经、澎湃新闻等新闻媒体的商业图表也越来越专业化。

图 11-1-1 展示了部分商业图表案例，来源于《华尔街日报》(The Wall Street Journal)、《商业周刊》(Business Week)、《经济学人》(The Economist)。其中，《经济学人》商业图表的显著特征就是图表的左上角标有一个红色的矩形；《华尔街日报》商业图表的显著特征就是图表右下角会标注“THE WALL STREET JOURNAL”。本节就教大家使用 Python 中的 matplotlib 绘制商业图表。

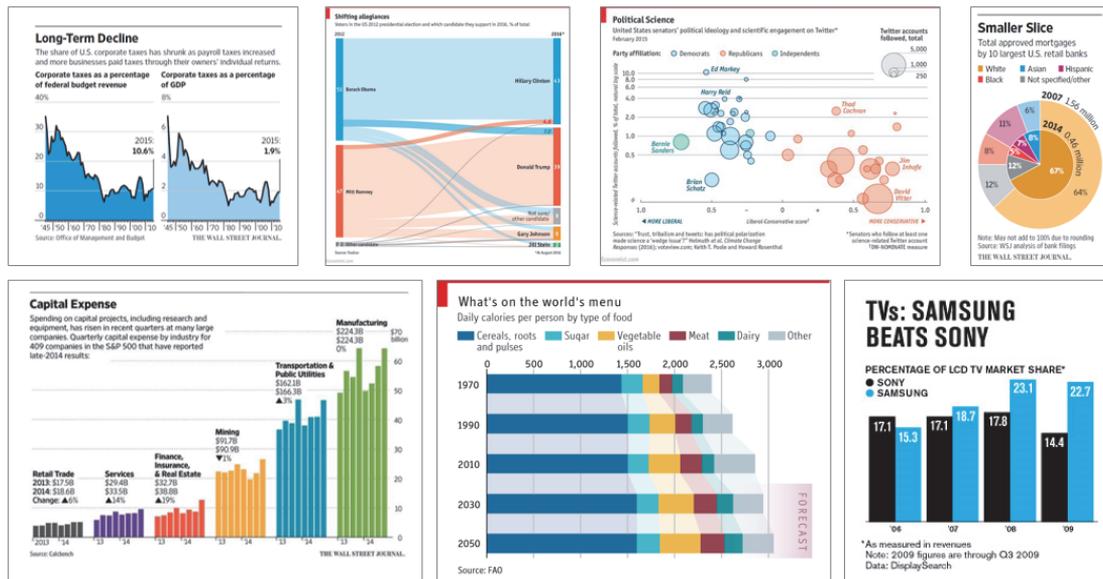
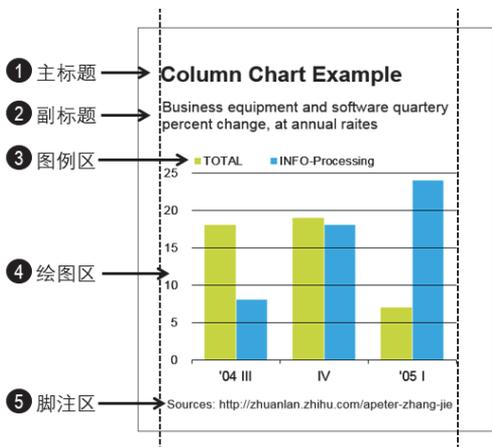


图 11-1-1 商业图表案例

11.1.1 商业图表绘制基础

仔细观察这些商业图表，我们可以发现《华尔街日报》《商业周刊》《经济学人》等商业其刊的固有格式，如图 11-1-2 所示。经典的商业图表都有一套固有的图表风格与颜色主题。相对我们平常直接绘制的图表（包括绘图区和图例区），商业图表还包括主标题、副标题以及脚注区。其中，主标题、副标题以及脚注区可以作为图表的背景信息，帮助读者了解图表所要表达的其他数据信息。这些图表的不同区域往往都左对齐，然后上下左右都留有一定的空白（margin）。



