

“智”在掌握 享受未来生活



跨境电商实战赛数据分析与营销比赛
项目报告

项目摘要

随着物联网、人工智能、和无线通信技术的发展，智能家居设备变得更加高效、可靠和易于使用。随着全球环保意识的提升,越来越多的消费者倾向于选择能够帮助他们减少能耗和碳排放的智能家居产品。与此同时，智能家居类产品被定位为高端市场，针对的是追求高品质生活的消费者群体，这自然导致了较高的客单价。而高客单价往往意味着高利润，因此智能家居品类无疑也是跨境电商企业采用“精品类运营”模式的较好选择。

本项目使用 C 题提供的数据，即 AliExpress 全球速卖通平台上某一真实智能家居品类店铺 2022 年 5 月—2023 年 4 月的实际订单数据，综合使用 Python、Excel、石方智数平台进行数据分析并提出营销建议。具体而言，本报告在第一部分调研智能家居行业现状；第二部分对数据进行预处理，包括匹配数据、清洗数据、转化数据、特征编码、扩展数据一系列操作；第三部分分析店铺运营情况，包括淡旺季分析、产品分析、客户分析、销售预测；第四部分为运营提升分析，包括滞销产品处理、备货建议、直通车建议、供应商选择；第五部分为数字营销实践，为本报告分析店铺进行具体的营销规划。

关键词：跨境电商 智能家居 数据分析 数字营销

目录

一、行业背景调研	1
（一）市场规模分析	1
（二）智能家居产品分析	3
（三）智能家居品牌分析	5
（四）用户体验	7
（五）未来趋势预测	7
二、数据预处理	8
三、店铺运营情况分析	9
（一）淡旺季分析	9
（二）产品分析	11
（三）客户分析	14
（四）销售预测	19
四、运营提升分析	23
（一）滞销产品处理	23
（二）备货建议	23
（三）直通车	25
（四）供应商选择	25
（五）价格优化	27
五、数字营销实践	28
（一）目标市场选择	28
（二）精准营销策略	28
（三）回报策略	31
（四）可视化大屏展示	32
附录	33

一、行业背景调研

随着科技的进步和消费者对于便捷、舒适生活的追求，智能家居行业得到了快速发展，并逐渐渗透到人们的日常生活中。智能家居是指通过互联网技术和智能设备，将家庭设备、家庭电器、家庭安全等各种家居设施连接，实现智能控制、自动化管理和远程控制的一种家居生活方式。其主要运用通信（智能电视、智能家庭影院）、传感器（温度、湿度、光照、气体传感器）、安全（智能门锁）、机器视觉（智能监控）、无线充电（无线充电器）、人工智能（智能音响）六大技术，其中传感器机器视觉及 AI 技术最为核心。智能家居较传统家居在控制方式、设备互联、功能、自动化程度、安全性、便捷性和舒适性等方面有显著的优势。智能音箱、智能电视、智能门锁、智能照明、智能家电等产品已经普及到许多家庭。这些产品不仅为人们提供了更加便捷的生活方式，同时也通过数据收集和分析，帮助用户更好地管理家庭事务，提高生活质量。

在全球范围内，智能家居市场呈现出多元化的发展趋势。不同国家和地区根据自身的文化背景和消费需求，发展出了各具特色的智能家居产品和服务。例如，一些地区注重家庭安全和监控，智能安防产品深受欢迎；而另一些地区则更加关注环保和节能，智能家居系统通过智能调节家电设备的工作状态，实现能源的有效利用。智能家居行业的发展也带动了相关产业链的发展。智能家居产品的制造、销售、安装、维护等环节都需要专业人员的支持。同时，智能家居技术的不断创新也推动了物联网、大数据、人工智能等相关领域的发展，为全球经济的发展注入了新的活力。历经单品智能、互联互通的阶段后，智能家居 3.0 阶段已基本实现设备间的联通与场景自动化，未来的演进方向将聚焦于系统智能化水平的整体提升。依托物联网、云平台、人工智能及边缘计算等多种支撑技术，将品类丰富的设备连网，在设备端搭载多样 AI 算法，使产品具备通过学习用户行为完成自主、判断决策，从而满足用户个性化、弹性化的使用需求；云平台与边缘计算的应用提升了智能家居系统的易用性与灵活性，构建设备间的控制管理，不断引导用户认知、理解全屋智能理念，推进空间整体智能化进程。

（一）市场规模分析

1.全球智能家居市场持续繁荣，预计将持续增长

如图 1 所示，全球智能家居市场规模及渗透率自 2017 年以来呈现逐年增长趋势，从 400 亿美元攀升至 2023 年的 1400 亿美元，彰显了科技的快速发展和人们对舒适生活的不断追求。据共研网预测，到 2026 年，这一市场规模将突破 2000 亿美元大关，达到 2078 亿美元，显示出行业发展的巨大潜力。据 TechInsights 报告预测，未来五年内，消费者在智能家居领域的支出将以接近 8% 的复合年增长率持续增长，到 2028 年预计将达到 1910 亿美元。这表明智能家居市场的消费需求将持续旺盛，且增长速度不断加快。在智能家居市场中，智能产品的单点解决方案更受消费者欢迎。以北美市场为例，智能恒温器、智能灯泡、智能插头、联网安全摄像头和语音控制智能扬声器等品类备受欢迎。这些产品的出现满足了消费者对智能家居功能明确、操作简便的需求。

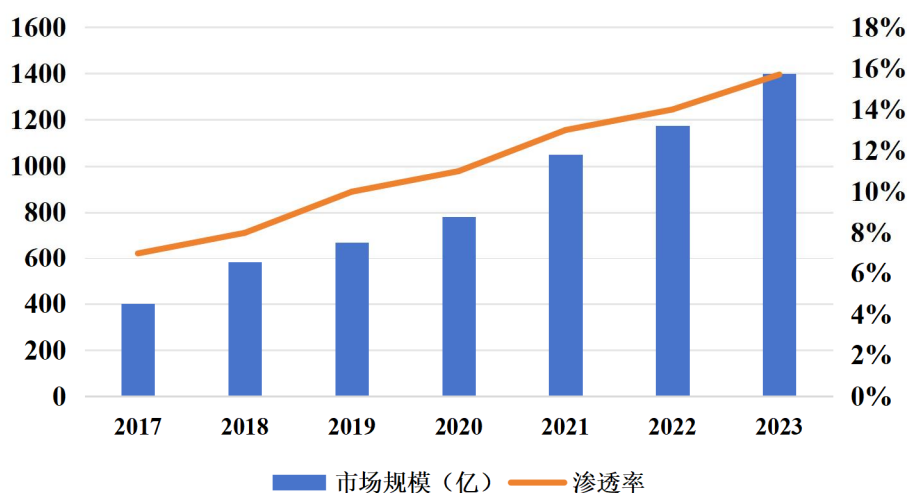


图 1.全球智能家居市场规模

（数据来源：共研网）

2.全球智能家居行业目前整体市场渗透率不高，呈现出明显的地区差异

根据艺恩咨询发布的研究报告，2022 年智能家居渗透率超过 30%的主要为欧美国家，其中英国、美国的渗透率均超过 40%，为全球渗透率最高的前两大国家，市场规模较大（见图 2）。在全球智能家居市场中，北美市场保持领先地位，2023 年占据最大市场份额，约为 353.6 亿美元。中国市场因起步晚于欧美发达国家，渗透率相对较低，约 15%。东南亚只有马来西亚、泰国和印度尼西亚渗透率超 10%，约 13%。东南亚市场因消费者年龄结构整体呈现年轻化，预计未来将有较大的市场潜力。非洲各国渗透率均没有超过 10%。可见，智能家居产业主流市场仍以欧美市场为主，东南亚国家发展迅猛潜伏重大市场机遇，非洲仍在起步阶段。

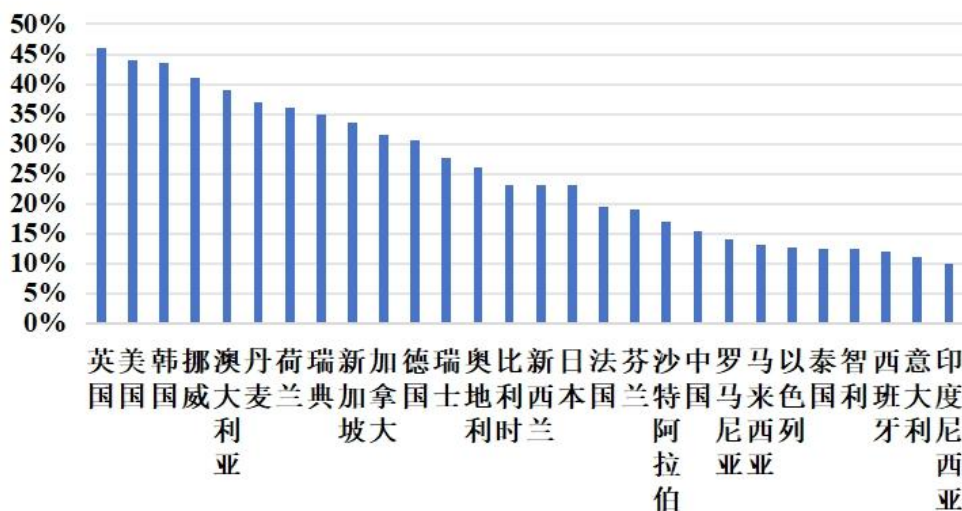


图 2.主要国家的智能家居渗透率

（数据来源：Statista）

欧洲市场中，娱乐相关智能家居设备渗透率最高，市场广阔且未饱和。在欧洲，娱乐相关的智能家居设备渗透率独占鳌头。随着技术的不断演进，智能家居设备有望呈现出更加多元化的趋势，不同地区将涌现出满足各自特定需求的高差异化产品。整体来看，智能家居设备市场仍然充满潜力，持有率尚未触及饱和点。以法国为例，尽管 95%的 16-64 岁网络用户中，仅有 20.3%购买过智能手表，12.3%购买过智能家居产品，但在智能互联的推动下，法国智能家居设备

市场仍然存在巨大的增长空间。

美国市场以娱乐为主，智能照明出货量增速领先。在美国，智能家居市场的主流产品同样以娱乐为主，辅以安全节能设备。知名品牌如 Samsung、LG、AT&T 在市场份额上差距不大，且大多数品牌经营的智能家居产品类目超过 10 个，与智能家居万物互联的概念高度契合。值得一提的是，智能照明在全球范围内已成为智能家居的第三大品类，其出货量增速领先。根据 IDC 数据，2022 年全球智能家居市场中，智能照明出货量占比达到 10.8%，并有望在 2023 年至 2027 年间以 13.6% 的复合年增长率持续增长，出货量预计将从 2023 年的 1.0 亿台增长至 2027 年的 1.7 亿台。

东南亚智能家居市场新锐品牌发展空间大，智能家电占比超 50%。在东南亚智能家居市场，robot 品牌占据超过 10% 的市场份额，而大部分其他品牌的市场份额均低于 5%，显示出行业市场集中度相对较低，为新锐品牌提供了发展空间。其中，中国的 ECOVACS 和小米等品牌跻身前五。在细分品类收入方面，智能家电占比超过 50%，满足了人们日常娱乐生活的需求，并存在高客单价产品。预计到 2027 年，行业整体收入将翻倍，同时安全、舒适照明类产品的占比也将有所提升。截止 2022 年，家庭娱乐类智能家居产品用户数量最多，达到 1000 万人，而安全类产品预计在未来将成为用户数量最多的细分类目，预计用户数将超过 2000 万。

中国智能家居照明出货量快速增长，出口提升幅度大。在中国，智能家居照明出货量经历了快速增长。从 2021 年至 2023 年，出货量从 1910 万台增长至 3379 万台，复合年增长率高达 33%。这一增长不仅反映了国内市场的强劲需求，也显示了中国智能家居照明产品在国际市场上的竞争力不断增强，出口提升幅度大，提升空间仍然广阔。

（二）智能家居产品分析

在智能家居行业产业链中，上中下游分别对应不同的行业。上游行业基本包括：工业设计、芯片、传感器、通讯模块、语音控制与交互技术、智能控制器等；而中游行业的产品更为常见，基本包括：智能窗帘、智能门锁、智能空气净化器、智能洗衣机、扫地机器人、智能电视等；中游行业占据较大的市场份额；下游行业包括各种销售渠道。基于本次分析的数据，其中包括的产品大概分为以下几个主要类别：红外遥控器、空气质量传感器、智能门窗传感器、家居安防摄像机、智能灯、智能开关、智能插座、温湿度传感器，处于智能家居行业产业链的中下游行业（如图 3 所示）。此外，Statista 数据显示，智控系统和智能安全产品将成为未来智能家居市场中增长最快的品类，表明这些领域具有巨大的发展潜力。

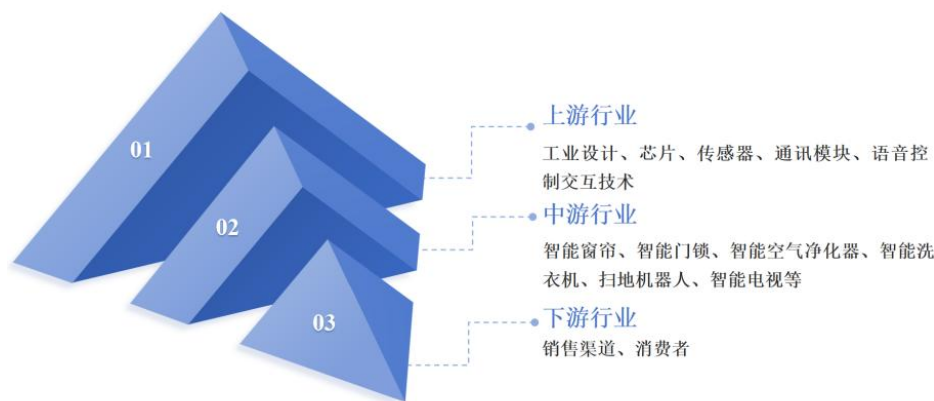


图 3.智能家居行业上中下游行业概况

基于本次分析数据涉及的主要产品类型，并参考 Statista，Grand View Research，Verified Market Research，Strategy Analytic，华安证券研究所等做进一步分析：

1.智能控制器

全球智能控制器技术已经相对成熟。智能控制器作为实现智能化的关键组件之一，自然也得到了广泛应用。图 5 为智能控制器的产品图片，智能控制器可以通过手机、平板电脑等移动设备进行远程操控，方便用户随时随地控制家中的电器设备，极大地提升了生活的便利性。其具有定时、感应、场景等多种控制方式，可以根据用户的需求自动调节电器设备的运行状态，有效节约能源，降低碳排放。由图 4 可知，英美等国智能家居渗透率较高。

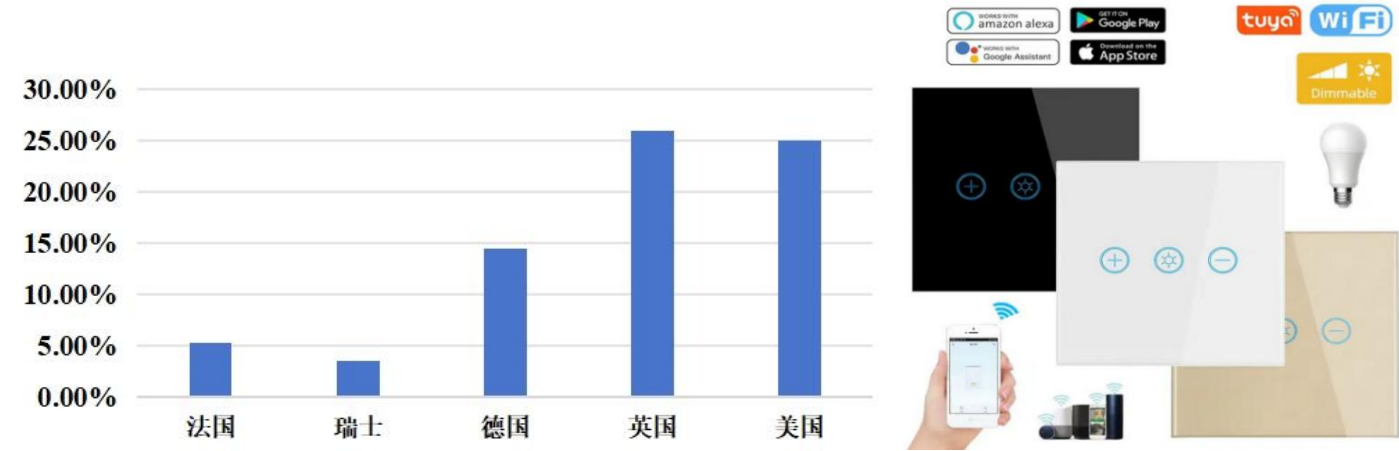


图 4.智能控制器在欧美国家渗透率

图 5.智能控制器产品图

2.智能照明

发达国家智能照明技术比较成熟。目前涂鸦品牌的智能灯可以让用户实现通过手机 APP 远程控制，实现开灯、关灯、调光等操作。这为用户提供了极大的便利，可以随时随地控制家里的灯光。此外，目前的智能照明产品不仅可以调节亮度、色温，还可以与音乐、影视等娱乐设备联动，为用户提供更加丰富的使用体验。如图 6 所示，智能照明在欧美国家的渗透率差异不大，图 7 为智能控制器的产品图片。

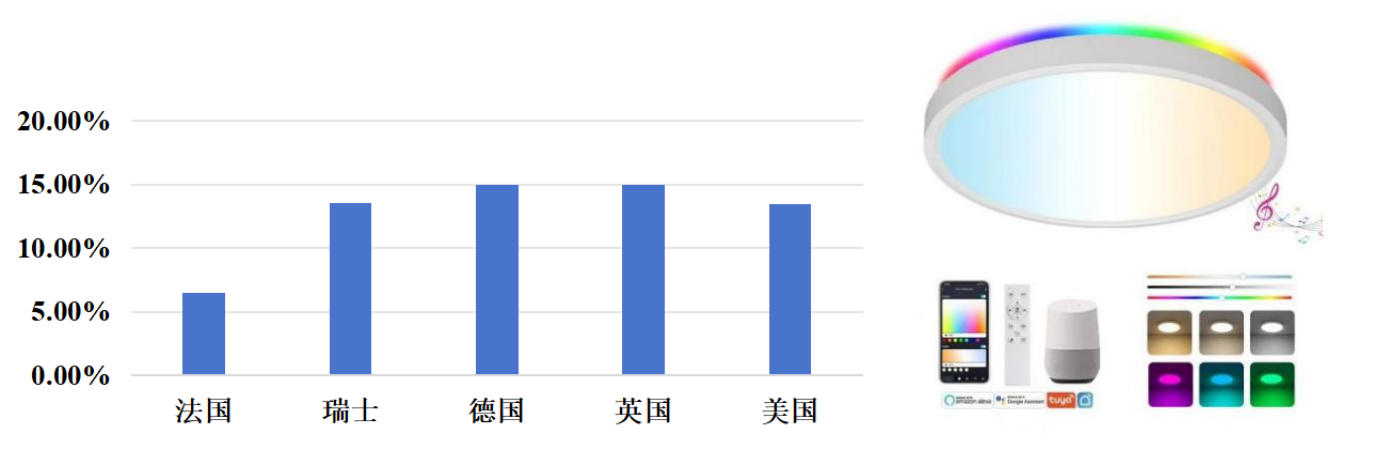


图 6.智能照明在欧美国家渗透率

图 7.智能灯产品图

3.智能安防

图 8 显示了智能安防在欧美国家的渗透率，图 9 为智能安防的产品图片。智能安防系统通常具有远程监控、实时报警、智能分析等功能，使用户可以随时随地了解安全状况，方便快捷。除了家庭安全领域，智能安防还在商业、工业、金融等领域得到广泛应用，如智能门禁系统、智能监控系统等，提高了商业运营的安全性和效率。

