

001.商品库存情况

2020年5月11日 8:04

销售行业卖的是商品，如果商品准备不充足或不合理，必然会影响销售业绩

1.1 新建度量值与辅助列

2020年5月11日 8:04

1、基础度量值，将需要计算的字段分别新建度量值

总数量 = SUM('明细'[销量])

总金额 = SUM('明细'[销售额])

总库存 = SUM('明细'[库存数量])

2、计算销售零售金额与库存零售金额

商品并不一定按标准价格进行销售，很多时候有打折的情况

可以使用新建列，如果你担心内存浪费，就用以下方法建立度量值

销售零售额 = sumx('明细','明细'[销量]*'明细'[零售价])

库存金额 = sumx('明细','明细'[库存数量]*'明细'[零售价])

3、计算款式数量，库存为0的款式不计算

款式数量 = CALCULATE(DISTINCTCOUNT('明细'[货号]),'明细'[库存数量]>0)

4、计算零星款式数量，库存少于6的款式

零星款式数量 = CALCULATE(DISTINCTCOUNT('明细'[货号]),'明细'[库存数量]>0 && '明细'[库存数量]<6)

5、计算货龄区间 【新建列】

货龄 = DATEDIFF('明细'[上市日期],date(2018,6,30),DAY)+1

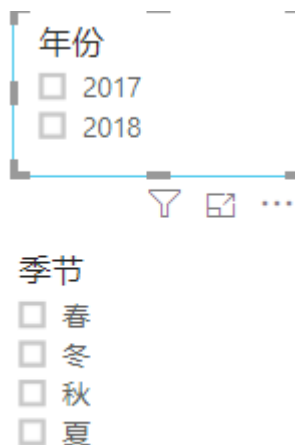
6、计算周转天数 【新建列】

周转天数=

```
switch(
    TRUE(),
    '明细'[货龄]<=30,"30天以下",
    '明细'[货龄]<=60,"30-60天",
    '明细'[货龄]<=90,"60-90天",
    '明细'[货龄]<=120,"90-120天",
    "120天以上"
)
```

1.2 切片器：年份、季节【调整顺序】

2020年5月11日 9:08



季节	顺序
春	1
夏	2
秋	3
冬	4

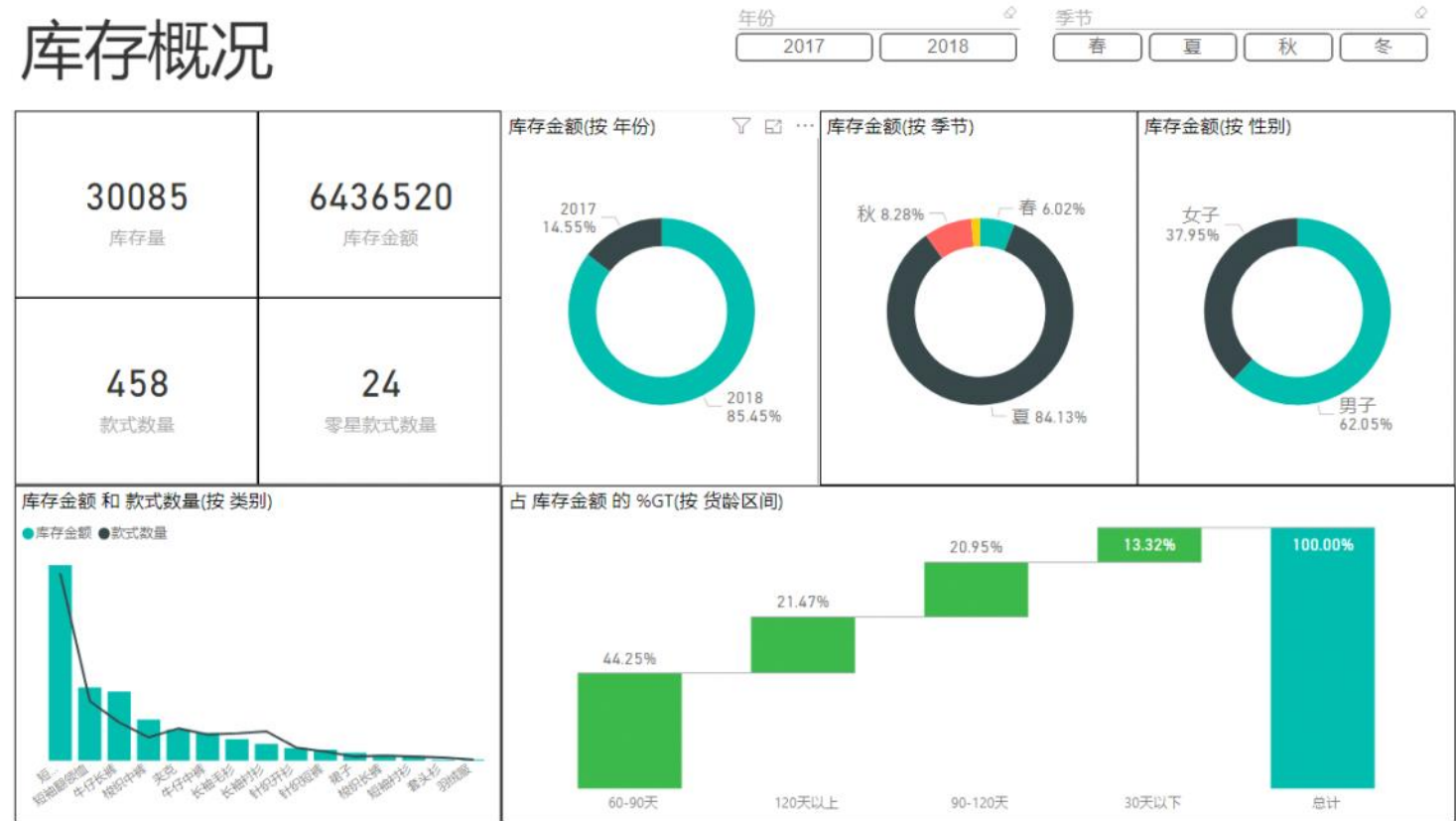
输入数据新建表

【新建列】 季节顺序 = RELATED(' 季节顺序' [顺序])

【建模】 - 【按列排序】 - 【季节顺序】

1.3 可视化

2020年5月11日 19:11



库存金额 和 款式数量(按 类别)

占 库存金额的 %GT(按 货龄区间)

1.3.1 卡片图

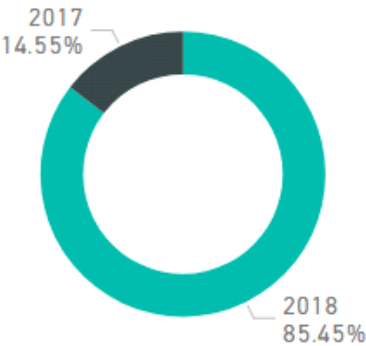
2020年5月11日 19:12



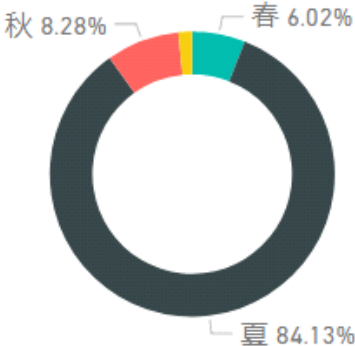
1.3.2 环形图

2020年5月11日 19:12

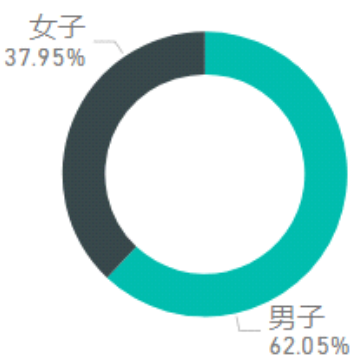
库存金额(按 年份)



库存金额(按 季节)



