

Python 在财务中的综合应用

项目导读

数字经济时代，以“大智移云物区”等技术为代表的新经济驱动财会产业链数字化升级，业务、财务、税务数据逐渐融合形成数字网络，会计核算职能逐渐弱化，对财务人员的财务决策和管理能力需求逐渐提升。传统的财务分析工具在对海量的、细颗粒度的、多维度的数据体系进行分析时，不能很好地满足企业财务管理需求。本项目的两个综合案例，将完整呈现图 9-1 中从数据采集、数据加工、数据分析和数据应用，到为企业提供管理决策的流程。

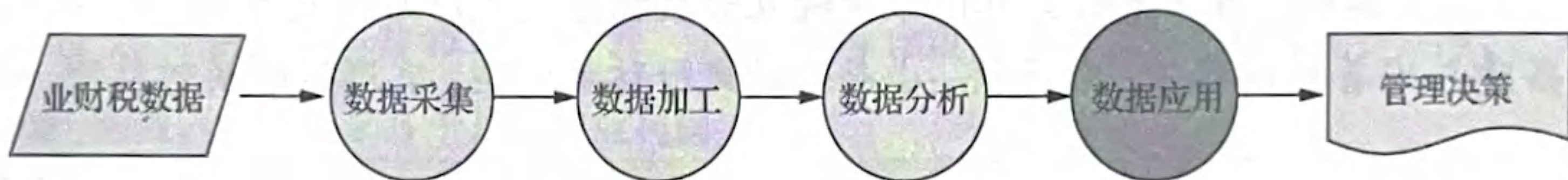


图 9-1 数据分析完整流程

任务一 上市公司财务指标可视化分析与应用

在传统的财务分析方法中，我们通过对企业盈利能力、营运能力、成长能力、偿债能力、现金流量等不同维度指标的计算，可以综合评价一个企业的财务状况。而在数字经济时代，随着大数据体系的逐步建立，这些数据都很容易从开放的数据源中获取，从而减少了大量简单重复的计算，更多的是将关注点放在数据的分析和数据价值的挖掘上。获取上市公司年报数据的常用途径如图 9-2 所示。



图 9-2 获取上市公司年报数据的常用途径

业务场景 9-1 上市公司财务指标分析

以格力电器（股票代码：sz.000651）为例，通过完整的数据分析流程，对格力电器近 10 年来的盈利情况进行分析和可视化呈现。企业数据仍使用项目五中的数据接口方法获取。

思路分析

- 1 数据采集：通过 baostock 数据接口采集数据

- 2 数据加工：对采集数据进行整理、清洗和加工
- 3 数据分析：通过计算、模型构建、可视化等方式进行数据分析
- 4 数据应用：利用数据分析结果提供决策支持

代码实现

(1) 导入 baostock 数据接口、pandas 模块。

➤ 导入模块

```
1 import baostock as bs
2 import pandas as pd
```

(2) 利用 query_profit_data()数据接口获取数据（详细方法可查阅项目五）。

➤ 数据准备：获取数据

```
1 lg = bs.login() # 登录系统
2 result_list = []
3 # 放置一个列表用于接收数据接口返回的数据，取名“结果列表”
4 year_list =pd.date_range('20111231', periods=10, freq='1Y').strftime("%Y")
5 # 利用 date_range()函数生成年度数据，便于查询
6 for year in year_list:
7     # 查询季频盈利能力数据
8     return_data = bs.query_profit_data(code="sz.000651", year=year, quarter=4)
9     while (return_data.error_code == '0') & return_data.next():
10         result_list.append(return_data.get_row_data()) # 将查询结果添至结果列表
11 result_table = pd.DataFrame(result_list, columns=return_data.fields)
12 # 将结果列表转为 DataFrame 格式
13 bs.logout() # 退出系统
14 result_table # 输出结果
```

运行结果

	code	pubDate	statDate	roeAvg	npMargin	gpMargin	netProfit	epsTTM	MBRevenue	totalShare	liqaShare
0	sz.000651	2012-04-25	2011-12-31	0.338857	0.063704	0.180666	5297340543.070000	1.858462	76754746050.710000	2817888750.00	2780073375.00
1	sz.000651	2013-04-27	2012-12-31	0.332792	0.074972	0.262920	7445927982.700000	2.453456	91248254915.960000	3007865439.00	2986211564.00
2	sz.000651	2014-04-25	2013-12-31	0.354521	0.092185	0.322369	10935755177.190000	3.614082	108052844520.690000	3007865439.00	2986211564.00
3	sz.000651	2015-04-28	2014-12-31	0.359563	0.103469	0.361003	14252954811.960000	4.706051	122745036614.310000	3007865439.00	2986469263.00
4	sz.000651	2016-04-29	2015-12-31	0.273413	0.129149	0.324597	12623732620.220000	2.083279	87930981568.340000	6015730878.00	5972117234.00
5	sz.000651	2017-04-27	2016-12-31	0.304205	0.143345	0.327018	15524634903.870000	2.563440	93187780602.400000	6015730878.00	5971955954.00
6	sz.000651	2018-04-26	2017-12-31	0.375051	0.151791	0.328577	22508599044.090000	3.723833	132189595255.700000	6015730878.00	5969568329.00
7	sz.000651	2019-04-29	2018-12-31	0.333959	0.133145	0.302282	26379029817.060000	4.355711	170592428489.170000	6015730878.00	5970717628.00
8	sz.000651	2020-04-30	2019-12-31	0.245151	0.125293	0.275815	24827243603.970000	4.105343	156888659016.130000	6015730878.00	5969931253.00
9	sz.000651	2021-04-29	2020-12-31	0.196811	0.132457	0.261417	22279242195.270000	3.686187	130427766473.540000	6015730878.00	5969908834.00

(3) 证券数据平台证券宝提供的 query_profit_data()参数说明，可参阅表 5-1。对各数据列更改列名。

➤ 数据加工：更改列名

```
1 result_table.rename(columns={'code':'证券代码','pubDate':'发布日期','statDate':'
财报日期','roeAvg':'净资产收益率','npMargin':'销售净利率','gpMargin':'销售毛利率
','netProfit':'净利润','epsTTM':'每股收益','MBRevenue':'主营业务收入','totalShare':'总
股本','liqaShare':'流通股本'},inplace = True)
2 result_table
```


运行结果

	证券代码	发布日期	财报日期	净资产收益率	销售净利率	销售毛利率	净利润	每股收益	主营业务收入	总股本	流通股本
0	sz.000651	2012-04-25	2011-12-31	0.338857	0.063704	0.180666	5297340543.070000	1.858462	76754746050.710000	2817888750.00	2780073375.00
1	sz.000651	2013-04-27	2012-12-31	0.332792	0.074972	0.262929	7445927982.700000	2.453456	91248254915.960000	3007865439.00	2986211564.00
2	sz.000651	2014-04-25	2013-12-31	0.354521	0.092185	0.322369	10935755177.190000	3.614082	108052844520.690000	3007865439.00	2986211564.00
3	sz.000651	2015-04-28	2014-12-31	0.359563	0.103469	0.361003	14252954811.960000	4.706051	122745036614.310000	3007865439.00	2986469263.00
4	sz.000651	2016-04-29	2015-12-31	0.273413	0.129149	0.324597	12623732620.220000	2.083279	87930981568.340000	6015730878.00	5972117234.00
5	sz.000651	2017-04-27	2016-12-31	0.304205	0.143345	0.327018	15524634903.870000	2.563440	93187780602.400000	6015730878.00	5971955954.00
6	sz.000651	2018-04-26	2017-12-31	0.376051	0.151791	0.328577	22508599044.090000	3.723833	132189595255.700000	6015730878.00	5969668329.00
7	sz.000651	2019-04-29	2018-12-31	0.333959	0.133145	0.302282	26379029817.060000	4.355711	170592428489.170000	6015730878.00	5970717628.00
8	sz.000651	2020-04-30	2019-12-31	0.245151	0.125293	0.275815	24827243603.970000	4.105343	156888659016.130000	6015730878.00	5969931253.00
9	sz.000651	2021-04-29	2020-12-31	0.196811	0.132457	0.261417	22279242195.270000	3.686187	130427766473.540000	6015730878.00	5969908834.00

(4) 提取用于盈利分析的数据。由于接口提供的数据为文本格式，为了便于计算及处理，将其转换为数值格式。

➤ 数据加工：格式转换

```
1 profit_data = result_table.iloc[:, 0:9]
2 # 提取财务数据，用于分析
3 profit_data = profit_data.astype({'净资产收益率': 'float', '销售净利率': 'float', '销售毛利率': 'float', '净利润': 'float', '每股收益': 'float', '主营业务收入': 'float'})
4 # 将文本格式转化为数值格式
5 pd.options.display.float_format = '{:.4f}'.format#保留 4 位小数
6 profit_data
```

运行结果

	证券代码	发布日期	财报日期	净资产收益率	销售净利率	销售毛利率	净利润	每股收益	主营业务收入
0	sz.000651	2012-04-25	2011-12-31	0.3389	0.0637	0.1807	5297340543.0700	1.8585	76754746050.7100
1	sz.000651	2013-04-27	2012-12-31	0.3328	0.0750	0.2629	7445927982.7000	2.4535	91248254915.9600
2	sz.000651	2014-04-25	2013-12-31	0.3545	0.0922	0.3224	10935755177.1900	3.6141	108052844520.6900
3	sz.000651	2015-04-28	2014-12-31	0.3596	0.1035	0.3610	14252954811.9600	4.7061	122745036614.3100
4	sz.000651	2016-04-29	2015-12-31	0.2734	0.1291	0.3246	12623732620.2200	2.0833	87930981568.3400
5	sz.000651	2017-04-27	2016-12-31	0.3042	0.1433	0.3270	15524634903.8700	2.5634	93187780602.4000
6	sz.000651	2018-04-26	2017-12-31	0.3751	0.1518	0.3286	22508599044.0900	3.7238	132189595255.7000
7	sz.000651	2019-04-29	2018-12-31	0.3340	0.1331	0.3023	26379029817.0600	4.3557	170592428489.1700
8	sz.000651	2020-04-30	2019-12-31	0.2452	0.1253	0.2758	24827243603.9700	4.1053	156888659016.1300
9	sz.000651	2021-04-29	2020-12-31	0.1968	0.1325	0.2614	22279242195.2700	3.6862	130427766473.5400

(5) 为了更直观地看到财务指标的变化情况，将 2011—2020 年这 10 年的净资产收益率、销售净利率、销售毛利率绘制成条形图。

➤ 数据分析：绘制财报指标比较条形图

```
1 import matplotlib.pyplot as plt
2 plt.rcParams['font.family'] = 'SimHei' # 设置中文字体为黑体
3 plt.rcParams['axes.unicode_minus'] = False # 中文状态下负号正常显示
4 profit_data.plot(x='财报日期', y=['净资产收益率', '销售净利率', '销售毛利率'], kind='barh', title='2011-2020 年财报指标比较', figsize=(40, 25))
```


运行结果（见图 9-3）

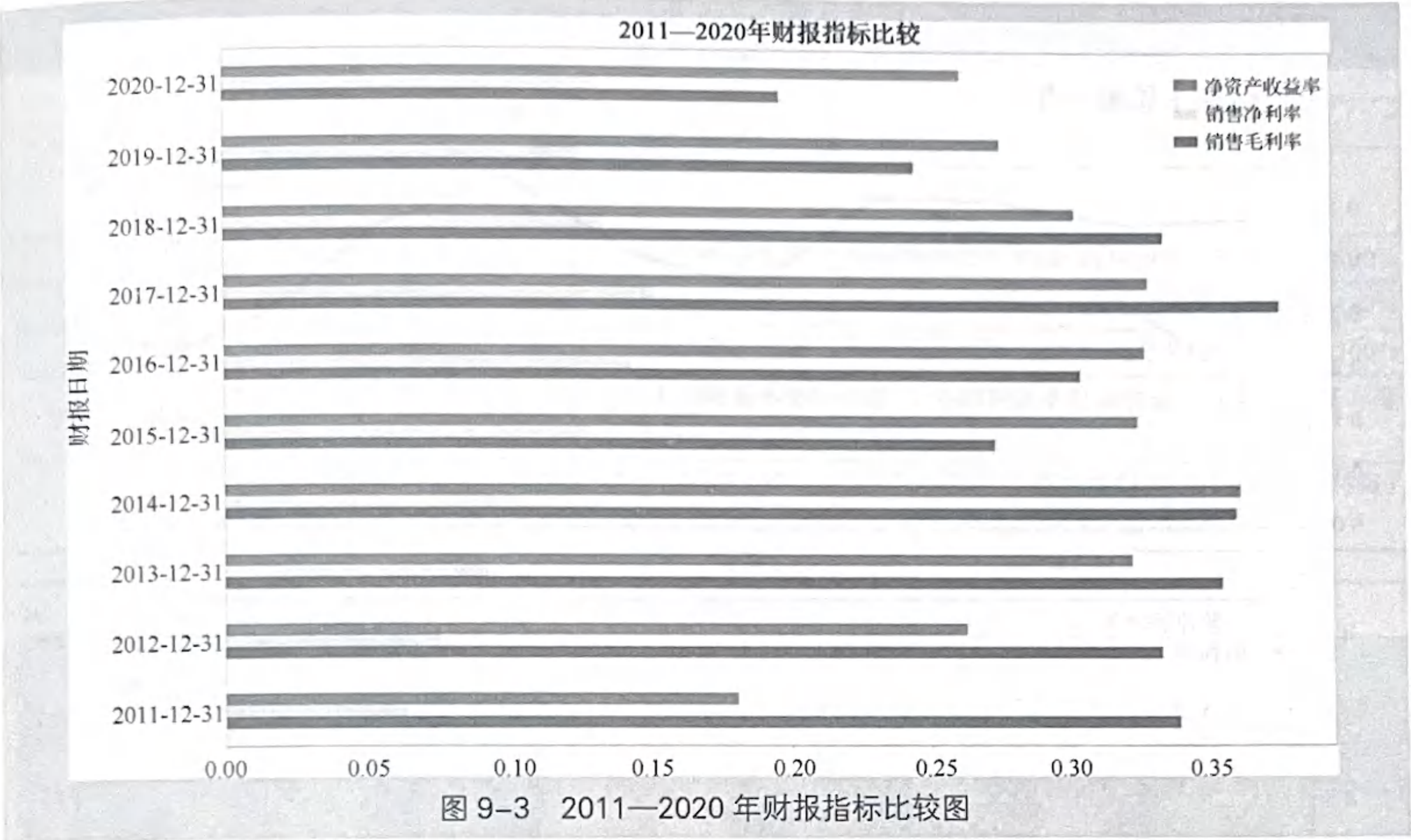


图 9-3 2011—2020 年财报指标比较图

（6）绘制 2011—2020 年主营业务收入和净利润比较的柱形图。

➤ 数据分析：绘制主营业务收入与净利润比较柱形图

```
1 profit_data.plot('财报日期',['主营业务收入','净利润'],secondary_y=['净利润'],kind='bar',title='2011—2020 年主营业务收入与净利润比较',figsize=(12,4),rot=0)
```