① **主标题**：标题区非常突出，往往占到整个图表面积的1/3甚至1/2。特别是主标题，往往使用大号字体和强烈对比效果，可以让读者首先捕捉到图表要表达的信息。

② **副标题**：副标题区往往会提供较为详细的信息，使用比主标题小一半的字号。

③ **图例区**：图例区位于副标题与绘图区之间，主要用于数据系列的标注与区分。但是，有时候会在绘图区中直接标注于数据系列上。

④ **绘图区**：绘图区为数据的可视化区域，绘图区的风格可以参考专业的商业图表绘制，主要体现在配色方案的选择上。

⑤ **脚注区**：脚注区一般使用Sources（数据来源）表明图表数据的来源。

图 11-1-2 商业图表范例（图表来源：《商业周刊》）

商业图表中最为重要的部分就是绘图区（plot area），绘图区的图表元素构成如图 11-1-3 所示，其具体组成如下所示。

（1）**数据系列（data series）**：使用点、线、面等不同图形表示数据系列，比如点类型的散点、气泡图，线类型的折线、曲线图，面类型的柱形、面积图等。

（2）**X 轴坐标（X number axis）**：数轴刻度应等距或具有一定规律性（如对数尺度），并标明数值。横轴刻度自左至右，数值一律由小到大。

（3）**Y 轴坐标（Y Number axis）**：数轴刻度应等距或具有一定规律性（如对数尺度），并标明数值。纵轴刻度自下而上，数值一律由小到大。

（4）**网格线（grid line）**：包括主要和次要的水平、垂直网格线 4 种类型，分别对应 Y 轴和 X 轴的刻度线。在折线图和统计直方图中，一般使用水平网格线作为数值比较大小的参考线。

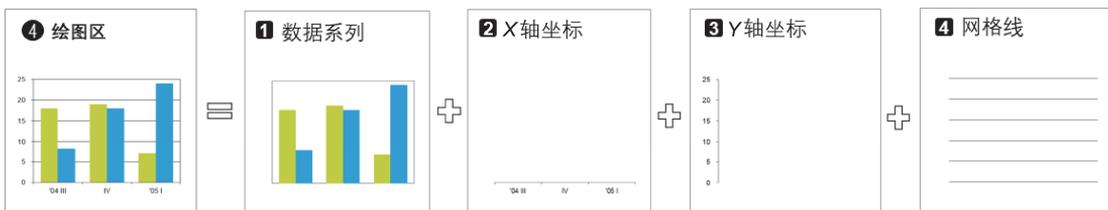


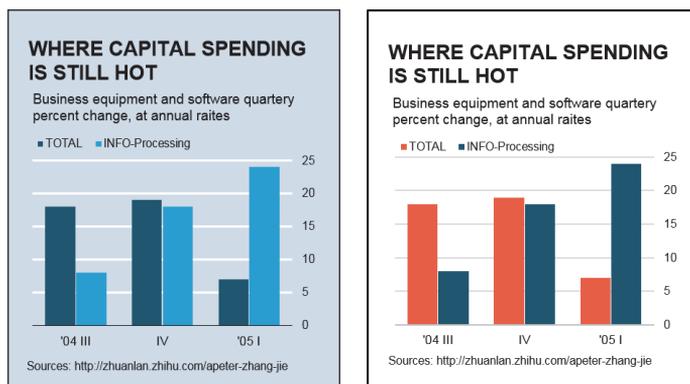
图 11-1-3 绘图区的图表元素组成

另外，绘图区的背景颜色是可以改变的。绘图区背景填充颜色的不同有时也是不同商业图表风格的重要特点。商业图表绘图区风格的设置主要包括绘图区的背景填充颜色，X 轴和 Y 轴坐标的颜色、标签位置与刻度线，以及网格线的颜色与粗细等。《华尔街日报》《商业周刊》《经济学人》三大

