

产品测试中 JAR 问题的分析方法

向采发

《市场研究》 2005 年第 9 期

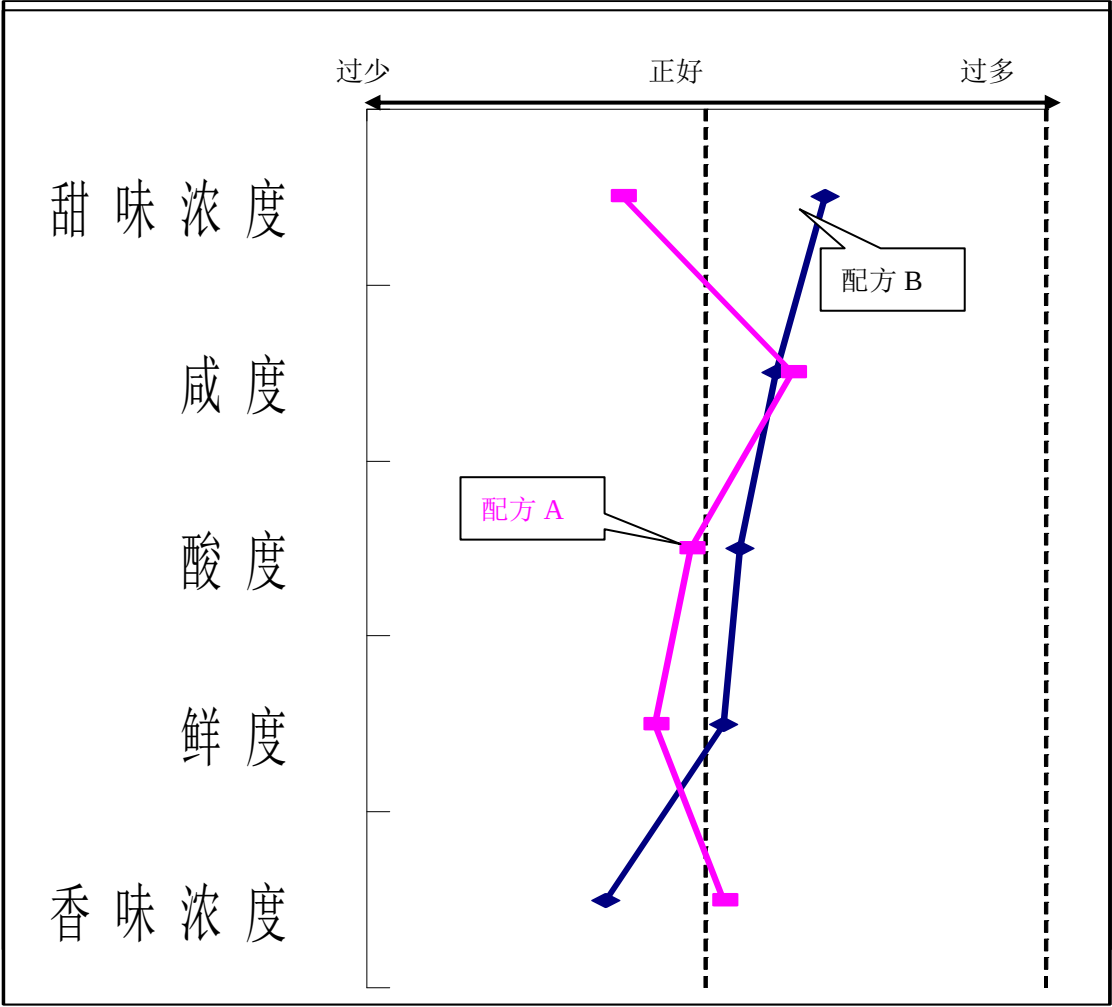
产品测试研究中，经常需要消费者对产品的感官属性进行评价，而人类通常对感官属性有一个最佳的适应点，例如：香味的浓淡程度，并不是香味越浓越好；甜味，也不是越甜越好。针对这种情况，在产品测试中，我们通常有两种方式来测度消费者对某个属性的感受：