运行结果（见图 9-4）



图 9-4 2011—2020 年主营业务收入与净利润比较柱形图

（7）同样地，为了更好地进行多维度数据分析，我们也可以同时绘制多张不同类型的子图。

➤ 数据分析：绘制多张不同类型的子图

```
1 figure,axes = plt.subplots(2,1,figsize=(12,8),sharex=True)
2 ax0 = profit_data.plot('财报日期',['净资产收益率','销售净利率','销售毛利率'],
title='2011—2020 年财报指标比较',ax=axes[0])
3 ax1 = profit_data.plot('财报日期','主营业务收入',kind='bar',title='2011—2020 年主营
```



```
业务收入与净利润比较',color='gold',ax=axes[1])
4 ax2 = profit_data.plot('财报日期','净利润',secondary_y=True,color='orangered',
ax=axes[1],style='--',marker='o',linewidth=2)
```

运行结果（见图 9-5）

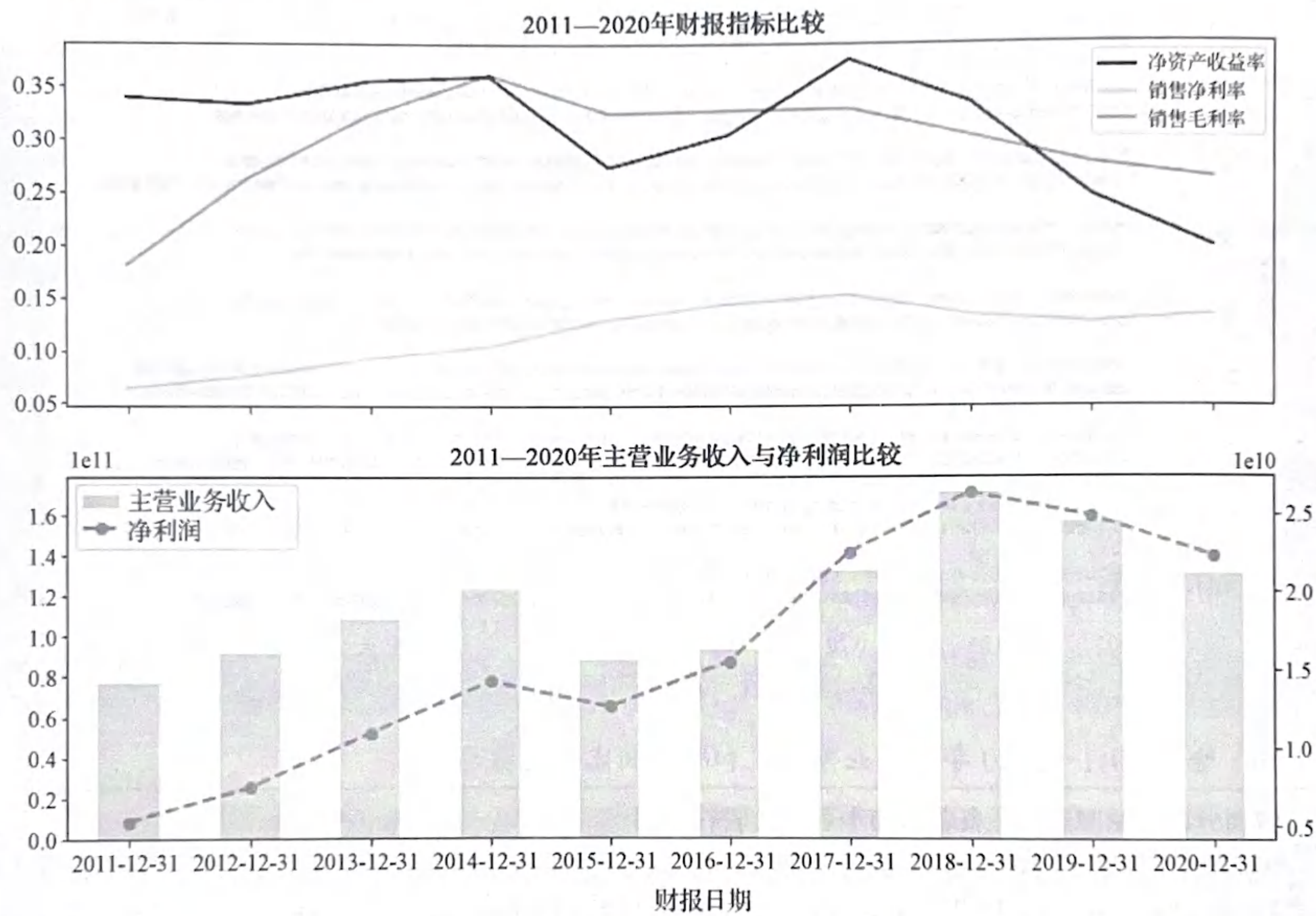


图 9-5 2011—2020 年盈利能力多维数据分析

(8) 用同样的方法，对格力电器的季频成长能力进行分析。

```
> 数据准备：通过 query_growth_data()获取数据
1 import baostock as bs
2 import pandas as pd
3 lg = bs.login() # 登录系统
4 result_list = []
5 # 放置一个列表用于接收数据接口返回的数据，取名“结果列表”
6 year_list =pd.date_range('20111231', periods=10, freq='1Y').strftime("%Y")
7 # 利用 date_range()函数生成年度数据，便于查询
8 for year in year_list:
9     # 查询季频成长能力数据
10     return_data = bs.query_growth_data(code="sz.000651", year=year, quarter=4)
11     while (return_data.error_code == '0') & return_data.next():
12         result_list.append(return_data.get_row_data()) # 将查询结果添加至结果列表
13 result_table = pd.DataFrame(result_list, columns=return_data.fields)
14 # 将结果列表转为 DataFrame 格式
15 bs.logout() # 退出系统
16 result_table # 输出结果
```

(9) 证券数据平台证券宝提供的 query_growth_data()参数说明，如表 9-1 所示。对各数据列更改列名。

表 9-1 季频成长能力 query_growth_data()参数说明

参数名称	参数描述	算法说明
code	证券代码	
pubDate	公司发布财报的日期	
statDate	财报统计季度的最后一天, 比如 2021-03-31、2021-06-30	
YOYEquity	净资产同比增长率	(本期净资产-上年同期净资产)/上年同期净资产的绝对值×100%
YOYAsset	总资产同比增长率	(本期总资产-上年同期总资产)/上年同期总资产的绝对值×100%
YOYNI	净利润同比增长率	(本期净利润-上年同期净利润)/上年同期净利润的绝对值×100%
YOYEPSBasic	基本每股收益同比增长率	(本期基本每股收益-上年同期基本每股收益)/上年同期基本每股收益的绝对值×100%
YOYPNI	归属母公司股东净利润同比增长率	(本期归属母公司股东净利润-上年同期归属母公司股东净利润)/上年同期归属母公司股东净利润的绝对值×100%

➤ 数据加工：更改列名

```
1 # 根据证券宝提供的参数说明, 将列名更改为中文
2 result_table.rename(columns={'code': '证券代码', 'pubDate': '发布日期', 'statDate': '财报日期', 'YOYEquity': '净资产同比增长率', 'YOYAsset': '总资产同比增长率', 'YOYNI': '净利润同比增长率', 'YOYEPSBasic': '基本每股收益同比增长率', 'YOYPNI': '归属母公司股东净利润同比增长率'}, inplace = True)
3 result_table
(10) 将数据接口提供的文本格式数据转换为数值格式数据。
```

➤ 数据加工：格式转换

```
1 growth_data = result_table.astype({'净资产同比增长率': 'float', '总资产同比增长率': 'float', '净利润同比增长率': 'float', '基本每股收益同比增长率': 'float', '归属母公司股东净利润同比增长率': 'float'})
2 # 将文本格式转化为数值格式
3 pd.options.display.float_format = '{:.2f}'.format # 保留 2 位百分数小数
4 growth_data
(11) 绘制 2011—2020 年成长能力多维分析图。
```

➤ 数据分析：绘制可视化图表

```
1 import matplotlib.pyplot as plt
2 plt.rcParams['font.family'] = 'SimHei' # 设置中文字体为黑体
3 plt.rcParams['axes.unicode_minus'] = False # 中文状态下负号正常显示
4 figure, axes = plt.subplots(5, 1, figsize=(12, 10), sharex=True)
5 ax0 = growth_data.plot('财报日期', ['净资产同比增长率'], title='2011-2020 年净资产同比增长率比较', kind='bar', color='gold', ax=axes[0])
6 ax1 = growth_data.plot('财报日期', ['总资产同比增长率'], title='2011-2020 年总资产同比增长率比较', kind='bar', color='red', ax=axes[1])
7 ax2 = growth_data.plot('财报日期', ['净利润同比增长率'], title='2011-2020 年净利润同比增长率比较', kind='bar', color='blue', ax=axes[2])
8 ax3 = growth_data.plot('财报日期', ['基本每股收益同比增长率'], title='2011-2020 年基本每股收益同比增长率比较', kind='bar', color='green', ax=axes[3])
9 ax4 = growth_data.plot('财报日期', ['归属母公司股东净利润同比增长率'], title='2011-2020 年归属母公司股东净利润同比增长率比较', kind='bar', color='orange', ax=axes[4])
```


运行结果（见图 9-6）

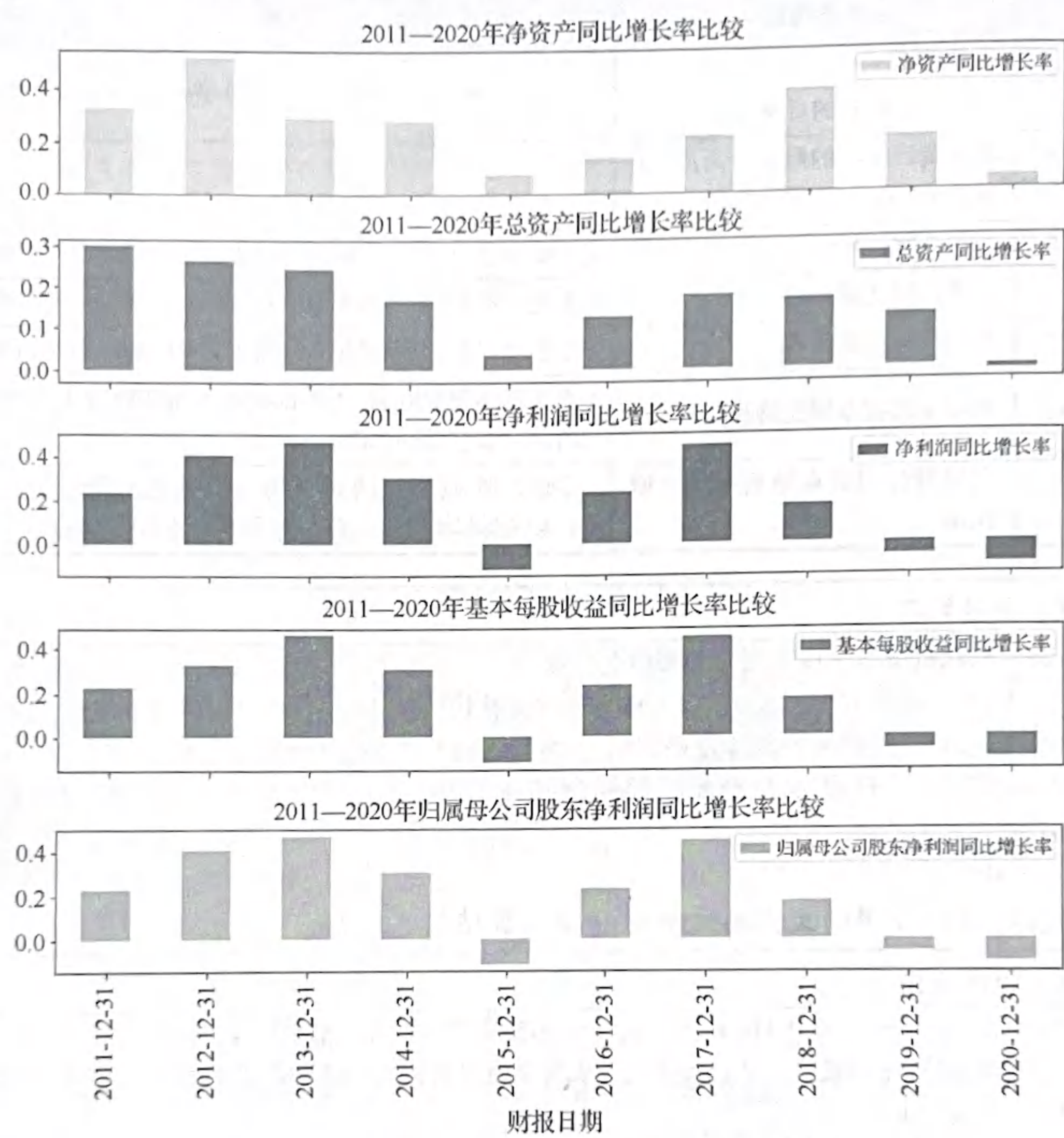


图 9-6 2011—2020 年成长能力多维数据分析

业务总结

在财务分析的入门阶段，分析的数据主要来源于财务报表、科目余额表、账务数据等，Excel 的函数和图表分析功能基本能够满足初级数据分析的需求，尤其是在数据量不大的时候，Excel 的操作便捷性可能优于 Python 的 pandas 模块等。随着数据可视化需求的增加，以 Power BI 为代表的商业智能软件，成了图表分析的利器。Python 则依靠功能强大的模块，比如 matplotlib 和 pyecharts，增加可视化图表的灵活性，可以更好地满足特殊的需求。业财融合将交易产生的海量业务数据、非财务信息同财务数据融合在一起，这时候就需要进行复杂的数据抽取、模型构建、数据挖掘等，Excel 很难满足这样的应用需求，而用 Python 读取庞大数据、处理加工高阶数据的优势和效率便体现出来。将得到的数据分析结果应用于管理决策中，才是数据分析的真正意义所在。

任务二 上市公司行业数据可视化分析与应用

企业业务、财务、税务数据中蕴含大量关于企业经营、管理和运行情况的信息。通过行业大数据分析，或者拉长数据分析的时间线，可以更清晰地确定企业在行业中的市场地位，也可以及时发现企业的异常波动，避免进入数据陷阱。

业务场景 9-2 上市公司行业数据分析

近年来，计算机通信行业发展迅猛。为了更好地了解该行业的发展状况，我们从中商产业研究院官网获取了 PC、服务器及硬件企业列表，如表 9-2 所示，拟使用 Python 获取相关企业的财报信息，开展对计算机通信行业的综合分析和比较。

表 9-2 PC、服务器及硬件企业相关信息

序号	股票代码	股票名称	公司全称	上市日期	主营业务
1	000021	深科技	深圳长城开发科技股份有限公司	1994-02-02	致力于为全球客户提供技术研发、工艺设计、生产加工、采购管理、物流支持等电子产品制造服务
2	000066	中国长城	中国长城科技集团股份有限公司	1997-06-26	从事网络安全与信息化、高新电子、电源、园区与物业服务及其他业务
3	000938	紫光股份	紫光股份有限公司	1999-11-04	提供技术领先的网络、计算、存储、云计算、安全和智能终端等全栈 ICT 基础设施及服务
4	000977	浪潮信息	浪潮电子信息产业股份有限公司	2000-06-08	服务器等云计算基础设施产品的研发、生产、销售
5	002236	大华股份	浙江大华技术股份有限公司	2008-05-20	为城市、企业、家庭数字化转型提供一站式智慧物联服务与解决方案
6	002415	海康威视	杭州海康威视数字技术股份有限公司	2010-05-28	以视频技术为核心打造从研发、制造到营销的完整价值链
7	002577	雷柏科技	深圳雷柏科技股份有限公司	2011-04-28	消费电子产品的自主研发、设计及销售
8	300042	朗科科技	深圳市朗科科技股份有限公司	2010-01-08	存储产品研发、生产和销售
9	600100	同方股份	同方股份有限公司	1997-06-27	信息技术和节能环保两大主营业务
10	600601	ST 方科	方正科技集团股份有限公司	1990-12-19	生产和销售 PCB 产品,提供网络宽带接入服务、IT 系统集成及解决方案、PC 产品的 FA 业务等