经典商业期刊图表绘图区元素的具体设置如图 11-1-4 所示。不同商业图表风格的双数据系列簇状柱形图如图 11-1-5 所示。

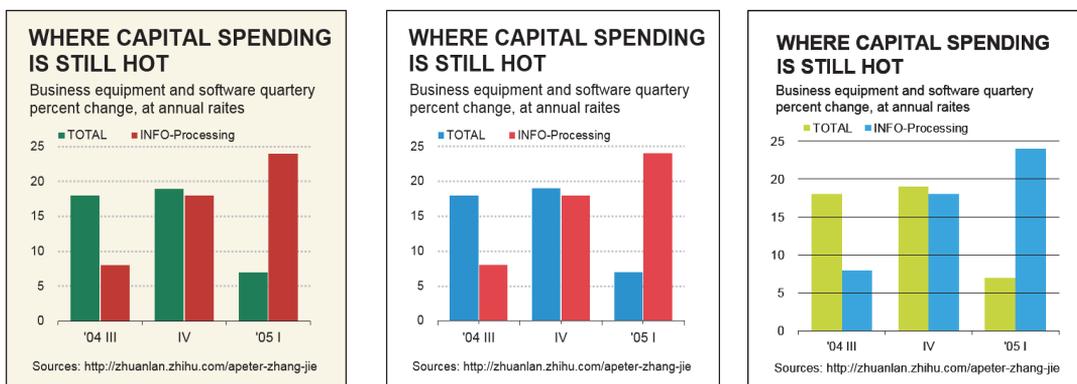
期刊	风格类型	1 绘图区			2 X轴坐标			3 Y轴坐标			4 网格线		
		填充颜色	类型	颜色	标签位置	刻度线	类型	标签位置	刻度线	类型	颜色	宽度 (磅)	
《经济学人》	[1]	206,219,231	0.75磅-实线	0, 0, 0	低	外部-主刻度线	无线条	高	无	实线	255,255,255	1.5	
	[2]	255,255,255	0.75磅-实线	0, 0, 0	低	外部-主刻度线	无线条	高	无	实线	191, 191, 191	1.5	
《华尔街日报》	[1]	248,242,228	0.75磅-实线	0, 0, 0	低	外部-主刻度线	无线条	低	无	圆点	191, 191, 192	1.5	
	[2]	255,255,255	0.75磅-实线	0, 0, 0	低	外部-主刻度线	无线条	低	无	实线	191, 191, 193	1.25	
《商业周刊》	2008	255,255,255	0.25磅-实线	0, 0, 0	低	外部-主刻度线	无线条	低	无	实线	0, 0, 0	0.25	

图 11-1-4 商业图表的绘图区元素设置



(a) 《经济学人》^[1]

(b) 《经济学人》^[2]



(c) 《华尔街日报》^[1]

(d) 《华尔街日报》^[2]

(e) 《商业周刊》

图 11-1-5 仿制的不同期刊风格的柱形图

除图表风格外，经典期刊的商业图表都有一套固有的颜色主题方案，图 11-1-6 展示了《经济学人》的颜色主题方案。这三种经典期刊中（见图 11-1-6~图 11-1-8），颜色主题方案多年始终保持不

变的是《经济学人》。《经济学人》的图表基本只用一个色系，或者做一些深浅明暗的变化；当数据系列增多时，会增加深绿色、深棕色等颜色。更多商业图表的颜色主题方案可以参考《Excel 数据之美：科学图表与商业图表的绘制》。

数据系列数	方案 1	方案 2	方案 3	方案 4
1	■	■	■	
2	■ ■	■ ■	■ ■	■ ■
3	■ ■ ■	■ ■ ■	■ ■ ■	
4	■ ■ ■ ■	■ ■ ■ ■	■ ■ ■ ■	■ ■ ■ ■
5	■ ■ ■ ■ ■	■ ■ ■ ■ ■	■ ■ ■ ■ ■	■ ■ ■ ■ ■
6	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■
>6	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■

图 11-1-6 《经济学人》的颜色主题方案

数据系列数	方案 1	方案 2	方案 3	方案 4
1	■	■	■	■
2	■ ■	■ ■	■ ■	■ ■
3	■ ■ ■	■ ■ ■	■ ■ ■	
4	■ ■ ■ ■	■ ■ ■ ■	■ ■ ■ ■	
5	■ ■ ■ ■ ■	■ ■ ■ ■ ■	■ ■ ■ ■ ■	■ ■ ■ ■ ■
6	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■
>6	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■

图 11-1-7 《华尔街日报》^[2]的颜色主题方案

数据系列数	方案 1	方案 2	方案 3	方案 4
1	■	■	■	■
2	■ ■	■ ■	■ ■	■ ■
3	■ ■ ■	■ ■ ■	■ ■ ■	
4	■ ■ ■ ■	■ ■ ■ ■	■ ■ ■ ■	
5	■ ■ ■ ■ ■	■ ■ ■ ■ ■	■ ■ ■ ■ ■	■ ■ ■ ■ ■
6	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■
>6	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■	■ ■ ■ ■ ■ ■ ■