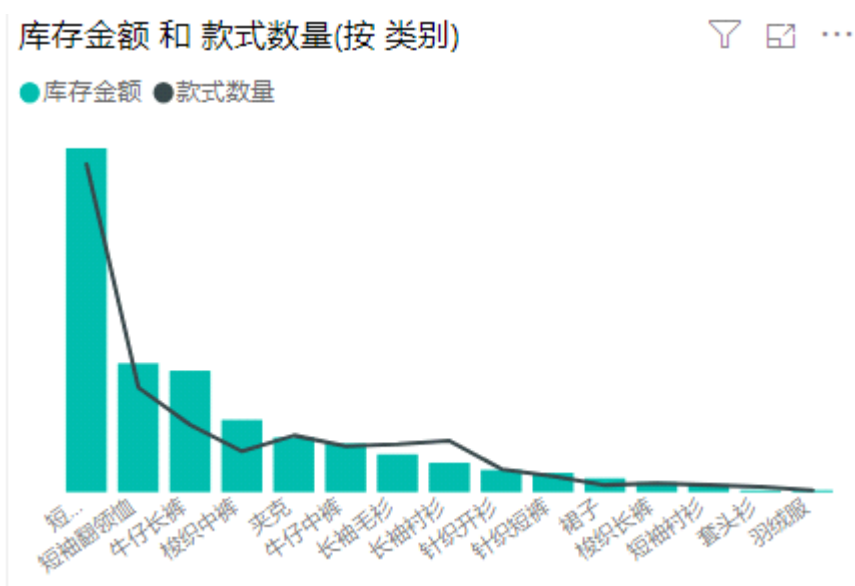
库存金额(按 性别)



图例：年份、季节、性别
值：库存金额

1.3.3 折线和簇状柱形图

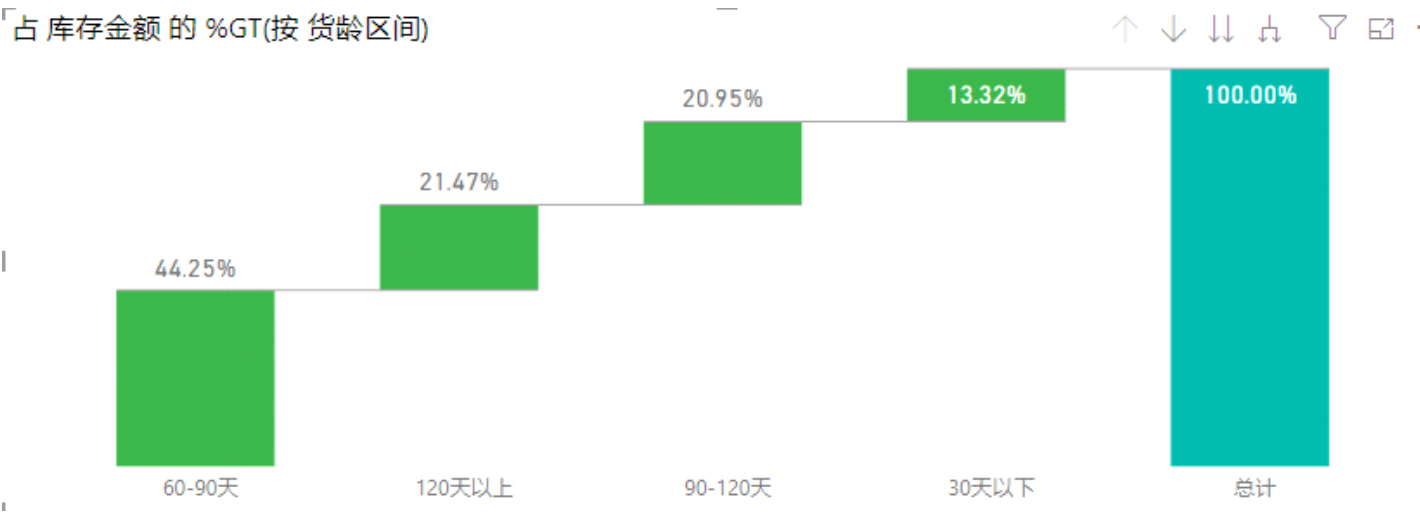
2020年5月11日 19:12



共享轴：类别
列值：库存金额
行值：款式数量

1.3.4 瀑布图

2020年5月11日 19:17



类别：周转天数，性别，类别

Y轴：库存金额%

向下钻取

002.存货销售情况

2020年5月12日 8:44

商品是销售的子弹，需要定期检查子弹是否充足，是否过多占用资金和库房，是否创造了足够的利润。

2.1 计算销售折扣与毛利率

2020年5月12日 8:48

折扣是零售业务非常重要的考核指标，折扣并不是越高越好，折扣高但消费者没有意愿购买则无法产生足够的利润，折扣也不是越低越好，当低过运营成本时，卖得越多亏损越多。

销售折扣 = $\text{DIVIDE}([\text{总金额}], [\text{销售零售额}])$

毛利是利润的源头，毛利和销售折扣与进货成本息息相关，假设进货折扣为零售价的40%

毛利率 = $\text{DIVIDE}([\text{销售折扣}] - 0.4, [\text{销售折扣}])$

注意：我这个是假设毛利率，在你们的实际分析中，毛利率=毛利额/营业收入*100%

2.2 存销比

2020年5月12日 8:55

存销比：存除以销，库存数量除以月均销售数量，用来评估目前的存货数量需要几个月可以消化完毕。它能评估目前仓库中的货品是否合理。

存销比 = DIVIDE([总库存],[总数量])

每个企业的业务逻辑不一样，可能对存销比的目标值不同，存销比并不是越高越好，越高证明库存越多，资金占用越大，风险越高。存销比也不是越低越好，太低说明库存不足，供应链不灵活，当前库存无法支撑销售目标。

计算存销比背景色，对存销比进行分组：

新建背景色度量值，可以对存销比进行分组，方便查看销售状态。当存销比小于等于2时显示绿色，小于等于4时显示黄色，大于4时显示红色

存销比背景色 = IF([存销比]<=2,"Green",IF([存销比]<=4,"Yellow","Red"))

2.3 上下装辅助列

2020年5月12日 8:55

【新建列】上下装 = if(find("裤",'明细'[类别],1,999)<999 || find("裙",'明细'[类别],1,999)<999,"下装","上装")

2.4 卡片图

2020年5月12日 10:14

1,878,286

销售额

2.55

存销比

76%

销售折扣

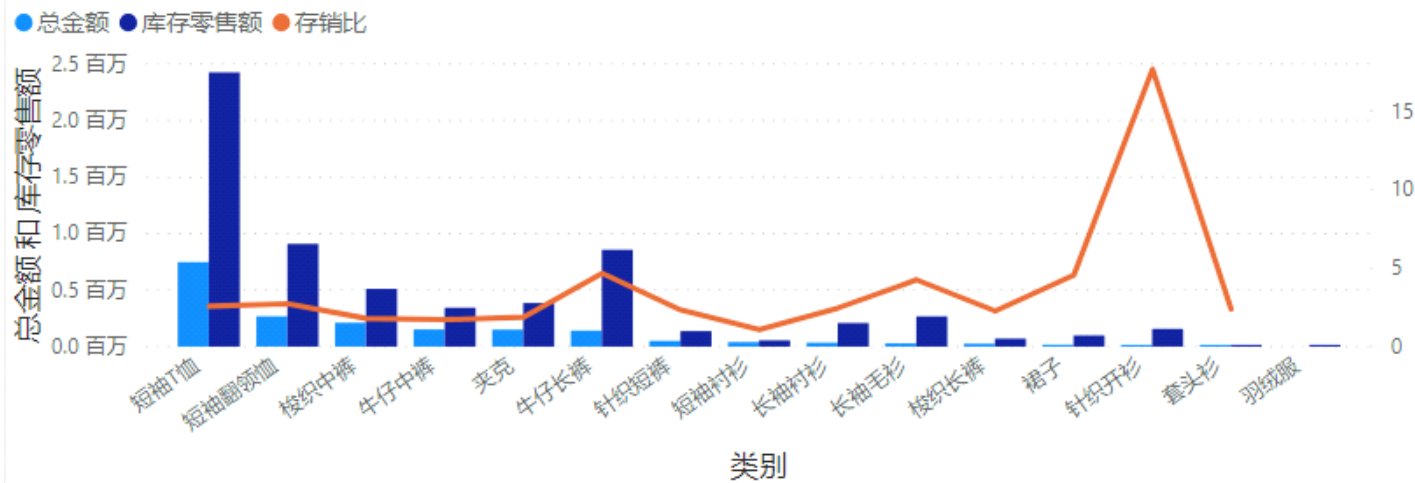
47.38%

毛利率

2.5 总金额，库存金额与存销比

2020年5月12日 10:14

总金额、库存零售额 和 存销比(按 类别)