代码实现

(1) 导入 pandas 模块、matplotlib 模块。

➤ 导入模块

```
1 import pandas as pd
2 pd.options.display.float_format = '{:.2f}'.format
3 import matplotlib.pyplot as plt
4 plt.rcParams['font.family'] = 'SimHei' # 设置中文字体为黑体
5 plt.rcParams['axes.unicode_minus'] = False # 中文状态下负号正常显示
```

(2) 查找行业数据，获得相应股票代码，即获取表 9-2 中所列示的股票代码。

➤ 爬取 PC、服务器及硬件行业上市企业的股票代码、股票名称，并以 DataFrame 展示

```
1 stockcodes=list()
2 company_names=list()
3 table=pd.read_html("https://s.askci.com/stock/a-ci0000001795-0/1")[0]
4 sr=table['股票代码']
5 company_names=company_names+table['股票名称'].tolist()
6 for i,v in sr.items(): # 对小于 6 位的股票代码前面补 0
7     digit_no=len(str(v))
8     s=""
```



```
9     for j in range(6-digit_no):
10         s=s+"0"
11         s=s+str(v)
12         stockcodes.append(s)
13 # 以 dataframe 形式来展示获取的股票代码和股票名称
14 ls=list()
15 ls.append(stockcodes)
16 ls.append(company_names)
17 df_company=pd.DataFrame(ls)
18 df_company
```

运行结果

	0	1	2	3	4	5	6	7	8	9
0	000021	000066	000938	000977	002236	002415	002577	300042	600100	600601
1	深科技	中国长城	紫光股份	浪潮信息	大华股份	海康威视	雷柏科技	朗科科技	同方股份	ST 方科

(3) 以海康威视（股票代码：002415）为例，我们可以从中商产业研究院网站查看海康威视的主要经济指标（见表 9-3）、主要盈利及营运能力分析（见表 9-4）、偿债能力分析（见表 9-5）、成本费用分析（见表 9-6）等数据信息。同时，该网站提供了三大报表和完整版的财务报告，便于对数据价值做进一步的挖掘。

表 9-3 海康威视主要经济指标 单位：亿元

年份	类别						
	营业收入	营业利润	利润总额	净利润	资产总计	负债合计	股东权益合计
2006	7.02	1.93	2.19	2.05	4.77	1.40	3.38
2007	11.85	3.43	3.90	3.65	8.72	3.27	5.45
2008	17.42	5.35	6.05	5.49	13.70	4.00	9.70
2009	21.02	6.58	7.66	7.06	21.97	7.00	14.97
2010	36.05	9.86	11.43	10.52	65.49	9.07	56.42
2011	52.32	14.72	17.27	14.82	83.17	15.03	68.14
2012	72.14	19.89	23.14	21.40	105.89	19.18	86.71
2013	107.46	29.47	33.86	30.77	140.72	29.59	111.13
2014	172.33	43.78	52.06	46.81	212.91	64.11	148.79
2015	252.71	54.94	67.50	58.82	303.16	110.19	192.97
2016	319.35	68.33	83.14	74.24	413.48	168.70	244.79
2017	419.05	104.43	104.87	93.78	515.71	209.67	306.04
2018	498.37	123.34	124.37	113.80	634.92	255.29	379.63
2019	576.58	137.08	137.55	124.65	753.58	298.85	454.73
2020	635.03	151.97	152.73	136.78	887.02	342.22	544.80

表 9-4 海康威视主要盈利及营运能力分析^①

年份	类别						
	盈利能力指标				营运能力指标		
	销售毛利率/%	营业利润率/%	总资产利润率/%	净资产收益率/%	存货周转率/次	应收账款周转率/次	总资产周转率/次
2006	27.64	31.2	45.91	—	2.98	8.18	1.47
2007	29.03	32.91	44.72	80.76	3.42	13.1	1.36
2008	30.83	34.73	44.16	76.27	3.87	15.31	1.27

^① 注：中商产业研究院网站展示盈利能力数据时，实际包含了存货周转率、应收账款周转率、总资产周转率三个反映营运能力的指标。本书按照财务管理理论，将标题调整为“盈利及营运能力”，下同。

续表

年份	类别						
	盈利能力指标				营运能力指标		
	销售毛利率/%	营业利润率/%	总资产利润率/%	净资产收益率/%	存货周转率/次	应收账款周转率/次	总资产周转率/次
2009	31.4	36.44	34.87	60.18	3.29	10.97	0.96
2010	27.24	31.71	17.45	27.35	3.42	9.21	0.55
2011	28.12	33.01	20.76	23.98	3.78	7.07	0.63
2012	27.59	32.08	21.85	27.70	4.1	5.67	0.68
2013	27.06	31.51	24.06	30.92	4.6	4.8	0.76
2014	24.51	30.21	24.45	36.27	5.15	4.81	0.81
2015	21.21	26.71	22.27	35.28	5.92	4.07	0.83
2016	21.4	26.03	20.11	34.56	5.61	3.3	0.77
2017	20.72	25.03	20.34	34.96	5.35	3.23	0.81
2018	20.43	24.96	19.59	33.99	5.15	3.18	0.78
2019	20.39	23.86	18.25	30.53	3.67	3.04	0.77
2020	19.9	24.05	17.22	27.72	2.99	2.93	0.72

表 9-5 海康威视偿债能力分析

年份	类别			
	资产负债率/%	股东权益比率/%	流动比率	速动比率
2006	29.35	70.86	3.21	2.19
2007	37.50	62.50	2.49	1.81
2008	29.20	70.80	3.22	2.66
2009	31.86	68.14	2.84	2.22
2010	13.85	86.15	6.69	5.95
2011	18.07	81.93	5.00	4.43
2012	18.11	81.89	4.65	4.00
2013	21.03	78.97	4.39	3.01
2014	30.11	69.88	3.13	2.23
2015	36.35	63.65	2.60	2.10
2016	40.80	59.20	3.01	2.33
2017	40.66	59.34	2.60	2.06
2018	40.21	59.79	2.17	1.88
2019	39.66	60.34	2.72	2.18
2020	38.58	61.42	2.39	1.97

表 9-6 海康威视成本费用分析 单位：元

年份	类别			
	营业成本	销售费用	管理费用	财务费用
2006	5.08 亿	4 786.64 万	5843.22 万	-39.29 万
2007	8.41 亿	1.18 亿	9704.06 万	-320.80 万
2008	12.05 亿	1.91 亿	1.45 亿	-550.09 万
2009	14.42 亿	2.27 亿	2.01 亿	-910.71 万
2010	26.23 亿	4.65 亿	3.73 亿	-1 900.88 万
2011	37.61 亿	6.26 亿	5.03 亿	-7 928.81 万
2012	52.24 亿	7.32 亿	7.71 亿	-7 172.42 万
2013	78.38 亿	9.27 亿	10.96 亿	-8 655.79 万
2014	130.10 亿	15.33 亿	16.46 亿	-8 202.94 万

续表

年份	类别			
	营业成本	销售费用	管理费用	财务费用
2015	199.10 亿	21.79 亿	22.11 亿	-1.53 亿
2016	251.01 亿	29.91 亿	31.09 亿	-2.25 亿
2017	332.24 亿	44.30 亿	10.11 亿	2.65 亿
2018	396.57 亿	58.93 亿	13.77 亿	-4.24 亿
2019	459.00 亿	72.57 亿	18.22 亿	-6.40 亿
2020	508.66 亿	73.78 亿	17.90 亿	3.96 亿

(4) 观察海康威视财报分析(年度)数据页面的网址 <https://s.askci.com/stock/financialanalysis> 可以发现,网址的最后一部分为海康威视的股票代码。因此,修改股票代码就能获取其他企业的数据。在第(2)步,我们已经获得了 PC、服务器及硬件行业 10 家企业的股票代码和名称,只需循环访问便可获得相关数据。下面以采集盈利及营运能力数据为例,获取该行业内 10 家企业 2020 年的数据,将其存放到新的 dataframe 即 df_profit 中。

➤ 获取行业内 10 家企业 2020 年的盈利及营运能力数据

```
1 df_profit=pd.DataFrame()
2 columns=list()
3 for v in stockcodes:
4     tables=pd.read_html("https://s.askci.com/stock/financialanalysis/"+v+"/",
header=None, index_col=None)
5     df=tables[1]
6     if len(columns)==0:
7         columns=df.iloc[0].tolist()
8     sr=df[df.iloc[:,0]=='2020'] # 获取 2020 年的盈利及营运能力数据
9     sr=sr.iloc[sr.shape[0]-1] # 转化为 series
10    df_profit.insert(df_profit.shape[1],v,sr) # 把 10 家企业 2020 年的盈利及营运数
据组合在一起 df_profit
11 df_profit=df_profit.T
12 df_profit.columns = columns
13 df_profit.index =company_names
14 df_profit
```

运行结果

类别\年份	销售毛利率 (%)	营业利润率 (%)	总资产利润率 (%)	净资产收益率 (%)	存货周转率 (次)	应收账款周转率 (次)	总资产周转率 (次)
深科技 2020	3.19	7.48	5.18	11.83	5.73	6.29	0.69
中国长城 2020	3.35	7.23	3.79	10.73	2.42	4.30	0.52
紫光股份 2020	5.61	6.36	6.52	6.54	5.48	6.81	1.03
浪潮信息 2020	2.17	2.67	4.40	12.06	5.71	6.91	1.64
大华股份 2020	7.79	16.05	11.61	22.09	3.46	2.03	0.72
海康威视 2020	19.90	24.05	17.22	27.72	2.99	2.93	0.72
雷柏科技 2020	7.56	12.13	4.64	4.51	6.08	4.38	0.38
朗科科技 2020	4.09	5.61	7.09	6.84	6.68	12.48	1.26
同方股份 2020	-4.12	1.47	0.61	0.63	2.22	4.01	0.42
ST 方科 2020	-14.57	-14.36	-9.55	-70.25	2.89	4.51	0.66

说明

表中“类别\年份”为网站数据抓取结果，这种表述不规范，但为保证代码运行的准确性，保留这一列名，下同。建议读者在学习过程中，更关注利用 Python 解决复杂财务问题的思路和方法。

查看字段列表

1 columns

运行结果

```
[ '类别\\年份',  
  '销售毛利率 (%)',  
  '营业利润率 (%)',  
  '总资产利润率 (%)',  
  '净资产收益率 (%)',  
  '存货周转率 (次)',  
  '应收账款周转率 (次)',  
  '总资产周转率 (次)']
```

数据清洗，替换掉百分号，并选择部分指标进行比较

```
1 dt_pro=df_profit.replace({ '%': '' }, regex = True ) # 替换掉百分号  
2 dt_pro=dt_pro.iloc[:,2:].astype(float) # 转换成 float  
3 dt_pro.columns=['营业利润率', '总资产利润率', '净资产收益率', '存货周转率', '应收账款周  
转率', '总资产周转率']  
4 dt_pro
```

运行结果

	营业利润率	总资产利润率	净资产收益率	存货周转率	应收账款周转率	总资产周转率
深科技	7.48	5.18	11.83	5.73	6.29	0.69
中国长城	7.23	3.79	10.73	2.42	4.30	0.52
紫光股份	6.36	6.52	6.54	5.48	6.81	1.03
浪潮信息	2.67	4.40	12.06	5.71	6.91	1.64
大华股份	16.05	11.61	22.09	3.46	2.03	0.72
海康威视	24.05	17.22	27.72	2.99	2.93	0.72
雷柏科技	12.13	4.64	4.51	6.08	4.38	0.38
朗科科技	5.61	7.09	6.84	6.68	12.48	1.26
同方股份	1.47	0.61	0.63	2.22	4.01	0.42
ST 方科	-14.36	-9.55	-70.25	2.89	4.51	0.66

(5) 为了解行业基本布局和市场竞争情况，可绘制折线图，通过对比盈利及营运能力指标了解行业概况。

对比行业内 10 家企业 2020 年度的各项盈利及营运指标

```
1 plt.figure(figsize=(15,5))  
2 plt.rcParams['font.size'] = 14 # 全局 14 号字体  
3 for i in range(6):  
4     sr=dt_pro[dt_pro.columns[i]]  
5     ls=list()  
6     for idx,v in sr.items():  
7         ls.append(float(v)) # 将“年度”以后的列都从 string 类型转化成 float 类型
```



```
8 plt.plot(dt_pro.index, ls, '-', label=columns[1:][i])
9 plt.xlabel(u"盈利及营运指标") # x 轴标签
10 plt.ylabel(u"数值") # y 轴标签
11 plt.title(u"2020 年度行业盈利及营运能力比较") # 标题
12 plt.legend(loc=4) # 图例在右下角
13 plt.show()
```

运行结果 (见图 9-7)

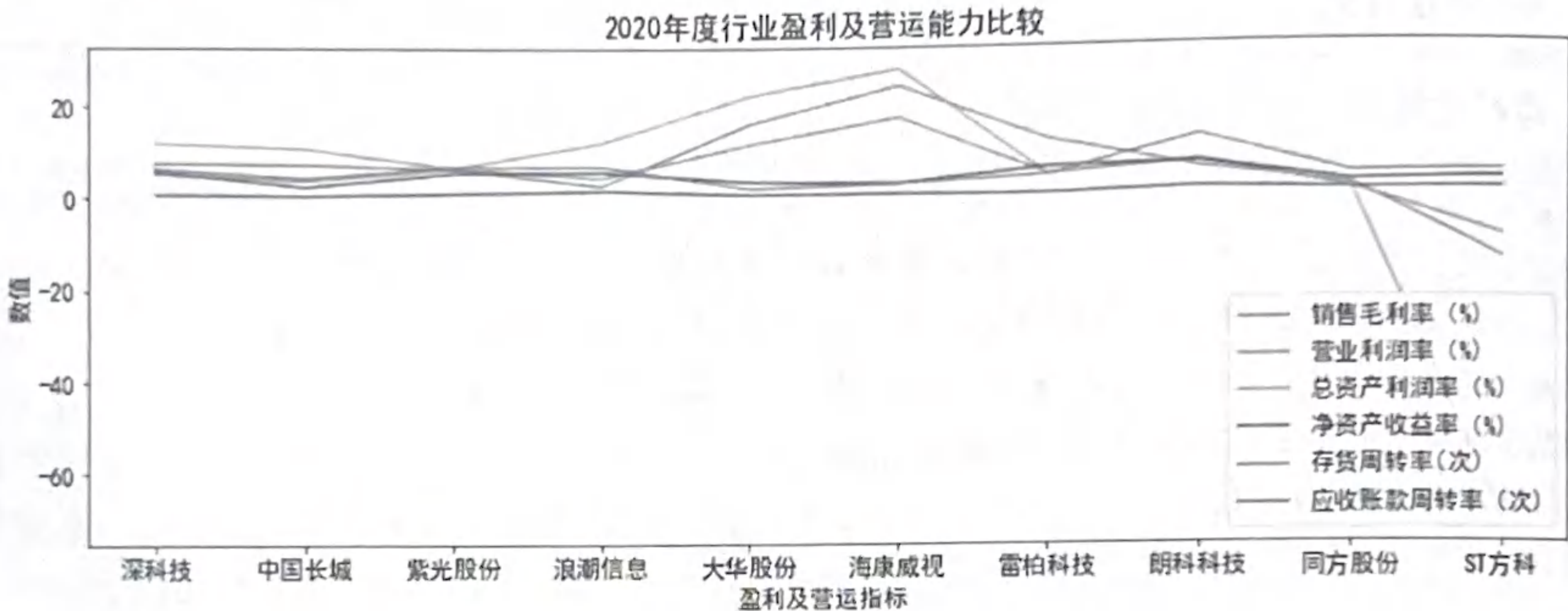


图 9-7 2020 年度行业盈利及营运数据比较

从盈利及营运能力上看，在 PC、服务器及硬件行业的 10 家上市企业中，海康威视和大华股份的盈利及营运能力表现良好。

(6) 为了更深入地分析行业竞争状况，选取所有企业各项盈利及营运能力指标的最优值，作为行业标杆参照数据，比较盈利及营运能力突出的海康威视、大华股份与行业标杆之间的差距。