图 11-1-8 《商业周刊》的颜色主题方案

技能 《经济学人》^[1]风格的柱形图

绘制商业期刊风格的图表，主要是在 Python 自动生成的图表的基础上，添加主副标题以及脚注；图表风格的具体设置可以参照图 11-1-4。图 11-1-5(a)所示的《经济学人》^[1]风格的柱形图的具体实现代码如下所示。

```

01 import matplotlib.pyplot as plt
02 import numpy as np
03 import pandas as pd
04 plt.rcParams["font.sans-serif"]='Arial'
05 #plt.rcParams["font.sans-serif"]='SimHei' #汉字显示设定
06 plt.rcParams['axes.unicode_minus']=False
07 plt.rcParams['axes.facecolor']='#CFDBE7'
08 plt.rcParams['savefig.facecolor']='#CFDBE7'
09 plt.rc('axes',axisbelow=True) #使网格线置于图表下层
10
11 df=pd.read_excel("多数据系列柱形图.xlsx",sheet_name="原始数据")
12 x_label=np.array(df["Quarter"])
13 x=np.arange(len(x_label))
14 y1=np.array(df["TOTAL"])
15 y2=np.array(df["INFO-Processing"])
16
17 width=0.35
18 fig=plt.figure(figsize=(5,4.5),dpi=100,facecolor='#CFDBE7')
19 plt.bar(x,y1,width=width,color='#01516C',label='TOTAL') #调整 y1 轴位置、颜色, label 为图例名称
20 plt.bar(x+width,y2,width=width,color='#01A4DC',label='INFO-Processing') #调整 y2 轴位置、颜色, label 为图例名称
21 plt.xticks(x+width/2,x_label,size=12) #设置 X 轴刻度、位置、大小
22 plt.yticks(size=12) #设置 Y 轴刻度、位置、大小
23 plt.grid(axis="y",c='w',linewidth=1.2) #设置 Y 轴网格线的颜色与粗细
24 #显示图例, loc 图例显示位置(可以用坐标方法显示), ncol 图例显示几列, 默认为 1 列,frameon 设置图形边框
25 plt.legend(loc=(0,1.02),ncol=2,frameon=False)
26 ax = plt.gca() #获取整个绘图区的句柄
27 ax.spines['top'].set_color('none') #设置上'脊梁'为无色
28 ax.spines['right'].set_color('none') #设置右'脊梁'为无色
29 ax.spines['left'].set_color('none') #设置左'脊梁'为无色
30 ax.yaxis.set_ticks_position('right') #Y 轴放置在右边
31 #添加主标题
32 plt.text(0.,1.25,s='WHERE CAPITAL SPENDING\nIS STILL HOT',transform=ax.transAxes, weight='bold',size=20)
33 #添加副标题
34 plt.text(0,1.12,s='Column charts are used to compare values\nacross categories by using vertical
bars.',transform=ax.transAxes,
35 weight='light',size=15)
36 #添加脚注
37 plt.text(0,-0.15,s='Sources: http://zhuanlan.zhihu.com/apeter-zhang-jie',transform=ax.transAxes,weight='light',size=10)
38 #图表的导出
39 plt.savefig('商业图表_经济学人 1.pdf',bbox_inches='tight',pad_inches=0.3)
40 plt.show()

```

11.1.2 商业图表绘制案例①

我们平时常用的条形图，也常常出现在商业图表中，但是往往会把 Y 轴标签省去，而将对应的数据名称放置在条形的上方，如图 11-1-9 所示。同时，省去 X 轴数值坐标，而将条形的数值直接放置在条形的右边。这样做的好处是可以节省图表的面积，尤其是当 Y 轴标签很长的时候。

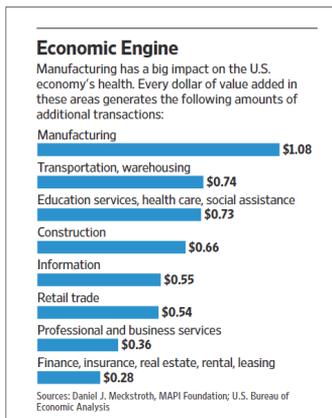


图 11-1-9 Y 轴标签省去的条形图（来源：《华尔街日报》）

技能 绘制 Y 轴标签省去的条形图

条形图一般需要降序展示，所以先使用 `sort_values()` 函数做降序处理，然后使用 `plt.barh()` 函数绘制条形，再使用 `plt.text()` 添加数据的类别和数值标签，如图 11-1-10(a) 所示。最后，省去 X 轴和 Y 轴，添加主、副标题以及脚注，如图 11-1-10(b) 所示，其具体代码如下所示。