六字码

类别

列序列

在此处添加数据字段

列值

占 总金额的 %GT

占 库存零售额的 %GT

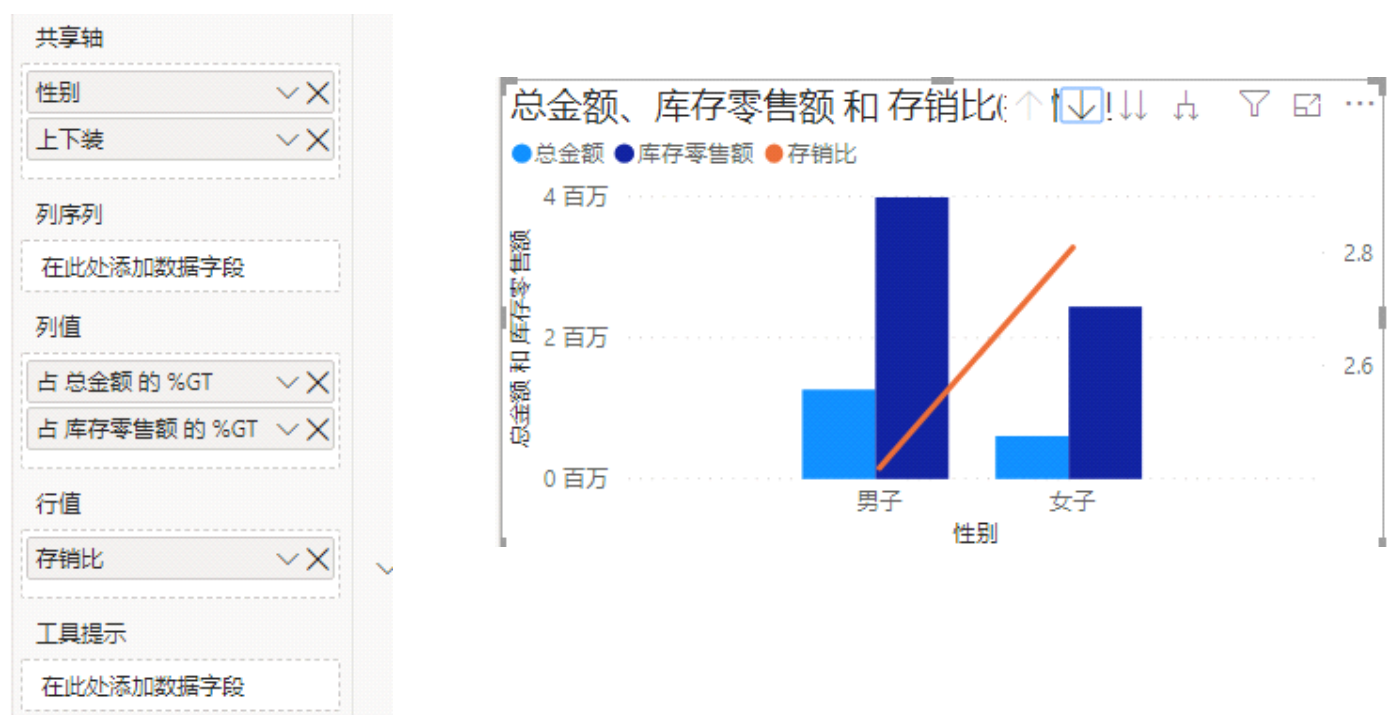
行值

存销比

T B + B 二

2.6 总金额，库存零售额和存销比

2020年5月12日 10:31

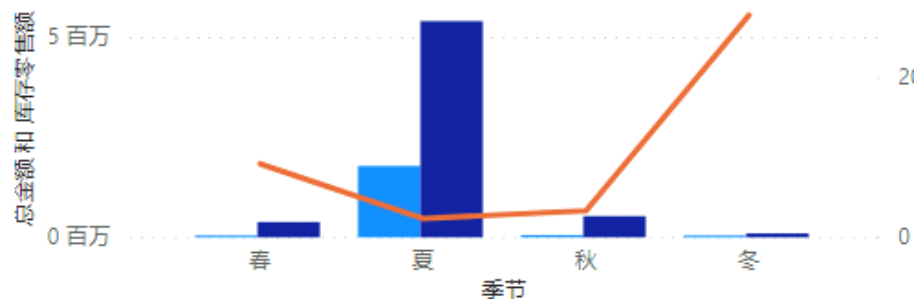


2.7 总金额，库存零售金额与存销比

2020年5月12日 10:34

总金额、库存零售额 和 存销比(按 季节)

● 总金额 ● 库存零售额 ● 存销比



共享轴

季节

列序列

在此处添加数据字段

列值

占 总金额的 %GT

占 库存零售额的 %GT

行值

存销比

工具提示

2.8 明细表

2020年5月12日 10:36

货号	性别	类别	年份	季节	上下装	销售零售额	总数量	总库存	存销比
STY0001	男子	短袖T恤	2017	夏	下装	1024	8	0	0.00
STY0002	男子	短袖T恤	2017	夏	下装	624	3	14	4.67
STY0003	男子	短袖翻领恤	2017	夏	下装	1368	6	12	2.00
STY0004	男子	牛仔中裤	2017	夏	上装	1368	6	0	0.00
STY0005	男子	短袖翻领恤	2017	夏	下装	684	3	12	4.00
STY0006	女子	牛仔长裤	2017	夏	上装	10458	21	158	7.52
STY0007	男子	短袖翻领恤	2017	夏	下装	278	1	9	9.00
STY0008	女子	短袖T恤	2017	夏	下装	0	0	0	
STY0009	男子	短袖T恤	2017	夏	下装	25070	115	301	2.62
STY0010	男子	短袖T恤	2017	夏	下装	592	4	8	2.00
STY0011	女子	短袖T恤	2017	夏	下装	1480	10	0	0.00
STY0012	女子	短袖T恤	2017	夏	下装	0	0	10	
STY0013	男子	短袖T恤	2017	夏	下装	592	4	6	1.50
STY0014	男子	短袖T恤	2017	夏	下装	1504	8	0	0.00
STY0015	男子	短袖T恤	2017	夏	下装	1128	6	9	1.50
STY0016	女子	短袖翻领恤	2017	夏	下装	1240	5	54	10.80
STY0017	女子	梭织中裤	2017	夏	上装	2068	11	18	1.64
STY0018	男子	短袖翻领恤	2017	夏	下装	2480	10	7	0.70
STY0019	男子	牛仔中裤	2017	夏	上装	2544	8	0	0.00

值	
货号	√ ×
性别	√ ×
类别	√ ×
年份	√ ×
季节	√ ×
上下装	√ ×
销售零售额	√ ×
总数量	√ ×
总库存	√ ×
存销比	√ ×

2.8.1 库存量前10名

2020年5月12日 10:40

筛选器

是(全部)

货号

是(全部)

筛选类型 ①

前 N 个

显示项目:

上 10

按值

总库存

应用筛选器

2.8.2 存销比条件格式

2020年5月12日 10:42

背景色 - 销售零售额

格式模式 字段值 ▼ [了解详细信息](#)

依据为字段

存销比背景色 ▼

003.存货仓位

2020年5月12日 14:14

3.1 添加图表

2020年5月13日 8:08

加载项可能访问个人和文档信息。使用加载项，即表示你同意其权限、许可条款和隐私策略。

synopic pancel



为你推荐 ▾

分类

全部

KPI



Synoptic Panel by OKViz ⚙️

Create custom maps with areas connected to the data.