2021年各国智能照明渗透率

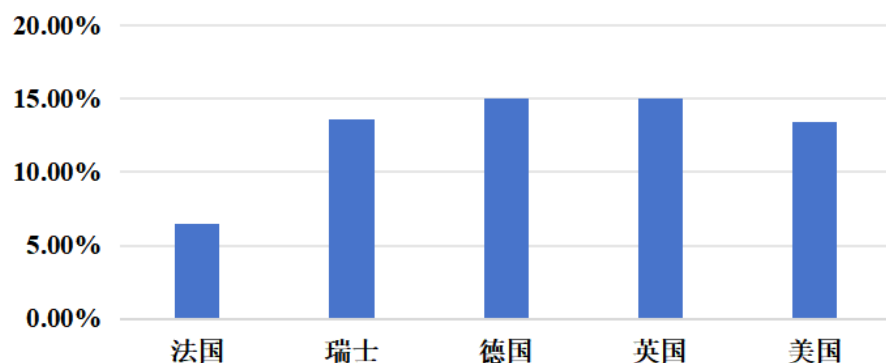


图 8.智能安防在欧美国家渗透率



图 9.智能安防产品图

由于欧美国家的经济发展水平较高，消费者的购买力较强，且普遍追求高品质的生活，伴随着智能家居产品的价格也在逐年下降，更多的消费者能够接触和购买到这些产品，所以各种智能产品在欧美发达国家渗透率较高。目前，智能家居各类子产品正在获得更广阔的市场规模，在技术日趋成熟的将来，智能家居产品价格虽然可能会下降，但其可以更多地向发展中国家的市场进军。

（三）智能家居品牌分析

对比产品线后，涂鸦（Tuya）品牌产品的供应商支持原材料溯源，且所有生产线均配备质检，相比于其他的供应商质量更有保障、更具经营经验。在品牌影响力对比方面，采用关键词搜索方法，涂鸦产品的品牌影响力大于各竞品。并且涂鸦品牌的供应商更具经营经验（16 年）。但是，相比于其他竞品，涂鸦的产品不具备价格优势：以智能灯为例，五种竞品的智能灯的价格区间在 14.34 元到 63.09 元，而涂鸦产品价格区间为 80.36 元到 98.93 元。以家居安防摄像机为例，竞品价格在 43.02 元到 114.70 元之间，涂鸦品牌产品价格区间在 86.03 元到 119.36 元，知名品牌小米的价格区间为 164.88 元到 247.32 元；由此可见涂鸦正在向高质量、高知名度的中高端品牌迈进。

表 1 智能家居竞品分析

品牌/供应商	营业经验	主营产品类目	评分	主要市场
Pcba (义乌市极光智能科技有限公司)	9 年	智能家居系统解决方案、智能网关、智能开关、NB-IOT 系统、传感器	3.3/5.0	中国 25% 西欧 25% 北美 25%
Ci-tech (深圳市西奈科技有限公司)	2 年	智能插头、WiFi 插头、智能开关、WiFi 开关、智能摄像头	5.0/5.0	中东 20% 非洲 20% 东南亚 10%
SONOFF (哈尔滨世外进出口有限公司)	5 年	智能家居产品、Sonoflex 产品、开窗器、太阳能发电系统	5.0/5.0	中美洲 20% 中国 10% 西欧 10%
POE (深圳圳市专业安防技术有限公司)	4 年	网络摄像机、模拟摄像机、网络视频录像机、数字视频录像机	4.6/5.0	南美洲 40% 西欧 20% 东亚 10%
Armee (深圳百景安全有限公司)	7 年	烟雾报警器、气体检测报警、门传感器报警	4.9/5.0	中国 69% 北美市场 26% 南美洲 2%

由于本店大多数产品都是来自涂鸦品牌，所以我们对其做单独的进一步分析。涂鸦品牌智能家居的主要市场集中在发达国家，其中欧洲市场是涂鸦品牌最重要的市场。但由于价格较高，处于中高档消费市场。欧洲市场对智能家居的需求可能更偏向于高端、智能化程度高的产品。接下来我们对涂鸦做 SWOT 分析：

1. 优势(Strengths)

- (1) **技术先进：**利用最新的物联网技术，可以实现远程控制、自动化控制、语音控制等功能，为用户带来前所未有的便利。如涂鸦 智能开关、涂鸦 ZigBee/WiFi 温湿度传感器，可以与谷歌助手互联；涂鸦 WiFi 智能门禁传感器可以通过手机 APP 控制等。
- (2) **舒适度高：**用户可以根据生活习惯和需求，自动调节家居环境，如温度、湿度、照明等，比如涂鸦 WiFi 智能灯，可以通过谷歌助手调节适宜的亮度，提高生活舒适度。
- (3) **安全可靠：**Smart Home Security Protection 类产品具备高度安全性，可以有效保护家庭财产安全；同时还可以通过实时监控家庭成员的生活状态，提高家庭安全系数。

2. 劣势(Weaknesses)

- (1) **价格较高：**涂鸦产品价格普遍较高，对于一些中低收入家庭来说，难以承受。
- (2) **安装复杂：**Gateway Analyzer、2 way Relay Module 等类智能电子元器件类产品需要专业的安装人员，安装过程较为复杂，普通用户难以自行完成。
- (3) **依赖网络：**涂鸦产品需要连接互联网才能正常使用，一旦网络出现问题，智能家居设备的功能将受到限制。

3. 机会(Opportunities)

- (1) **政策支持：**政府对于物联网、人工智能等新兴产业给予政策支持，有利于智能家居行业的发展。类 ChatGPT 应用可整合多类设备，借助 Template Starter Kit 优化和和自然语义理解式对话提供个性化服务。涂鸦品牌产品可以通过与 Open AI 等大语言模型结合，提升产品的智能性、理解能力，更好地满足用户的个性化需求。
- (2) **市场需求：**市场需求随着人们生活水平的提高不断增长。涂鸦品牌应挖掘除欧洲和南美地区的其他发达国家市场。
- (3) **创新空间大：**智能家居行业仍处于发展初期，还有许多未开发的市场和未满足的用户需求，具有很大的创新空间。例如，通过融合行业 AI 大数据算法，类 ChatGPT 应用可分析来自智能设备的数据，预测设备寿命，减少消费者损失；且可以通过大量数据训练分析用户语气变化，调节灯光柔和水平，为用户提供情绪价值。

4. 威胁(Threats)

- (1) **竞争激烈：**智能家居行业竞争激烈，SONOFF、geya 等中小品牌，小米、华为等大品牌都在争夺市场份额，导致涂鸦品牌产品利润空间必然受到压缩。
- (2) **技术更新快：**智能家居领域技术更新迅速，企业需要不断投入研发成本来保持市场竞争力。

(3) **数据安全问题：**涂鸦智能设备涉及到用户的隐私数据，保障数据安全成为一大挑战。

(四) 用户体验

目前，智能家居还存在一些问题，导致用户体验不佳。如图 10 所示，无法精确理解需求、自主智能弱、无法满足自身个性化需求是智能家居最主要的问题。据唯奥、华安证券研究所调查表明，智能家居用户偏好的交互模式中，语音交互、手机 APP 自动感应最受用户喜欢。而按键交互、手势交互很少受到用户青睐。由此可知，越智能的交互模式越能受到用户的青睐。目前，智能家居产品的用户体验正变得更加丰富、智能化和个性化。智能家居用户体验从单一的遥控操作到智能语音控制、自动场景联动和个性化需求满足的综合体验不断演进。具体而言：

- 1.加强了操控的便携性与交互性。例如扫地机器人导航模式不断迭代更新，从随机碰撞模式发展至导航模式，增加清洁，同时从单一的清扫功能迭代至清扫、拖地、烘干一体化，并同时具备语音/触控/手机 APP 交互模式。
- 2.加强了安全性。智能门禁从卡片门禁发展至生物、手机、密码识别模式一体化。
- 3.加强了节能性。智能灯具从点控开关模式发展至自动感应模式，且具备自动调节光暗功能，帮助用户实现节能需求

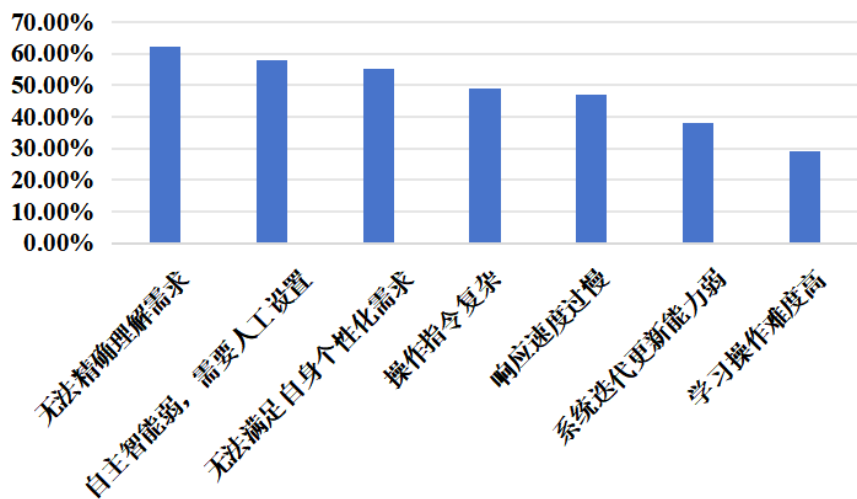


图 10.用户认为存在的问题

(资料来源：唯奥，华安证券研究所)

(五) 未来趋势预测

以具备智慧家居入口意义的智能音响为例，过去实现了语音助手、家居控制功能，但趋向产品同质化严重，市场缺乏创新，增长放缓。可能是因为：用户体验达到瓶颈（产品智能化、交互能力不足）、安全性待提升、兼容性存在问题。目前，智能家居市场中视频娱乐设备占据更大的市场份额，视频娱乐类产品未来难有爆发式增长。智能控制器等便捷人们需求的品类目前持有率不高，未来或将增长较快，在全球范围内仍然存在很大市场空间。

未来智能家居将更加注重场景价值的纵向深化，不断提升现有功能精度、扩大应用范围，为决策提供更多有力的辅助依据；注重横向拓展——不断聚类产品功能并挖掘新的使用场景，实现场景的系统化服务，并加强场景与场景间

的联动功能。技术的迭代为纵向深化提供可能，数据只有经过有效转换才能成为决策依据，在持续扩大感知范围与维度的基础上，借助 AI 能力实现有效分析、向决策大脑传递可靠的决策依据；技术的沉淀可以为用户提供更多价值应用场景，“举一反三”创新服务场景，在实现基础监看、管理的同时，能通过设备的自主决策实现无人化的事件处理，从真正意义上完成智能家居与用户生活的交互融合，降低智能家居的使用门槛。

二、数据预处理

原始数据通常包含噪声、缺失值和异常值，这些都可能影响模型的学习效果。数据预处理是确保数据分析成功的关键步骤，它直接关系到模型的准确性、效率和可靠性。通过有效的数据预处理，可以从杂乱无章的原始数据中提取有价值的信息，为后续的数据分析和模型训练打下坚实的基础。本报告综合使用 Python、Excel、石方智数平台进行数据预处理，涉及 Python 代码的内容在附录中展示。

本报告的数据预处理主要分为以下 5 个步骤：

1.匹配数据

根据列名，把四张 Excel 表的数据匹配到一张表内。我们发现表“202211-202301”的数据缺失“收货地址”、“手机”、“联系电话”列数据，所以这一时间段的数据没有这四个特征变量。

2.清洗数据

“负责人（业务员）”列所有值都是空值，故删除该特征变量。

3.转化数据

所分析内容涉及货币单位均为美元，为了简化分析，去掉了“店铺优惠、产品总金额、订单金额、预计增值税、物流费用”列的货币单位，改为纯数字。

4.特征编码

为便于分析，设置虚拟变量。“平台是否代征代缴”列：平台已税=1；平台未税=0；null=2。

“交易状态”列：等待买家付款=1，等待买家收货=2，等待您发货=3，订单关闭=4，冻结中=5，交易完成=6，买家申请取消订单=7

5.扩展数据

为便于后续数据分析，将“下单时间”列分成“下单日期”与“下单时间”两列。将“商品信息”列分为“产品描述”与“产品属性”分为两列。“商品编码”列分为“编码信息”与“数量信息”两列。

添加“发货速度”列，使用付款时间与发货时间的间隔，单位为小时。添加列“是否延期发货”，如果发货期限晚发货时间，说明没有超时，赋值 0；如果发货期限早发货时间，说明超时，赋值为 1；如果为空值，赋值为 2。

通过数据预处理，使得数据在相同的尺度上，帮助算法更快地收敛，提高模型的训练效率和性能，并且减少数据中的冗余信息，使数据分析和机器学习项目更加灵活和高效。

三、店铺运营情况分析

（一）淡旺季分析

1. 季节分析

每个季节的销量和销售总额的计算主要涉及对销售数据的分析和统计。首先，需要对每个季节进行定义，例如春季、夏季、秋季和冬季。然后，根据销售记录和销售额数据，将销售额按季节进行分类和累加。计算每个季节的销售总额，可以通过将该季节内所有销售额相加来得到。类似地，计算每个季节的销量，可以通过将该季节内所有销售量相加来得到。最后，计算销售总额，可以将所有季节的销售总额相加。这样，就可以得到每个季节的销量和销售总额的计算结果。这些数据对于企业了解产品销售情况、制定销售策略和做出决策具有重要的参考价值。

由图 11 可知，春季为旺季，秋季为淡季。图 12 展示了各月份具体的销量，帮助我们做更加具体的分析。可以发现，智能家居产品总体来说春冬季节销量好，夏秋季节销量差，为典型的季节性产品。

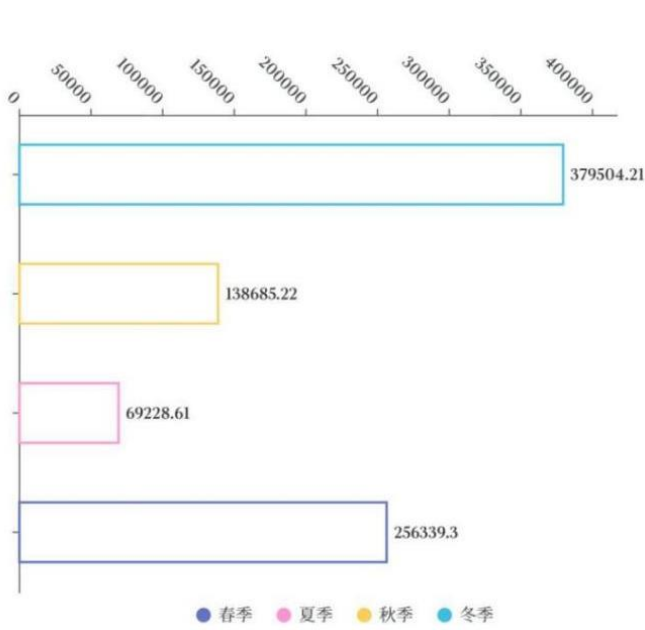


图 11. 季节销量分析

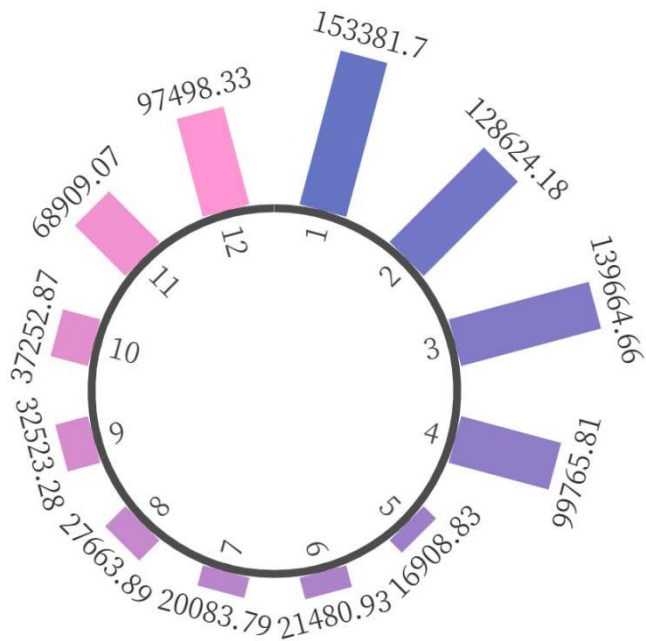
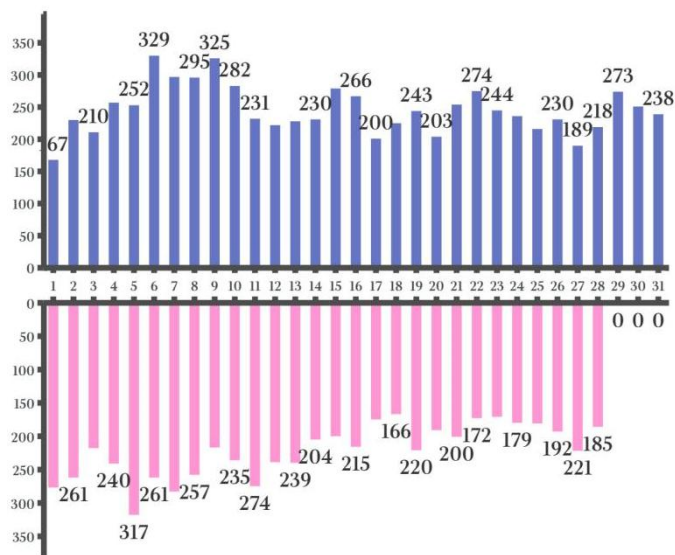


图 12. 月销量分析

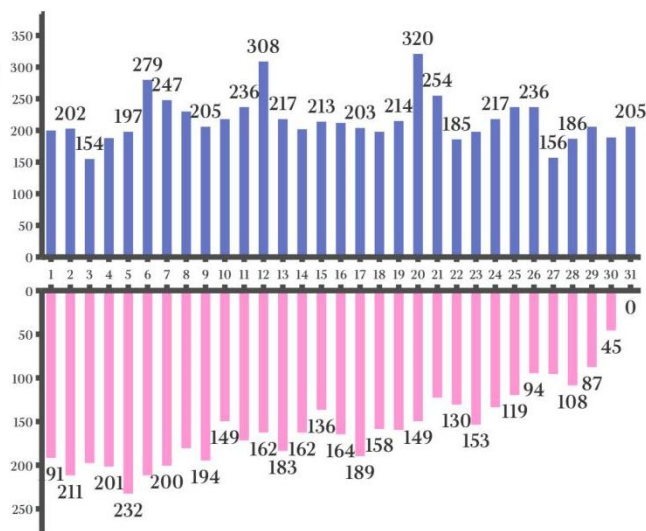
2. 日度分析

接下来我们取旺季月份，即 1-2 月、3-4 月、11-12 月做更加具体的日度销量分析，结果如图 13-16 所示。进一步，我们筛选出订单量最高的前 10 热销日期，并将其绘制成折线图，如图 19 所示，1 月的销量十分亮眼，销量前 8 的日子均在一月，最高订单量可达 393 单/天。可见 1 月是十分重要的营销节点。可以发现，虽然受季节影响，产品有淡旺季，但是日销量高低与节假日、促销活动有着紧密的关系。