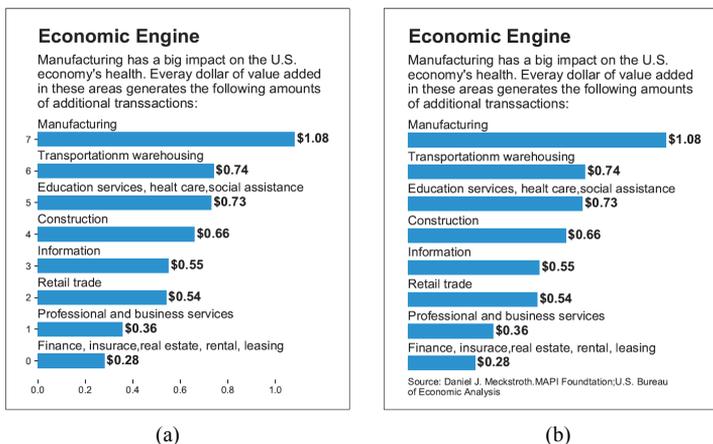


图 11-1-10 Y 轴标签省去的条形图绘制过程

```

01 import matplotlib.pyplot as plt
02 import numpy as np
03 import pandas as pd
04 df=pd.DataFrame(dict(group=["Manufacturing","Transportationm warehousing","Education services, healt care,social
assistance",
05                             "Construction","Information","Retail trade","Professional and business services",
06                             "Finance, insurace,real estate, rental, leasing"],
07                             price=[1.08,0.74,0.73,0.66,0.55,0.54,0.356,0.28]))
08 df=df.sort_values(by=["price"],ascending=True)
09 x_label=np.array(df['group'])
10 y=np.array(df['price'])
11 x_value=np.arange(len(x_label))
12 height=0.45
13 fig=plt.figure(figsize=(5,5))
14 plt.xticks([])
15 plt.yticks([])
16 ax = plt.gca() #获取整个表格边框
17 ax.spines['top'].set_color('none') #设置上'脊梁'为无色
18 ax.spines['right'].set_color('none') #设置右'脊梁'为无色
19 ax.spines['left'].set_color('none') #设置左'脊梁'为无色
20 ax.spines['bottom'].set_color('none') #设置下'脊梁'为无色
21 plt.barh(x_value,color='#0099DC',height=height,width=y,align="center")
22 for a,b,label in zip(y,x_value,x_label): #给条形图加标签, 需要使用 for 循环
23     plt.text(0, b+0.45, s=label, ha='left', va= 'center',fontsize=13,family='sans-serif')
24     plt.text(a+0.01, b, s="$"+ str(round(a,2)), ha='left', va= 'center',fontsize=13.5,family='Arial',weight ="bold")
25
26 plt.text(0,1.3,s='Economic Engine',transform=ax.transAxes,weight='bold',size=20,family='Arial')
27 plt.text(0,1.05,s="Manufacturing has a big impact on the U.S.\neconomy's health. Everay dollar of value
28 added\nin these areas generates the following amounts\nof additional transsactions: ",
29         transform=ax.transAxes,weight='light',size=14,family='sans-serif')
30 plt.text(0,-0.05,s='Source: Daniel J. Meckstroth.MAPI Foundation;U.S.
31 Bureau\nof Economic Analysis',transform=ax.transAxes,weight='light',size=10,family='sans-serif')
32 #plt.savefig('商业图表_条形图.pdf',bbox_inches='tight', pad_inches=0.3)
33 plt.show()

```

11.1.3 商业图表绘制案例②

商业图表往往会在平常绘制的图表基础上，更加注重图表的美观性，如图 11-1-11 所示的带面积填充连接的堆积柱形图，就是在普通堆积柱形图的基础上，在相邻的堆积柱形之间添加填充面积引导，更加便于展示不同数据系列之间的数据累加情况。