★★★★☆

添加



导入Csv文件后，（注：不局限于一张SVG图表，可同时导入多张图表到Powerbi，适用于商场楼层分布）

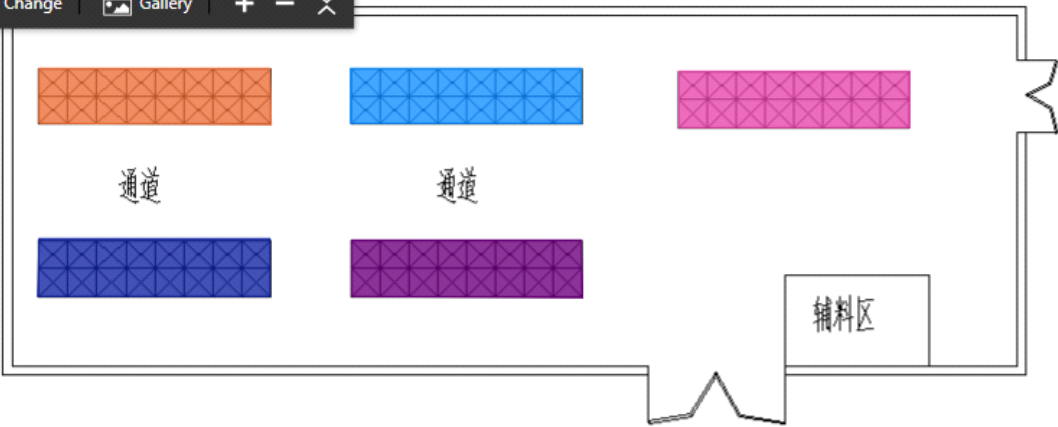
格式设置在Data Colors标签下面

3.2 完成

2020年5月13日 8:34

总库存(按 仓位)

Change Gallery + - X



通道 通道 辅料区

货号	仓位	总库存
		0
STY0002	1号仓位	14
STY0009	1号仓位	301
STY0010	1号仓位	8
STY0012	1号仓位	10
STY0013	1号仓位	6
STY0015	1号仓位	9
STY0022	1号仓位	13
STY0023	1号仓位	18
STY0025	1号仓位	8
STY0028	1号仓位	37
STY0029	1号仓位	16
STY0031	1号仓位	24
STY0033	1号仓位	20
STY0034	1号仓位	14
STY0038	1号仓位	21
STY0039	1号仓位	16
STY0174	1号仓位	27
STY0207	1号仓位	47
STY0208	1号仓位	18
STY0219	1号仓位	59
STY0220	1号仓位	10
STY0226	1号仓位	174
STY0231	1号仓位	3
STY0236	1号仓位	29
STY0237	1号仓位	382
STY0238	1号仓位	44
STY0247	1号仓位	80
STY0250	1号仓位	55
STY0251	1号仓位	30
STY0253	1号仓位	31
STY0254	1号仓位	25
STY0257	1号仓位	16
总计		30085

季节

☐ 春☐ 夏☐ 秋☐ 冬

性别

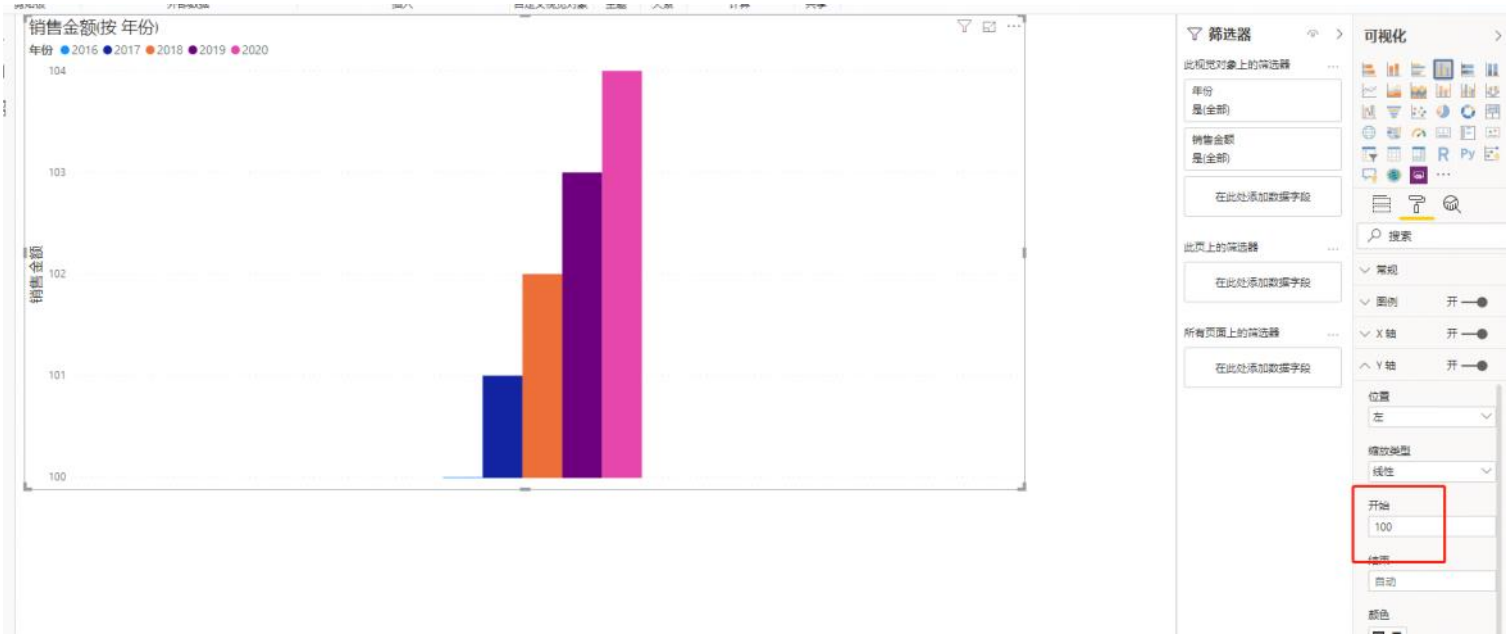
☐ 男子☐ 女子

类别

☐ 短袖T恤☐ 短袖衬衫☐ 短袖翻领恤☐ 夹克☐ 牛仔长裤☐ 牛仔中裤☐ 裙子☐ 梭织长裤☐ 梭织中裤☐ 套头衫☐ 羽绒服☐ 长袖衬衫

004.图表都是骗人的

2020年5月12日 22:22



005.帕累托分析

2020年5月12日 22:50

帕累托分析法又称28原则，是一种得到广泛应用的统计学分析方法，具体来说，指20%的投入可以产出80%的效果。
20%的产品带来80%的销售额。
20%的客户带来80%的利润。
80%的损失是由两名员工造成的。

帕累托图算法方面最重要的部分是一条累计上涨的曲线，从0开始直到100%，因此核心问题是对滚动的累计销售金额的计算，从而算出销售占比的累计变化。使用折线和簇状柱形图即可，关键在于80%恒线的添加

5.1 累计销售金额

2020年5月12日 23:34

(1) 计算销售金额，这是基础度量值

销售金额 = SUM('明细'[销售金额])

(2) 计算累计销售金额（例如：1+2+3+...+99）

累计销售金额 = calculate(计算器,筛选器)

计算器：[销售金额]

筛选器：filter(allselected('明细'),sum('明细'[销售金额])<=[销售金额])

忽略当前视觉对象的筛选条件，显示所有货号的明细表，但是外部切片器的筛选条件不进行忽略，可以自行筛选。

累计销售金额=calculate([销售金额],filter(allselected('明细'),sum('明细'[销售金额])<=[销售金额]))

(3) 计算总销售额

总销售额 = calculate([销售金额],allselected('明细'))

(4) 计算累计销售占比

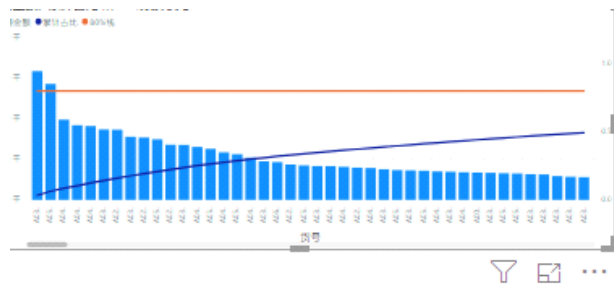
累计占比 = DIVIDE([累计销售金额],[总销售额])

(5) 80%线

80%线 = 0.8

5.2 帕累托图

2020年5月12日 23:34



筛选器

此视觉对象上的筛选器

80%线
是(全部)

货号
是(全部)

累计占比
是(全部)

销售金额
是(全部)

在此处添加数据字段













此页上的筛选器

在此处添加数据字段

所有页面上的筛选器

在此处添加数据字段

可视化



共享轴
货号

列序列
在此处添加数据字段

列值
销售金额

行值
累计占比
80%线

工具提示

5.2.1 数据标签设置

2020年5月13日 0:21

全部都打开太乱了，可以将累计占比显示为“开”，其它数据字段设置为关

5.2.2 切片器

2020年5月13日 0:41

年份, 季节, 性别, 类别

5.3 一步到位的累计占比

2020年5月12日 22:44

一步到位的累计占比=

VAR 累计销售金 =calculate(XXXXXXXX)