由于抓取的网页数据是文本格式的，且可能存在缺失值和异常值，因此需要先对其进行清洗和加工。对所有企业的盈利及营运数据进行清洗，将百分号去除，方便其从字符串转化为浮点数进行运算。获取不同盈利及营运指标的最优值 max_value(盈利及营运指标中的最大值)，作为行业标杆参照数据。

➤ 使用 describe()函数获取行业盈利及营运数据的最小值、均值、最大值等信息

```
1 dt_pro_st=dt_pro.describe() # 描述统计
2 dt_pro_st
```

运行结果

	营业利润率	总资产利润率	净资产收益率	存货周转率	应收账款周转率	总资产周转率
count	10.00	10.00	10.00	10.00	10.00	10.00
mean	6.87	5.15	3.27	4.37	5.46	0.80
std	10.04	6.93	27.07	1.72	2.94	0.40
min	-14.36	-9.55	-70.25	2.22	2.03	0.38
25%	3.41	3.94	5.02	2.92	4.08	0.56
50%	6.79	4.91	8.79	4.47	4.45	0.70
75%	10.97	6.95	12.00	5.73	6.68	0.95
max	24.05	17.22	27.72	6.68	12.48	1.64

➤ 读取大华股份和海康威视的各项盈利能力指标、营运能力指标数据及相关行业数据

```
1 pro_st=dt_pro_st.loc[["mean","50%","max"],:]
2 pro_compare=pd.concat([pro_st,dt_pro.loc["大华股份":"海康威视",]])
3 pro_compare
```


运行结果

	营业利润率	总资产利润率	净资产收益率	存货周转率	应收账款周转率	总资产周转率
mean	6.87	5.15	3.27	4.37	5.46	0.80
50%	6.79	4.91	8.79	4.47	4.45	0.70
max	24.05	17.22	27.72	6.68	12.48	1.64
大华股份	16.05	11.61	22.09	3.46	2.03	0.72
海康威视	24.05	17.22	27.72	2.99	2.93	0.72

➤ 提取上述返回结果中各项数据的最大值

```
1 max_value=pro_compare.loc['max'].tolist()
2 max_value.append(max_value[0]) # 将第一项数据复制到最后, 为绘制雷达图做准备
3 max_value
```

运行结果

[24.05, 17.22, 27.72, 6.68, 12.48, 1.64, 24.05]

(7) 得到行业标杆参照数据后, 和前述操作一样, 修改访问页面网址的股票代码部分, 分别获取两家企业 2020 年度的盈利及营运指标数据, 存入列表 ls_index 中。

➤ 将大华股份和海康威视的盈利及营运指标数据按照前述方法处理

```
1 ls_index=list()
2 dahua=pro_compare.loc['大华股份'].tolist()
3 dahua.append(dahua[0])
4 ls_index.append(dahua)
5 haikang=pro_compare.loc['海康威视'].tolist()
6 haikang.append(haikang[0])
7 ls_index.append(haikang)
8 ls_index
```

运行结果

[[16.05, 11.61, 22.09, 3.46, 2.03, 0.72, 16.05],
[24.05, 17.22, 27.72, 2.99, 2.93, 0.72, 24.05]]

➤ 提取相关字段

```
1 labels=columns[2:8]
2 labels.append(columns[2]) # 将第一项数据复制到最后, 为绘制雷达图做准备
3 labels
```

运行结果

['营业利润率(%)',
'总资产利润率(%)',
'净资产收益率(%)',
'存货周转率(次)',
'应收账款周转率(次)',
'总资产周转率(次)',
'营业利润率(%)']

➤ 准备绘图数据

```
1 unt=2*3.1415926/6 # 7个点(闭合图形), 共6端
2 theta=list()
3 angle=0
4 for i in range(7):
```



```
5     angle=i*unt
6     theta.append(angle) # 将间隔角度列表赋值给 theta
7     rate_company=list() # 存放两家企业参考标杆的百分比数据
8     for idx,ls in enumerate(ls_index):
9         r=list()
10        for i,v in enumerate(max_value):
11            ratio=float(ls[i])/float(v)*100
12            r.append(ratio)
13        rate_company.append(r)
14    rate_company # 以 DataFrame 格式展示列表
15    df_ratio=pd.DataFrame(rate_company,columns=labels,index=['大华股份','海康威视'])
16    df_ratio
```

运行结果

	营业利润 率 (%)	总资产利润率 (%)	净资产收益率 (%)	存货 周转率 (次)	应收账款周 转率 (次)	总资产周转 率 (次)	营业利润率 (%)
大华股份	66.74	67.42	79.69	51.80	16.27	43.90	66.74
海康威视	100.00	100.00	100.00	44.76	23.48	43.90	100.00

➤ 绘制雷达图

```
1    ls_company=[['002236','大华股份','b'],['002415','海康威视','r']] #大华股份以蓝色展
示, 海康威视以红色展示
2    for idx,v in enumerate(ls_company):
3        r=rate_company[idx]
4        plt.polar(theta,r,v[2]+"-",lw=2)
5        plt.ylim(0,100)
6        ls=list()
7        for vv in theta:
8            ls.append(vv*180/3.1415926)
9        plt.thetagrids(ls, labels) # 生成标签
10       plt.fill(theta, r, facecolor=v[2], alpha=0.25) #填充
11       ttl=v[1]
12       plt.title(ttl, x=-0.2) # 设置标题, 且标题位置在左上角
13       plt.show()
```

运行结果 (见图 9-8)

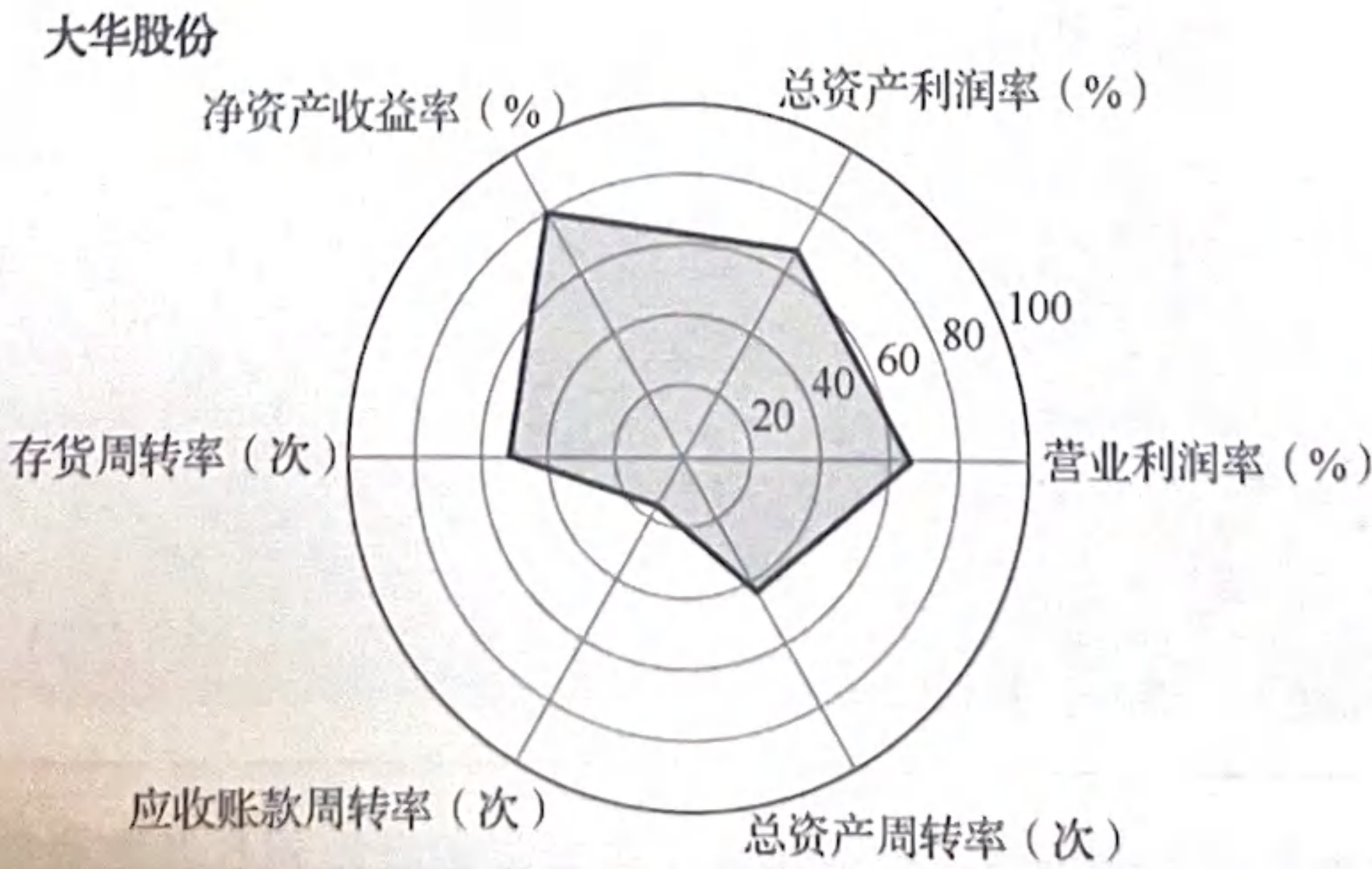


图 9-8 2020 年大华股份和海康威视的盈利及营运能力表现

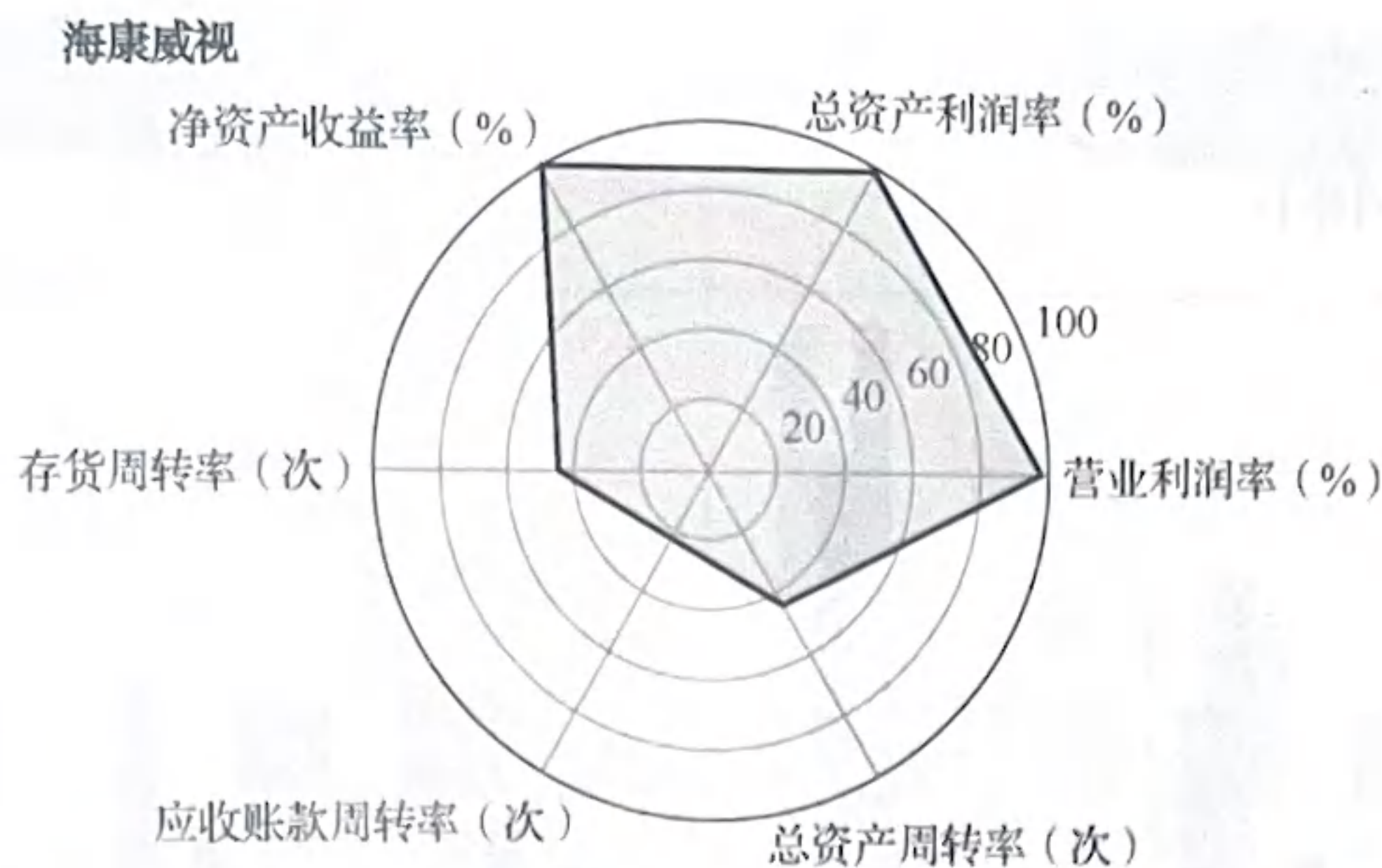


图 9-8 2020 年大华股份和海康威视的盈利及营运能力表现 (续)

(8) 按照以净资产收益率为核心的杜邦分析体系, 从图 9-8 中可以看出, 海康威视的盈利及营运能力表现强劲, 大华股份虽与其同处行业前列, 但总体表现弱于海康威视。为了更好地展示两者的对比情况, 可以将雷达图绘制在同一画布上。