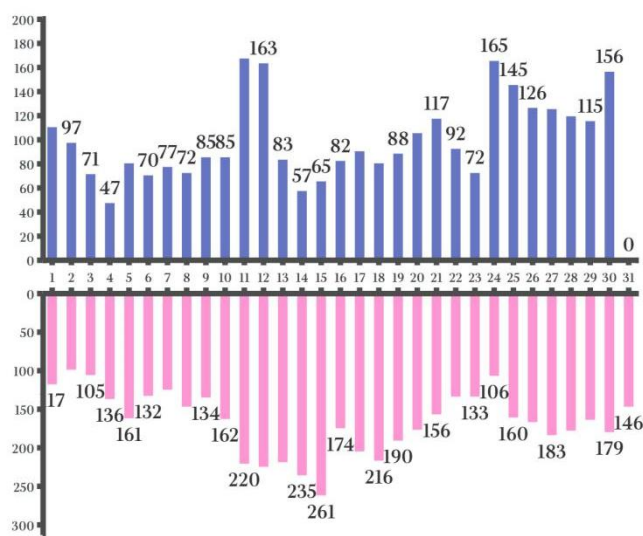
● 一月销量 ● 二月销量

图 13.1-2 月日度销量



● 三月销量 ● 四月销量

图 14.3-4 月日度销量



● 十一月销量 ● 十二月销量

图 15.前十大促日期销量

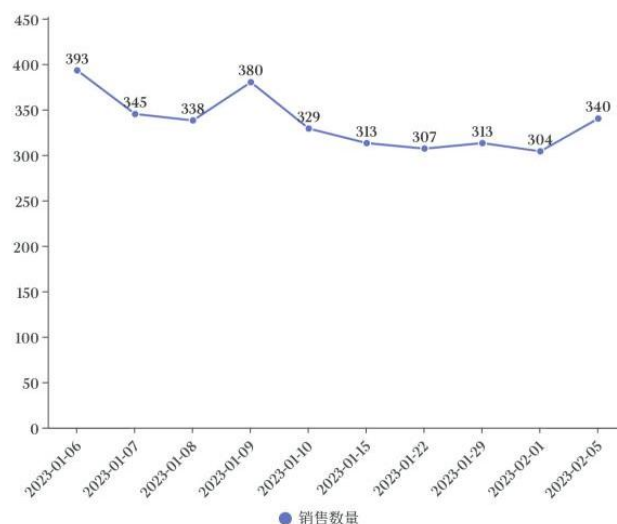


图 16.11-12 月日度销量

3.建议

(1) 在春、冬季即将来临之时增加广告投放，夏秋季节减少宣传。热销季节前营销以曝光为主要目标，提升品牌认知度和关键词的热度，同时结合多种广告产品和广告投放方式，帮助增加流量入口，商品的销售转化率。在淡季时的整体策略应该为尽量收缩，减少资金浪费。

(2) 创造节日营销关联，抓住营销节点。具体而言，在冬季：黑色星期五（11 月的第四个星期五）是美国传统的购物节日，阿里巴巴通常也会参与，并推出大量的商品折扣和促销活动；12 月的圣诞节期间，是一个做圣诞促销活动的机会；1 月作为一年的开端，许多家庭会购置新的产品，并且俄罗斯东正教教会使用儒略历在 1 月 7 日过圣诞节，西班牙在 1 月 20 日过圣巴斯蒂安节，由后文可知，这两个国家也是本店产品的主要销售国，所以店铺应该配合节日进行营销宣传活动。在春季：据日度分析可知，二月中上旬的销量高于二月下旬，这可能是因为 2 月 14 日的情人节；阿里

巴巴通常在 3 月至 4 月之间有春季购物节促销活动，吸引消费者购买新的季节性商品，本店铺也可以根据本文的第五部分进行相应的方案设计进行营销。

(二) 产品分析

本文主要对销量前十产品进行分析。具体而言，先删除订单异常数据（订单状态为订单关闭、冻结中、买家申请取消订单的数据），再使用 Python 统计销量前十的商品，统计结果如图 17 所示。可以发现涂鸦 ZigBee/WiFi 家用温湿度传感器、涂鸦智能迷你 WiFi 灯开关、WiFi 温湿度安全报警器、涂鸦 WiFi 智能门开闭探测器的销量明显高于其他产品。所以我们选取这四种商品来进行进一步分析，为了使可读性更强，我们使用 A、B、C、D 来代表这四个商品。

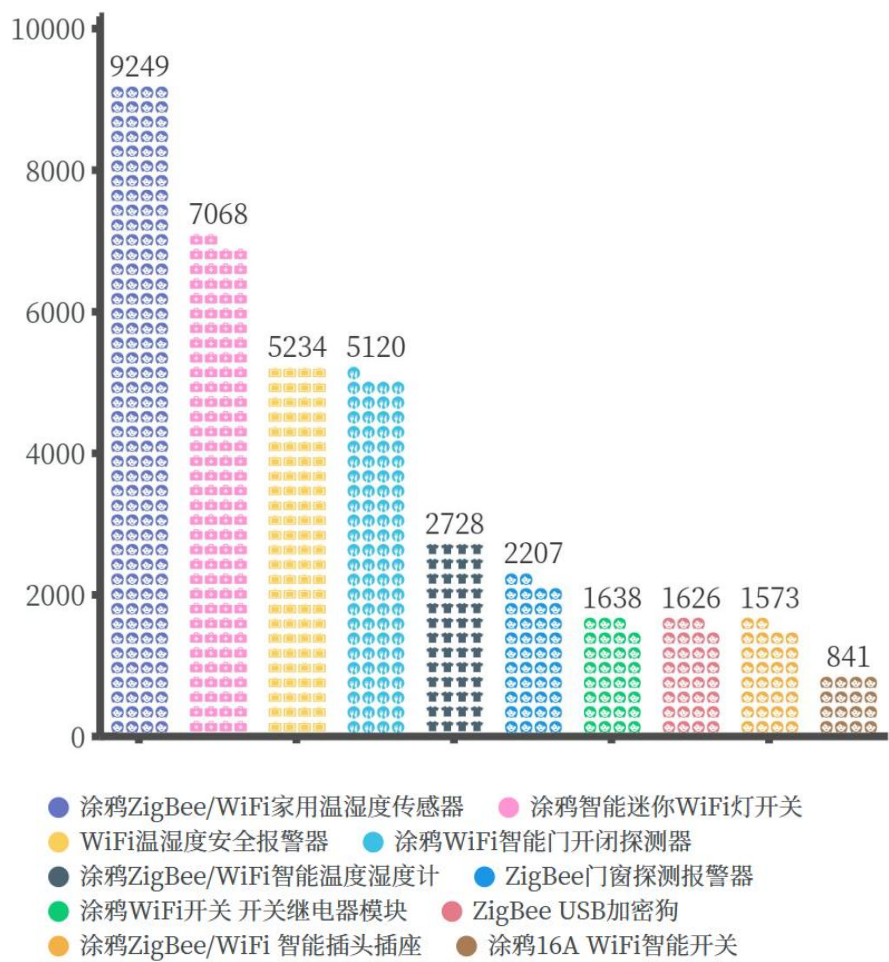


图 17.前十产品销量商品

1.热销产品季节分析

由图 18 可知，热销商品也均为典型的季节性产品，春冬季销量显著高于其他季节。为了能对热销商品有更加详细的理解，我们针对春冬季节再做日度分析，粉色代表冬季，蓝色代表春季，如图 19-22 所示。

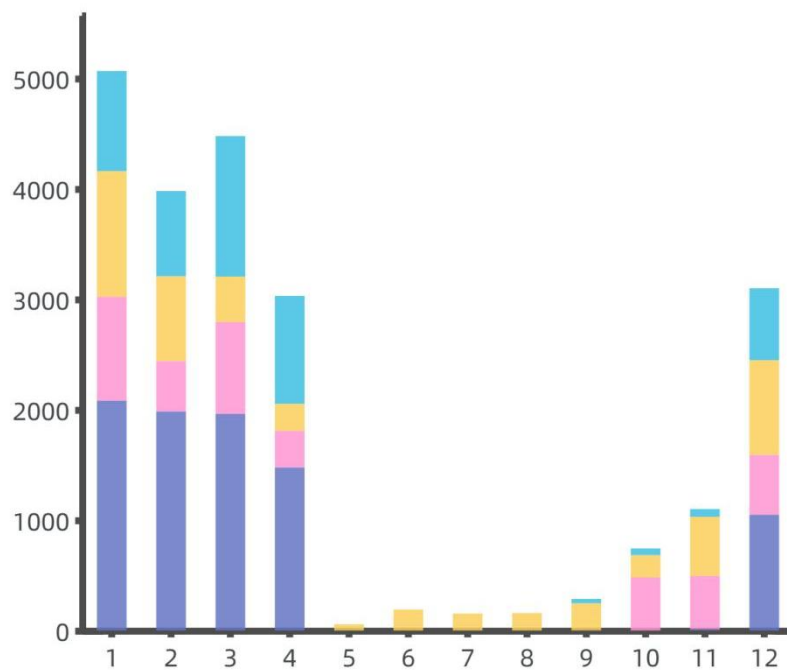


图 18.大促产品季节性分析

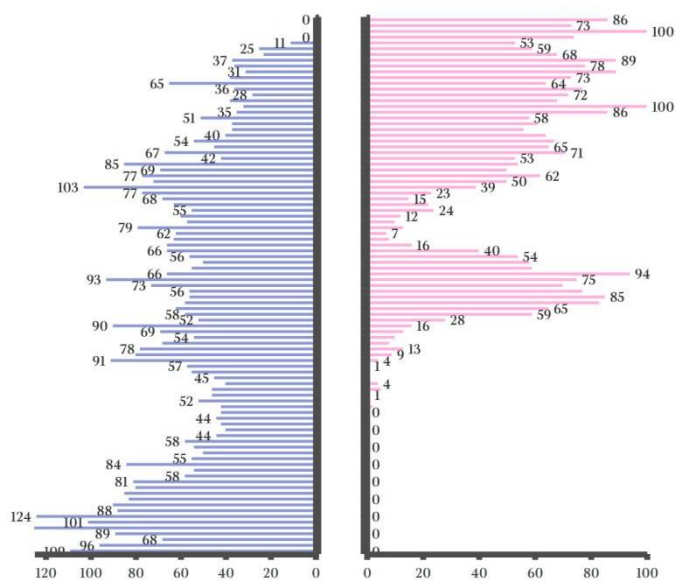


图 19.A 产品春冬日度分析

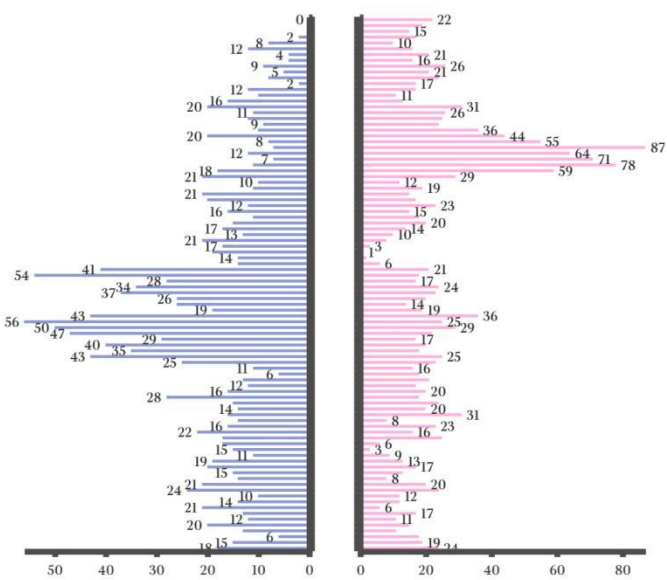


图 20.B 产品春冬日度分析

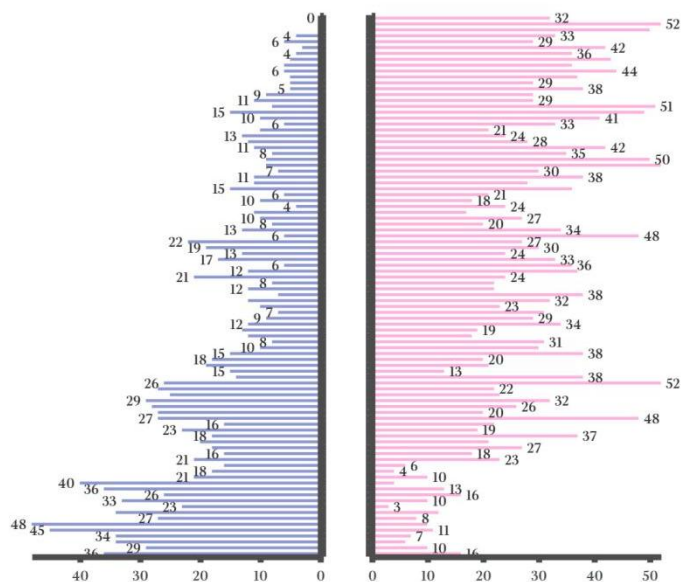


图 21.C 产品春冬日度分析

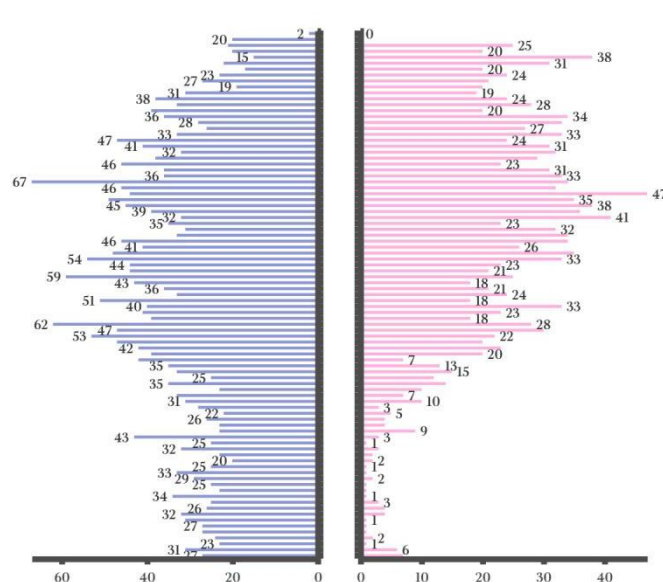


图 22.D 产品春冬日度分析

2.热销产品属性分析

（1）**发货地分析：**该部分主要通过“产品属性”中的“Ships From”进行分析。如表 2 所示，**中国是最大的发货来源。**但是 WiFi 温湿度安全报警器（C 商品）主要从俄罗斯发货。在总体上，捷克也贡献了较多的发货量。

表 2 货源地统计

国家	A	B	C	D	合计
中国	6475	2841	1408	3937	14661
捷克	1006	219	254	280	1759
西班牙	542	139	102	227	1010
俄罗斯	298	838	3136	197	4469
波兰	70	2	8	41	121
法国	118	10	44	39	211
比利时	56	15	10	26	107

（2）**款式分析：**由于涂鸦智能迷你 WiFi 灯开关（B 商品）无款式数据，所以我们仅对三类商品做分析。如表 3 所示，一种商品的不同款式销量相差极大。可以发现，涂鸦 ZigBee/WiFi 家用温湿度传感器（A 商品）、WiFi 温湿度安全报警器（C 商品）、涂鸦 WiFi 智能门开闭探测器（D 商品）**销量最好的都是款式 1**，款式 7 的销量都较为惨淡。具体而言，A 商品的款式 1、3、4，C 商品的款式 1、2，D 商品的款式 1、2，销量都在 1000 件以上，应重点关注。

表 3 热销商品款式分析

	款式 1	款式 2	款式 3	款式 4	款式 5	款式 6	款式 7	款式 8	款式 9	款式 10
A	2987	0	2381	1647	0	853	0	258	0	439
C	1504	1224	806	586	333	176	37	66	13	210
D	1651	1052	578	616	0	363	0	139	0	348

（3）**功能分析：**涂鸦迷你 WiFi 灯开关（B 商品）的商品信息还展示了该产品的功能。如图 23 所示，**无功率监控器与有功功率监控器的销量远大于其他技术产品**，有快速配对功能的产品被选择的最少。

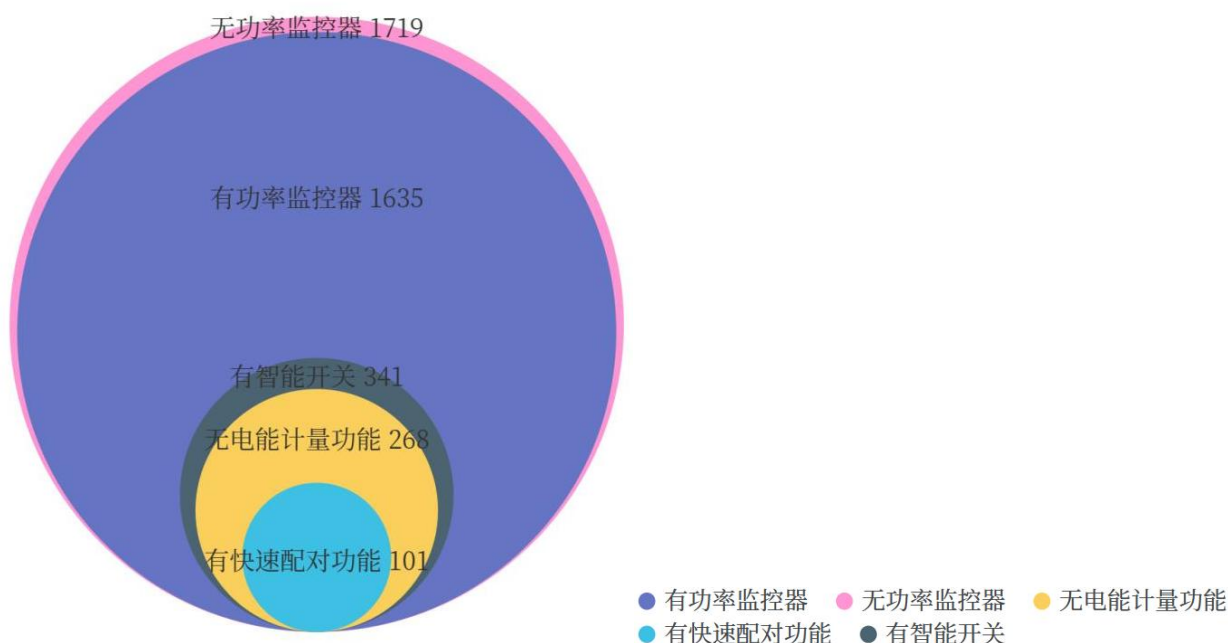


图 23.迷你 WiFi 智能开关属性

3.建议

（1）将同一时期销量不同产品搭配成套餐销售：在 1-2 月，A 产品和 C 产品可以搭配打折销售来弥补 A 产品在冬季销量的不足，B 产品和 D 产品可以搭配打折销售来弥补 D 产品在冬季销量的不足。在 3-4 月，A 产品和 B 产品可以搭配打折销售来弥补 B 产品在春季销量的不足，在 3-4 月，C 产品和 D 产品可以搭配打折销售来弥补 C 产品在春季销量的不足。

（2）应首先考虑选择中国的供应商，其次考虑捷克的供应商，但 WiFi 温湿度安全报警器（C 商品）也可以从俄罗斯进货。

（3）将最高销量款式作为商品上架的主图，保证重点款式的库存：对于涂鸦 ZigBee/WiFi 家用温湿度传感器（A 商品），应将款式 1 作为产品主图，吸引客户点击查看，保证款式 1、3、4 的库存；类似的，应将款式 1 作为 WiFi 温湿度安全报警器（C 商品）的主图，保证款式 1、2 的库存；涂鸦 WiFi 智能门开闭探测器（D 商品）应将款式 1 图片作为主图，同时保证款式 1、2 的库存。

（4）对具有高销量功能的商品采取积极的备货策略：涂鸦迷你 WiFi 灯开关（B 商品）的商品无功率监控器与有功率监控器都较高，所以对无功率监控器功能、有功率监控器的 B 商品都要积极进货，相对来说可以多无功率监控器的 B 商品。“有快速配对功能”虽然在技术上较为前卫，但被选择的并不多，可以考虑少进货。

（三）客户分析

1.客户类型分析

在现代营销理论中，客户价值理论是指客户对企业来说能够创造何种价值，有的客户可以为企业创造更多的价值，这类客户被企业称作为 VIP 客户或者是忠诚度较高的客户，也有部分客户给企业创造的价值较低。学者指出企业应该深入了解客户价值产生创造的过程，并积极利用客户价值理论指导自己的营销活动，以在营销活动中取得更高的有效性。客户价值往往是一个不断发展和变化的过程，客户价值属于动态概念也就是说一个客户在购买和使用产品的过程中会在使用前期、使用中期和使用后期做出改变。为了更为准确的测量客户价值，学者们创造了多种测量客户价