VAR 总销售额 = calculate(XXXXXX)

return

divide(累计销售额, 总销售额)

5.4 前20%销售占比矩阵

2020年5月13日 0:42

帕累托分析或者二八分析并不是严格意义上的20%和80%，结合行业特征可灵活设定界限。例如我们可以对每个类别20%销售占比，可以观察到哪些品类销售结构相对正常，哪些品类Top款销售占比过低，比较分散，需要做出调整。

5.4.1 计算前20%累计销售金额和销售占比度量值

2020年5月12日 22:55

(1) 计算商品种类 = calculate(distinctcount('明细'[货号]),'明细'[销售金额]>0)

(2) 20%种类数量 = Round([计算商品种类]*0.2,0)

(3) 计算前20%销售金额

前20%销售金额 = SUMX(TOPN([20%种类数量],all('明细'[货号]),[销售金额],DESC),[销售金额])

(4) 计算前20%销售占比

前20%销售占比 = divide([前20%销售金额],[销售金额])

5.4.2 矩阵

2020年5月12日 23:17

行

性别

类别

列

在此处添加数据字段

值

计算商品种类

20%种类数量

销售金额

前20%销售额

前20%销售占比

006.购物篮分析

2020年5月13日 2:28

6.1 复制一张商品表

2020年5月14日 9:28

在PowerQuery里面复制一张商品表，命名为“关联商品表”。这两者之间是相互匹配与比对的关系

商品表用于体现结果，关联商品表用于过程计算

注意：复制后的查询 将与原查询失去关联，原查询更改数据处理步骤不影响复制后的查询步骤。

6.2 汇总销售明细

2020年5月14日 9:29

分组依据

指定要按其进行分组的列以及一个或多个输出。

☐ 基本 ☒ 高级

商品编码

▼

小票流水号

▼

日期

▼

添加分组

新列名	操作	柱
销量	求和	销售数量
金额	求和	销售金额
毛利	求和	毛利额

添加聚合

确定

取消

这一步就是把一维数据做数据透视

6.3 使用合并查询把小票流水V过来

2020年5月14日 9:29

小票流水号分别引入商品表和关联商品表，用来判断两边的产品存在于哪笔购物小票中

选中商品表，合并查询，通过商品编码把小票流水V过来，然后只保存小票流水

同理，关联商品表也是这样操作

6.4 合并商品表与关联商品表

2020年5月14日 9:29

选中商品表，合并查询。只保留商品编码，并重命名为关联商品编码

合并

选择表和匹配列以创建合并表。

商品表

商品编码	商品名称	售价	进价	小票流水号
101	鸡蛋	2.5	3	A2020301001
101	鸡蛋	2.5	3	A2020301002
101	鸡蛋	2.5	3	A2020301003
101	鸡蛋	2.5	3	A2020301004
102	西红柿	7.9	5	A2020301001

关联商品表

商品编码	商品名称	售价	进价	小票流水号
101	鸡蛋	2.5	3	A2020301001
101	鸡蛋	2.5	3	A2020301002
101	鸡蛋	2.5	3	A2020301003
101	鸡蛋	2.5	3	A2020301004
102	西红柿	7.9	5	A2020301001

联接种类

左外部(第一个中的所有行, 第二个中的匹配行)

☐ 使用模糊匹配执行合并

模糊匹配选项

✓ 所选内容匹配第一个表中的 239 行(共 239 行)。

确定

取消

6.5 剔除商品编码本身

2020年5月14日 10:24

自定义列

添加从其他列计算的列。

新列名

剔除两边相同商品编码

自定义列公式 ①

```
= if [商品编码]=[关联商品编码] then "N" else "Y"
```

可用列

商品编码
商品名称
售价
进价
小票流水号
关联商品编码

<< 插入

[了解 Power BI Desktop 公式](#)

✓ 未检测到语法错误。

确定

取消

筛选只选Y后，这列就可以删除了

6.6 去除关联商品表加载

2020年5月14日 10:31

将关联商品表加载去掉勾选，因为此查询是一个过渡数据，需要的数据已经链接至商品表查询中了，设置完成关闭并应用

6.7 建立关系

2020年5月14日 11:18



6.8 计算销售笔数、共同销售笔数、所占比例

2020年5月14日 10:32

(1) 计算销售总笔数，对销售单编号非重复计数即可

销售总笔数 = `DISTINCTCOUNT('销售表'[小票流水号])`

(2) 计算共同销售笔数，因为是A商品对每一个B商品的共同笔数，所以此处直接计数

共同销售笔数 = `COUNTA('商品表'[关联商品编码])`

(3) 计算A和B共同交易笔数在A的交易笔数中的占比，将两个度量值直接相除即可

所占比例 = `Divide([共同销售笔数],[销售总笔数])`

其实，销售总笔数与共同销售笔数都是中间计算过程，为了能讲清楚，所以拆分了，你也可以直接写所占比例公式。

6.9 占比的条件格式

2020年5月14日 11:19

所占比例格式 = if([所占比例]>0.3,"Green",IF([所占比例]>=0.2,"Yellow","Red"))

6.10 可视化：矩阵

2020年5月14日 11:19



所占比例-右-背景色-格式模式-字段值-占比条件格式

6.11 表

2020年5月14日 11:19

表：

商品名称

关联商品名称

其它的项目根据自己需要定

007.畅销商品分析

2020年5月15日 9:33

在分析畅销商品时，最忌讳的就是只按销售数量或销售金额降序确定畅销品，这样可能带来极大的账务风险。

一、售罄率高不代表畅销

售罄率 = 某段时间销售数量/订货数量*100%

例如：毛巾就订了2条，一下子就卖完了。

二、销售数量高也不代表畅销

例如：A鞋进货100双，7天卖了50双；B鞋进货50双，7天卖了40双；这样根本看不出谁是畅销品，因为B鞋只进了50双，可能卖到30双时已经断码了，影响了成交率。如果把A当成畅销品，你追加一大笔订单，也可能会采购了很多消费者不需要的商品。

三、销售数量与售罄率综合考虑能否说明商品畅销？

例如：30天后，A鞋总共卖了98双，售罄率为98%；B商品销售了49双，售罄率也是98%；如同销售10天时，B鞋的销售数据已无法真实反映消费者需求，销售30天时，A和B售罄率也全部无法反映真实情况，当货品卖到50%-70%时，可能断码现象已经产生了。A在10天后销售也被抑制，比真实值偏低。

所以，销售数量和售罄率结合也无法反映商品是否畅销

四、结合时间因素判断商品是否畅销

判断商品是否畅销，需要引入时间因素，在越短时间内销量越高（以新品为例，不考虑折扣因素），则商品畅销。

当商品供应充足，尺码齐全，但是售罄率没有达到50%，销售数量可以反映消费者真实需求。这时我们可以做出判断，售罄率越快达到50%并且该期间销量越高，则越畅销。通过计算达到售罄50%之前平均每天的销量进行排名，即可判断什么是真正的畅销品。