绘制大华股份 vs 海康威视盈利及营运能力对比雷达图

```
1 style=["+", "*"]
2 for idx,v in enumerate(ls_company):
3     r=rate_company[idx]
4     plt.polar(theta,r,v[2]+style[idx],lw=2,label=v[1])
5     plt.ylim(0,100)
6     plt.fill(theta, r, facecolor=v[2], alpha=0.25) # 填充
7 plt.thetagrids(ls, labels)
8 plt.title("大华 VS 海康盈利及营运能力对比",y=-0.3) # 设置标题及其位置
9 plt.legend(bbox_to_anchor=(1.25, 0), loc=3)
10 plt.show()
```

运行结果 (见图 9-9)

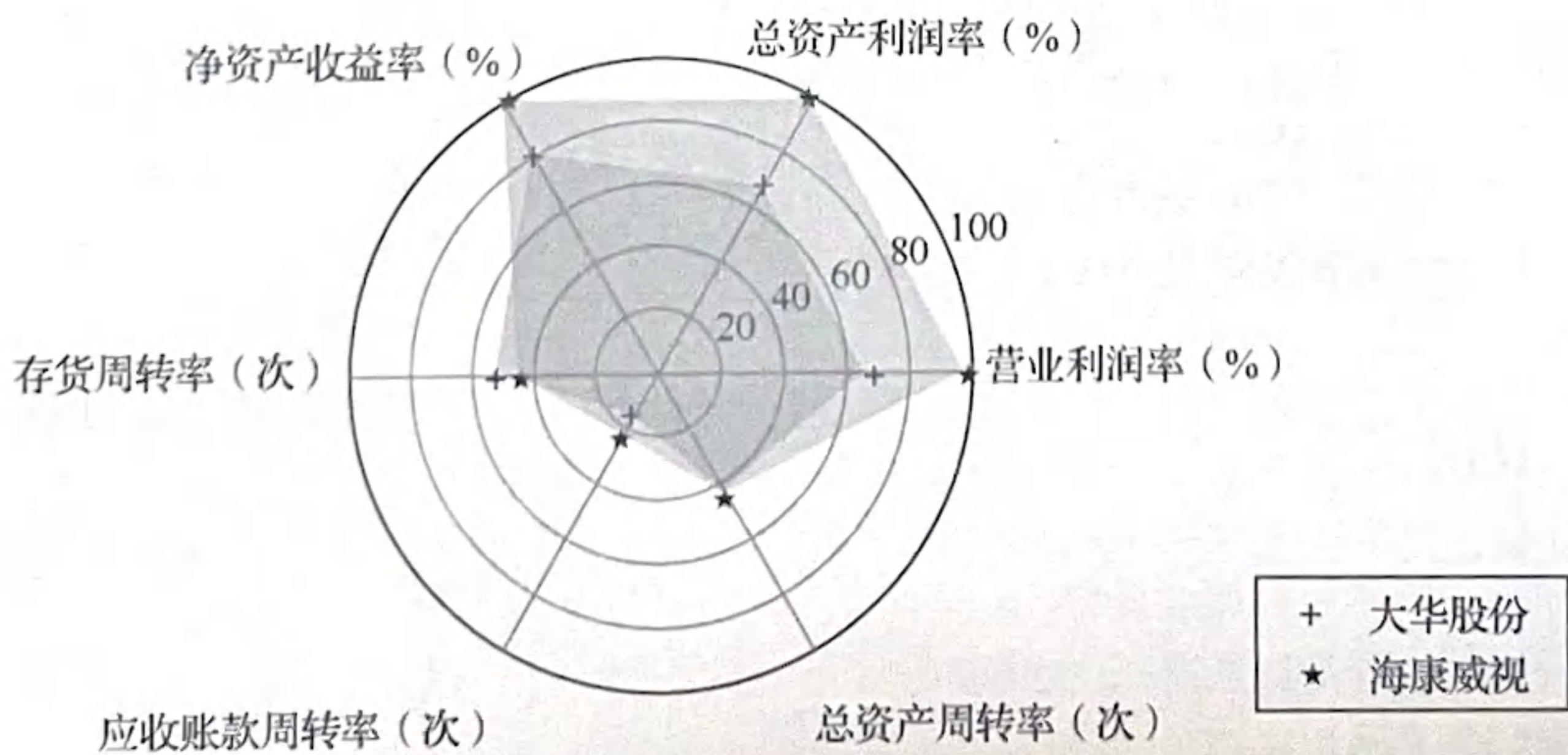


图 9-9 2020 年大华股份 vs 与海康威视盈利及营运能力对比

(9) 将重要财务指标及描述统计数据以柱形图显示。

大华股份 vs 海康威视财务指标对比

```
1 pro_compare_T=pro_compare.T
2 pro_compare_T["2020 年财务指标"]=pro_compare_T.index
3 pro_compare_T
4 pro_compare_T.plot("2020 年财务指标",['mean', '50%', 'max', '大华股份', '海康威视'],
```



```
kind="bar",figsize=(16,5))
5 plt.show()
```

运行结果（见图 9-10）

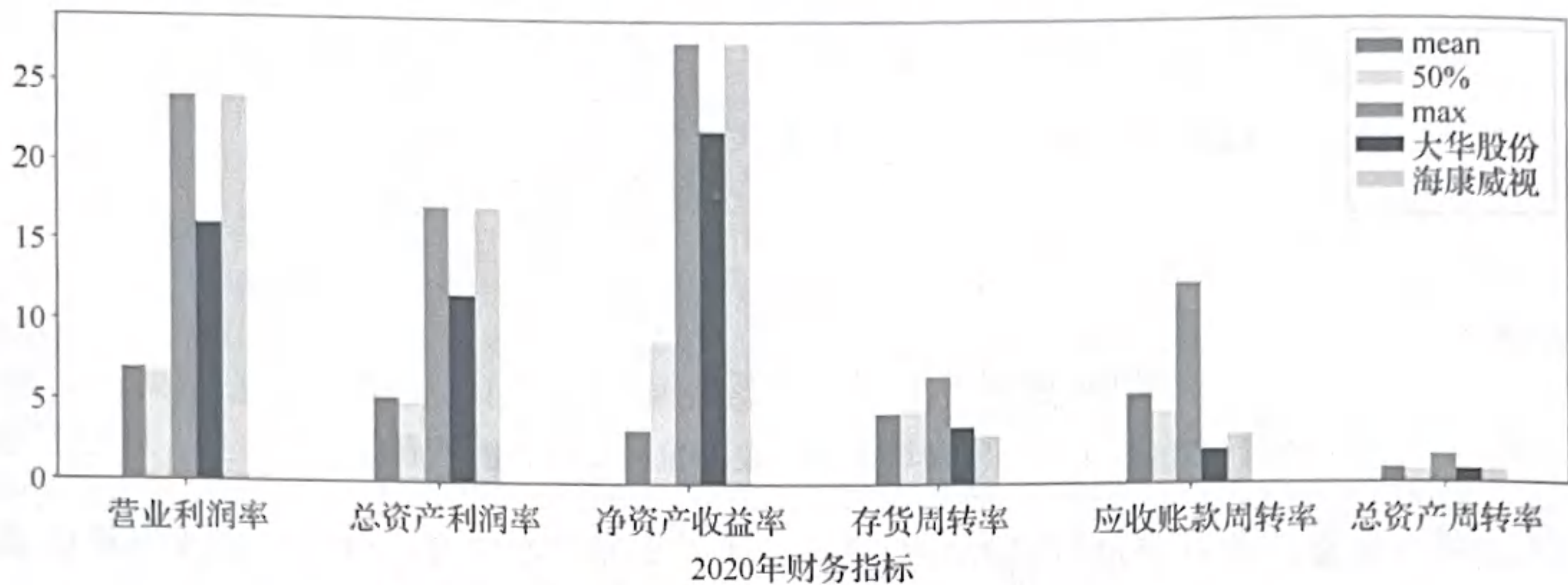


图 9-10 2020 年大华股份 vs 海康威视财务指标对比

（10）将海康威视 2020 年度的财务指标数据与行业均值、行业中位数、大华股份及行业最大值进行对比。

```
> 将海康威视的财务指标数据与行业均值、行业中位数、大华股份及行业最大值进行对比
1 plt.figure(figsize=(25,15))
2 plt.rcParams['font.size'] = 14 # 全局 14 号字体
3 plt.subplot(2,2,1)
4 plt.bar(x=dt_pro_st.columns,height=dt_pro_st.loc["mean",:],label="行业均值",
width = 0.6, alpha = 0.7,color="b" )
5 plt.plot(pro_compare.columns,pro_compare.loc["海康威视",:], "r",linestyle = '--',
marker = '*', label = '海康威视')
6 plt.title("2020 年海康威视 vs 行业均值")
7 plt.subplot(2,2,2)
8 plt.bar(x=dt_pro_st.columns,height=dt_pro_st.loc["50%",:],label="行业中位数",
width = 0.6, alpha = 0.7 ,color="g")
9 plt.plot(pro_compare.columns,pro_compare.loc["海康威视",:], "r",linestyle = '--',
marker = '*', label = '海康威视')
10 plt.title("2020 年海康威视 vs 行业中位数")
11 plt.subplot(2,2,3)
12 plt.bar(x=dt_pro_st.columns,height=pro_compare.loc["大华股份",:],label="大华股份",
width = 0.6, alpha = 0.9 )
13 plt.plot(pro_compare.columns,pro_compare.loc["海康威视",:], "r",linestyle = '--',
marker = '*', label = '海康威视')
14 plt.title("2020 年海康威视 vs 大华股份")
15 plt.subplot(2,2,4)
16 plt.bar(x=dt_pro_st.columns,height=pro_compare.loc["max",:],label="行业最大值",
width = 0.6, alpha = 0.9 ,color="y")
17 plt.plot(pro_compare.columns,pro_compare.loc["海康威视",:], "r",linestyle = '--',
marker = '*', label = '海康威视')
18 plt.title("2020 年海康威视 vs 行业最大值")
19 plt.show()
```


运行结果（见图 9-11）

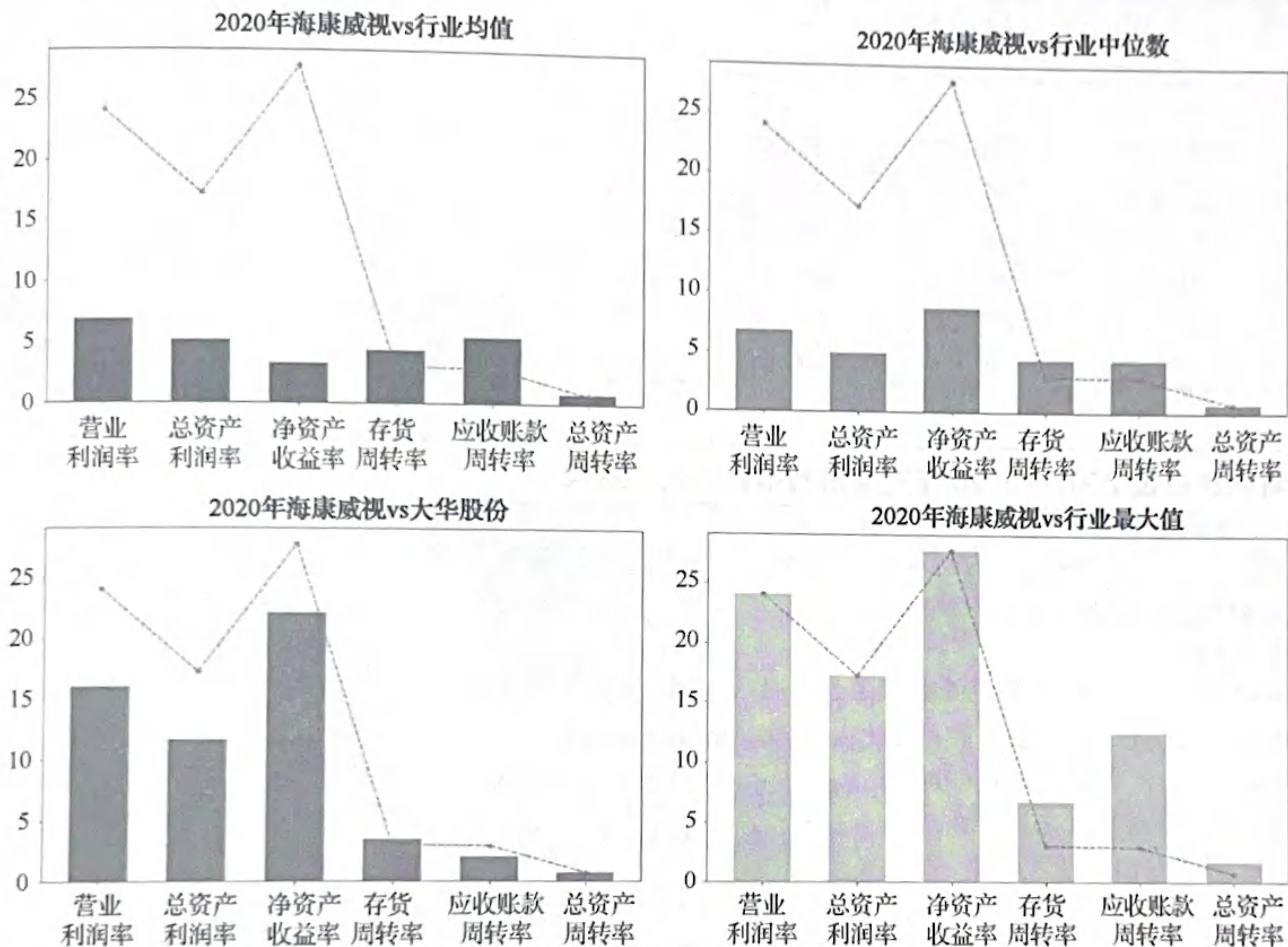


图 9-11 2020 年度海康威视的各项财务指标与行业均值、行业中位数、大华股份及行业最大值的比较

（11）结合杜邦分析法的财务指标体系做进一步分析。相关计算公式如下。

- ① 净资产收益率=资产净利率×权益乘数
=（净利润/总资产）×（总资产/总权益资本）×100%
- ② 资产净利率=销售净利率×资产周转率
=（净利润/营业总收入）×（营业总收入/总资产）×100%

由图 9-11 可知，海康威视的多项财务指标处于行业标杆水平。根据以上公式可以发现，海康威视的盈利及营运贡献主要来自其较高的销售净利率。下面我们对海康威视做进一步分析，查看其 2007—2020 年的净资产收益率、总资产利润率、营业利润率的发展曲线。

```
> 获取数据
1 tables=pd.read_html("https://s.askci.com/stock/financialanalysis/002415/",
header=0)
2 df=tables[1]
3 df=df[df['类别\年份']>2006]
4 df
```

运行结果

类别\年份	销售毛利率 (%)	营业利润率 (%)	总资产利润率 (%)	净资产收益率 (%)	存货周转率 (次)	应收账款周转率 (次)	总资产周转率 (次)
1 2007	29.03	32.91	44.72	80.76	3.42	13.10	1.36
2 2008	30.83	34.73	44.16	76.27	3.87	15.31	1.27
3 2009	31.40	36.44	34.87	60.18	3.29	10.97	0.96
4 2010	27.24	31.71	17.45	27.35	3.42	9.21	0.55
5 2011	28.12	33.01	20.76	23.98	3.78	7.07	0.63