值的方式，其中 RFM 法就是比较典型的代表。RFM 分析，是一种将用户分层、进而针对不同用户群体进行精细化运营的方法。可以通过对用户进行精细化运营，不断将用户转化为重要价值用户。RFM 的三个字母，分别代表了三个维度：R（Recency）：最近一次消费时间。反映了用户最近消费的热度，用以衡量用户是否流失。理论上，最近一次消费时间越长，流失概率越高；F（Frequency）：一段时间内用户的消费频率。反映了用户对于产品、品牌的忠诚程度。理论上，一定时间内的购买频率越高，用户忠诚度越高；M（Monetary）：累积消费金额。反映了用户的购买力。

本报告使用 RMF 分析对店铺买家做客户类型分析，并如表 4 分为以下八类：

表 4RMF 客户分类

	R	F	M
重要价值客户	高	高	高
一般价值客户	高	高	低
重要发展客户	高	低	高
一般发展客户	高	低	低
重要保持客户	低	高	高
一般保持客户	低	高	低
重要挽留客户	低	低	高
一般挽留客户	低	低	低

首先剔除订单状态为“订单关闭”、“冻结中”、“买家申请取消订单”的数据以及无买家姓名数据，最后用于分析数据共 39096 条，共 31663 个不同的买家。M 值使用“产品总金额”，因为数据统计的最后下单日期为 2023/4/31，所以在 R 值计算时，设定的参考日期为 2023/5/01。根据数据的分布特征，本报告使用中位数来判定 RMF 值的高低，R 的中位数为 88 天，F 的中位数为 1 次，M 的中位数为 18.26 美元。

如图 24 所示，在所有客户类型中，占比最高的是“一般挽留客户”，比例高达 30.96%；占比最低的是“一般保持客户”，比例仅为 1.39%。在高价值客户中，占比最高的是“重要挽留客户”，比例为 17.71%；占比最低的是“重要保持客户”。这说明在样本数据时间区间内，本店铺多数客户价值不高，即最后一次消费时间较为久远，消费频次不高，且累积消费金额较低。占比 6.70%重要价值用户却贡献了 83.62%的销售额，说明重要价值用户对店铺来说至关重要。重要保持客户占比 4.00%，贡献了 5.84%的销售额；重要挽留客户占比 17.71%，贡献了 3.99%的销售额；重要发展客户占比 12.92%，贡献了 2.77%的销售额。

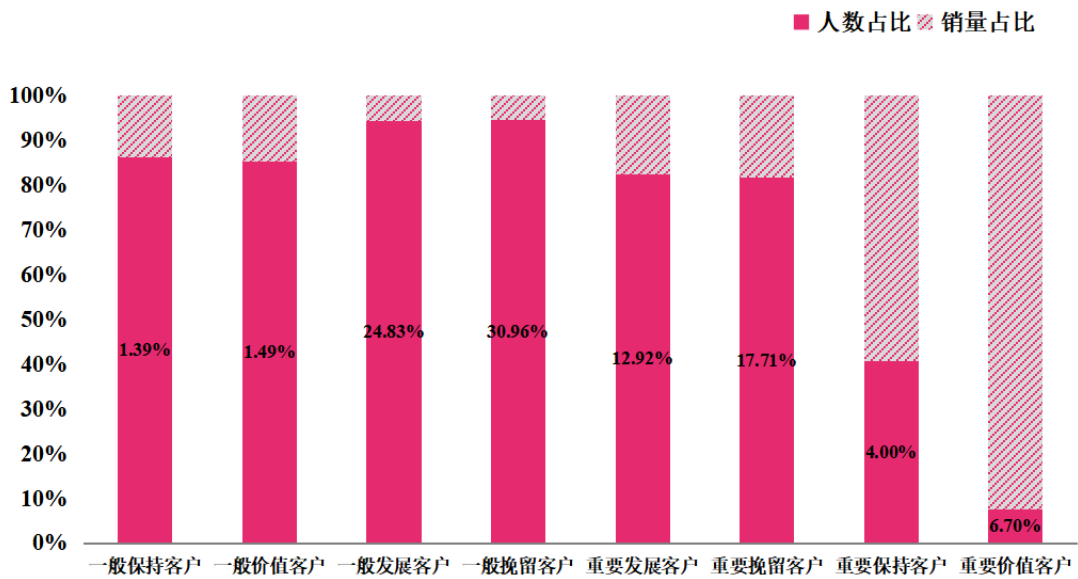


图 24.不同类型客户的消费情况

2.复购率分析

剔除订单状态为“订单关闭”、“冻结中”、“买家申请取消订单”的数据，剩下 39098 条数据。发货后的购买行为算复购，复购率=复购次数大于 0 的买家/买家总人数。共有 31,663 个不同的买家，3,273 个买家复购，复购率约为 10.34%。由于复购超过 10 次的买家极少，参考意义不大，所以图 25 仅展示复购 1-10 次的客户数量及贡献的销量。复购 1 次的买家最多，高达 2672 人，对总销量贡献了 15.23%。如表 5 所示，在复购次数前十的商品中可分为五类，家用温湿度传感器、智能灯开关、智能门窗传感器、USB 加密狗、智能灯。可以发现，复购率高的产品同时都是热销产品，这几类产品的共同的特点是更为智能化，可以通过手机 APP 或谷歌助手来控制。这种便捷性显著提升了客户生活的幸福感。

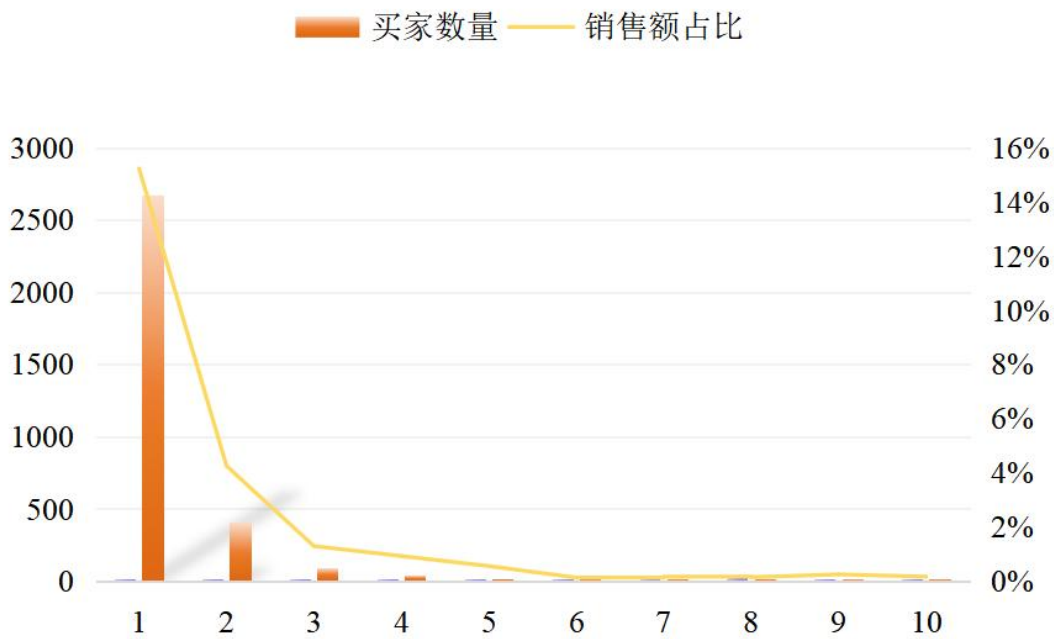


图 25.不同复购次数客户帕累托图

表 5 复购次数前十名产品类型

复购次数	产品类型	特点
490	家用温湿度传感器	可联网 与谷歌助手兼容
207	智能灯开关	可联网 语音控制 迷你 涂鸦品牌
125	智能门窗传感器	APP 控制
111	USB 加密狗	适用于多个场景
80	智能灯	可联网 与谷歌助手兼容

3.国别分析

本报告首先统计了各国消费者的购买次数，结果如图 26 所示，俄罗斯、巴西、德国、法国、西班牙消费者的购买次数最多。本报告还统计了重要客户、复购人数综合前五的国家，如表 6 显示，本店铺的重要客户集中在欧洲和南美地区，尤其是需要关注俄罗斯、德国、法国地区客户的需求。所以，我们对俄罗斯、德国、法国做进一步分析。

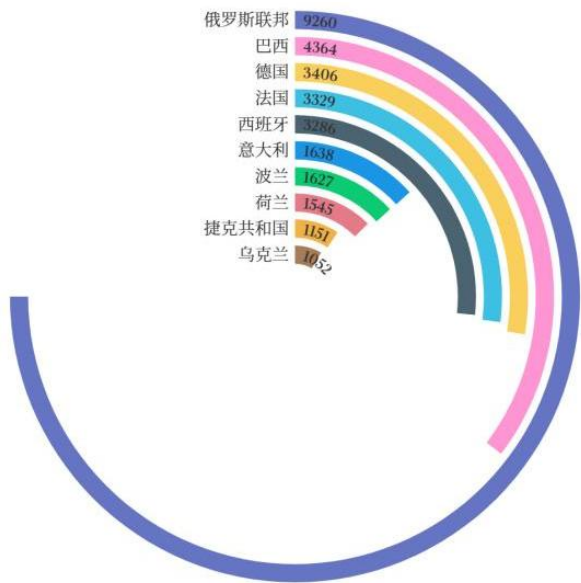


图 26.各国消费次数

表 6 各国客户类型

国家	重要价值客户	重要保持客户	重要挽留客户	重要发展客户	复购人数
俄罗斯	280	318	1027	297	697
德国	226	130	496	504	315
法国	195	107	539	461	272
巴西	115	107	604	266	222
西班牙	171	87	396	367	238

(1) 俄罗斯

首先对物流进行分析：如图 27 所示，RU 表示与俄罗斯邮政直接合作，OD 表示按需配送，PUDO 表示按需在自提点取包裹。可以发现，有两种物流与俄罗斯邮政有直接合作关系的，有四种是阿里巴巴集团提供的物流服务。在所有物流中，选择次数最多的是“卖方送货方式-RU”，高达 7512 次，说明俄罗斯客户偏爱与当地邮政直接合作的卖方送货方式。接下来对俄罗斯消费者的产品偏好进行分析：产品购买次数最高的前五名如图 28 所示。可知 WiFi 温湿度安全报警器、涂鸦 ZigBee/WiFi 家用温湿度传感器、涂鸦智能迷你 WiFi 灯开关、ZigBee 水泄漏传感器、涂鸦 WiFi 智能门窗传感器是俄罗斯最热销的产品。

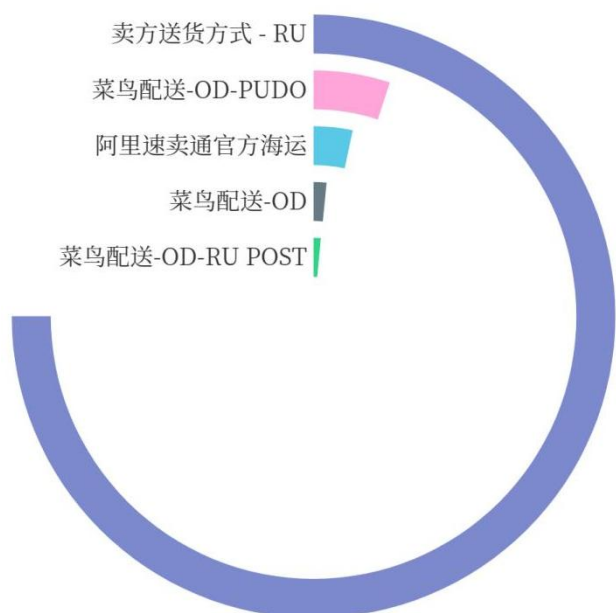


图 27.俄罗斯物流偏好

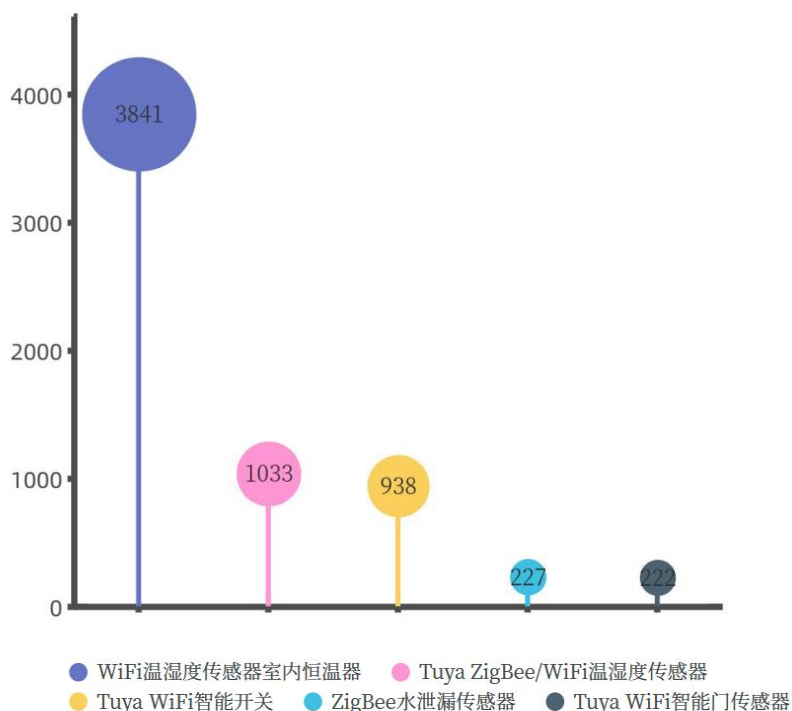


图 28.俄罗斯产品偏好

(2) 德国

德国的消费者与俄罗斯消费者呈现出不同的物流偏好。如图 29 所示，“阿里速卖通标准送货”被选择次数最多，高达 1776 次。“阿里速卖通官方海运”也高达 694 次，而“卖方送货方式-ES”仅排第三，被选择 390 次。说明德国消费者比较信任阿里巴巴官方物流。德国消费者产品购买次数最高的前五名如图 30 所示，可知涂鸦 ZigBee/WiFi 家用温湿度传感器、涂鸦 WiFi 智能门窗传感器、涂鸦智能迷你 WiFi 灯开关、智能插头插座、USB 加密狗是德国最热销的产品。

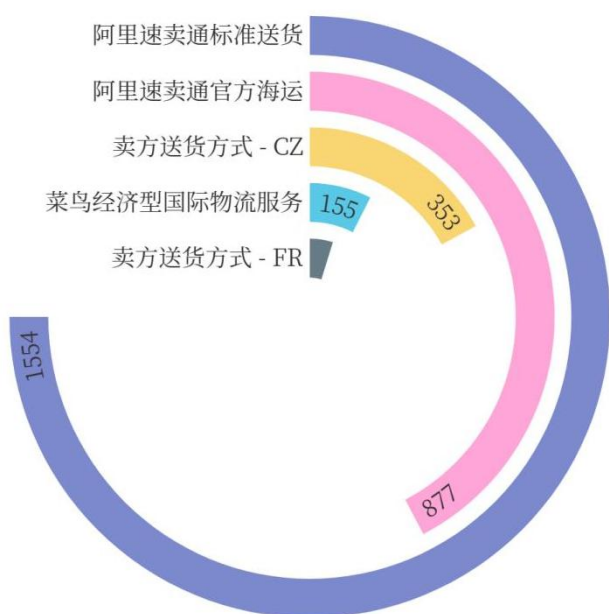


图 29.德国物流偏好

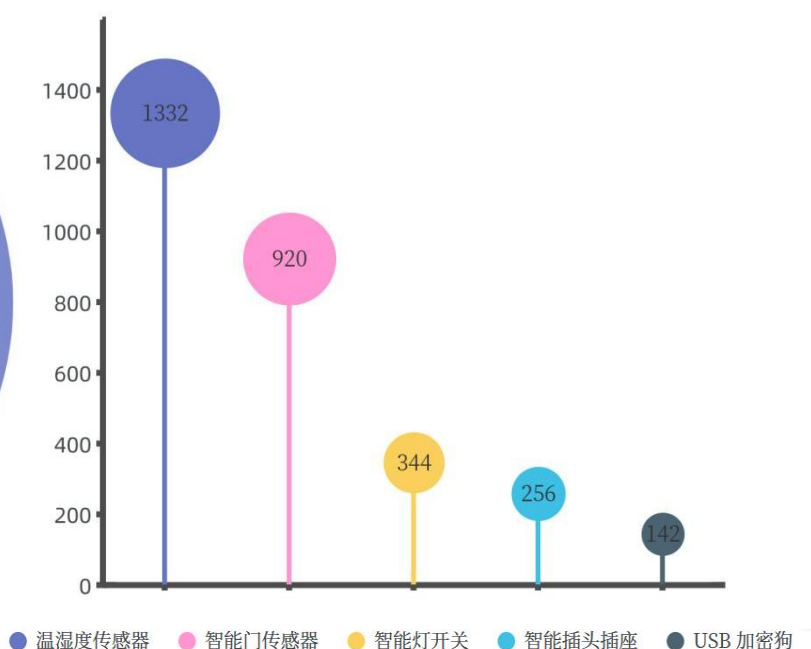


图 30.德国产品偏好

(3) 法国

如图 31 所示，法国的消费者选择次数最多的物流方式是“阿里速卖通标准送货”，高达 1554 次；“阿里速卖通官方海运”也高达 877 次，而卖方送货方式仅排第三和第五，说明法国消费者和德国消费者一样，比较信任阿里巴巴

官方物流。法国消费者产品购买次数最高的前五名如图 32 所示。和德国一样涂鸦 ZigBee/WiFi 家用温湿度传感器、涂鸦 WiFi 智能门窗传感器、涂鸦智能迷你 WiFi 灯开关、智能插头插座、USB 加密狗是法国最热销的产品。

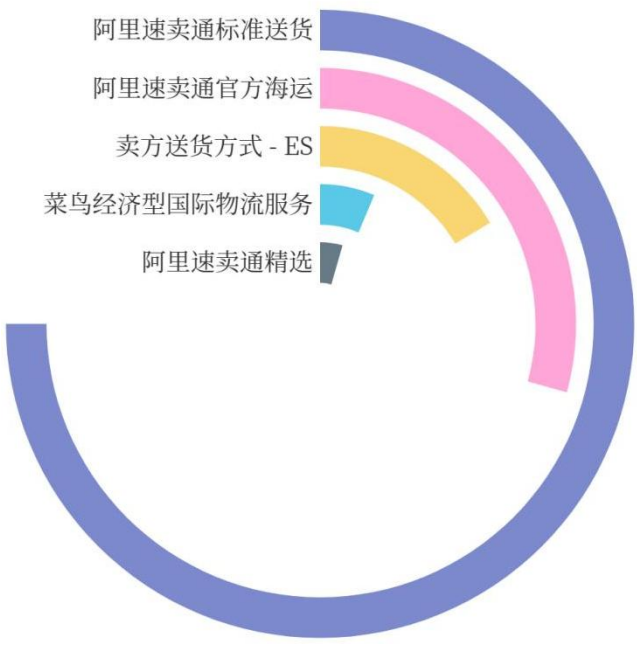


图 31.法国物流偏好

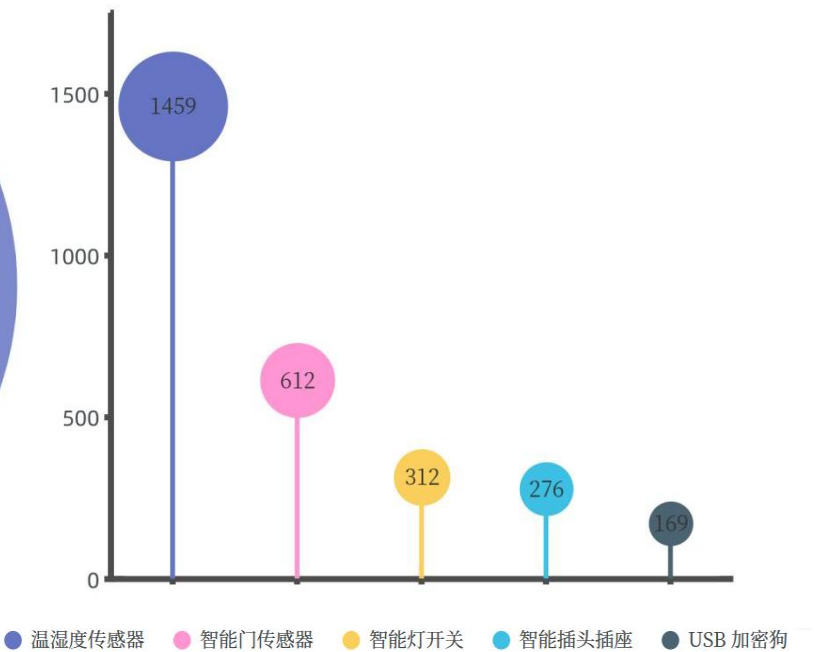


图 32.法国产品偏好

4.建议：

（1）**店铺应对不同类型的客户开展不同的策略。**对重要价值用户：应提供 VIP 服务，比如专属优惠、定制产品、优先客服等，以保持其忠诚度和活跃度。对重要保持客户：该类型客户是一段没来的忠实客户，建议通过个性化邮件、社交媒体互动等方式与他们保持联系，并提供回归优惠来提高复购率。对重要挽留客户：该类型客户曾经消费金额高，这种用户即将流失，建议通过调查问卷了解他们的需求和不满因素，然后提供解决方案或优惠，努力挽回。对重要发展客户：该类型客户消费频率低，可以通过推送相关产品的营销活动、提供首次使用优惠、引入会员制度等策略来提高其消费频率。