达到售罄50%平均每天销量 = 达到50%售罄时的销量/达到50%售罄时销售天数*100%

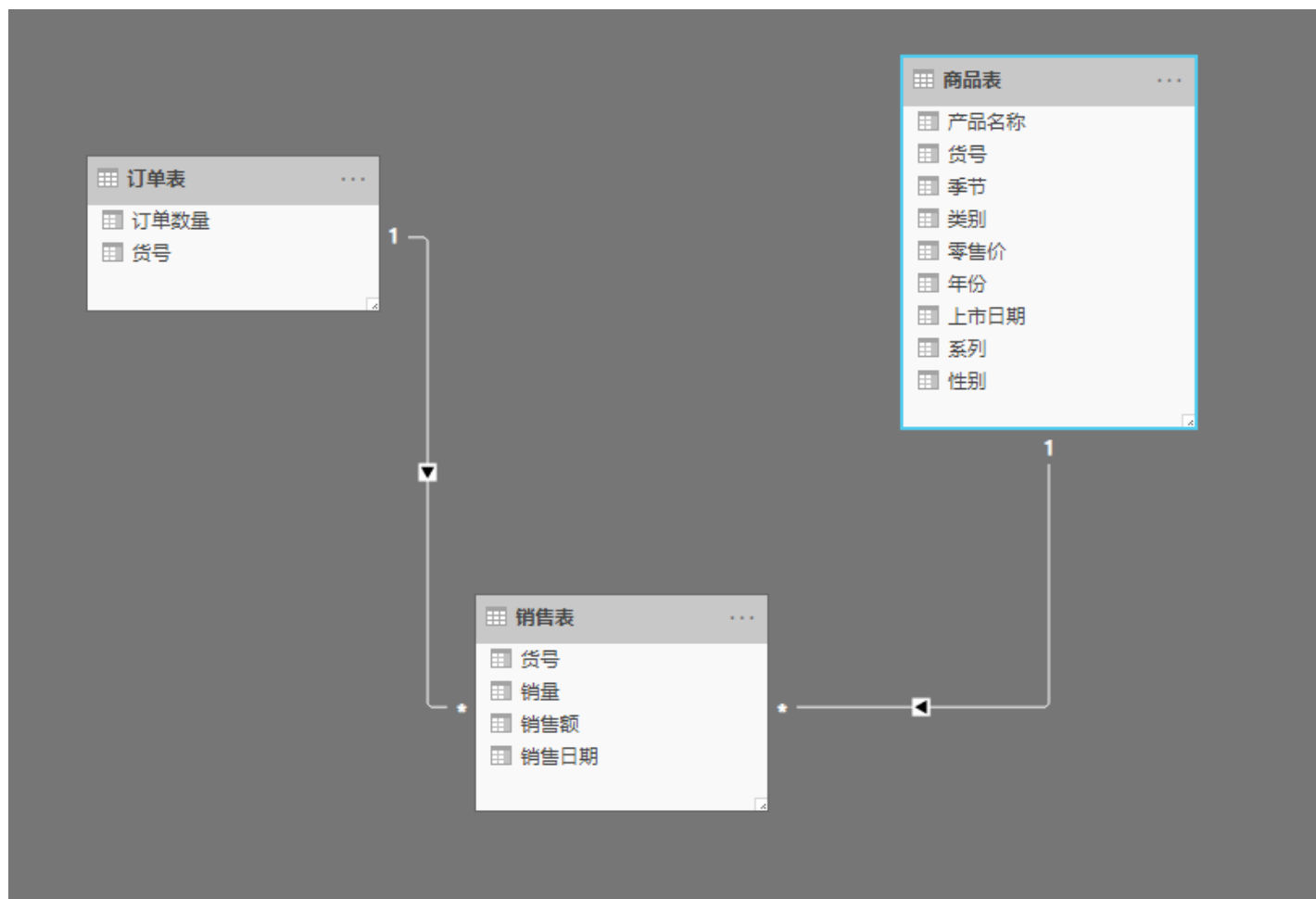
当商品到截止日期仍没有达到50%的售罄，可以认定商品被一直充分供应，公式中的销售天数为截止日期减去该商品的到货日期。

五、总结

什么是畅销品？要综合考虑销量、售罄率、时间三个因素。

7.1 建立关系

2020年5月15日 9:58



7.2 计算达到50%售罄时的销售天数

2020年5月15日 9:58

要计算某产品什么时候售罄达到50%，首先要计算产品的滚动累计销量，然后滚动销量与订单总数做对比，当达到50%时停止计算。

在【销售表】中添加列

(1) 计算滚动销量

滚动销量 =

```
SUMX (
  FILTER (
    FILTER ( '销售表', '销售表'[货号] = EARLIER ( '销售表'[货号] ),
      '销售表'[销售日期] <= EARLIER ( '销售表'[销售日期] )
    ),
    '销售表'[销量]
  )
```

'销售表'[销售日期] <= EARLIER ('销售表'[销售日期])
第一天直接照抄数量
第二天加上第一天
第三天加上第一和第二天
.....以此类推

(2) 计算售罄率

售罄率 = 某段时间销售数量/订货数量*100%

售罄率 = DIVIDE('销售表'[滚动销量], RELATED('订单表'[订单数量]))

多端找一端，销售表找订单表

(3) 计算售罄率大于50%的最小销售日期

1. 当该商品一直没有达到50%时返回最后一个销售日期

MAXX (FILTER ('销售表', '销售表'[货号] = EARLIER ('销售表'[货号]), '销售表'[售罄率]) < 0.5, 返回: MAX ('销售表'[销售日期]))

2. 当该商品达到50%时返回最小的一个销售日期

MINX (FILTER (FILTER ('销售表', '销售表'[货号] = EARLIER ('销售表'[货号]), '销售表'[售罄率] >= 0.5), '销售表'[销售日期])

3. 最后用IF语句判断

if(售罄率<0.5, 返回最大销售日期, 返回售罄率大于0.5那张表中的最小销售日期)

4. 写完整

售罄率大于50%的最小销售日期 =

```
IF (
  MAXX ( FILTER ( '销售表', '销售表'[货号] = EARLIER ( '销售表'[货号] ), '销售表'[售罄率] )
    < 0.5,
  MAX ( '销售表'[销售日期] ),
  MINX (
    FILTER (
      FILTER ( '销售表', '销售表'[货号] = EARLIER ( '销售表'[货号] ),
        '销售表'[售罄率] >= 0.5
      ),
      '销售表'[销售日期]
    )
  )
```

(4) 引入上市日期

上市日期 = RELATED ('商品表'[上市日期])

(5) 计算达到50%售罄时销售天数

达到50%售罄时销售天数 = DATEDIFF ('销售表'[上市日期], '销售表'[售罄率大于50%的最小销售日期], DAY) + 1

(6) 计算达到50%售罄时的销量

达到50%售罄时的销量 =

```
SUMX (
  FILTER (
    FILTER ( '销售表', '销售表'[货号] = EARLIER ( '销售表'[货号] ),
      '销售表'[销售日期] <= '销售表'[售罄率大于50%的最小销售日期]
    ),
    '销售表'[销量]
  )
```

先筛选一个货号，
日期小于等于售罄率大于50%最小销售日期的销量数量之和

7.3 统计畅销款

2020年5月15日 9:58

越快达到50%售罄的商品越畅销，在达到50%售罄之前，该商品不受断码等因素影响，可以反映真实的销售状况，使该期间的销售数量可以倒推出该商品的理想销售数量，从而进行畅销款排名。

1、新建表：

畅销款 =

SummarizeColumns (

'销售表'[货号],

'销售表'[上市日期],

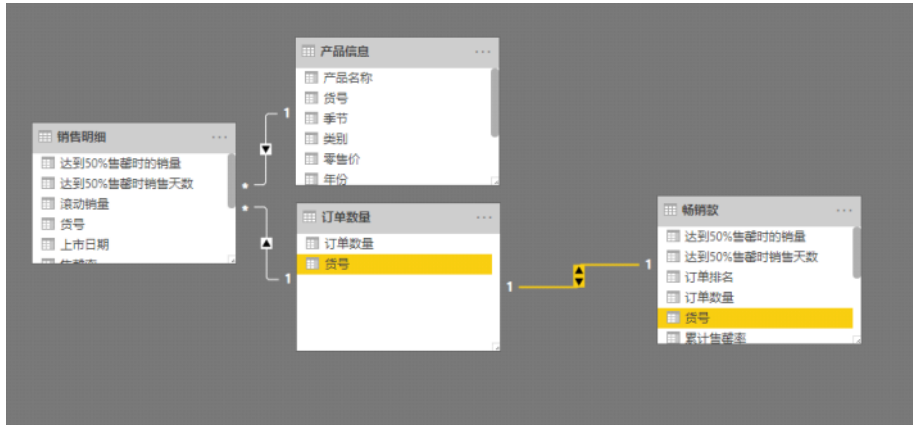
'销售表'[售罄率大于50%的最小销售日期],

'销售表'[达到50%售罄时销售天数],

'销售表'[达到50%售罄时的销量],

"累计销量", SUM ('销售表'[销量])

)



2、引入订单数量到畅销款表中

订单数量 = Related('订单表'[订单数量])

3、计算累计售罄率

售罄率 = 某段时间销售数量/订货数量*100%

累计售罄率 = DIVIDE('畅销款'[累计销量], '畅销款'[订单数量])

4、计算理想平均周销量

理想平均周销量 = divide('畅销款'[达到50%售罄时的销量], '畅销款'[达到50%售罄时销售天数]) *7

注：达到50%售罄时的销量为真实反映顾客需求的销量，换算成周便于比较

5、计算排名

订单排名 = RANKX(ALL('畅销款'), '畅销款'[订单数量])

累计销量排名 = RANKX(ALL('畅销款'), '畅销款'[累计销量])


修正排名 = RANKX(ALL('畅销款'), '畅销款'[理想平均周销量])



注：使用ALL函数确保排名范围为全部产品，Rankx选取不同排名字段

7.4 可视化：表

2020年5月15日 11:26

值	
货号	▽ ×
产品名称	▽ ×
上市日期	▽ ×
售罄率大于50%的最小排	▽ ×
订单数量	▽ ×
累计销量	▽ ×
累计售罄率	▽ ×
达到50%售罄时销售天数	▽ ×
达到50%售罄时的销量	▽ ×
理想平均周销量	▽ ×
订单排名	▽ ×
累计销量排名	▽ ×
修正排名	▽ ×

货号  

按 修正排名 底端对...  