6	2012	27.59	32.08	21.85	27.70	4.10	5.67	0.68
7	2013	27.06	31.51	24.06	30.92	4.60	4.80	0.76
8	2014	24.51	30.21	24.45	36.27	5.15	4.81	0.81
9	2015	21.21	26.71	22.27	35.28	5.92	4.07	0.83
10	2016	21.40	26.03	20.11	34.56	5.61	3.30	0.77
11	2017	20.72	25.03	20.34	34.96	5.35	3.23	0.81
12	2018	20.43	24.96	19.59	33.99	5.15	3.18	0.78
13	2019	20.39	23.86	18.25	30.53	3.67	3.04	0.77
14	2020	19.90	24.05	17.22	27.72	2.99	2.93	0.72

➤ 查看海康威视 2007—2020 年主要盈利指标的发展趋势

```
1 plt.figure(figsize=(15,5))
2 plt.rcParams['font.size'] = 14 # 全局 14 号字体
3 # 抓取年份列表和主要盈利指标数据
4 year=list()
5 ROA=list() # 总资产利润率(Return On Assets)
6 ROE=list() # 净资产收益率(Return On Equity)
7 OPR=list() # 营业利润率(Operating Profit Ratio)
8 for idx,row in df.iterrows():
9     year.append(row[0])
10    val=0
11    if row[4]=='--':
12        val=0
13    else:
14        val=float(row[4].split("%")[0])
15    ROE.append(val)
16    OPR.append(float(row[2]))
17    ROA.append(float(row[3]))
18 plt.plot(year, ROE, marker='o', mec='b', mfc='b',color='b',label=u"净资产收益率")
19 plt.plot(year, OPR, marker='^', mec='y', mfc='y',color='y',label=u"营业利润率")
20 plt.plot(year, ROA, marker='*', mec='r', mfc='r',color='r',label=u"总资产利润率")
21 plt.legend() # 让图例生效
22 plt.xlabel(u"年份") # x 轴标签
23 plt.ylabel(u"百分比") # y 轴标签
24 plt.title(u"海康威视 2007—2020 年主要盈利及营运能力表现") # 标题
25 plt.show()
```

运行结果（见图 9-12）

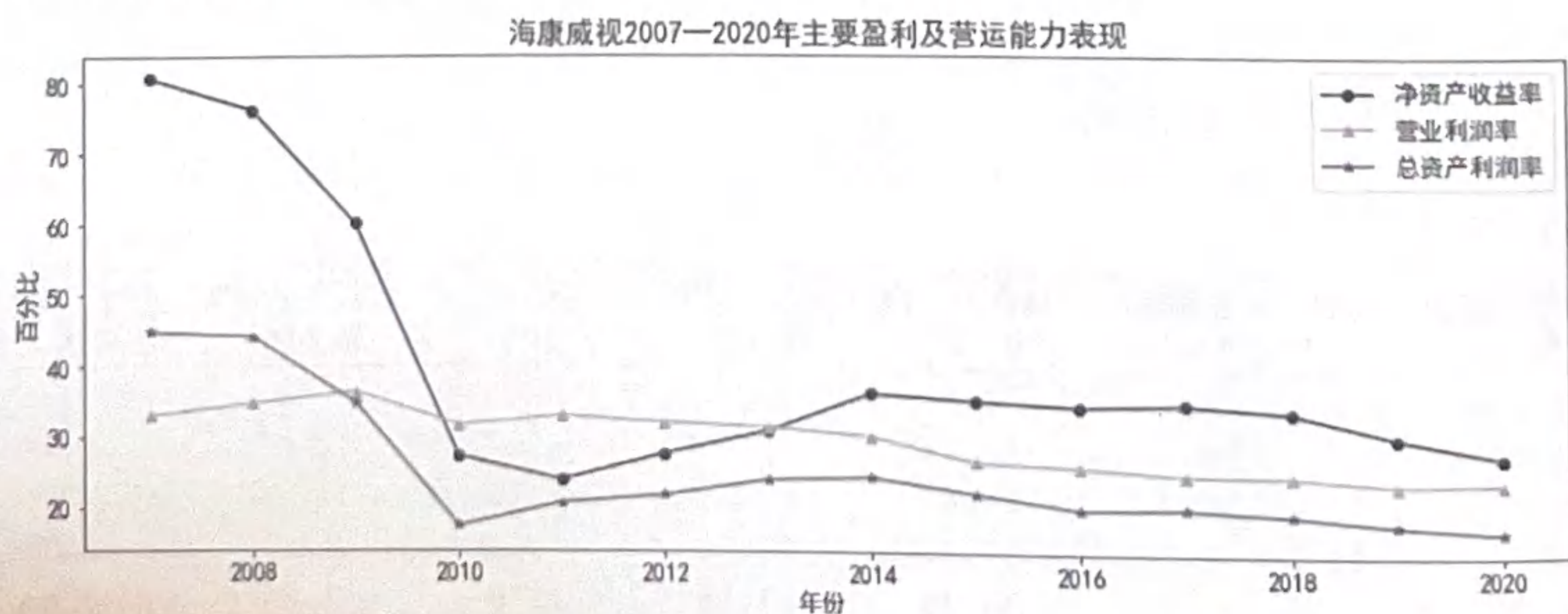


图 9-12 海康威视 2007—2020 年的主要盈利及营运能力

(12) 玫瑰图可以更为直观地展示数据的变化趋势, 且美观度优于基础图表。下面绘制玫瑰图, 展示海康威视 2006—2020 年营业利润率的变化情况。

➤ 获取从 2006 年到 2020 年营业利润率的数值对应关系

```
1 N=15 # 从 2006 年到 2020 年一共 15 年
2 year=list()
3 for i in range(N):
4     year.append(str(2006+i))
5 value=tables[1]["营业利润率(%)"].tolist()
6 data = {'year': year, 'value':value}
7 df = pd.DataFrame(data)
8 df.T
```

运行结果

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
year	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
value	31.20	32.91	34.73	36.44	31.71	33.01	32.08	31.51	30.21	26.71	26.03	25.03	24.96	23.86	24.05

➤ 调用自定义函数, 绘制玫瑰图

```
1 pi=3.1415926
2 angle=list()
3 width=2*pi/15
4 for i in range(15):
5     angle.append(i*width) # 每个扇形的起始角度
6 plt.figure(figsize=(15, 8)) # 创建画布
7 ax = plt.subplot(projection='polar')
8 # 删除不必要的内容
9 ax.set_ylim(-4, df['value'].max()) # 中间空白
10 ax.set_theta_zero_location('N') # 设置极坐标的起点(即 0 度)在正上方向
11 ax.grid(False) # 不显示极轴
12 ax.spines['polar'].set_visible(False) # 不显示极坐标最外的圆形
13 ax.set_yticks([]) # 不显示坐标间隔
14 ax.set_thetagrids([]) # 不显示极轴坐标
15 colors = ['blue', 'blueviolet', 'orange', 'brown', 'green', 'pink', 'turquoise',
16 'tomato']
17 # 绘画
18 ax.bar(angle, df['value'], width=width, color=colors, alpha=1)
19 for i in range(N):
20     ax.text(angle[i]+0.08, # 角度
21             df['value'][i], # 长度
22             df['year'][i], # 文本
23             rotation=angle[i] * 180 / pi, # 文字角度
24             rotation_mode='anchor', # this parameter is a trick
25             alpha=1,
26             fontweight='bold', size=18
27             )
28 plt.title("海康威视 2006-2020 年的营业利润率", loc = 'center')
29 plt.show()
```


运行结果（见图 9-13）

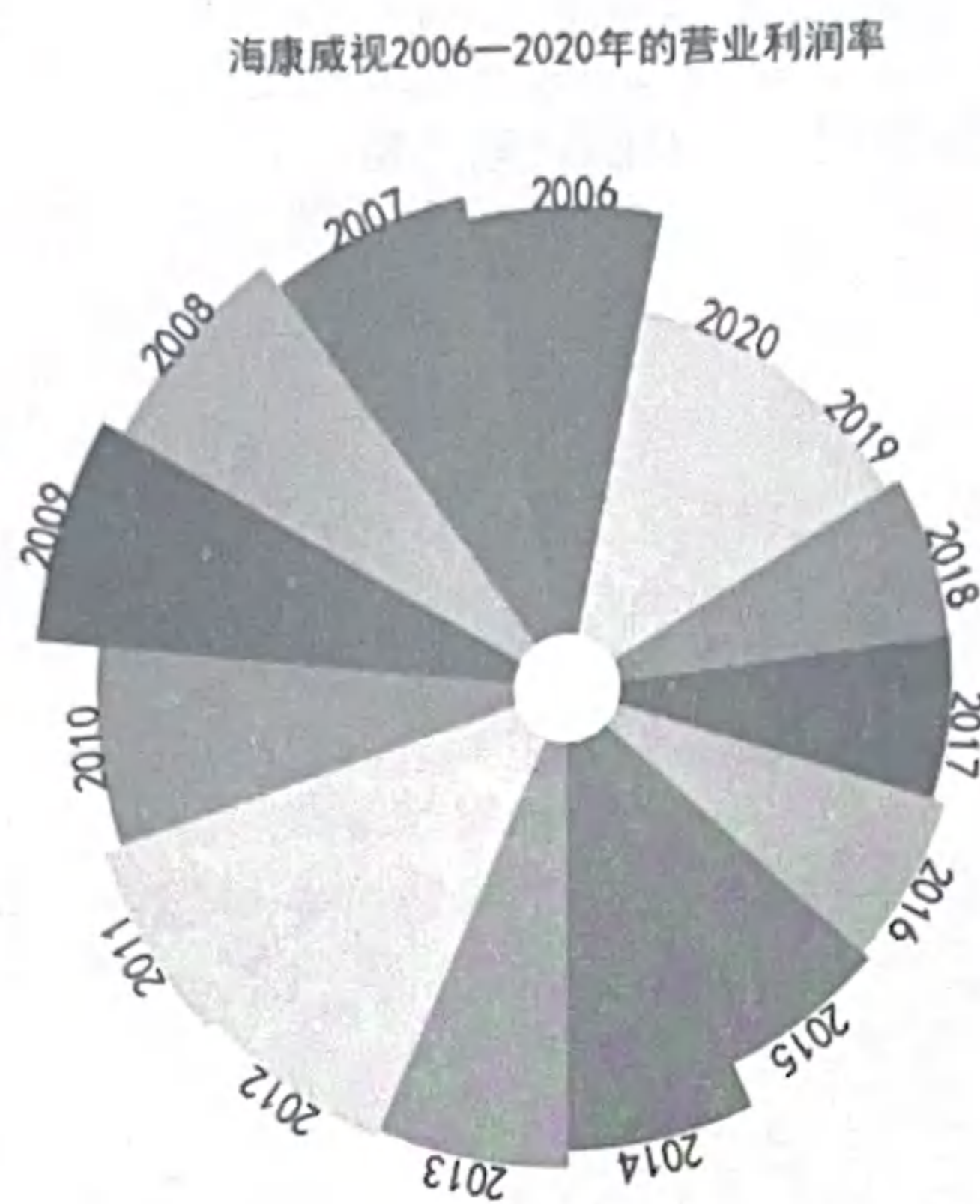


图 9-13 海康威视 2006—2020 年的营业利润率

（13）图 9-7 显示了 2020 年海康威视的盈利能力表现最佳，因此，在完成对海康威视历史数据的纵向比较后，下面对其 2020 年 4 个季度的表现进一步深入分析。通过中商产业研究院季度财务数据查询页面，可以看到海康威视 2019—2021 年的主要经济指标如表 9-7 所示。

表 9-7 海康威视 2019—2021 年的主要经济指标（按季度）

月份	类别						
	营业收入/亿元	营业收入 同比/%	营业利润/亿元	营业利润 同比/%	利润总额/亿元	利润总额 同比/%	净利润/亿元
202106	339.02	39.68	72.59	29.74	72.82	29.6	68.55
202103	139.88	48.35	29.00	56.59	29.10	55.53	24.35
202012	635.03	10.14	151.97	10.86	152.73	11.04	136.78
202009	420.21	5.48	96.88	2.59	97.27	2.51	85.77
202006	242.71	1.45	55.95	9.11	56.19	8.94	46.67
202003	94.29	-5.16	18.52	0.54	18.71	0.59	15.53
201912	576.58	15.69	137.08	11.14	137.55	10.6	124.65
201909	398.39	17.86	94.43	16.94	94.89	15.95	80.94
201906	239.23	14.6	51.28	8.99	51.58	8.07	42.24
201903	99.42	6.16	18.42	-14.13	18.60	-13.73	15.18

读取海康威视 2020 年各季度财务数据，对营业收入、营业利润、利润总额、净利润等数据进行分析。

➤ 读取海康威视 2020 年各季度财务数据

```
1 tables=pd.read_html("https://s.askci.com/stock/financialanalysis/002415/quarter/",
header=0)
2 df_quarter=tables[0]
3 df_quarter.set_index(['类别\年份'],inplace=True)
4 df_quarter=df_quarter.loc[[202012,202009,202006,202003]]
5 df_quarter
```


运行结果

类别\年份	营业收入 (元)	营业收入同 比(%)	营业利润 (元)	营业利润同 比(%)	利润总额 (元)	利润总额 同比(%)	净利润(元)
202012	635.03 亿	10.14	151.97 亿	10.86	152.73 亿	11.04	136.78 亿
202009	420.21 亿	5.48	96.88 亿	2.59	97.27 亿	2.51	85.77 亿
202006	242.71 亿	1.45	55.95 亿	9.11	56.19 亿	8.94	46.67 亿
202003	94.29 亿	-5.16	18.52 亿	0.54	18.71 亿	0.59	15.53 亿

➤ 提取营业收入、营业利润绝对值数据

```
1 df_quarter=df_quarter[['营业收入(元)','营业利润(元)']]
2 df_quarter
```

运行结果

类别\年份	营业收入 (元)	营业利润 (元)
202012	635.03 亿	151.97 亿
202009	420.21 亿	96.88 亿
202006	242.71 亿	55.95 亿
202003	94.29 亿	18.52 亿

(14) 将营业收入和营业利润数据分别绘制环形图，查看 2020 年各季度营业收入、营业利润的贡献情况。

➤ 绘制不同季度在不同指标的贡献图

```
1 # 查看不同指标下，海康威视公司在不同季度的财务情况
2 plt.figure(figsize=(15, 15))
3 plt.rcParams['font.size'] = 14
4 for i in range(2):
5     ax = plt.subplot(221+i)
6     # 获取以下几个维度的数据
7     labels=["第 4 季度","第 3 季度","第 2 季度","第 1 季度"]
8     sr=df_quarter[df_quarter.columns[i]]
9     x=list()
10    for v in sr.tolist():
11        value=float(v.split("亿")[0])
12        if value<0:
13            value=0    # 去除负数
14            x.append(value)
15    ax.pie(x=x,    # 绘图数据
16          labels = labels,    # 添加标签
17          autopct = '%.1f%%',    # 设置百分比的格式，这里保留一位小数
18          wedgeprops = {'width': 0.5, 'edgecolor':'green'},    # 边框显示绿色
19          )
20    ax.set_title("各季度贡献-"+df_quarter.columns[i])
21 plt.show()
```