（2）**对于复购率高的商品，推出再次购买的限时优惠券。**家用温湿度传感器、智能灯开关、智能门窗传感器、USB 加密狗、智能灯是复购率最高的五类商品，在消费者收货后，发放限时优惠券，刺激消费者再次购买。

（3）**对不同国家实行不同的广告投放策略。**虽然俄罗斯、德国、法国都是欧洲国家，但是俄罗斯与德法的产品偏好有明显的差异。所以在做产品推广时，应注意各个国家的特性。

（四）销售预测

为了使分析结果可读性更强，我们主要对水泄漏传感器、红外遥控器、空气质量传感器、智能门窗传感器、家居安防摄像机、智能灯、智能开关、智能插座、温湿度传感器九类商品进行销售预测分析。为了便于建模分析，我们利用 Python 将空缺日期的销量填充为 0。由于预测模型的误差是逐步累积的，考虑到随着时间的增加预测模型的误差越来越大，我们将 7 天为一个周期进行计算（该部分代码如在附录中展示）。

1.模型建立

为了更好的预测结果，我们选取了三种数学模型进行建模，最终根据数据的合理性选取模型，下面是三种模型：

(1) ARIMA 模型

ARIMA 模型实质上为组合模型，由自回归（AR）模型、移动平均（MA）模型和差分模型三部分组成，ARIMA 模型如式(1)所示：

$$ARIMA(p,d,q) = AR(p) + difference(d) + MA(q) \quad (1)$$

AR 模型的理论假设是现值是历史值的加权求和，定义如式(2)：

$$X_t = \varphi_1 X_{t-1} + \varphi_2 X_{t-2} + \dots + \varphi_p X_{t-p} + \varepsilon_t \quad (2)$$

可以证明，AR 模型的特征方程的特征根落在单位圆之外才能保证模型具有平稳性。MA 模型代表了白噪声的移动平均，定义如式(3)：

$$X_t = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \quad (3)$$

MA 模型总是平稳的，若结合了 d 阶差分模型，则 ARIMA 模型平稳的充分必要条件是 AR 模型平稳。综上所述，ARIMA 模型的理论定义如式(4)：

$$\nabla^d X_t = \varphi_1 X_{t-1} + \varphi_2 X_{t-2} + \dots + \varphi_p X_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \quad (4)$$

(2) LSTM 模型

如图 33 所示，LSTM 网络的每一个单独的单元模块都由输入门、记忆门、遗忘门和输出门组成，在模型参数上新增单元的状态 c_t ，遗忘门负责选择性记忆前一个单元的信息 a_{t-1} ，可通过参数的选择来更改其记忆能力，遗忘门计算公式如式(5)：

$$\nabla^d X_t = \varphi_1 X_{t-1} + \varphi_2 X_{t-2} + \dots + \varphi_p X_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \quad (5)$$

其中 σ 表示 sigmoid 函数， W_f 和 b_f 为权重和偏置，LSTM 将输入变成了 $[a_{t-1}, x_t]$ 的形式，将两者压缩在一个向量内。记忆门则负责保留多少的信息给 c_f ，公式如式(6)：

$$\begin{cases} \tilde{c}_t = \tanh(W_c * [a_{t-1}, x_t] + b_c) \\ \Gamma_i^t = \sigma(W_i * [a_{t-1}, x_t] + b_i) \\ c_t = \Gamma_f^t * c_{t-1} + \Gamma_i^t * \tilde{c}_t \end{cases} \quad (6)$$

由上式可见，当前单元的状态 c_t 既包含了上一个单元的状态 c_{t-1} 和信息 a_{t-1} ，也包含了本单元的输入 x_t 。输出门负责将本单元的信息与状态传入下一个单元，输出公式如式(7)：

$$\begin{cases} \Gamma_o^t = \sigma(W_o * [a_{t-1}, x_t] + b_o) \\ a_t = \Gamma_o^t * \tanh(c_t) \end{cases} \quad (7)$$

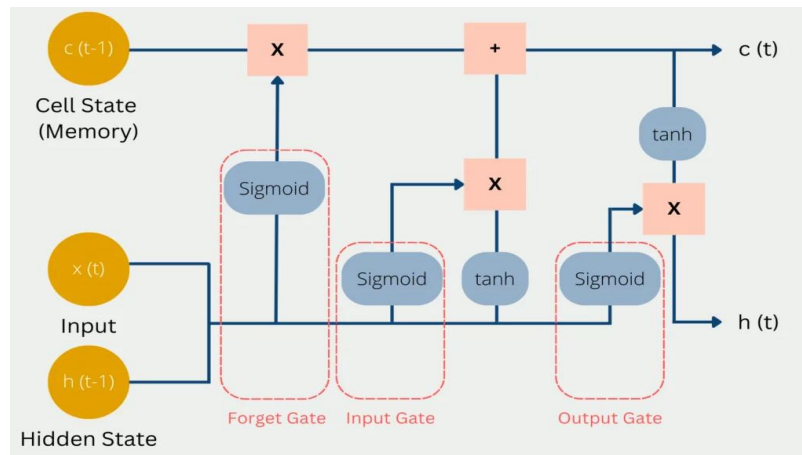


图 33.LSTM 模型结构

(3) Prophet 模型

Prophet 算法是一种差异化预测模型，即模型可以分解为趋势项、季节期项、假日效应等组成部分：

$$P(t)=g(t)+s(t)+h(t)+\varepsilon_t \tag{8}$$

其中， $g(t)$ 函数是分析时间序列的增长趋势的，里面有两个函数可供选择，一个是基于逻辑回归函数，另一个是基于分段线性函数。 $s(t)$ 是周期项，时间序列可以包含各种类型周期的季节性趋势。Prophet 使用傅里叶级数来模拟时间序列的周期性。 $h(t)$ 是节假日项，节假日或一些重大事件会对时间序列产生较大的影响，这些时间点往往不具有周期性。Prophet 将每个假期在不同时间点的影响模拟为一个独立的模型。

为了更好地预测数据，找到合适的参数设置，我们在代码中设置了以 MAPE、MSE 为参考标准的网格寻参。

2.预测结果

为了保证结果的合理性，我们使用平均绝对百分误差（MAPE）和均方误差（MSE）衡量模型预测精度。MAPE 考虑的是预测误差占真实值的比重加权，MSE 用于量化预测值与实际值之间的差异，MAPE 计算方式如公式(9)所示：

$$MAPE = \sum_{i=1}^n (|(F_i - X_i) / X_i| * 100\%) / N \tag{9}$$

其中， F_i 表示的是第 i 个预测值， X_i 代表的第 i 个观察到的实际值， N 表示的是预测对象的总数量。

MAPE 计算方式如公式(10)所示，

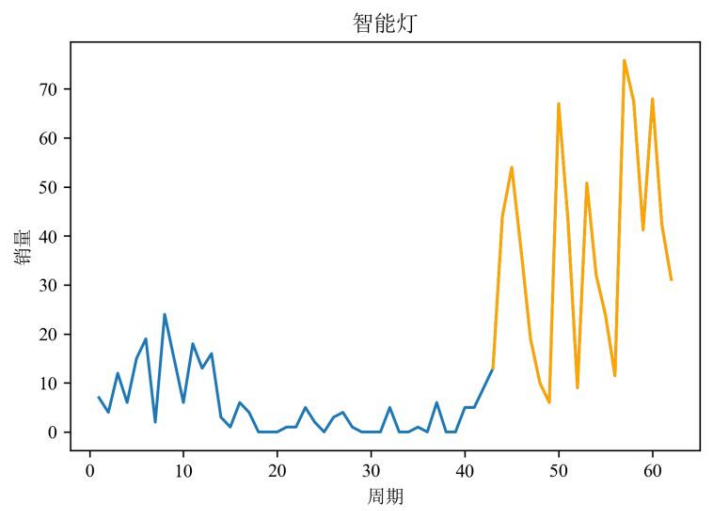
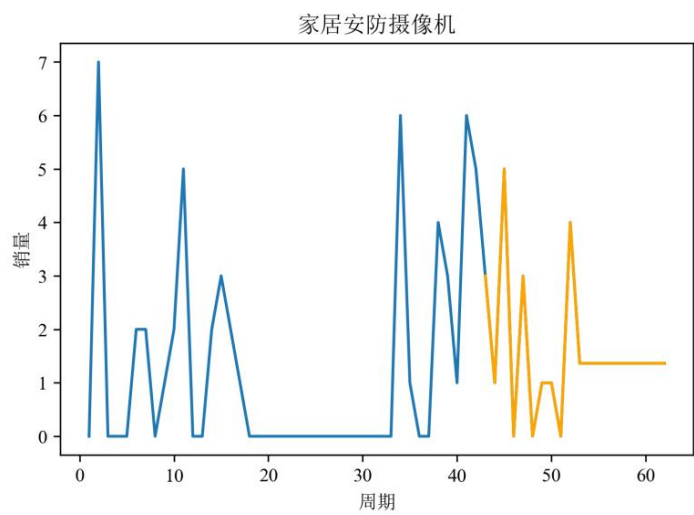
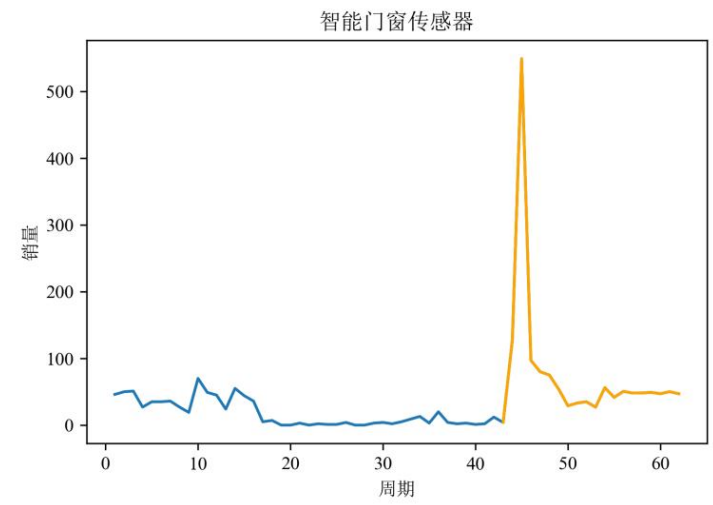
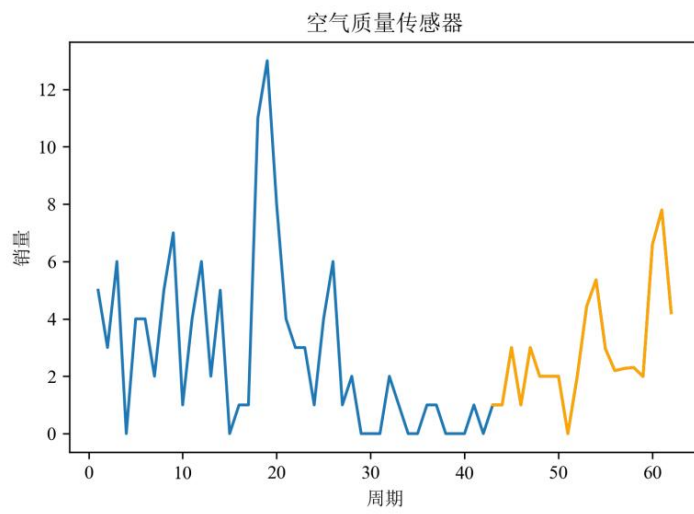
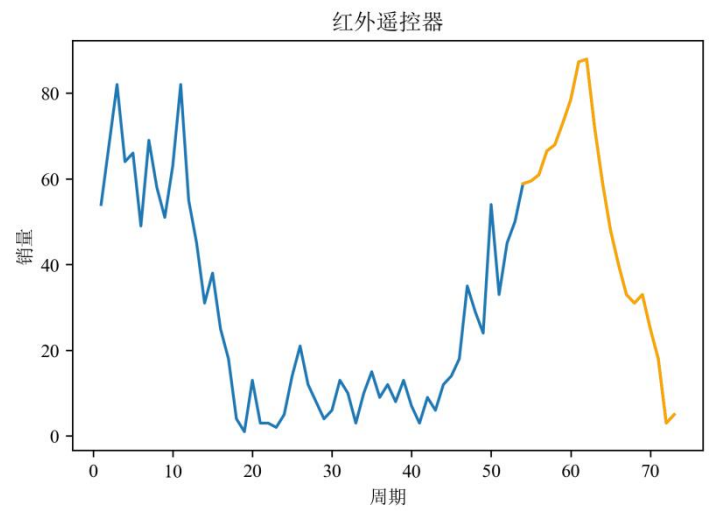
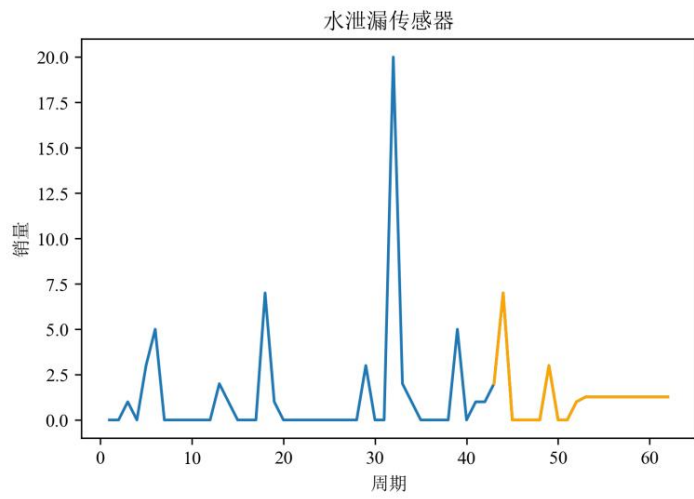
$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \bar{Y}_i)^2 \tag{10}$$

其中， n 是样本数量， Y_i 是第 i 个观察到的实际值， \bar{Y}_i 是第 i 个预测值。

MAPE 和 MSE 的值越小，表示预测模型的预测值与实际值之间的差异越小，即模型的预测精度越高。我们选择 MAPE 和 MSE 最小的模型进行预测，模型选择结果如表 7 所示。为了更加直观地展现结果，我们将预测价格可视化，如图 34-41 所示，其中蓝色部分表示已有数据，黄色部分表示预测值。

表 7 模型选择

产品	ARIMA 模型		LSTM 模型		Prophet 模型		最终选择
	MAPE	MSE	MAPE	MSE	MAPE	MSE	
水泄漏传感	1.52e-05	0.4993	4562264.21	595.51	36773934.22	1.98	ARIMA 模型
智能门窗传	33787.15	28106.20	118.21	296.38	1049074.10	292.30	LSTM 模型
智能开关	32.57	91091.3	20.35	11127.59	20.35	11227.72	LSTM 模型
智能插座	259.01	4412.9	115876.15	175.87	24.32	115.66	Prophet 模型
温湿度传感	168.56	86570.0	2333451.05	4444.64	30.96	3980.84	Prophet 模型
智能灯	5422.44	450.12	18886230.16	141.38	98.54	58.81	Prophet 模型
空气质量传	7419182.83	0.8382	2867.58	1.11	41.88	0.48	Prophet 模型
家居安防摄	无穷大	3.1380	20719732.08	25.98	1890.46	2.26	Prophet 模型
红外遥控器	24046105.49	463.7273	9335.25	144.73	17.07	13.64	Prophet 模型



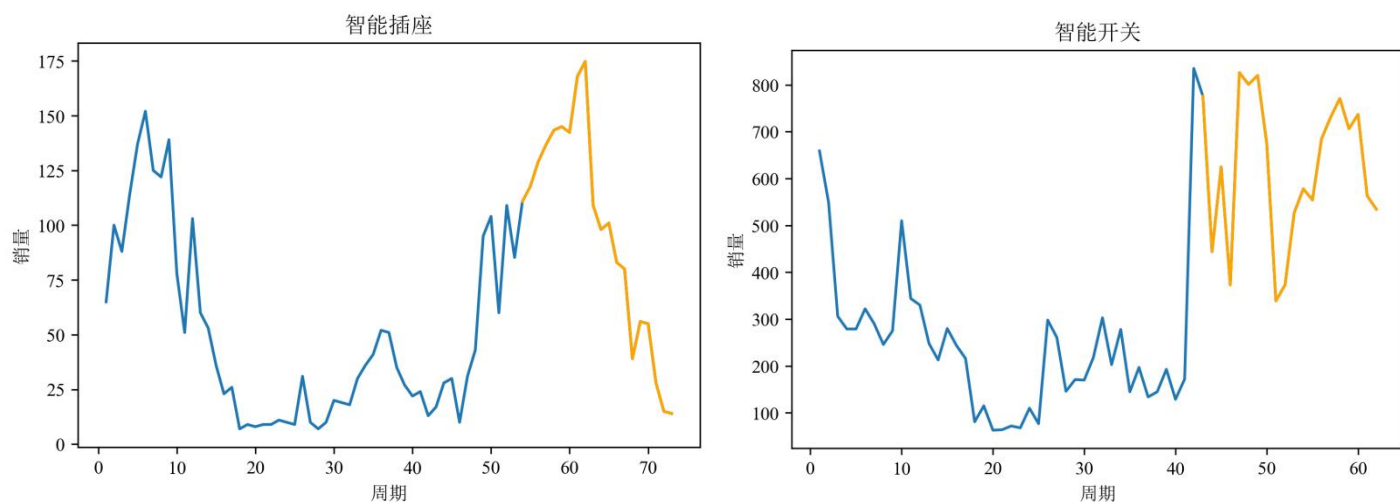


图 34-41.主要商品销售预测

3.建议

- (1) 商品均有很强的周期性，应根据预测销量周期性地积极调整备货数量。
- (2) 对未来销量高的产品充足备货并进行调研。通过预测发现家具安防摄像机和空气质量传感器销量比较低迷，建议商家可以采取低备货量策略，以免对仓库造成浪费。对于温湿度传感器和智能开关销量较高的商品多备货，以满足更多顾客的需求。