筛选类型 ①

前 N 个 ▼

显示项目:

下 ▼ 10

按值

修正排名 ▼ ×

应用筛选器

7.5 总结

2020年5月15日 11:37

1. 累计售罄率高的商品不一定排名靠前，因为该商品“售罄”已失真。
2. 按销量排名和按照修正排名也有很大不同，因为销量受订货量影响，订单数量小会影响销量。
3. 修正排名是相对准确的畅销款排名，可为后期订货提供参考
4. 案例中，排名与订单前10名重合度只有50%，说明订单下单不是很准确

注意：数据分析仅供参考，实际工作中有很多问题。团购、退货等

008.生命周期分析

2020年5月15日 9:35

数据表中新建列：

天数 = [日期]-RELATED('信息表'[发布日期])

拿到每个视频发布了多少天

9.1 计算发布日期

2020年5月17日 8:05

发布日期 = firstdate('数据表'[日期])

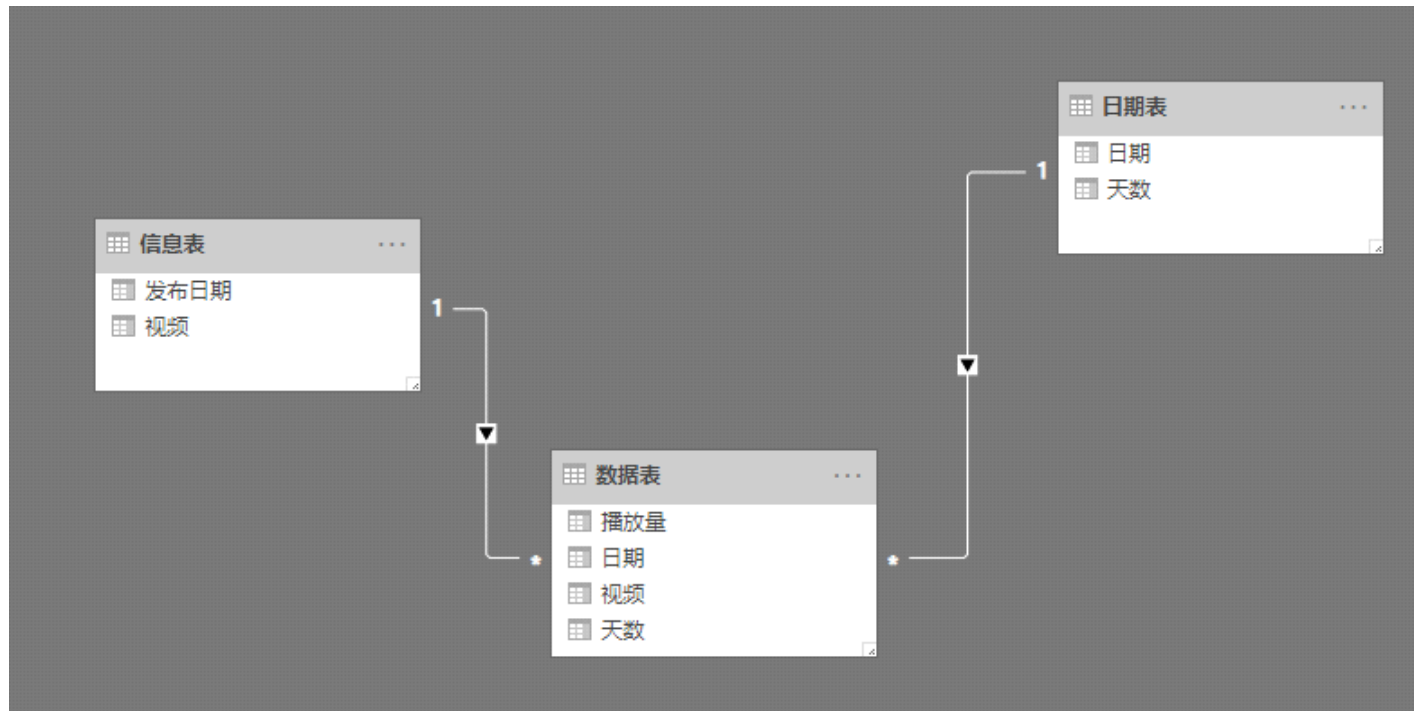
9.2 计算天数

2020年5月17日 8:05

天数 = [日期]-RELATED('信息表'[发布日期])

9.3 建议关系

2020年5月17日 8:05



9.4 累计播放量

2020年5月17日 8:06

累计销售量 = IF(ISBLANK([播放总量]),BLANK(),CALCULATE([播放总量],FILTER(ALL('日期表'),'日期表'[天数]<=MAX('日期表'[天数]))))

记得先要建立[播放总量]的度量值

9.5 可视化 折线图

2020年5月17日 8:13

轴

天数

图例

视频

值

累计销售量

工具提示

009.陈列面积规划

2020年5月15日 9:33

陈列指把商品按照一定规律展示给顾客，激发顾客的购买欲望。
陈列是一种综合艺术，好的陈列可以增强顾客的购物体验，并将商品信息快速地传递给顾客。

零售企业的陈列不仅要具备美感，而且需要为销售服务，如果陈列将橱窗搭配得非常好看，则会吸引顾客进店。
但是当顾客询问尺码信息时，如果店铺断码，并没有相应的库存，则会造成成交流失。

因此陈列需要与店铺货品数据相结合才能最大化产出。

民用级零售服务行业最忌讳的：

- 1、搞卫生，尘土问题应该应该从根源上解决。门朝东或西，食用用品紧靠蔬菜阵列区，这都是自己给自己找活干，增加人员成本（追求死角卫生）
- 2、频繁更换陈列位置，促销商品加大陈列没有错，优化陈列也没问题，但是不要将陈列调整做为日常工作，影响顾客体验
- 3、将只会干活的人在没培训和学习的时候提升到管理层，他们不懂科学，不信科学。他们常说老子打天下时你在哪？你跟我讲科学？勿忘十年浩劫

商品	陈列数量	日均销量	送货时间	最低库存数量
1号商品	8	25	3	75+8
2号商品	16	3	3	9+16

9.1 设置权重参数

2020年5月17日 9:25

根据 **品种数**，**一段时间内的销售金额**，**库存金额** 三个方面考虑，分别赋予不同因子不同权重，将三者进行加权计算，得出建议陈列面积。

我们关注哪个因素就把哪个因素的滑杆提高，建立陈列占比也将会偏重该因子。

一般情况下考虑库存和销售的因素比重较多，品种数也不能不考虑，库存大了就要促销消化库存或退货，今后要注意订货数量或者直接清品库存量小的无法大面积陈列。

中间的图表是建议陈列面积，向下钻取可以看分配

矩阵为各系列的明细

某一因素或指标相对于某一事物的重要程度，其不同于一般的比重，体现的不仅仅是某一因素或指标所占的百分比，强调的是因素或指标的相对重要程度，倾向于贡献度或重要性。通常，权重可通过划分多个层次指标进行判断和计算，常用的方法包括层次分析法、模糊法、模糊层次分析法和专家评价法等



权重和 = [库存金额权重 值]+[销售金额权重 值]+[品种数权重 值]

9.2 建立基础度量值

2020年5月17日 9:25

销售数量 = SUM('销售'[销量])

销售额 = SUM('销售'[销售金额])

库存量 = SUM('销售'[库存数量])

库存额 = SUM('销售'[库存金额])

品种数 = CALCULATE(DISTINCTCOUNT('销售'[货号]),'销售'[库存数量]>0)