运行结果（见图 9-14）

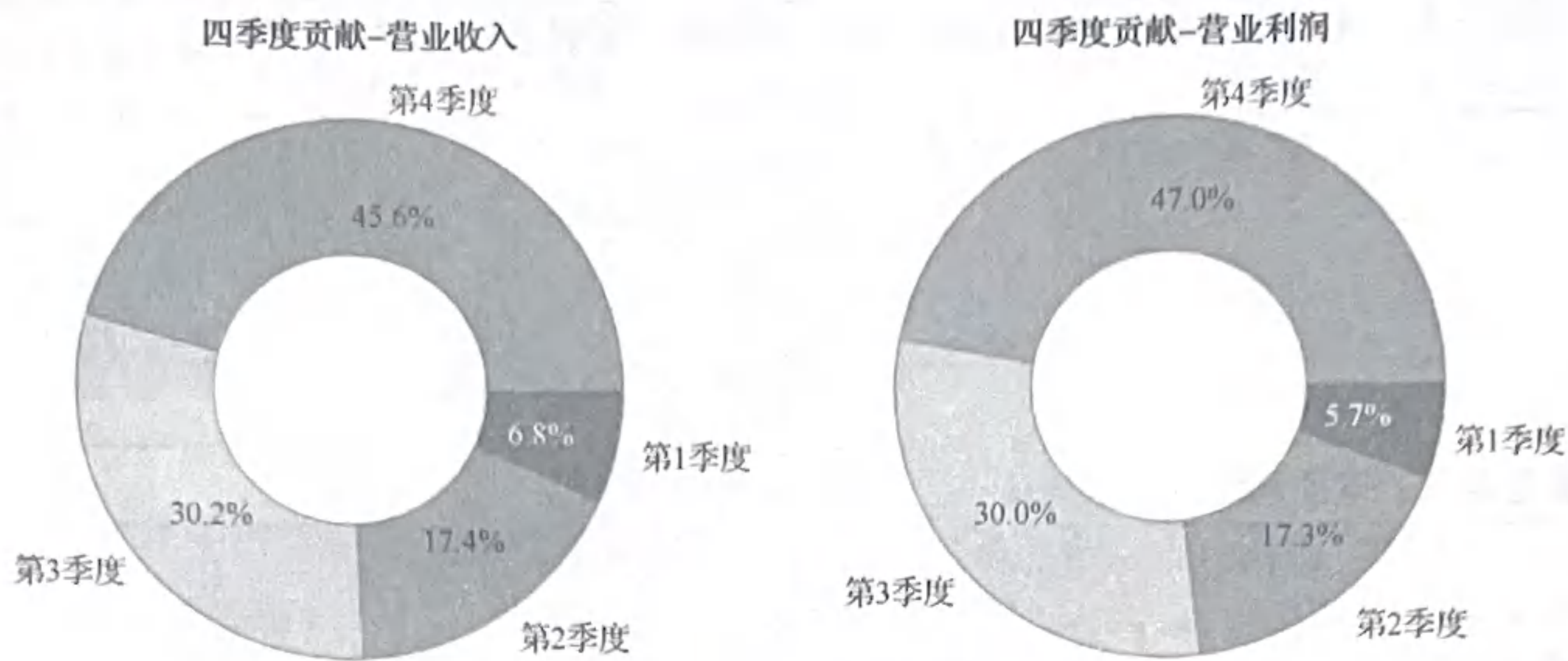


图 9-14 海康威视 2020 年各季度盈利贡献对比

由图 9-14 可以看出，海康威视 2020 年 4 个季度的营业收入和营业利润贡献存在显著差异。从营业收入指标看，4 个季度在稳步增长；从营业收入同比数据看，第 1 季度没有增长，但从第 3 季度起开始强劲反弹；营业利润指标与营业收入指标呈现相同的走势，在原材料普遍上涨的外部环境下，可以对其成本结构做更深入的分析；但在营业利润同比的贡献率上，第 3 季度不如第 2 季度，需要对其其他数据做更深入的分析 and 挖掘，找出原因。

（15）同样，可以分别获取季度盈利能力指标和偿债能力指标数据展开分析。

① 获取 2020 年各季度的盈利能力指标数据。

➤ 获取 2020 年各季度的主要盈利能力指标

```
1 plt.figure(figsize=(15,5))
2 plt.rcParams['font.size'] = 14 # 全局 14 号字体
3 df_profit=tables[1]
4 df_profit=df_profit[df_profit["类别\年份"]>=202003]
5 df_profit=df_profit[df_profit["类别\年份"]<=202012]
6 df_profit
```

运行结果

类别\年份	销售毛利率 (%)	营业利润率 (%)	总资产利润率 (%)	净资产收益率 (%)	存货周转率 (次)	应收账款周转率 (次)	总资产周转率 (次)
3 202012	19.90	24.05	17.22	27.72%	2.99	2.93	0.72
4 202009	18.64	23.15	12.28	18.12%	1.96	2.52	0.53
5 202006	18.42	23.15	7.81	9.75%	1.08	2.27	0.34
6 202003	16.77	19.84	2.56	3.28%	0.42	1.84	0.13

② 筛选出总资产利润率、净资产收益率、营业利润率，绘制主要盈利指标比较图。

➤ 绘制 2020 年各季度主要盈利能力指标表现图

```
1 # 抓取年份列表和主要盈利能力及营运能力指标数据
2 quarter=list()
3 q=0
4 ROA=list(0 for n in range(4)) # 总资产利润率(Return On Assets)
5 ROE=list(0 for n in range(4)) # 净资产收益率(Return On Equity)
6 OPR=list(0 for n in range(4)) # 营业利润率(Operating Profit Ratio)
7 for idx,row in df_profit.iterrows():
8     q+=1
```



```
9     quarter.append(str(q))
10     val=0
11     val=float(row[4].split("%")[0])
12     ROE[4-q]=val
13     OPR[4-q]=float(row[2])
14     ROA[4-q]=float(row[3])
15 plt.plot(quarter, ROE, marker='o', mec='b', mfc='b',color='b',label=u"净资产收
益率")
16 plt.plot(quarter, OPR, marker='^', mec='y', mfc='y',color='y',label=u"营业利润率")
17 plt.plot(quarter, ROA, marker='*', mec='r', mfc='r',color='r',label=u"总资产利润率")
18 plt.legend()      # 让图例生效
19 plt.xlabel(u"季度")      # X 轴标签
20 plt.ylabel(u"百分比")      # Y 轴标签
21 plt.title(u"海康威视 2020 年各季度主要盈利指标表现")      # 标题
22 plt.show()
```

运行结果（见图 9-15）

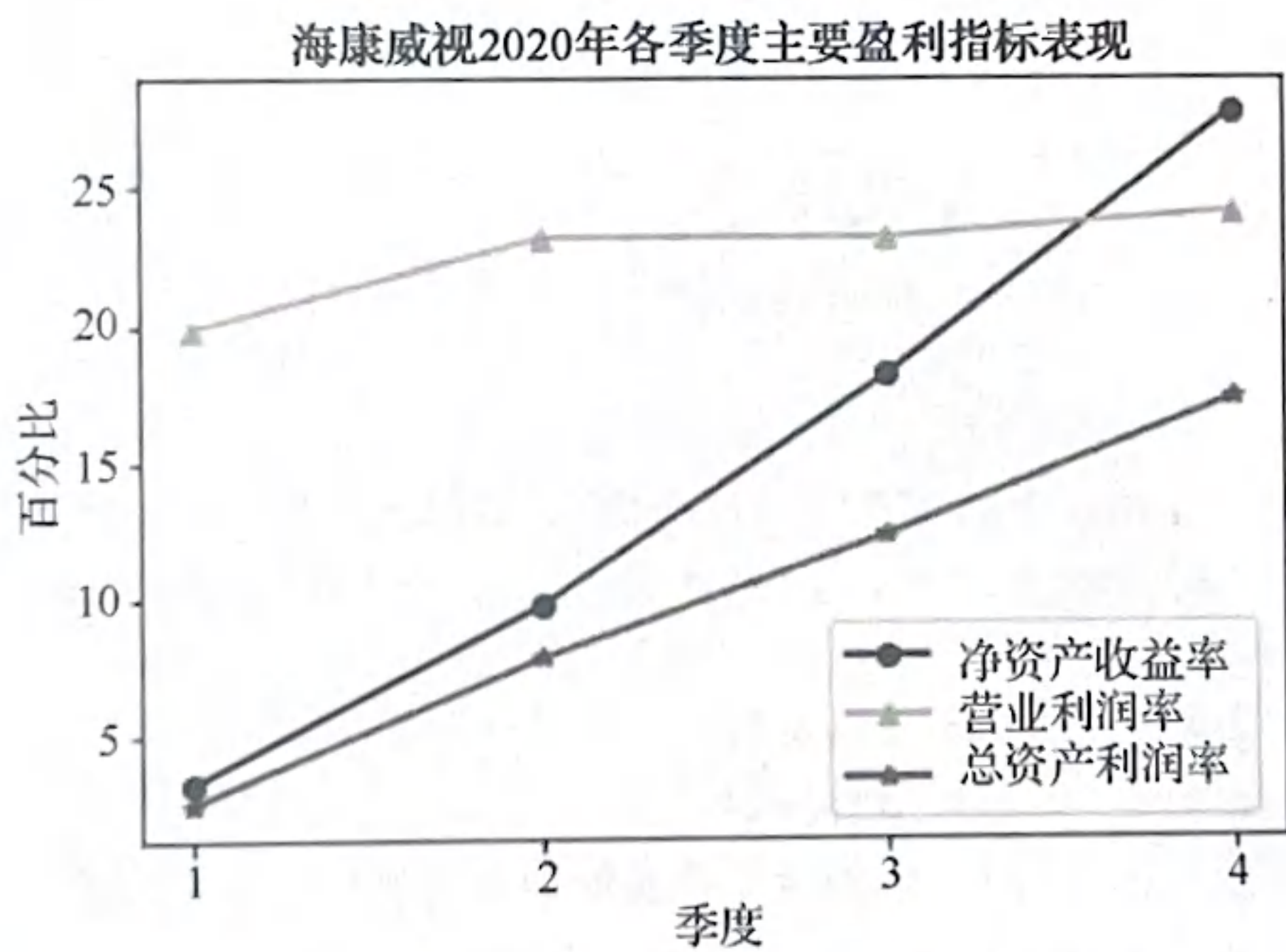


图 9-15 海康威视 2020 年各季度主要盈利能力指标表现

③ 获取偿债能力指标数据并绘图。

➤ 获取 2020 年各季度的主要偿债能力指标

```
1 df_debt=tables[2]
2 df_debt=df_debt[df_debt["类别\年份"]>=202003]
3 df_debt=df_debt[df_debt["类别\年份"]<=202012]
4 df_debt
```

运行结果

	类别\年份	资产负债率 (%)	股东权益比率 (%)	流动比率	速动比率
3	202012	38.58	61.42	2.39	1.97
4	202009	38.96	61.04	2.71	2.16
5	202006	38.49	61.51	2.79	2.17
6	202003	35.71	64.29	3.09	2.36

④ 根据以上数据，绘制折线图。

➤ 绘制 2020 年各季度主要偿债能力指标表现图

```

1 plt.figure(figsize=(15, 5))
2 quarter=list()
3 DAR=list(0 for n in range(4))      # 资产负债率(debt to asset ratio)
4 ER=list(0 for n in range(4))      # 股东权益比率(Equity Ratio)
5 CR=list(0 for n in range(4))      # 流动比率(current ratio)
6 AR=list(0 for n in range(4))      # 速动比率(Acid-test Ratio)
7 q=0
8 for idx,row in df_debt.iterrows():
9     q+=1
10    quarter.append(str(q))
11    DAR[4-q]=float(row[1])
12    ER[4-q]=float(row[2])
13    CR[4-q]=float(row[3])
14    AR[4-q]=float(row[4])
15 ax=plt.subplot(121)
16 ax.plot(quarter, DAR, marker='o', mec='b', mfc='b',color='b',label=u"资产负债率")
17 ax.plot(quarter, ER, marker='^', mec='y', mfc='y',color='y',label=u"股东权益比率")
18 ax.legend()      # 让图例生效
19 ax.set_xlabel(u"季度")      # x 轴标签
20 ax.set_ylabel(u"百分比")      # y 轴标签

21 ax=plt.subplot(122)
22 ax.plot(quarter, CR, marker='*', mec='r', mfc='r',color='r',label=u"流动比率")
23 ax.plot(quarter, AR, marker='.', mec='g', mfc='g',color='g',label=u"速动比率")
24 ax.legend()      # 让图例生效
25 ax.set_xlabel(u"季度")      # x 轴标签
26 ax.set_ylabel(u"数值")      # y 轴标签
27 plt.suptitle(u"海康威视 2020 年各季度主要偿债能力表现")      # 标题
28 plt.show()

```

运行结果（见图 9-16）

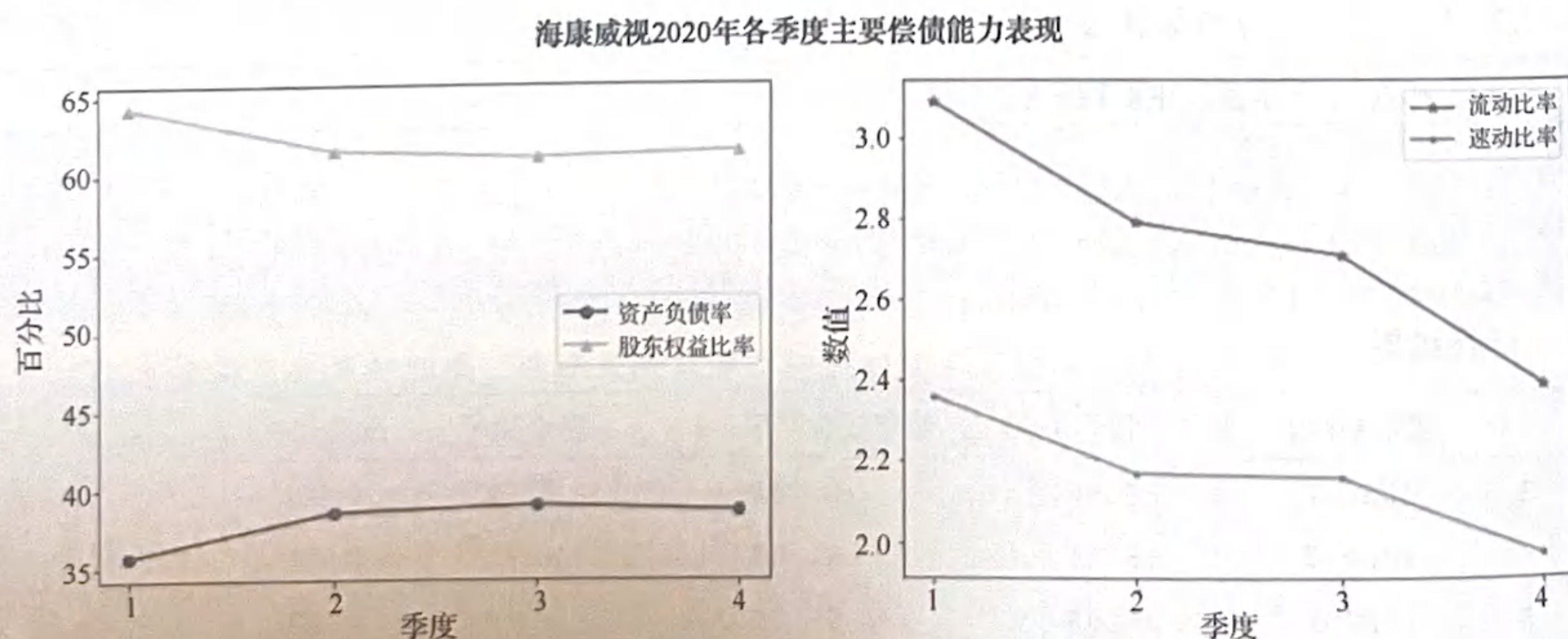


图 9-16 海康威视 2020 年各季度主要偿债能力指标表现

(16) 获取 2020 年各季度的成本费用指标。

➤ 获取 2020 年各季度成本费用指标

```
1 # 成本费用分析
2 df_cost=tables[3]
3 df_cost=df_cost[df_cost["类别\年份"]>=202003]
4 df_cost=df_cost[df_cost["类别\年份"]<=202012]
5 df_cost
```