- 1、将属性感觉表现转化为快感指标(Hedonic scale)，比如消费者对属性表现的喜欢程度。这是很多研究公司常用的一种手段，提问方式可能是这样的：请问您对该产品气味浓淡的喜欢程度如何？请用 1 到 5 分来评价，1 分表示非常不喜欢，5 分表示非常喜欢。这种方式的优点是：评价量表（喜欢程度）是有次序的，通常可以利用显著性检验/相关/回归分析等方式来进行统计分析；但缺点是：即使被访者回答不喜欢，也难以从评价中得知气味是偏浓呢，还是偏淡，如果偏淡，大致需要改进多少？因此通常还需要借助其他手段来补救。
- 2、用“正好(Just-About-Right，简称 JAR 问题)”的尺度方式来提问。同样对上面问题，我们提问的方式变为：请问你认为该产品气味的浓淡程度如何？请用 1~5 分来评价，1 分表示太淡，5 分表示太浓，3 分表示正好。在这种方式下，评价得分并非越高越好，而是越靠近 3 分越好。JAR 问题的优势是，可以借助于消费者的个人感知标准，确定最优的属性改进方向和大致改进尺度。因此 JAR 问题可以帮助产品开发人员直观地诊断产品的缺陷，确定产品属性改进方向和程度。但其缺点是：这种评价方式下，研究人员通常缺乏有力的统计分析手段，通常的显著性检验/相关/回归分析难以进行，在多个 JAR 属性中，难以确定这些因素的相对重要性和改进优先度。

虽然存在种种不便，但是 JAR 问题由于其对产品开发改进指导的直观性，仍然受到产品研发人员的青睐，目前，在食品/饮料、纺织品、个人护理用品等领域中，JAR 问题是产品测试的常用尺度。而研究分析人员经过不懈的努力，也发展出了对 JAR 问题一系列分析手段，兹举例如下

一、常规的均值和图形分析

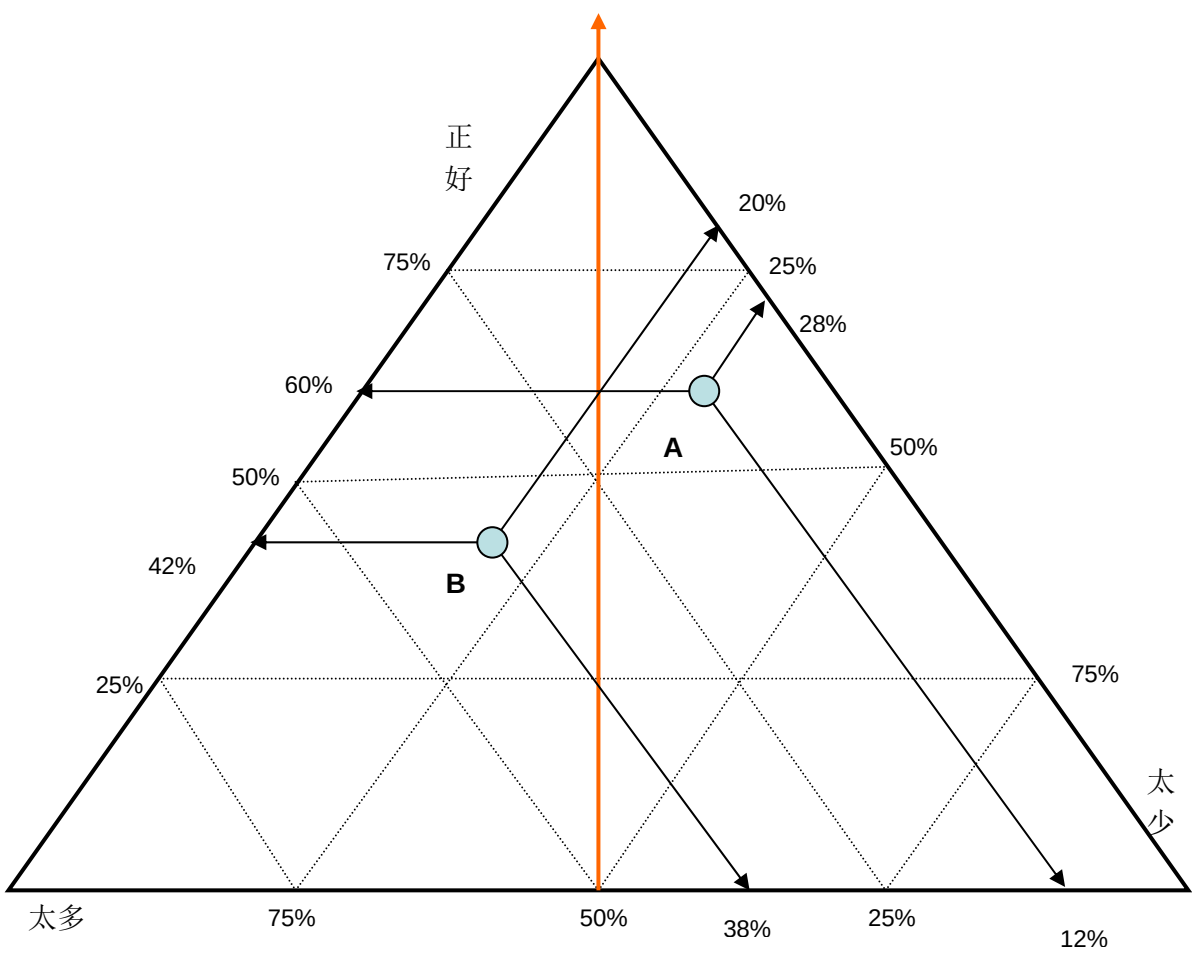
蛇形图可以将属性的平均得分通过折线图表示出来，这样可以看到每个属性离中心点的大致偏向。这种方式直观简洁，一目了然，是最常用的方法之一。