四、运营提升分析

(一) 滞销产品处理

通过分析订单数据发现，恒温散热器阀门、蓝牙指按式机器人开关、家居安防摄像机是滞销最严重的商品。针对这种情况，我们建议做出如下处理方式：

- 1.“顺手带一件”。让消费者能以便宜的价格加购滞销产品。可以在店铺其他商品页面和付款页面都加上选项，显示滞销商品原价与折后价，消费者可以在购买其他商品时考虑加购一件打折后的滞销商品。
- 2.将滞销产品出售给清仓批发商或转售商。通过该方式可以降低库存成本，用减少库存积压带来的资金压力。
- 3.停止生产或移除产品线。如果滞销产品的销售情况长期不佳，并且经过多次尝试仍无法改善，可能需要考虑这种方式。

(二) 备货建议

采购备货是根据当前企业销量、库存、备货时效等情况，预测未来一段时间的备货需求，并由采购负责下单，物流进行中转调拨，仓储进行收货上架等流程完成整个备货需求执行的业务过程。目的是为了满足不同企业销售运营的产品供应需要，提高库存周转率，同时减少缺货导致的客户体验差、减少滞销带来的资金占用压力，达到采销平衡的核心目标。不同的商品特征，需要匹配不同的备货策略。我们选择 ABC 管理法和 XYZ 库存管理方法结合分析。

ABC 管理法是管理库存的经典方法。通过计算每个商品的销售收入在所有商品产生的总收入中的累积百分比贡

献进行排名，来对商品进行分类，它基于帕累托原则（80/20 规则）：A 类商品贡献了 80%的销售收入，这些商品需要严格控制库存，避免缺货。B 类商品贡献了接下来的 10%的销售收入，对于这类商品，库存控制可以适当放松。剩下的 10%收入由 C 类商品贡献，但 C 类商品种类最多，分配到的进货成本和仓储资源优先级最低。如表 9 所示，XYZ 库存管理方法是一种基于产品销售频率和重要性的库存分类方法。这种方法将库存产品分为三个类别：X 类、Y 类和 Z 类，每个类别都有不同的管理策略。将 ABC 与 XYZ 组合起来，如图 42 所示，可以把所有商品可以分为 AX、AY、AZ、BX、BY、BZ、CX、CY 和 CZ 九个类别。

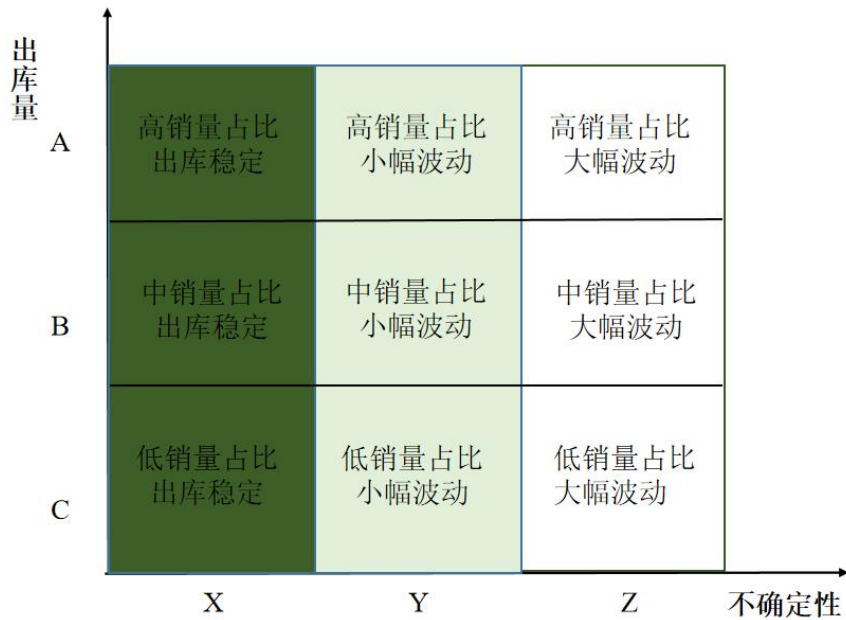


图 42.组合备货商品分类

本店铺商品分类以及相应的备货建议如表 8 所示。需要注意的是，因为本店铺产品基本都有较强的季节性，所以出库量稳定只是在本店相对稳定。总体来说，即使处于 X 类（出库稳定）的产品，也应把备货重点放在春冬季节，夏秋季节不至于紧缺即可。

表 8 备货商品分类

类别	商品	备货建议
AX	涂鸦 ZigBee/WiFi 家用温湿度传感器 WiFi 温湿度安全报警器	采用积极的库存管理，保持较高的库存水平，以实现规模经济和减少补货频率
AY	ZigBee USB 加密狗 涂鸦 WiFi 开关 开关继电器模块 涂鸦智能灯	采用灵活的库存管理，需要更频繁的库存复查
AZ	涂鸦 ZigBee/WiFi 智能插头插座 涂鸦 WiFi 智能门开闭探测器 ZigBee 门窗探测报警器	采用谨慎库存管理，重点关注市场动态和客户需求
BX	ZigBee 水泄漏传感器 涂鸦 16A WiFi 智能开关	采用适度库存策略，可以采用定期复查的方式来调整库存策略
BY	家居安防摄像机	实施安全库存策略，根据市场趋势和销售数据灵活调整库存
BZ	红外遥控器	采取保守库存策略，考虑实施更灵活的生产计划和紧急补货机制
CX	运动传感器	库存管理可以相对宽松，但仍需注意库存的周转速度
CY	蓝牙指按式机器人开关	按需采购来减少库存风险
CZ	恒温散热器阀门	按需采购和最小化库存，减少库存积压

（三）直通车

直通车即 pay for performance，按点击付费，通过自助设置多维度展示产品信息，获得大量曝光吸引潜在买家。首先，我们应该确定预算目标。如图 43 所示，本店铺目前处于成长阶段，应专注于优化转化率、精准引流、打造爆品。根据前文的分析，本店铺的重点市场在欧洲和南美州。南美洲的出价时间从 20 点开始，溢价系数在溢价后高于行业平均出价；欧洲 16 点开始，24 点关，按 18：00-20：00 观察行业 TOP10 出价，要比行业平均出价高。为了展示地更清晰，不同的产品的具体投放方案如表 9 所示。需要注意的是，应从宽泛匹配改为精准词组匹配，以避免与同类商品同质性冲突，触及更精准流量。

具体投放过程中，本报告有如下建议：第一，一个产品建议只放一条计划推广；第二：不同产品类目尽量分计划推；第三，智能计划不建议加太多词，只添加单独出价的核心词即可，收集关键词的渠道主要有：平台搜索下拉框、发品时的智能推荐词、直通车系统推荐词、同行使用的关键词、访客详情等，注意要“冷热搭配”，热度高和热度低的词相结合，做到关键词全面覆盖。

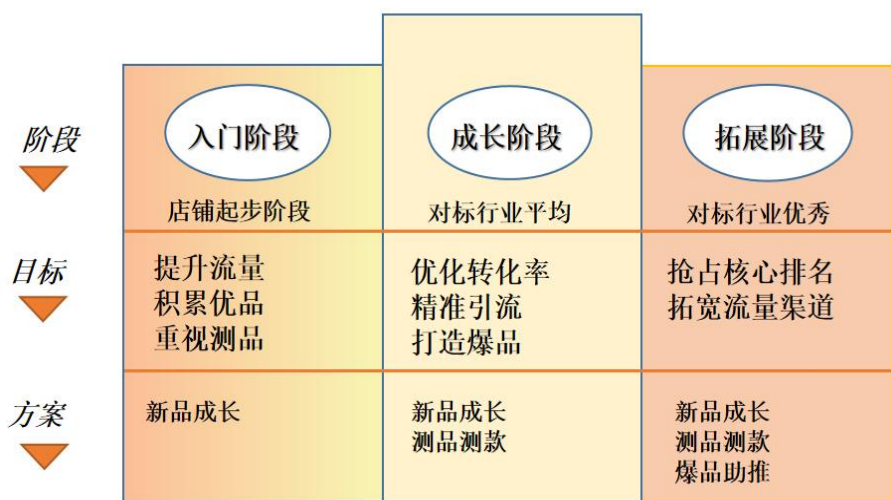


图 43.不同阶段店铺的投放方案

表 9 不同产品推广方案

	新品成长	测品测款	爆品助推	关键词推广	定向推广
目标	测试新产品的潜力如何	为目前销量最优产品获取流量	对店铺最优产品进行持续推广	通过词的覆盖来获取精准流量	对转化率表现不错的产品堆积流量
选词	可不加	前期系统匹配，后期采用核心词根	关注核心词/询盘词	尽可能覆盖产品类目全部三星以上词	关注核心词/询盘词/买家搜索词
选品	新品/未测试过的老品	精准流量即将打造的爆品	数据最佳的爆品或准爆品	数据较好的产品	数据较好的产品

（四）供应商选择

在挑选供应商时可以采用 5R 原则，如图 44 所示，适时（Right Time）、实质（Right Quality）、适量（Right Quantity）、适价（Right Price）、适地（Right Place）的从供应商手中挑选产品。



图 44.5R 原则

在前文分析可知，本店铺的发货地主要在中国，所以我们从 1688 官网上寻找支持外贸订单、加工定制的中国厂家。经过我们的仔细筛选，生产产品较为匹配且综合评价较好的六家可备选供应商的信息如表 10 所示。和其他厂家对比来看，我们认为**深圳市文汇科技发展有限公司**是最佳的供货商选择。

深圳市文汇科技发展有限公司专业研发生产销售智能家居类产品，品类齐全，现货充足，尤其在智能插座、智能开关、智能通断器和无线控制类产品有着较高的品控和质量，可根据客户定制开发，承接各种 OEM、ODM 订单，**有丰富的海内外客户合作经验**。该公司还是 1688 网站 2024 上榜三钻工厂、智能服务家居排行榜第一名，拥有十项生产专利。最近 90 天累计成交笔数高达 4175 笔，最近 30 天 48 小时内揽收率高达 95.45%，满意率达 97.83%。

表 10 供应商对比

序号	供货商	基本信息					经营状况		
		成立时间	地址	厂房面积 (m^2)	诚信等级	对外贸易 资质证书 数量	回头率 (%)	准时履约 率 (%)	年交易额 (万元)
1	深圳市文汇科技发展有限公司	2018	广东深圳	2115	AAA	6	57	100	2000 以上
2	深圳市奥莱斯特塑胶电子有限公司	2010	广东深圳	660	AA	1	16	100	2000 以上
3	厦门柏益加节能科技有限公司	2015	福建厦门	2180	AAA	2	72	100	101~500
4	温州市正尚电子有限公司	2011	浙江温州	2600	AAA	3	36	89	2000 以上
5	深圳市创刻安防技术有限公司	2015	广东深圳	无数据	AAA	1	60	100	2000 以上
6	深圳市睿易安智能有限公司	2008	广东深圳	无数据	A	1	54	无数据	2000 以上

（五）价格优化

市场经济中的任何参与者无一不在追求各自经济利益最大化。价格策略是一种典型的市场经济行为，因此也必须瞄准利益最大化目标。然而产品有不同特性单一的定价方法可能会造成一定损失。定价又是关乎其企业形象、利润获取以及展示自身竞争姿态的重要讯号，因此，执行差异化定价策略势在必行。竞争因素在真实的零售市场中是必然存在且无可避免的，而且显而易见地会影响商品的需求。我们将考虑竞争情形应用于后续的价格优化模型。

首先我们对本店销售的商品类型进行归类。根据费雪的分类标准，以使用时长、商品价值与重复购买率将产品划分为易耗品与耐用品，总结来说，耐用品有以下特点：（1）产品的单位价值相对来说较高（2）产品的生命周期较长，可以长期反复使用（3）重复购买率低。对于耐用品，消费者往往更加关注产品的质量、口碑等。所以消费者在对商品进行购买决策前，总会存在一个“货比三家”的过程，消费者的选择偏好受到诸如商品价格、商品质量与售后服务等多种特征的影响。正因如此，对于零售企业来说，怎样刻画消费者的需求与选择，从而把握消费者的喜好并进行定价，是向消费者传递服务理念，塑造消费者忠诚度，从而提升其市场回报的重要一环。

参考 2021 年柴昭《竞争情况下在线零售商不同类型产品需求与定价模型研究》，我们建议选择 MNL(Multinomial Logit Model) 刻画消费者购买商品时选择决策行为，从而进行价格优化。

MNL 模型可以用来刻画线上零售商的内部与外部竞争，也就是描述消费者在线上零售商 S 提供的某一个商品组中，或者是不同零售商之间进行选择，该商品组或者零售商的组合，也就是上面提到的选择集合，商品组一共包含件商品或 n 个不同的零售商。下面，具体描述模型的效用与选择概率的计算。在选择过程前，消费者对于商品效用的刻画，一般会包括两个部分：可以观测到的效用以及不可观测到的效用。效用用 u_{ij} 来表示 $u_{ij} = \mu_{ij} + \epsilon_{ij}$ 。其中 i 表示的是消费者， j 表示的是选择集合中的元素，在本文中也就是产品 j 。本文将可观测部分的需求用线性形式表示： $\mu_{ij} = \alpha_j + \beta_j p_j$ ，这表明价格是影响消费者进行决策选择的重要因素。

具体来说，消费者 i 选择购买产品 j 的效用如式(11)：

$$u_{ij} = \alpha_j + \beta_j p_j + \epsilon_{ij} \quad (11)$$

消费者 i 外部选择的效用（不购买任何一件产品的效用）如式(12)：

$$u_{i0} = \mu_0 + \epsilon_{i0} \quad (12)$$

不失一般性，本文假设外部选择中 $\mu_0 = 0$ ，基于这一假设，我们有式(13)：

$$u_{i0} = \epsilon_{i0} \quad (13)$$

在上述效用刻画中，涉及到的符号说明如下： α_j 是消费者对于产品 i 的偏好； β_j 是产品 j 的价格弹性系数，即在其他因素均相等的情况下，对数价格每增长 1 单位而导致的效用的改变； p_j 是产品 j 的销售价格。

假设 ϵ_{ij} 与 ϵ_{i0} 独立同分布于耿贝尔分布（Gumbel Distribution）。耿贝尔分布的累积分布函数是 $F(x) = e^{-e^{-x/\eta + \gamma}}$ ，其中 $\gamma = 0.5772$ 是欧拉常数，来确保波动项 ϵ 的均值为 0。在以上效用的刻画下，按照随机效用理论，消费者在进行购买时，选择的是使得自身所获效用最大的产品。因此，其选择产品 j 的概率为式(14)：

$$q_j = P(u_{ij} > u_{ik}, k \neq j) = \exp(\alpha_j + \beta_j p_j) / (1 + \sum_{k=1}^n \exp(\alpha_k + \beta_k p_k)) \quad (14)$$

当购买任何一件商品的效用低于不购买商品的效用时，外部选择（不购买任何一件产品）的概率为式(15)：

$$q_0 = 1 / (1 + \sum_{k=1}^n \exp(\alpha_k + \beta_k p_k)) \quad (15)$$

而如果知道该品类产品的总的市场规模 M ，用市场规模 M 乘以产品选择概率，就是产品对应的需求 D_j ，如式(16)所示：

$$D_j = M * q_j = M * \exp(\alpha_j + \beta_j p_j) / (1 + \sum_{k=1}^n \exp(\alpha_k + \beta_k p_k)) \quad (16)$$

我们可以对店铺各品类的产品数据进行梳理,以商品组的总访问量作为商品组的总的市场规模。正如式(13)与式(14)分别描述的消费者选择产品 j 与不进行购买的概率公式所列,我们发现二者相除可以消除分母的多项式,达到式(17)简化的效果。

$$D_j/D_0 = (M * q_j)/(M * q_0) = \exp(\alpha_j + \beta_j p_j) \quad (17)$$

对上面的式子两边同时取对数,可以得到线性形式的式(18):

$$\text{Log}(D_j/D_0) = \log(\exp(\alpha_j + \beta_j p_j)) = \alpha_j + \beta_j p_j \quad (18)$$

在这样的情况下,对模型的参数进行估计的时候,只需要对销量 D 与未购买的数量之比进行对数处理,便可以采用线性回归进行拟合。这样看来,MNL 模型得到的需求,与回归模型部分指数形式的需求模型颇为接近。由于目前的订单数据缺少部分关键数据,我们无法直接进行价格优化的最终结果计算,但是该模型可作为参考。

五、数字营销实践

(一) 目标市场选择

涂鸦品牌智能家居的主要市场集中在发达国家,其中欧洲市场是涂鸦品牌最重要的市场。但由于价格较高,处于中高档消费市场。欧洲市场对智能家居的需求可能更偏向于高端、智能化程度高的产品,最热卖的商品是温湿度传感器、温湿度探测器,可见对于生活品质追求较高。

智能家居产品的目标客户群体大部分是年轻人、高端消费者等。在营销中需要侧重于科技感、智能控制、节能环保、安全可靠等方面,突出这些特点,吸引消费者。同时,在巩固原有欧洲市场的同时,也可以通过搜索引擎优化(SEO)引起潜在用户的主意,多挖掘东亚市场,相对应地推出一些性价比较高的产品。

(二) 精准营销策略

现代营销学之父菲利普·科特勒的《营销管理》中首先提出“精准影响”概念,定义为通过大数据、高速通讯等现代信息技术,汇总及分析用户的个性、适应性等价值数据建立用户沟通服务体系对企业的营销模式进行优化并实施回报率高的精准策略。研究发现,精准营销主要包含是在充分了解用户信息基础上建立用户画像,关注用户需求和喜好从而有针对性的进行营销策略,从而可达到低成本可持续的企业收益目标。精准营销需要基于先进的信息技术系统实现,包含围绕客户数据的网络平台、各类媒介技术、数据库中心和配套数据算法、人工智能等,深度挖掘目标用户个性化需求,精准匹配媒体投放和产品销售,保证业务与管流程规范化和效益最大化。图 45 为精准营销模型。

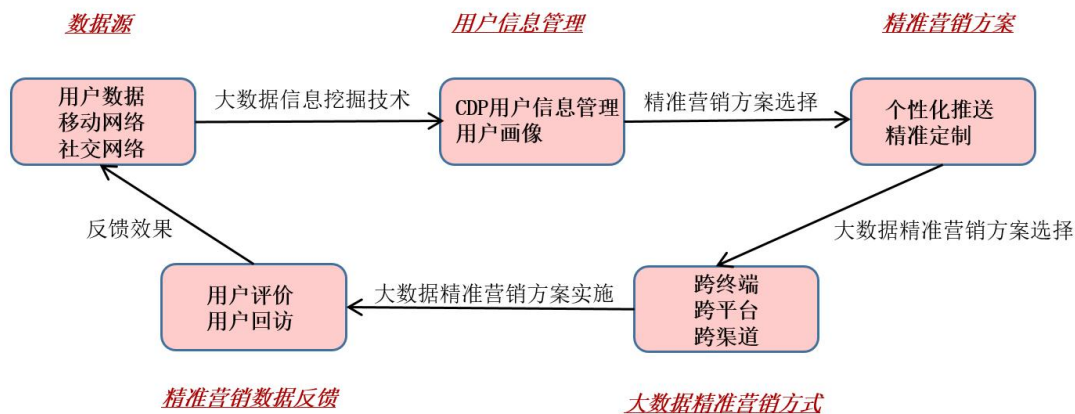


图 45.精准营销模型

精准营销具备如下特征：用户标签性，企业通过不同途径和方式吸引目标用户，对其运用数字标签进行分类和计算，为精准营销策略做好基础准备工作；沟通有效性，企业对不同标签用户群体的需求，制定与针对性的沟通策略，达到高效信息交互目的；沟通经济型，使用社交媒体平台及网络技术，如电话、微信等，减少用户对接沟通成本；成果数字化，企业可以借助科学统计方法分析其投放内容、广告观看、点击次数，活跃用户、成交量和成交额，以实现用户的消费特征分类匹配和实时管控营销策略成本，获得更多营销收益。根据精准营销策略，我们提出以下优化方案：