运行结果

	类别\年份	营业成本（元）	销售费用（元）	管理费用（元）	财务费用（元）
3	202012	508.66 亿	73.78 亿	17.90 亿	3.96 亿
4	202009	341.89 亿	52.34 亿	13.35 亿	1.55 亿
5	202006	198.01 亿	34.20 亿	8.65 亿	-2.28 亿
6	202003	78.48 亿	13.76 亿	3.50 亿	-2.09 亿

(17) 根据以上数据绘制柱形图。

➤ 绘制柱形图

```
1 OC=list(0 for n in range(4)) # 营业成本(又称运营成本 Operating Cost)
2 SE=list(0 for n in range(4)) # 销售费用(Selling Expense)
3 AE=list(0 for n in range(4)) # 管理费用(Administrative Expense)
4 FE=list(0 for n in range(4)) # 财务费用(Financing Expense)
5 delta=0.2
6 qt=pd.Series([1,2,3,4])
7 q=0
8 for idx,row in df_cost.iterrows():
9     q+=1
10    OC[4-q]=float(row[1].replace("亿",""))
11    SE[4-q]=float(row[2].replace("亿",""))
12    AE[4-q]=float(row[3].replace("亿",""))
13    FE[4-q]=float(row[4].replace("亿",""))
14 plt.figure(figsize=(15, 8))
15 plt.rcParams['font.size'] = 18 # 全局 14 号字体
16 plt.bar(qt-2*delta, OC, label="营业成本",width=delta)
17 plt.bar(qt-delta, SE, label="销售费用",width=delta)
18 plt.bar(qt, AE, label="管理费用",width=delta)
19 plt.bar(qt+delta, FE, label="财务费用",width=delta)
20 plt.legend(bbox_to_anchor=(1.05, 0), loc=3) # 让图例生效
21 plt.xticks(qt) # x 轴上的刻度显示, 可以把 x 改成字符串列表
22 plt.xlabel(u"季度") # x 轴标签
23 plt.ylabel(u"成本费用") # y 轴标签
24 plt.title(u"海康威视 2020 年各季度成本费用构成") # 标题
25 plt.show()
```


运行结果（见图 9-17）

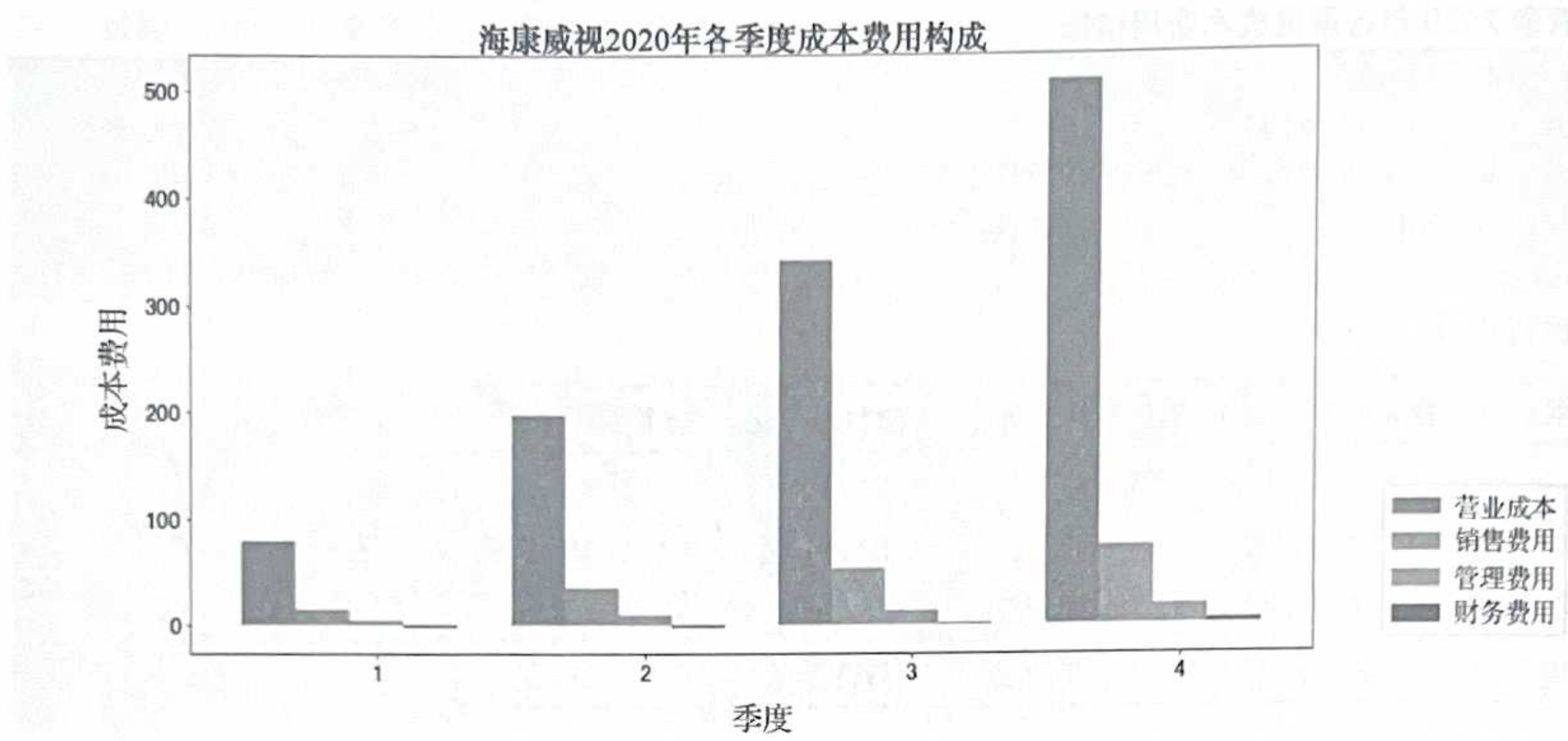


图 9-17 海康威视 2020 年各季度主要成本费用构成

(18) 将以上财务指标绘制到一张图上呈现。

➤ 综合显示

```
1 # 三图综合
2 # 查看不同指标下，海康威视公司在各季度的财务情况
3 plt.figure(figsize=(15, 10))
4 plt.rcParams['font.size'] = 12

5 ax = plt.subplot(231)
6 # 获取以下维度的数据
7 labels=["第 4 季度","第 3 季度","第 2 季度","第 1 季度"]
8 sr=df_quarter[df_quarter.columns[0]]
9 x=list()
10 for v in sr.tolist():
11     value=float(v.split("亿")[0])
12     if value<0:
13         value=0 # 去除负数
14     x.append(value)
15 ax.pie(x=x, # 绘图数据
16     labels = labels, # 添加标签
17     autopct = '%.1f%%', # 设置百分比的格式，这里保留一位小数
18     wedgeprops = {'width': 0.5, 'edgecolor':'green'}, # 边框显示绿色
19 )
20 ax.set_title("各季度贡献-"+df_quarter.columns[0])

21 ax = plt.subplot(234)
22 sr=df_quarter[df_quarter.columns[1]]
23 x=list()
24 for v in sr.tolist():
25     value=float(v.split("亿")[0])
26     if value<0:
```



```

26     value=0    # 去除负数
27     x.append(value)
28 ax.pie(x=x,    # 绘图数据
29     labels = labels,    # 添加标签
30     autopct = '%.1f%%',    # 设置百分比的格式, 这里保留一位小数
31     wedgeprops = {'width': 0.5, 'edgecolor': 'green'},    # 边框显示绿色
32     )
33 ax.set_title("各季度贡献-"+df_quarter.columns[1])

34 ax = plt.subplot(232)
35 ax.plot(quarter, ROE, marker='o', mec='b', mfc='b', color='b', label=u"净资产收益率")
36 ax.plot(quarter, OPR, marker='^', mec='y', mfc='y', color='y', label=u"营业利润率")
37 ax.plot(quarter, ROA, marker='*', mec='r', mfc='r', color='r', label=u"总资产利润率")
38 ax.legend()    # 让图例生效
39 ax.set_xlabel(u"季度")    # x 轴标签
40 ax.set_ylabel(u"百分比")    # y 轴标签
41 ax.set_title(u"海康威视 2020 年各季度主要盈利能力表现")    # 标题

42 ax = plt.subplot(433)
43 ax.plot(quarter, DAR, marker='o', mec='b', mfc='b', color='b', label=u"资产负债率")
44 ax.plot(quarter, ER, marker='^', mec='y', mfc='y', color='y', label=u"股东权益比率")
45 ax.legend()    # 让图例生效
46 ax.set_xlabel(u"季度")    # x 轴标签
47 ax.set_ylabel(u"百分比")    # y 轴标签
48 ax.set_title(u"海康威视 2020 年各季度主要偿债能力表现")    # 标题

49 ax = plt.subplot(436)
50 ax.plot(quarter, CR, marker='*', mec='r', mfc='r', color='r', label=u"流动比率")
51 ax.plot(quarter, AR, marker='.', mec='g', mfc='g', color='g', label=u"速动比率")
52 ax.legend()    # 让图例生效
53 ax.set_xlabel(u"季度")    # x 轴标签
54 ax.set_ylabel(u"数值")    # y 轴标签

55 ax = plt.subplot2grid(shape = (2,3), loc = (1,1), colspan = 2)
56 ax.bar(qt-2*delta, OC, label="营业成本", width=delta)
57 ax.bar(qt-delta, SE, label="销售费用", width=delta)
58 ax.bar(qt, AE, label="管理费用", width=delta)
59 ax.bar(qt+delta, FE, label="财务费用", width=delta)
60 ax.legend()    # 让图例生效
61 ax.set_xticks(qt)
62 ax.set_xlabel(u"季度")    # x 轴标签
63 ax.set_ylabel(u"成本费用")    # y 轴标签
64 ax.set_title(u"海康威视 2020 年各季度成本费用构成")    # 标题
65 plt.show()

```


运行结果（见图 9-18）

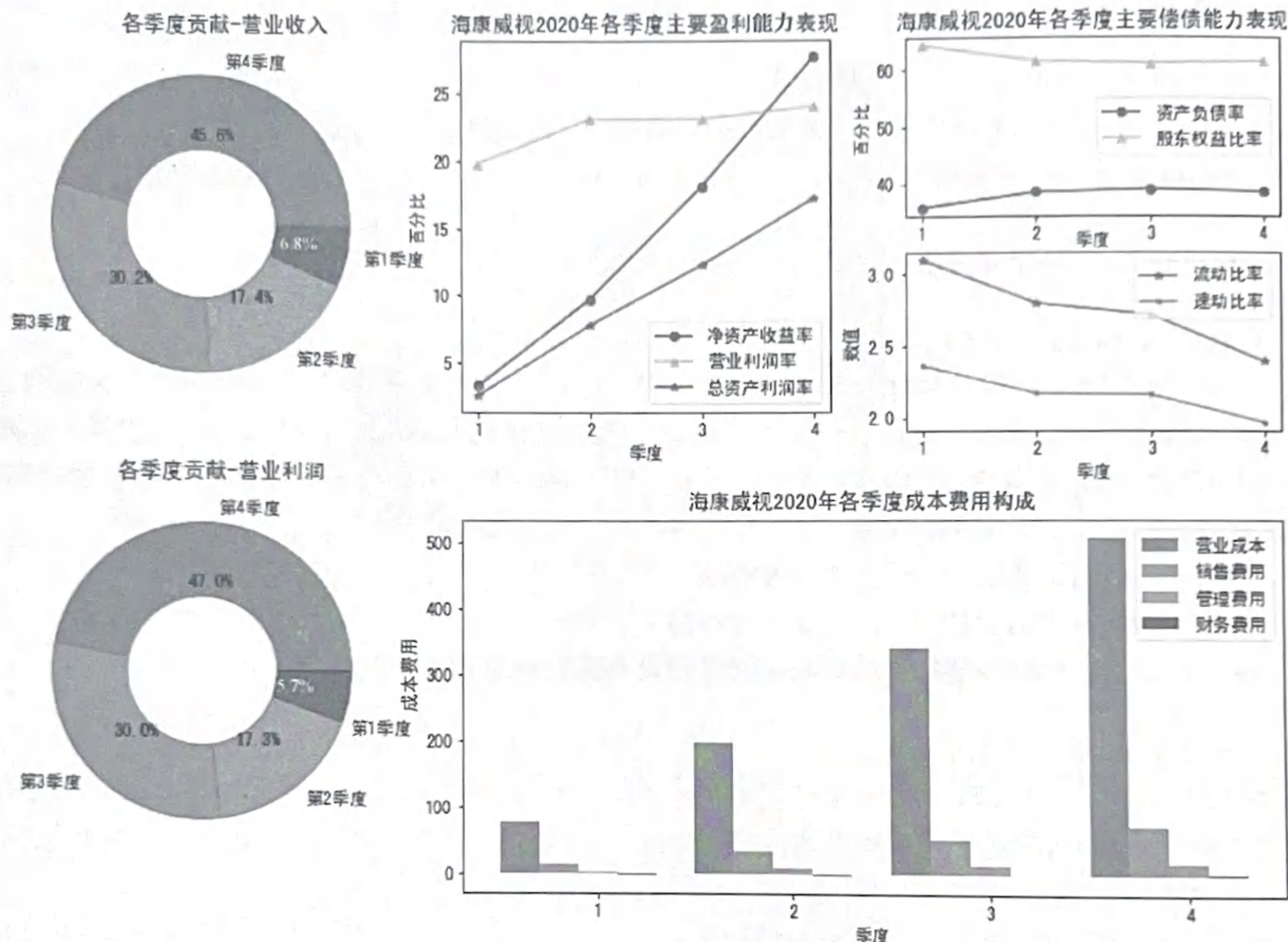


图 9-18 海康威视 2020 年各季度不同维度财务指标比较

业务总结

无论是企业决策者还是个人投资者，都应了解企业的经营情况、财务状况和业务范围等信息。过去，关于财务数据的分析主要集中在三大财务报表的解读上，耗费了大量时间计算财务指标。事实上，由于上市公司财务报表是一套数据集合，不仅具有标准化的、结构化的“四表一注”数据，还有很多非结构化的数据，以及许多不在财务报表中的数据，都需要进行深度解读和价值挖掘。随着 Python 在财务中的开发应用进一步深入，可以通过文本挖掘、机器学习、自然语言识别等方式，进一步挖掘完整版财务报告和非财务数据的数据价值。