二、三角图展示方法

三角图也是用来呈现 JAR 属性的一个好手段，可用于呈现单个属性的评价，也可用于多个属性的评价，以及多个产品的对比呈现。

三角图的制作很简单，采用等边三角形的方式，左边表示 JA 人群的比例，右边表示 TL 人群的比例，下方表示 TM 人群的比例，具体的刻度方式参见下图：



图：一个三角图示例

上图含义：对于属性 A，60%的人认为正好合适，28%的人认为过少，12%的人认为过多。而对于 B 配方，42%的人认为正好，20%的人认为过少，38%的认为过多。

将多个属性置于同一个三角图，可以方便地呈现产品的优点和缺点：

- 属性越靠近三角形的上尖端，则表示产品在该属性上表现越佳；
- 属性越靠近底端，则表明该属性的合适程度越低；
- 属性如果靠近左边（位于中心线左边），则表明认为该属性过强（多）的比例超过过弱（少）的比例；
如果靠近右边（中心线右边），则表明认为属性过弱（少）的比例超过过强（多）的比例。

三、Penalty 分析

Penalty 分析是 JAR 问题分析中最常用方法，它可以在多个属性中，找出每个属性的相对重要程度，并确定产品改进的优先次序。Penalty 分析的基本思想是：通过分析具体属性和对产品总评价之间的关联，确定每个具体属性对总体属性的影响程度，从而达到确定具体数据重要性的目的。在传统的属性评价中，这部分工作通常是通过回归分析和相关分析进行的。

Penalty 分析的过程

Penalty 分析需要以下两个方面的指标：

- **对产品的总体评价。**总体评价必须是等差或等比的尺度，比如对产品的总体喜欢程度用 1-7 分评价，分数越高越喜欢。
- **对具体属性的 JAR 偏好评价。**假定消费者评价了产品的某个具体属性，采用方式的是，中间值表示最合适。对一个 5 分量表而言：1 分表示太少，2 分稍微有点少，3 分正好，4 分稍微有点多，5 分表示太多。

于是根据每个属性，我们可以将消费者分为 3 类：

- 第 1 类（打 1~2 分者）：认为产品在属性上偏少的人，记为 TL；

- 第二类 (打 3 分者): 认为属性表现为正好的人 , 记为 JA ,
- 第三类 (打 4~5 分者): 认为属性表现为偏多的人 , 记为 TM。

对于总体评价指标 (总体喜欢程度), 记消费者对该产品的总体喜欢程度为 L。针对每个属性的三类人群 , 记每类人群对产品的总体喜欢程度 (平均值) 为

- $L(TL)$: 表示 TL 人群对产品的总体喜欢程度
- $L(TM)$: 表示 TM 人群对产品的总体喜欢程度
- $L(JA)$: 表示 JA 人群对产品的总体喜欢程度