1.关联策略

在“Tik Tok”“Instagram”“YouTube”等社交媒体平台快速成熟下，企业亟待探求升级多种渠道与用户进行多维度关联。一方面，社交媒体营销实施不会受到地域和时间的限制，另一方面，企业用户在需求构想、需求确定、产品设计、功能体验、价格定位、传播认可、服务反馈等阶段可实时参与企业进行信息交互和产品监督，有利于企业获得真实和持久的用户价值。

（1）建立矩阵式社交媒体关联布局

通过社交媒体矩阵式策略升级进行节奏式的营销布局，促进与用户重新建立长期而稳定的互动关系。根据各类社交平台凸显不同的社交属性及功能，本店铺需要进一步划分实施策略投放至不同平台，以获得不同营销效果。例如如在“Tik Tok”“Instagram”“YouTube”等平台及时社交、内容发布类型平台，可以通过用户数量已有一定积累的官方账号渠道，除固定周期的企业理论、品牌概念、新产品发布等信息发布以外，增加互动标签或专辑活动功能。例如，建立用户参与功能需求、产品设计征集、体验反馈等短周期的互动活动。让用户感知到本店铺产品与服务的创新是真正从用户需求出发，解决用户生活和工作的实际问题

（2）培养 KOL 测评分享用户

在“Tik Tok”“Instagram”“YouTube”等平台建立互动分享、体验有礼等互动玩法，吸引用户主动提供创新想法和内容，让企业获得有效的用户反馈；在电商及专业测评类型平台上，培养 KOL 及互动率高的用户，在不同渠道上进行专业分享和带领活动，拉近用户对产品的认知，产生带动力高的社群转化效应。

（3）良好利用平台引流功能

“Tik Tok”“Instagram”“YouTube”等社交平台近几年的发展已经积累大量的用户。这类平台可以利用自然的推荐算法给与账号进行用户引流和推荐，通过技术流、内容流、互动操作等视频内容传递、视频互动策略等可以推动这些平台输送更多用户流量，以吸引新用户建立联系，持续进行用户互动。

通过对各类型平台的针对性社交媒体营销策略矩阵排布，能够给用户深入人心的企业品牌和产品形象带来新的活力。通过给用户耳目一新的产品和服务，突破原本传统业务建立的一次性买卖关系和用户关系，通过新颖的营销

内容和持续互动方法，获得老用户持续的口碑影响价值和新用户吸引价值，为本店铺突破新的市场规模打好用户基础。

2.创造可持续节日营销关联

企业节日关怀可以促进用户感知品牌温度，成为各行业、企业炙手可热的营销工具之一。通过利用节日 IP，在提供用户更多产品和服务的同时，更好维护住与用户之间的关系，从而吸引更多用户加入品牌团体并产生效益，让用户产生期待并进行持续买单，更好的延续企业的营销价值。因此，本店铺可以通过各类社交媒体平台，建立自有品牌的专属节日营销策略，以拉近品牌与用户之间的亲密度。如图 46 所示，我们制作了宣传海报可最做为宣传参考。



图 46.节日营销海报示例

（1）建立固定时间用户回馈活动

在每年的固定时间推出品牌节日庆祝活动，包含用户节日赠礼或互动礼品给与品牌用户更优势的价格以及各类电商平台的重要价值客户、重要发展客户、重要保持客户服务体验等等措施，让用户感知到品牌是持续关注他们的需求并且能够根据他们的需要不断升级服务。通过专属品牌节日在社交媒体平台的发酵，不仅让新用户可以认可更多产品和服务，更给老用户带去温暖和深刻的印象，以此加强本店铺与用户间的持续关联作用，为以后的营销策略开展和产品业务扩展建立良好稳固的基础。

（2）借力社交媒体平台节日促销能力

当下阿里速卖通已成立每月固定或每年固定的社交电商促销活动，本店铺可以借助其平台机制进行方案优化和策略排布，帮助用户建立固定的互动习惯。运用社交媒体平台各月不同的激励政策，快速获得用户流量和用户需求，也使得产品、服务及成套方案信息以最便捷的方式到达了用户一侧。这不仅为企业节约了大量的人力和时间成本，更引导用户与企业建立新型供需关系，获取从产品使用前预期、使用期间感受和到使用之后的影响反馈等信息，从而在社交媒体平台无形的营销策略中，使得用户与企业的互动价值达到最大化。

3.关系策略

在当前信息日益透明化的社会，用户关系与企业经济利益的关联度越来越高，利用社交媒体维持良好的用户关系直接影响市场占有率。例如，小米公司通过自建平台和社交平台建立良好的“米粉”社群，持续维护小米公司通过手机市场带动智能产品、电脑、家电等市场开拓。由此可见，借助社交媒体营销策略能够与目标用户创造、建立和维持互动状态，优化和提升用户资源分配，将营销策略从信息传播功能升级至战略经营体系，进一步提升企业利益转化率。

（1）围绕用户数据升级用户关系

积极采用大数据计算匹配用户关系并进行关系状态管理及营销输出。技术能力的提升不仅应用于产品的更新换代和企业竞争力的提升，也在企业策略运营、用户运营等方面发挥着重要作用。在巨大的用户群体面前，单一进行传统的关系维护办法已经满足不了市场发展的需要，进行用户的合理分类、分层管理才能在多变的用户需求中建立持续的用户价值传递通道。通过能够明晰不同用户类型的多层级需求，从而根据预期数据进行矩阵营销策略实施，进一步提升用户口碑和忠诚度，激发更大品牌影响力。所以我们认为，应该建立用户关系数据管理通过社交媒体营销平台传播及触达效果数据为基础建立用户关系数据平台并运营管理。根据用户在社交媒体平台上的行为动向数据，挖掘潜在用户群体并关注与用户关系的深浅度，目标将沉默用户变为“新客”，将陌生用户和“新客”培养为“熟客”，进而围绕不同用户关系，对沉默用户制定需求刺激策略，为新用户采取吸引策略，为老用户建立维护策略及VIP专属服务策略等。

（2）采用精准营销、私域营销丰富关系网络

精准的策略投放和用户瞄准能够提高企业营销资源的合理分配，发挥营销投入的最大价值。提升营销资源的利用率和营销成本的投入产出比十分重要。首先，根据前文的客户分析进行精准营销投放，对不同用户层级和行为预期，快速匹配针对性的营销计划，通过精准营销策略触达至各类社交媒体平台的用户，在信息冗杂的市场中快速调动用户消费欲望，建立用户互动关联，创造销售可能性。

同时，可以借助自营平台或微信社交平台进行关系用户私域引流和忠实用户私域营销策略。基于社交和分享的基础功能可以延伸维护多个用户群，从用户的社交关系中直接引导产生消费。用户群内需设置专属营销人员、客服及售后专员，时刻关注用户的需求并紧跟消费路径，提供专属价格或策略的智能家居产品或服务购买链接，从营销传播至消费再到售后过程经历让忠实用户感受到贵宾级服务待遇，从而有助于铺设更多品牌用户的好友营销网络。通过借助用户管理和用户运营进行营销资源合理分配和用户价值的高效提炼，进而采用精准营销、私域营销等方式建立更深层次的私域用户关系网络，从而为企业节省更多的人力、时间和企业资源等成本，提高营销转传播的效率和效果，维护企业用户关系的稳定性。

（三）回报策略

企业的经营目标始终是围绕经济利益进行的，为用户创造有价值的产品和服务并从用户角度获得相应回报是推动正常市场运转和发展的良性循环。智能家居产品实现在家庭环境下各家电设备联动控制功能一般都会配套手机端APP同步显示和操控，因此在产品销售阶段和运营阶段可以进行功能增值和二次营销，有计划的采取免费与收费相结合的收益回报策略，从吸引用户到获得用户好感，从营销信息传递升级成为有价值环的良性营销策略。

1. 铺设社交媒体裂变反馈的优惠路径

社交媒体平台的分享属性给社交媒体营销策略的实施创造了用户裂变、内容裂变价值裂变等高效工具。我们的社交营销思路主要是有效利用用户社交，在营销初级阶段，可以通过渠道优惠券或分享优惠抵扣形式给与用户一定购买折扣，通过让步少许利益来获取更多用户营销机会和品牌曝光频率。在营销深入阶段，建立忠实用户分享链接，通过专属分享链接为忠实用户和被分享者提供不同的优惠措施，从而实现社交带货的营销价值；营销进阶阶段，给与精准用户和私域用户特定的优惠方案和返利路径，使得品牌用户关系的得到进一步维护，用户社交价值时刻紧抓在企业手里通过社交媒体建立裂变的用户传递网络，结合不同渠道的优惠策略更易在短时间内积累用户量，建立良好的流量基础，为智能家居产品铺设口碑形象价值通道，并可以进一步运用社交媒体营销策略反复吸引用户、感染用户。

2.增加增值服务

各行业的竞争赛道已经不是原来的单一方向，智能家居行业的市场已经存在传统家电厂商、互联网企业、细分领域产品等多个行业激烈的竞争。各类行业的跨界合作已成为当下用户流量互补的良好渠道，通过共享各个行业的资源，为合作企业带来双赢的用户价值。

用户与智能家居产品的交互涉及到生活、工作及娱乐等方方面面，因此，智能家居产品配套 APP 或有屏产品交互界面上可以增加相关的增值服务。例如，智能冰箱产品使用界面上可以增加食谱推荐、营养搭配、生鲜配送等一键完成功能，给与用户配套的增值生态服务。

基于智能家居产品的娱乐属性，智能产品与各类娱乐内容的结合得以实现和推广可以利用 APP 及交互界面上增加操控、语音交互、内容传播等运营功能，通过提供一定的免费的音乐、电台、新闻、故事等内容共享服务和设置合理付费门槛的精华内容和功能，满足用户的基本使用产品需求之外，实现智能家居产品的二次变现能力。例如，智能音箱产品上实现音乐、有声书等内容的会员购买，通过用户购买获取内容资源的分成收益等等。

不同行业、不同品牌的跨界合作为智能家居产品提供了更广阔的价值创造平台，采取附加良性回报的社交媒体营销策略，不仅让用户认可智能家居产品的品质和功能，培养用户持续输出价值的影响力，使得本店铺不断提升品控和服务，在与用户建立的新型回报关系上实现更多企业价值产出和用户价值收益。

（四）可视化大屏展示



附录

1.数据预处理

```
import pandas as pd

# 加载数据

df1 = pd.read_excel("C:\\Users\\华硕\\Desktop\\三创赛\\题目数\\202302-202304.xlsx")
df2 = pd.read_excel("C:\\Users\\华硕\\Desktop\\三创赛\\题目数据\\202211-202301.xlsx")
df3 = pd.read_excel("C:\\Users\\华硕\\Desktop\\三创赛\\题目数据\\202208-202310.xlsx")
df4 = pd.read_excel("C:\\Users\\华硕\\Desktop\\三创赛\\题目数据\\202205-202307.xlsx")

# 根据列名合并表格，忽略列的原始排列顺序

merged_df = pd.concat([df1, df2, df3, df4], axis=0, ignore_index=True)

# 合并后的 DataFrame 保存为新的 Excel 文件

merged_df.to_excel("C:\\Users\\华硕\\Desktop\\三创赛\\题目数据", index=False)


# 读取 Excel 文件

file_path = "C:\\Users\\华硕\\Desktop\\三创赛\\题目数据\\合并数据.xlsx"

df = pd.read_excel(file_path)


# 定义一个函数来去除"US $"并只保留数字

def remove_us_dollar(value):
    if isinstance(value, str):
        return value.replace('US $', '').strip()
    return value

# 应用这个函数到 D 列和 O 列

df['D'] = df['D'].apply(remove_us_dollar)
df['O'] = df['O'].apply(remove_us_dollar)
df['S'] = df['S'].apply(remove_us_dollar)
df['W'] = df['W'].apply(remove_us_dollar)
df['Y'] = df['Y'].apply(remove_us_dollar)


df['V'] = df['V'].replace({"等待买家付款": 1, "等待买家收货": 2, "等待您发货": 3,
"订单关闭": 4, "冻结中": 5, "交易完成": 6, "买家申请取消订单": 7})

df['N'] = df['N'].replace({"平台已税": 1, "平台未税": 0, "null": 2})


data = pd.read_excel(file_path)


# 将 "付款时间 "和 "发货时间 "列转换为日期时间格式
```

```

## 使用 errors='coerce' 来处理转换错误，将其转换为 NaT
data['付款时间'] = pd.to_datetime(data['付款时间'], format='%Y-%m-%d %H:%M', errors='coerce')
data['发货时间'] = pd.to_datetime(data['发货时间'], format='%Y-%m-%d %H:%M', errors='coerce')

#以小时计算差异
data['用时(小时)'] = (data['发货时间'] - data['付款时间']).dt.total_seconds() / 3600

import pandas as pd
# 定义一个函数，以根据指定的条件分配值
def assign_value_corrected(row):
    if pd.isnull(row['发货时间']) or pd.isnull(row['发货期限']):
        return 2 # One of the dates is missing
    elif row['发货时间'] <= row['发货期限']:
        return 0 # Not overdue
    else:
        return 1 # Overdue
# 将该函数应用于每行，以创建新的列
new_data['状态'] = new_data.apply(assign_value_corrected, axis=1)

```

2.店铺运营情况分析

```

import pandas as pd
from collections import defaultdict
import matplotlib.pyplot as plt
from openpyxl import Workbook

# 分别获取月份和季节的数据
data = pd.read_csv('data.csv', usecols=['下单日期', '订单状态'], parse_dates=True)
month_dict = defaultdict(int)
season_dict = defaultdict(int)
day_dict = defaultdict(int)

for _, row in data.iterrows():
    status = int(row[1])
    # 剔除异常数据
    if status in [4, 5, 7]:
        continue
    date = row[0]
    date_data = date.split('/')

```

```

month = int(date_data[1])
month_dict[month] += 1
# 更改数字以具体分析月份
if month == 4:
    day = int(date_data[2])
    day_dict[day] += 1

for k, v in month_dict.items():
    season_dict[(k-1)//3] += v

# 取消注释代码以分别作图
# 绘制月份销量柱状图
# month_list = month_dict.items()
# month_list = sorted(month_list)
# x, y = zip(*month_list)
# plt.bar(x=x, height=y, width=0.4)
# plt.show()

# 绘制季度销量柱状图
# labels = ['spring', 'summer', 'autumn', 'winter']
# season_list = season_dict.items()
# season_list = sorted(season_list)
# x, y = zip(*season_list)
# plt.xticks(x, labels=labels)
# plt.bar(x, y)
# plt.show()

# 绘制日度销量柱状图
# day_list = day_dict.items()
# day_list = sorted(day_list)
# x, y = zip(*day_list)
# plt.bar(x=x, height=y, width=0.4)
# plt.show()

# wb = Workbook()
# ws = wb.active
# ws.title = "四月日度分析"
# ws.cell(row=1, column=1, value='日期')

```



```

# ws.cell(row=1, column=2, value='销量')

# row = 2

# day_list = day_dict.items()

# day_list = sorted(day_list)

# for item in day_list:

#     ws.cell(row=row, column=1, value=item[0])

#     ws.cell(row=row, column=2, value=item[1])

#     row += 1

# wb.save('四月日度分析.xlsx')


import pandas as pd
from collections import defaultdict
import matplotlib.pyplot as plt
import matplotlib.pyplot as pyplot
from openpyxl import Workbook


data = pd.read_csv('new_data.csv', usecols=['下单日期', '商品信息', '数量'], parse_dates=True)
product_dict = defaultdict(int)


for _, row in data.iterrows():
    # 异常数据已剔除
    try:
        split_data = row[1].split(',')
        name = split_data[0].strip()
        num = int(row[2])
    except:
        continue
    product_dict[name] += num


# 得到销量最高的十个产品
product_dict = sorted(product_dict.items(), key = lambda kv:(kv[1], kv[0]), reverse=True)
top_product_list = product_dict[: 10]


# 给相应产品分配产品编码，便于绘图
code_dict = defaultdict(str)
for i, item in enumerate(top_product_list):
    code = chr(ord('A') + i)
    code_dict[code] = item[0]

```

```

# 绘制热销产品柱状图

# x, y = zip(*top_code_list)

# plt.bar(x=x, height=y, width=0.4)

# plt.show()


# 分析热销商品的季节性和属性
# 修改该变量以分析不同产品

selected_product = 3

month_dict = defaultdict(int)
country_dict = defaultdict(int)
color_dict = defaultdict(int)
day_dict = defaultdict(int)

for _, row in data.iterrows():

    try:

        split_data = row[1].split('(')
        name = split_data[0].strip()

    except:

        continue

    if name == top_product_list[selected_product][0]:

        date = row[0].split('/')
        month = int(date[1])
        month_dict[month] += 1

        property = split_data[1][:-2].split(':')
        country = property[-1]
        country_dict[country] += 1

        if selected_product == 0:

            color = property[2].split()[1].split(',')[-1][0]

        elif selected_product == 1:

            color = property[2].split(',')[-1][0]

        color_type = property[2].split(',')[-1][0]
        color_dict[color_type] += 1

    # 分析日度销量

    if month == 1:

        day = int(date[2])
        day_dict[day] += 1


# wb = Workbook()

```

```

# ws = wb.active
# ws.title = "D 商品月度分析"
# ws.cell(row=1, column=1, value='月份')
# ws.cell(row=1, column=2, value='销量')
# row = 2
# month_list = month_dict.items()
# month_list = sorted(month_list)
# for item in month_list:
#     ws.cell(row=row, column=1, value=item[0])
#     ws.cell(row=row, column=2, value=item[1])
#     row += 1
# wb.save('D 商品月度分析.xlsx')

# 绘制国家销量柱状图
# country_list = country_dict.items()
# x, y = zip(*country_list)
# plt.bar(x=x, height=y, width=0.4)
# plt.title(top_product_list[selected_product][0])
# plt.show()

# 绘制月份销量柱状图
# month_list = month_dict.items()
# month_list = sorted(month_list)
# x, y = zip(*month_list)
# plt.bar(x=x, height=y, width=0.4)
# plt.title(top_product_list[selected_product][0])
# plt.show()

# wb = Workbook()
# ws = wb.active
# ws.title = "D 一月日度分析"
# ws.cell(row=1, column=1, value='日期')
# ws.cell(row=1, column=2, value='销量')
# row = 2
# day_list = day_dict.items()
# day_list = sorted(day_list)
# print(day_list)
# for item in day_list:

```

```
# ws.cell(row=row, column=1, value=item[0])
# ws.cell(row=row, column=2, value=item[1])
# row += 1
# wb.save('D 一月日度分析.xlsx')
```

3.绘图

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

# 读取 Excel 文件数据
data = pd.read_excel(r'D:\桌面\数据预处理后.xlsx')
plt.rcParams['font.sans-serif'] = ['SimSun'] # 指定使用宋体字体
plt.rcParams['axes.unicode_minus'] = False # 解决负号 '-' 显示为方块的问题

# 创建一个新的列来存储产品名称
data['产品名称'] = ""

# 遍历产品信息，解析产品名称
for index, row in data.iterrows():
    # 根据产品的结构提取产品名称
    product_info = row['商品信息']
    product_name = re.search(r'^([^\()]*)', product_info).group(1)

    # 将提取到的产品名称存储到新的列中
    data.at[index, '产品名称'] = product_name.strip()

# 将下单日期列转换为日期类型
data['下单日期'] = pd.to_datetime(data['下单日期'], format='%Y/%m/%d')

# 分析下单日期数据，找出电商大促日期
daily_order_count = data['下单日期'].dt.date.value_counts().sort_index()

# 根据实际情况确定大促日期的条件
promotion_threshold = 300 # 假设每日订单数超过 300 单的日期为大促日期

promotion_dates = daily_order_count[daily_order_count > promotion_threshold].index.tolist()
# 打印电商大促日期
# print("电商大促日期: ", promotion_dates)
```

```

# 筛选出大促节日的订单
promotion_data = data[data['下单日期'].isin(promotion_dates)]

# 进行促销产品分析，计算促销产品的销售数量
promotion_product_analysis = promotion_data.groupby('产品名称').agg({'产品名称': 'count'}).rename(
    columns={'产品名称': '销售数量'}).reset_index()