平均单品库存量 = DIVIDE([库存量],[品种数])

9.3 计算建议陈列占比

2020年5月17日 9:25

一、计算建议陈列占比

品种数占比 = $\text{DIVIDE}([\text{品种数}], \text{CALCULATE}([\text{品种数}], \text{ALL}('销售')))$

销售额占比 = $\text{DIVIDE}([\text{销售额}], \text{CALCULATE}([\text{销售额}], \text{ALL}('销售')))$

库存额占比 = $\text{DIVIDE}([\text{库存额}], \text{CALCULATE}([\text{库存额}], \text{ALL}('销售')))$

二、占比与权重相结合计算加权

加权 = $[\text{品种数占比}] * [\text{品种数权重 值}] + [\text{销售额占比}] * [\text{销售金额权重 值}] + [\text{库存额占比}] * [\text{库存金额权重 值}]$

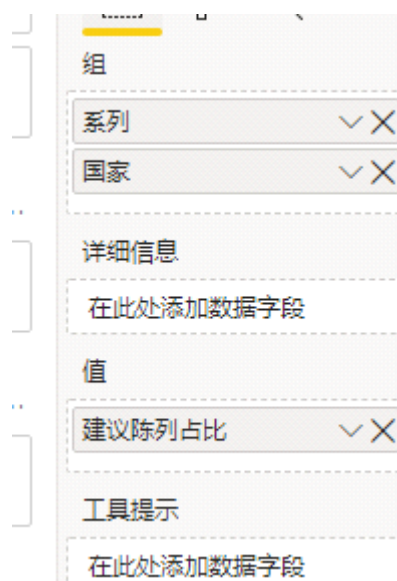
三、计算建议陈列占比

建议陈列占比 = $\text{DIVIDE}([\text{加权}], \text{CALCULATE}([\text{加权}], \text{ALLSELECTED}('销售')))$

使用ALLSELECTED而不是ALL，这是由于我们只清除该图表范围的筛选器计算总值，外部筛选器仍对其发生作用。就是百分比的显示永远是正确的

9.4 可视化

2020年5月17日 13:07



树形图



矩阵

010.单品陈列规划

2020年5月17日 11:26

各系列陈列面积确定后，需要确定各个子区域具体单品怎么进行陈列。每个卖场板块都有相对较好的位置，这些位置的顾客视线佳，用来陈列最好的货品，也有中等的位置，还有较差的位置，顾客较少注意到。

陈列位置的好坏很容易区分，首先是顾客必经之路，堆头端架，在货架上顾客抬手就能拿到的地方就是黄金位置。所以食用盐一定放在最下面。

考虑商品的因素就太多了，这里我们简化一下：三个因素

销售状况，库存状况，上市日期。

换句话说，销售越好，库存越足，越新的商品就是好的商品。（注意，实际中每个企业的实际情况不一样，你要针对自己企业的实际情况分析）

三个因素的重要程序可能不同，同样，我们可以设立权重综合进行排名。

10.1 单品排名分组【新建列】

2020年5月18日 16:23

单品排名细化到不同地区不同系列的组内排名，首先从销售金额，库存金额，上市日期三个维度分别进行排名，依据不同权重计算出组内总排名，然后按照组内排名位置对名词进行相应分组。

(1) 新建国家和系列合并列

国家&系列 = '销售'[国家] & '销售'[系列]

单品排名计算的是每个性别每个系列内部的款式排名，合并国家和系列两个字段以备排名使用。

(2) 从销售金额，库存金额，上市日期三个维度进行排名

单品销售额排名 = RANKX(FILTER('销售','销售'[国家&系列]=EARLIER('销售'[国家&系列])), '销售'[销售金额])

单品库存额排名 = RANKX(FILTER('销售','销售'[国家&系列]=EARLIER('销售'[国家&系列])), '销售'[库存金额])

单品上市日期排名 = RANKX(FILTER('销售','销售'[国家&系列]=EARLIER('销售'[国家&系列])), '销售'[上市日期])

(3) 加权排名

加权指数 = '销售'[单品销售排名]*0.3 + '销售'[单品库存额排名]*0.4 + '销售'[单品上市日期排名]*0.3

单品综合排名 = RANKX(FILTER('销售','销售'[国家&系列]=EARLIER('销售'[国家&系列])), '销售'[加权指数],,ASC)

对三个维度的排名分别赋予合适的权重计算加权指数，计算完成后使用加权指数对货品进行综合排名

我们用升序排列，因为加权指数越小排名越靠前，这与销售金额越大排名越靠前是一种相反的趋势，因此需要新增参数改变排名方式。

(4) 单品排名分组

内部品种数 = COUNTAX(FILTER('销售','销售'[国家&系列]=EARLIER('销售'[国家&系列])), '销售'[货号])

单品排名分组 =

Var a =

DIVIDE('销售'[单品综合排名], '销售'[内部品种数])

return

IF(a<=0.2,"前20%",IF(a<=0.5,"20%-50%", "后50%"))

使用COUNTAX+FILTER配合判断每个国家每个系列中有多少单品数。

将单款排名与内部款式数相除，使用if判断该款式在内部的位置

使用VAR新增名次变量，有助于缩减if函数的书写长度

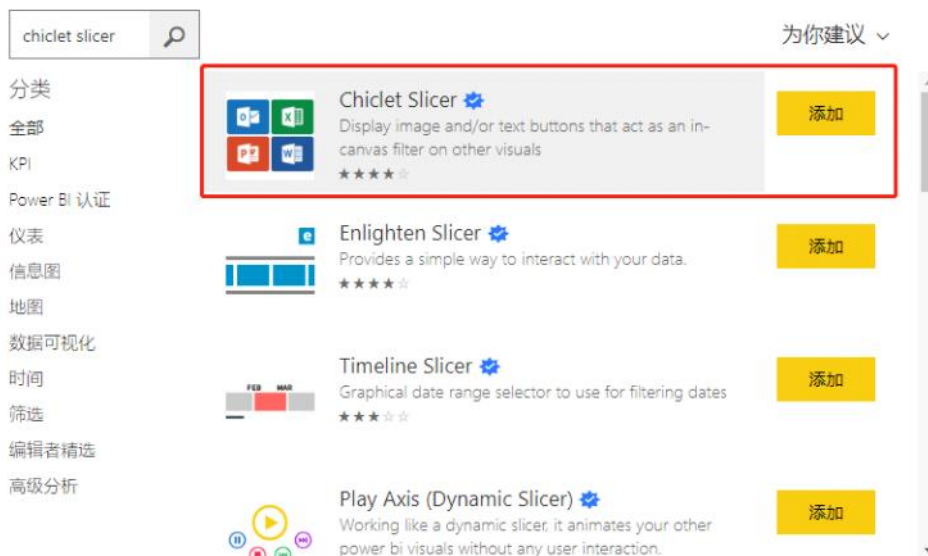
10.2 制作图片切片器

2020年5月18日 16:24

Power BI 视觉对象

应用商店 | 我的组织

加载项可能访问个人和文档信息。使用加载项，即表示你同意其权限、许可条款和隐私策略。



配图 =

IF(

'销售'[库存数量]>0,

SWITCH(

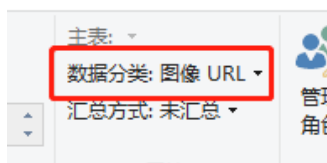
'销售'[单品排名分组],

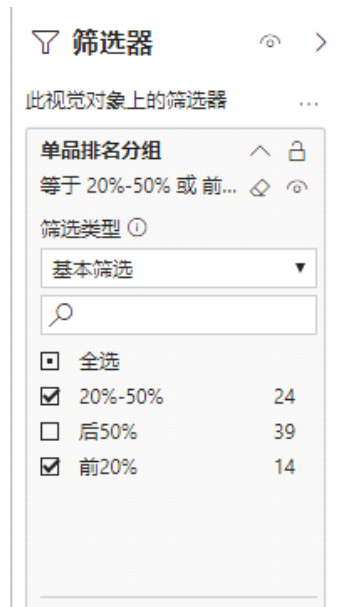
"前20%","http://img.jf258.com/uploads/2014-08-28/151432314.jpg",

"20%-50%","http://img.jf258.com/uploads/2014-08-28/151432926.jpg",

BLANK()

))





10.3 制作图片展示表

2020年5月18日 16:24