但是对于 X 轴为时间序列的堆积柱形图，可以将数据系列按平均数值做排序处理后，使数值越

大的数据系列越贴近 X 轴, 这样便于比较不同数据系列的数值大小。所以在图 11-1-11 中, “Asia-Pacific” 应该放置在最下面, 然后依次是 “North America” “Western Europe” 等。

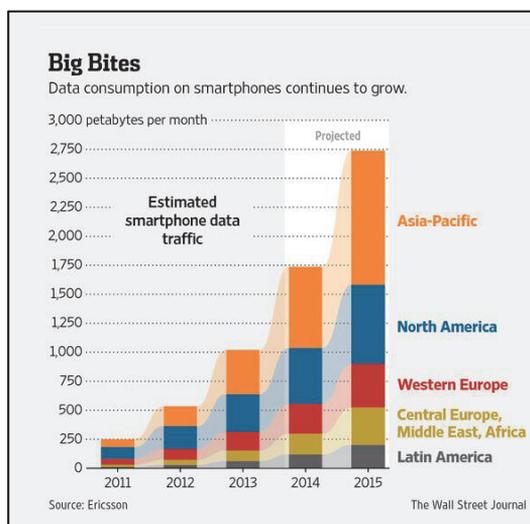


图 11-1-11 带面积填充连接的堆积柱形图 (来源:《华尔街日报》)

其实, 带面积填充连接的堆积柱形图就是两个图层的叠加: 上层的堆积柱形图(见图 11-1-12(a)) 和下层的堆积面积图(见图 11-1-12(b)), 叠加效果如图 11-1-12(c)所示, 这样基本实现图 11-1-11 绘图区的数据系列展示效果。其关键在于如何根据堆积柱形图的数据构造堆积面积图的数据。

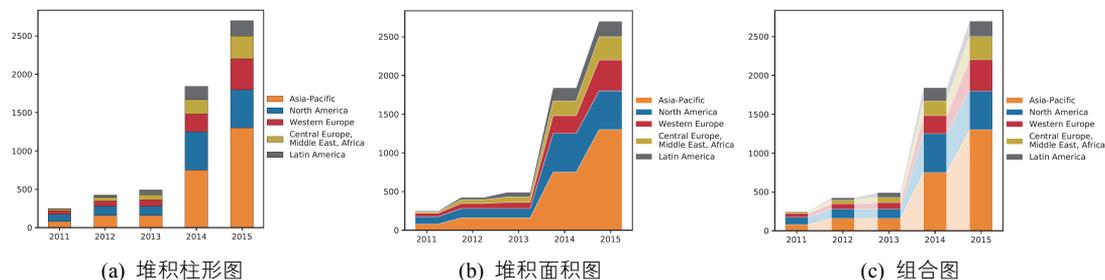


图 11-1-12 带面积填充连接的堆积柱形图的绘制过程

技能 绘制带面积填充连接的堆积柱形图

图 11-1-13 所示为使用 matplotlib 仿制的带面积填充连接的堆积柱形图, 其具体实现代码如下所示。堆积柱形图和堆积面积图的共有部分数据为数据系列的数值以及柱形宽度 (width)。使用 matplotlib 包的 `stackplot()` 函数可以绘制如图 11-1-12(b) 所示的底层的堆积面积图; `bar()` 函数可以绘制