于是我们可以计算每个属性 (比如说甜度) 的 Penalty 指数(简记为 P)

- $P(TL)=L(TL)-L(JA)$, TL 人群的 Penalty 指数
- $P(TM)=L(TM)-L(JA)$, TM 人群的 Penalty 指数

Penalty 的大小反应了某个属性的偏向对总体指标的影响力度 , 通常称为 Mean drops。如果将 Penalty 指数乘以该偏向的人群比例 , 则可以计算出由于这个属性偏向某个方向 , 而导致消费者总体评价下降的总分数 , 常称为 Total mean drops。

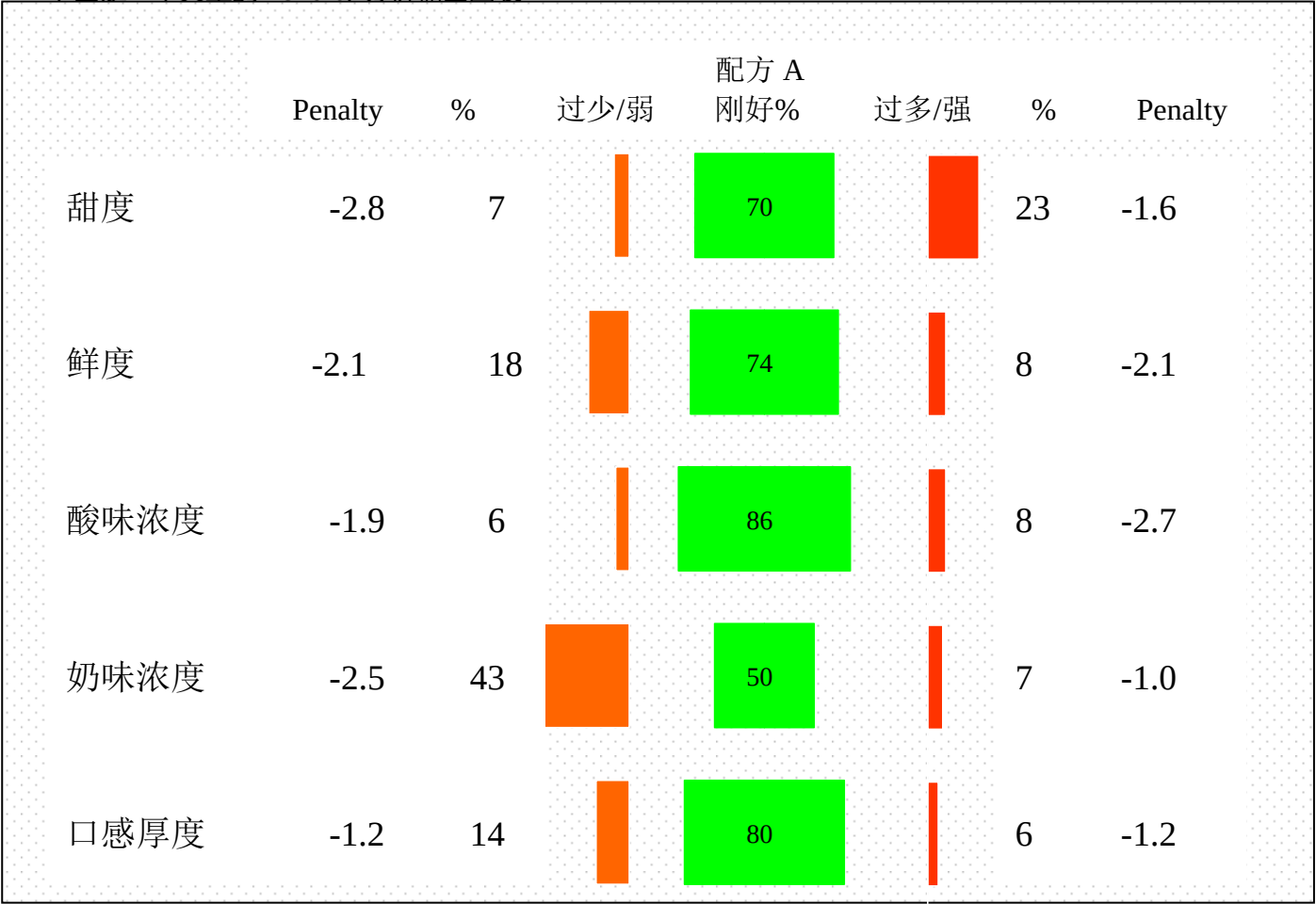
Penalty 指数反映了属性对总体评价的影响程度 , 因此它是一个可以用来衡量属性重要程度的标志 , 也被称为 Penalty 重要性。借助于象限分析等手段 , 可以确定产品属性的改进优先度和改进方向。