# 按销售数量降序排序
promotion_product_analysis = promotion_product_analysis.sort_values('销售数量', ascending=False)
# 仅保留前 10 个产品
promotion_product_analysis = promotion_product_analysis.head(10)
# 保存为 Excel 文件
promotion_product_analysis.to_excel(r'C:\Users\87202\Desktop\季节性产品分析\promotion_product_analysis.xlsx',
index=False)
# 保存为 Excel 文件
promotion_dates_count = daily_order_count[daily_order_count.index.isin(promotion_dates)].head(10).reset_index()
promotion_dates_count.columns = ['日期', '销售数量']
promotion_dates_count.to_excel(r'C:\Users\87202\Desktop\季节性产品分析\promotion_dates_count.xlsx', index=False)
# 创建包含两个子图的图表
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(14, 6))

# 绘制柱状图
ax1.bar(promotion_product_analysis['产品名称'], promotion_product_analysis['销售数量'])
ax1.set_xticks(range(len(promotion_product_analysis['产品名称']))) # 设置刻度位置
ax1.set_xticklabels(promotion_product_analysis['产品名称'], rotation=90)
ax1.set_xlabel('产品名称')
ax1.set_ylabel('销售数量')
ax1.set_title('前 10 个大促产品销售数量')

# 绘制折线图
ax2.plot(range(len(promotion_dates_count['日期'])), promotion_dates_count['销售数量'], linestyle='-', marker='o',
color='orange')
ax2.set_xticks(range(len(promotion_dates_count['日期']))) # 设置刻度位置
ax2.set_xticklabels(promotion_dates_count['日期'], rotation=45)
ax2.set_xlabel('日期')
ax2.set_ylabel('销售数量')
ax2.set_title('前 10 个大促日期销售数量')

```

```

# 调整子图之间的间距
plt.subplots_adjust(wspace=0.3)

plt.show()

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

# 读取 Excel 文件数据
data = pd.read_excel(r'C:\Users\87202\Desktop\季节性产品分析\数据预处理后.xlsx')

plt.rcParams['font.sans-serif'] = ['SimSun'] # 指定使用宋体字体
plt.rcParams['axes.unicode_minus'] = False # 解决负号 '-' 显示为方块的问题

# 删除不必要的空格
data['州/省'] = data['州/省'].str.strip()

# 将产品总金额列转换为数值类型
data['产品总金额'] = data['产品总金额'].astype(float)

# 按州/省分组并计算销售额之和
state_sales = data.groupby('州/省')['产品总金额'].sum()

# 排序并选择销售额前 10 的州/省
top_10_states = state_sales.sort_values(ascending=False).head(10)

# 保存为 Excel 文件
top_10_states.to_excel(r'C:\Users\87202\Desktop\季节性产品分析\top_10_states_analysis.xlsx', index=True)

# 绘制柱状图
plt.figure(figsize=(16, 14)) # 调整图表尺寸
plt.bar(top_10_states.index, top_10_states.values)
plt.xlabel('州/省')
plt.ylabel('销售额')
plt.title('销售额前 10 的州/省')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

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

```



```

# 读取 Excel 文件数据
data = pd.read_excel(r'C:\Users\87202\Desktop\季节性产品分析\数据预处理后.xlsx')

plt.rcParams['font.sans-serif'] = ['SimSun'] # 指定使用宋体字体
plt.rcParams['axes.unicode_minus'] = False # 解决负号 '-' 显示为方块的问题

# 最频繁使用的前 10 个物流公司
# 去除物流公司字段中的空值
data = data.dropna(subset=['买家选择物流'])

# 删除不必要的空格
data['买家选择物流'] = data['买家选择物流'].str.strip()

# 计算每个物流公司的使用频率
logistics_freq = data['买家选择物流'].value_counts().head(10)

# 创建一个 DataFrame 保存物流公司名称和使用频率
df_logistics_freq = pd.DataFrame({'物流公司名称': logistics_freq.index, '使用频率': logistics_freq.values})

# 保存为 Excel 文件
df_logistics_freq.to_excel(r'C:\Users\87202\Desktop\季节性产品分析\logistics_freq_analysis.xlsx', index=False)

# 绘制柱状图
plt.figure(figsize=(16, 15)) # 调整图表尺寸
plt.bar(df_logistics_freq['物流公司名称'], df_logistics_freq['使用频率'])
plt.xlabel('物流公司')
plt.ylabel('使用频率')
plt.title('最频繁使用的前 10 个物流公司')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

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

# 读取 Excel 文件数据
data = pd.read_excel(r'C:\Users\87202\Desktop\季节性产品分析\数据预处理后.xlsx')

```

```

plt.rcParams['font.sans-serif'] = ['SimSun'] # 指定使用宋体字体
plt.rcParams['axes.unicode_minus'] = False # 解决负号 '-' 显示为方块的问题

# 订单量最高的前 10 个国家
# 删除不必要的空格
data['收货国家'] = data['收货国家'].str.strip()
# 计算每个国家的订单数量
country_orders = data['收货国家'].value_counts().head(10)
# 创建一个 DataFrame 保存国家名称和订单数量
df_country_orders = pd.DataFrame({'国家名称': country_orders.index, '订单数量': country_orders.values})
# 保存订单数量到 Excel
df_country_orders.to_excel(r'C:\Users\87202\Desktop\季节性产品分析\country_orders_quantity.xlsx', index=False)
# 设置图表大小
plt.figure(figsize=(14, 10))

# 绘制柱状图
plt.bar(df_country_orders['国家名称'], df_country_orders['订单数量'])
plt.xlabel('国家')
plt.ylabel('订单数量')
plt.title('订单量最高的前 10 个国家')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

import pandas as pd
import matplotlib.pyplot as plt

# 读取 Excel 文件数据
data = pd.read_excel(r'C:\Users\87202\Desktop\季节性产品分析\数据预处理后.xlsx')
plt.rcParams['font.sans-serif'] = ['SimSun'] # 指定使用宋体字体
plt.rcParams['axes.unicode_minus'] = False # 解决负号 '-' 显示为方块的问题

# 将下单日期和付款时间转换为 DateTime 类型
data['下单日期'] = pd.to_datetime(data['下单日期'])
data['付款时间'] = pd.to_datetime(data['付款时间'])

# 提取下单日期中的月份作为季节
data['季节'] = data['下单日期'].dt.month.map({

```

```

1: '冬季', 2: '冬季', 3: '春季',
4: '春季', 5: '春季', 6: '夏季',
7: '夏季', 8: '夏季', 9: '秋季',
10: '秋季', 11: '秋季', 12: '冬季'
})

# 计算每个季节的销量和销售总额
season_sales = data.groupby('季节').agg({'订单金额': sum})

# 保存季节销售总额到 Excel
season_sales.to_excel(r'C:\Users\87202\Desktop\季节性产品分析\season_sales_analysis.xlsx')

# 绘制销售总额柱状图
fig, ax = plt.subplots()
season_sales['订单金额'].plot(kind='bar', ax=ax, color='r', width=0.4, label='销售总额')

# 设置图表标签
plt.xlabel('季节')
plt.ylabel('销售总额')
plt.title('季节销售总额')
plt.legend()

plt.tight_layout()
plt.show()

import pandas as pd
import matplotlib.pyplot as plt

# 读取 Excel 文件数据
data = pd.read_excel(r'C:\Users\87202\Desktop\季节性产品分析\数据预处理后.xlsx')
plt.rcParams['font.sans-serif'] = ['SimSun'] # 指定使用宋体字体
plt.rcParams['axes.unicode_minus'] = False # 解决负号 '-' 显示为方块的问题

# 将下单日期和付款时间转换为 DateTime 类型
data['下单日期'] = pd.to_datetime(data['下单日期'])
data['付款时间'] = pd.to_datetime(data['付款时间'])

# 提取下单日期中的月份作为季节

```

```

data['季节'] = data['下单日期'].dt.month.map({
    1: '冬季', 2: '冬季', 3: '春季',
    4: '春季', 5: '春季', 6: '夏季',
    7: '夏季', 8: '夏季', 9: '秋季',
    10: '秋季', 11: '秋季', 12: '冬季'
})

# 计算每个季节的销售数量
season_sales = data.groupby('季节').size().reset_index(name='销售数量')

# 保存季节销售数量到 Excel
season_sales.to_excel(r'C:\Users\87202\Desktop\季节性产品分析\season_sales_quantity.xlsx', index=False)

# 绘制销售数量柱状图
fig, ax = plt.subplots()
season_sales.plot(kind='bar', x='季节', y='销售数量', ax=ax, color='b', width=0.4, label='销售数量')

# 设置图表标签
plt.xlabel('季节')
plt.ylabel('销售数量')
plt.title('季节销售数量')
plt.legend()

plt.tight_layout()
plt.show()

```

4.销售预测

```

# 假设 sales_data 是一个已经加载的 DataFrame
sales_data = pd.read_excel(r'C:\Users\20439\Desktop\all.xlsx', sheet_name='Gateway Analyzer')

# 创建一个包含所有日期的日期范围
date_range = pd.date_range(start='2022-04-30', end='2023-04-30')

# 创建一个空的 DataFrame 来存储填充后的数据
filled_sales_data = pd.DataFrame(index=date_range, columns=['Sales'])

# 初始化所有销量为 0
filled_sales_data['Sales'] = 0

```

```

# 遍历 sales_data 的行，更新销量数据
for index, row in sales_data.iterrows():
    date_obj = row['Date'] # 假设'Date'是包含日期的列名
    sales = row['Sales'] # 假设'Sales'是包含销量的列名
    # 更新销量数据
    filled_sales_data.loc[date_obj, 'Sales'] = sales

# 打印结果
print(filled_sales_data)

import pandas as pd

# 假设 filled_sales_data 是一个已经填充好的 DataFrame
# filled_sales_data = ...

# 设置输出文件的路径
output_file_path = r'C:\Users\20439\Desktop\forecast\Gateway Analyzer.csv'

# 将 DataFrame 保存为 CSV 文件
filled_sales_data.to_csv(output_file_path) # 设置 index=False 以避免将索引也保存到文件中

# 读取时间数据的格式化
def parser(x):
    return datetime.strptime(x, '%Y/%m/%d')

# 转换成有监督数据
def timeseries_to_supervised(data, lag=1):
    df = DataFrame(data)
    columns = [df.shift(i) for i in range(1, lag + 1)] # 数据滑动一格，作为 input，df 原数据为 output
    columns.append(df)
    df = concat(columns, axis=1)
    df.fillna(0, inplace=True)
    return df

# 转换成差分数据
def difference(dataset, interval=1):
    diff = list()
    for i in range(interval, len(dataset)):

```



```

        value = dataset[i] - dataset[i - interval]
        diff.append(value)
    return Series(diff)

# 逆差分
def inverse_difference(history, yhat, interval=1): # 历史数据, 预测数据, 差分间隔
    return yhat + history[-interval]

# 缩放
def scale(train, test):
    # 根据训练数据建立缩放器
    scaler = MinMaxScaler(feature_range=(-1, 1))
    scaler = scaler.fit(train)
    # 转换 train data
    train = train.reshape(train.shape[0], train.shape[1])
    train_scaled = scaler.transform(train)
    # 转换 test data
    test = test.reshape(test.shape[0], test.shape[1])
    test_scaled = scaler.transform(test)
    return scaler, train_scaled, test_scaled

# 逆缩放
def invert_scale(scaler, X, value):
    new_row = [x for x in X] + [value]
    array = numpy.array(new_row)
    array = array.reshape(1, len(array))
    inverted = scaler.inverse_transform(array)
    return inverted[0, -1]

# fit LSTM 来训练数据
def fit_lstm(train, batch_size, nb_epoch, neurons):
    X, y = train[:, 0:-1], train[:, -1]
    X = X.reshape(X.shape[0], 1, X.shape[1])
    model = Sequential()
    # 添加 LSTM 层
    model.add(LSTM(neurons, batch_input_shape=(batch_size, X.shape[1], X.shape[2]), stateful=True))
    model.add(Dense(1)) # 输出层 1 个 node
    # 编译, 损失函数 mse+优化算法 adam

```

```

model.compile(loss='mean_squared_error', optimizer='adam')
for i in range(nb_epoch):
    # 按照 batch_size, 一次读取 batch_size 个数据
    model.fit(X, y, epochs=1, batch_size=batch_size, verbose=0, shuffle=False)
    model.reset_states()
    print("当前计算次数: "+str(i))
return model

# 1 步长预测
def forecast_lstm(model, batch_size, X):
    X = X.reshape(1, 1, len(X))
    yhat = model.predict(X, batch_size=batch_size)
    return yhat[0, 0]

# 加载数据
df = pd.read_csv('C:/Users/20439/Desktop/week/Water Leakage Sensor.csv', header=0, parse_dates=[0], index_col=0)

# 如果您想要将 DataFrame 转换为 Series, 可以这样做:
series = df.squeeze()

# 让数据变成稳定的
raw_values = series.values
diff_values = difference(raw_values, 1)#转换成差分数据

# 把稳定的数据变成有监督数据
supervised = timeseries_to_supervised(diff_values, 1)
supervised_values = supervised.values

# 数据拆分: 训练数据、测试数据, 前 40 行是训练集, 后 12 行是测试集
train, test = supervised_values[0:-12], supervised_values[-12:]

# 数据缩放
scaler, train_scaled, test_scaled = scale(train, test)

# fit 模型
lstm_model = fit_lstm(train_scaled, 1, 900, 11) # 训练数据, batch_size, epoche 次数, 神经元个数

# 预测
train_reshaped = train_scaled[:, 0].reshape(len(train_scaled), 1, 1)#训练数据集转换为可输入的矩阵

```

```
lstm_model.predict(train_reshaped, batch_size=1)#用模型对训练数据矩阵进行预测
```

测试数据的前向验证，实验发现，如果训练次数很少的话，模型回简单的把数据后移，以昨天的数据作为今天的预测值，当训练次数足够多的时候

才会体现出来训练结果

```
predictions = list()
```

for i in range(len(test_scaled)):#根据测试数据进行预测，取测试数据的一个数值作为输入，计算出下一个预测值，以此类推

```
    # 1 步长预测
```

```
    X, y = test_scaled[i, 0:-1], test_scaled[i, -1]
```

```
    yhat = forecast_lstm(lstm_model, 1, X)
```

```
    # 逆缩放
```

```
    yhat = invert_scale(scaler, X, yhat)
```

```
    # 逆差分
```

```
    yhat = inverse_difference(raw_values, yhat, len(test_scaled) + 1 - i)
```

```
    predictions.append(yhat)
```

```
    expected = raw_values[len(train) + i + 1]
```

```
    print('Day=%d, Predicted=%f, Expected=%f' % (i + 1, yhat, expected))
```

性能报告

```
rmse = sqrt(mean_squared_error(raw_values[-12:], predictions))
```

```
print('Test RMSE: %.3f' % rmse)
```

绘图

```
import matplotlib.pyplot as plt
```

```
plt.rcParams['font.sans-serif'] = ['SimHei'] # 或者 'Microsoft YaHei'
```

```
plt.rcParams['axes.unicode_minus'] = False # 解决负号显示问题
```

```
plt.xlabel('周期数')
```

```
plt.ylabel('销量')
```

```
plt.title('门窗传感器')
```

```
pyplot.plot(predictions)
```

```
pyplot.show()
```

####尝试预测 20 周期

```
for i in range(len(test_scaled), len(test_scaled) + 20):
```

```
    # 使用上一次的预测值作为新的输入
```

```
    X = predictions[-1].reshape(1, 1, -1)
```

```
    yhat = forecast_lstm(lstm_model, 1, X)
```

```
    # 逆缩放
```

```

yhat = invert_scale(scaler, X, yhat)
# 逆差分
yhat = inverse_difference(raw_values, yhat, len(test_scaled) + i)
predictions.append(yhat)
print('Day=%d, Predicted=%f' % (i + 1, yhat))
# 输出所有预测结果
print(predictions)
# 绘图
import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif'] = ['SimHei'] # 或者 'Microsoft YaHei'
plt.rcParams['axes.unicode_minus'] = False # 解决负号显示问题
plt.xlabel('周期数')
plt.ylabel('销量')
plt.title('门窗传感器')
pyplot.plot(predictions)
pyplot.show()

def gridsearch_prophet(data, params):
    best_params = None
    best_mse = float('inf')
    # Generate all possible combinations of parameter values
    param_grid = ParameterGrid(params)
    for param in param_grid:
        # Initialize Prophet model with current parameter values
        model = Prophet(seasonality_mode=param['seasonality_mode'],
                        seasonality_prior_scale=param['seasonality_prior_scale'],
                        changepoint_prior_scale=param['changepoint_prior_scale'])
        # Fit the model on the data
        model.fit(data)
        # Make predictions for the next 5 steps
        future = model.make_future_dataframe(periods=5)
        forecast = model.predict(future)
        # Calculate mean squared error
        mse = ((data['y'] - forecast['yhat'])**2).mean()
        # Update best parameters if current MsE is lower
        if mse < best_mse:
            best_params = param
            best_mse = mse

```

```

return best_params

# Example usage
# data: Pandas DataFrame with 'ds'(date)and 'y'(target)columns
data = pd.read_csv('C:/Users/20439/Desktop/week/Water Leakage Sensor.csv',names=['Date', 'Sales'],skiprows=1,
header=None, parse_dates=['Date'], date_parser=lambda x: pd.to_datetime(x, format='%Y-%m-%d'))
data.rename(columns={'Date':'ds','Sales':'y'}, inplace=True)
# 定义参数网格

params={'seasonality_mode':['additive','multiplicative'],
        'seasonality_prior_scale':[0.01,0.1,1.0],
        'changepoint_prior_scale':[0.001,8.01,0.1]}

# 执行网格搜索并获取最佳参数
best_params = gridsearch_prophet(data, params)
# 使用最佳参数初始化 Prophet 模型
best_model = Prophet(seasonality_mode=best_params['seasonality_mode'],
                     seasonality_prior_scale=best_params['seasonality_prior_scale'],

changepoint_prior_scale=best_params['changepoint_prior_scale'])
# 在数据上拟合模型
best_model.fit(data)
# 对接下来的 10 个步骤进行预测
future = best_model.make_future_dataframe(periods=10)
forecast = best_model.predict(future)
# 打印接下来 10 个步骤的预测值
print(forecast[['ds','yhat']].tail(10))
forecast[['ds', 'yhat', 'yhat_lower', 'yhat_upper']].tail()
# 展示预测结果
best_model.plot(forecast)
# 预测的成分分析绘图，展示预测中的趋势、周效应和年度效应
best_model.plot_components(forecast)

# 设置字体
chinese_font = FontProperties(fname=r"C:\Windows\Fonts\simsum.ttc", size=12, weight='bold') # 宋体，加粗
english_font = FontProperties(fname=r"C:\Windows\Fonts\times.ttf", size=10, weight='light') # Times New Roman，加粗

# 读取 CSV 文件
data = pd.read_csv('C:/Users/20439/Desktop/预测/Smart Switch.csv')

```



```
# 设置图像的 DPI
plt.figure(dpi=300)

# 绘制所有数据的销售量
plt.plot(data['Index'], data['Sales'], label='All Sales')

# 获取最后 20 组数据的索引
last_20_indices = data['Index'].tail(20).index

# 绘制最后 20 组数据的销售量，用橙色线条突出显示
plt.plot(data.loc[last_20_indices, 'Index'], data.loc[last_20_indices, 'Sales'], color='orange', label='Last 20 Sales')

# 设置标题和轴标签
plt.title('智能开关', fontproperties=chinese_font, fontsize=12, fontweight='bold')
plt.xlabel('周期', fontproperties=chinese_font, fontsize=10, fontweight='bold')
plt.ylabel('销量', fontproperties=chinese_font, fontsize=10, fontweight='bold')

# 设置刻度标签的字体为 Times New Roman
plt.yticks(fontproperties='Times New Roman', size=10)
plt.xticks(fontproperties='Times New Roman', size=10)

# 显示图表
plt.show()
```