如图 11-1-12(a)所示的堆积柱形图；`annotate()`函数可以添加带引导线的文本；`text()`函数可以添加主/副标题等图表背景信息。

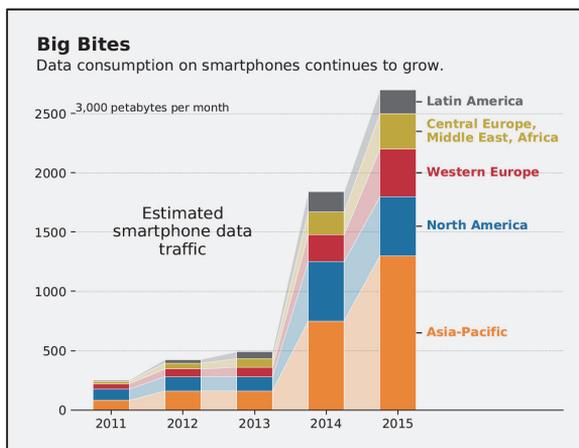


图 11-1-13 使用 matplotlib 仿制的带面积填充连接的堆积柱形图

```

01 from matplotlib import pyplot as plt
02 import pandas as pd
03 import numpy as np
04 plt.rcParams['axes.axisbelow']=True)
05 plt.rcParams['axes.facecolor']='#EFEFEF'
06 plt.rcParams['savefig.facecolor']='#EFEFEF'
07
08 df= pd.read_csv("商业图表_堆积柱形图.csv",engine='python',index_col=0)
09 meanRow_df=df.apply(lambda x: x.mean(), axis=0)      #对每个数据系列求均值
10 Sing_df=meanRow_df.sort_values(ascending=False).index #降序排序处理
11 n_row,n_col=df.shape
12 x_value=np.arange(n_row)   #构造 X 轴数值
13 colors=["#F28526","#0671A8","#C72435","#C3A932","#636466"]   #构造数据系列的填充颜色列表
14
15 width=0.5 #设定柱形的宽度
16 #构造面积图部分的数据
17 x=[] #x 为堆积面积图的 X 轴数值
18 for i in range(n_row):
19     x=x+[i-width/2,i+width/2]
20 df_area=pd.DataFrame(index=x)
21 for j in list(range(n_col))[:-1]:
22     y=[]
23     for i in range(n_row):
24         y=y+np.repeat(df.iloc[i,j],3).tolist()

```

```

25 df_area[df.columns[j]] = y #构造堆积面积图每个数据系列的 Y 轴数值
26
27 fig=plt.figure(figsize=(7,5),dpi=100,facecolor='#EFEFEF')
28 #绘制底层的堆积面积图
29 plt.stackplot(df_area.index.values, df_area.values.T,colors=colors,linewidth=0.1,edgecolor='w',alpha=0.25)
30 #绘制上层的堆积柱形图
31 bottom_y=np.zeros(n_row)
32 for i in range(n_row):
33     label=Sing_df[i]
34     plt.bar(x_value,df.loc[:,label],bottom=bottom_y,width=width,color=colors[i],label=label,edgecolor='w',
linewidth=0.25)
35     #添加带引导线的每个数据系列名称，并用跟数据系列柱形填充一样的颜色
36     plt.annotate(s=label,xy=(x_value[-1]+width/2*0.9,bottom_y[-1]+df.loc[:,label].values[-1]/2),
37                 xytext=(x_value[-1]*1.1,bottom_y[-1]+df.loc[:,label].values[-1]/2),c=colors[i],
38                 arrowprops=dict(facecolor='gray',arrowstyle='-'),verticalalignment='center',weight='bold')
39     bottom_y=bottom_y+df.loc[:,label].values
40 #设置图表风格，包括图表背景颜色、X 轴和 Y 轴格式以及网格线的格式
41 plt.xlim(-0.5,6.3)
42 plt.xticks(x_value,df.index,size=10) #设置 X 轴刻度与标签
43 plt.grid(which='major',axis="y",linestyle='--',linewidth='0.5',color='gray',alpha=0.5)
44 ax = plt.gca() #删除左边和顶部的绘图区域边框线
45 ax.spines['right'].set_color('none')
46 ax.spines['top'].set_color('none')
47 ax.spines['left'].set_color('none')
48 #添加图表的背景信息，包括 Y 轴坐标、图表说明以及主副标题
49 plt.text(-0.5,2500,s='3,000 petabytes per month',weight='light',size=9,verticalalignment='bottom')
50 plt.text(1,1500,s='Estimated\smartphone data\ntraffic',weight='light',size=13,verticalalignment='center',
horizontalalignment='center')
51 plt.text(-0.08,1.07,s='Big Bites',transform=ax.transAxes,weight='bold',size=15)
52 plt.text(-0.08,1.01,s='Data consumption on smartphones continues to grow.',transform=ax.transAxes,
weight='light',size=12)
53 #保存导出图表为 PDF 格式
54 #plt.savefig('商业图表_堆积柱形图.pdf',bbox_inches='tight',pad_inches=0.3)
55 plt.show()

```

11.2 学术图表绘制示例

图表在学术论文中是很重要的一部分。实验结果是论文的核心和主要部分，而实验结果一般以图表的形式呈现。读者经常通过图表来判断这篇文章是否值得阅读，所以每个图表都应该能不依赖正文而独立存在。所谓一图抵千言（A picture is worth a thousand words）。图表设计是否精确和合理