一个 Penalty 分析计算过程

JAR 刻度	人数	占总	产品总体喜	分类平均喜	Penalty 指数	
--------	----	----	-------	-------	------------	--

		体比 例	欢程度(平均 值)	欢程度	Mean drops	total mean drops
1 分-太少	8	8%	2.0	L(TL)=2.60	2.60-4.50	$(-1.90)*(8+12)$
2 分-稍少	12	12%	3.0		=-1.90	=-38
3 分-正好	50	50%	4.5	L(JA)=4.50		
4 分-稍多	20	20%	3.5	L(TM)=3.17	3.17-4.50	$(-1.33)*(20+10)$
5 分-太多	10	10%	2.5		=-1.33	=-40

下面是一个典型的 Penalty 分析输出图表



从上图可以看出，奶味浓度偏淡是该产品的最主要问题。在甜度方面，虽然有 23%的消费者认为过甜，但是过甜对消费者的总体喜欢影响不大，Penalty 指数仅为-1.6；而甜度太弱对消费者的总体喜欢影响很大，因此总体来看，该产品在甜度方面的改进需要并不迫切。

四、JAR 问题的统计分析

在统计分析方面，为了客户 JAR 尺度的非线性问题，目前研究者(R. Xiong 等,2004)这提出了采用哑变量（Dummy Variable）处理的方法，也有些研究者尝试采用优化线性尺度后回归的方法。其中哑变量处理的方法比较直观，在此作简要介绍：

采用哑变量处理的基本思路是：将 JAR 问题的两个方向测度转化为两个哑变量，具体来说，对于一个 5 分尺度的 JAR 变量，处理方式如下

原始变量尺度	哑变量		
X	Z ₁	Z ₂	Z=Z ₁ -Z ₂
1（太少）	-2	0	-2
2（有点少）	-1	0	-1
3（正好）	0	0	0
4（有点多）	0	1	-1
5（太多了）	0	2	-2

在进行过哑变量处理之后，原始变量事实上被拆分为两个单向变量，然后我们对这两个变量进行传统意义上的统计分析，比如主成分分析、回归分析等。

以回归分析为例，假定原来的回归模型为

$$Y=a+b_1X_1+b_2X_2+...+b_kX_k$$

经过哑变量处理后，回归模型转化为：

$$Y=a+(b_{11}Z_{11}+b_{12}*Z_{12})+(b_{21}*Z_{21}+B_{22}*Z_{22})+...+(B_{k1}*Z_{k1}+B_{k2}*Z_{k2})$$

经过哑变量处理之后，对于回归分析而言，哑变量处理之后的变量间共线性程度可能加剧，因此采用 PLS（偏最小二乘法）回归或者岭回归方法更加常用一些。

借助于回归分析模型，我们可以预测产品在改进后的最大喜欢程度（当所有测量属性都处于正好状态时的喜欢程度），并分析产品主要属性对产品喜欢程度的贡献。

结束语

总体来看，目前关于 JAR 问题的分析方法在产品测试中仍然是一个热门的话题，一些分析方法（比如采用 Bootstrap 方法[2]）仍在探讨和改进之中。由于所学有限，本文介绍中定有不妥或遗漏之处，尚请同仁不吝指正。

参考文献：

- 1、Jean-Francois Meullenet and Rui Xiong , *Bootstrap Penalty Analysis-Alternatives to distributions and. penalty analysis.*
- 2、R. Xiong and J.-F. Meullenet , *A Novel Dummy Variable Approach to Preference Mapping. of Likings from JAR Data.*

