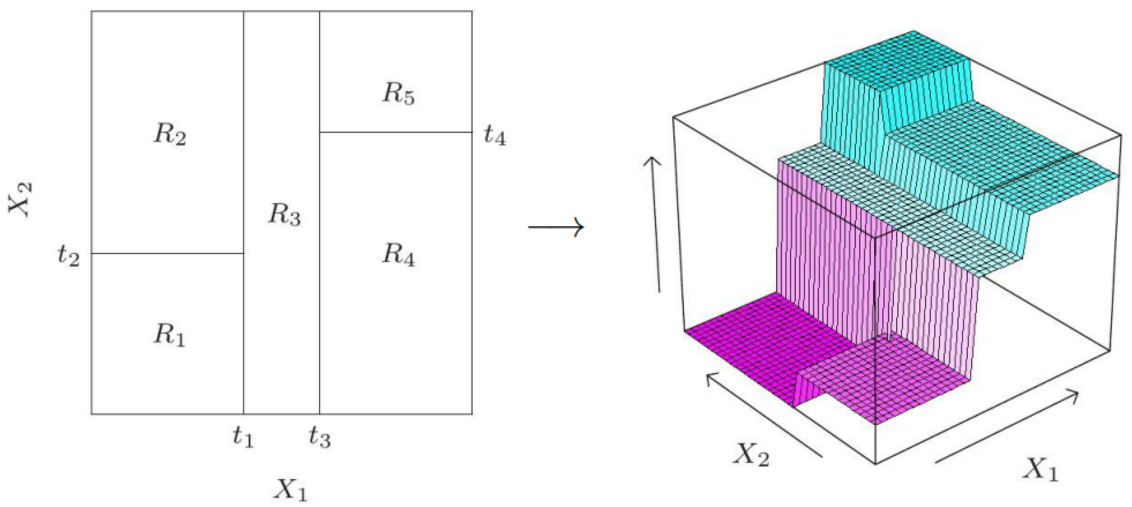
多变量决策树与核树

前言

到了这部分其实已经算完结了，但没有完全完结，因为我们需要做延伸。至于延伸什么，卖个关子（别骂了，我知道这样做很贱（

现在，***让我们将代数问题转化为图像问题，将每个属性“按死在坐标轴上”，于是乎，我们就得到了图像表示***。这种表示方式***不仅很好地表达了离散型数据，还顺带表示了连续型数据***（因为坐标轴本身就是连续的）。

让我们来看看这张图吧（假设是三维的，当然现实中我们的训练集可不止三类，这里方便理解）：



很直观对吧，但是为什么是直直的“平面”？原因就出在我们把属性把属性当坐标轴上，***在划分决策树时，这部分划分的平面叫做“分类平面”，每一段划分都直接对应了某个属性取值，诶！又回到了线性模型，还记得我们的分类问题么？我们希望找到一个合理的边界从而实现分类。***

分类边界的每一段都是与坐标轴平行的，这样的分类边界使得学习结果有较好的可解释性，但在学习任务的真实分类边界比较复杂时，必须使用很多段划分才能获得较好的近似，也就是如图下图所示

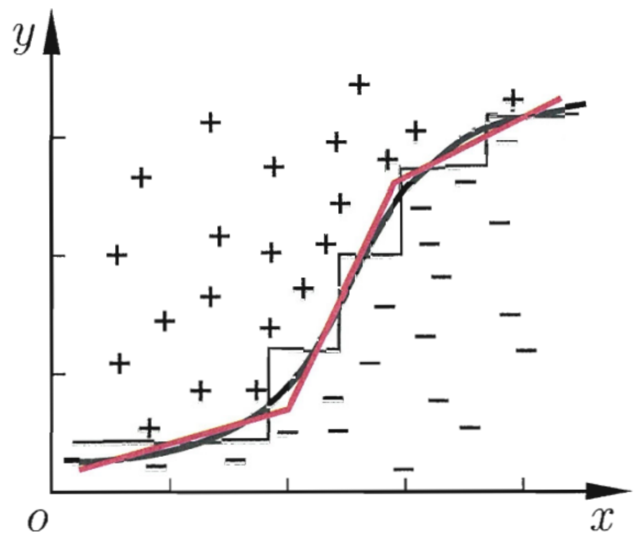
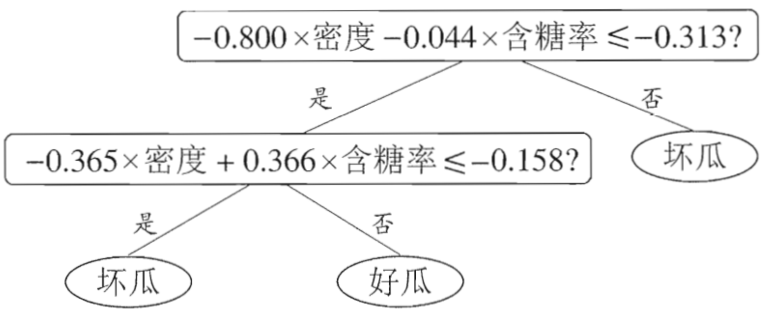


图 4.12 决策树对复杂分类边界的分段近似

此时的决策树会相当复杂，由于要进行大量的属性测试，预测时间开销会很大，怎么办...看到那条红线了么？那就是答案，我们可以使用“斜线”而不是“傻傻的平行于坐标轴的直线”，这条直线就是我们说的多变量决策树的关键！！！

决策树的最终点——多变量决策树

我们怎么表示这条直线呢？动动脑子（use 🧠），***你们不觉得这像是一个子节点由很多个属性同时决定的么？对没错，于是我们就表示出了这个树，比如下图（来自西瓜书）：***



也就是说，我们可以将 “线性模型带入决策树” 使其实现“多变量决策树”，有趣，知识串联了...

但是，代价是……

你可能已经想到：既然可以任意方向划分，那岂不是很爽？遗憾的是，**爽归爽，代价不小**：

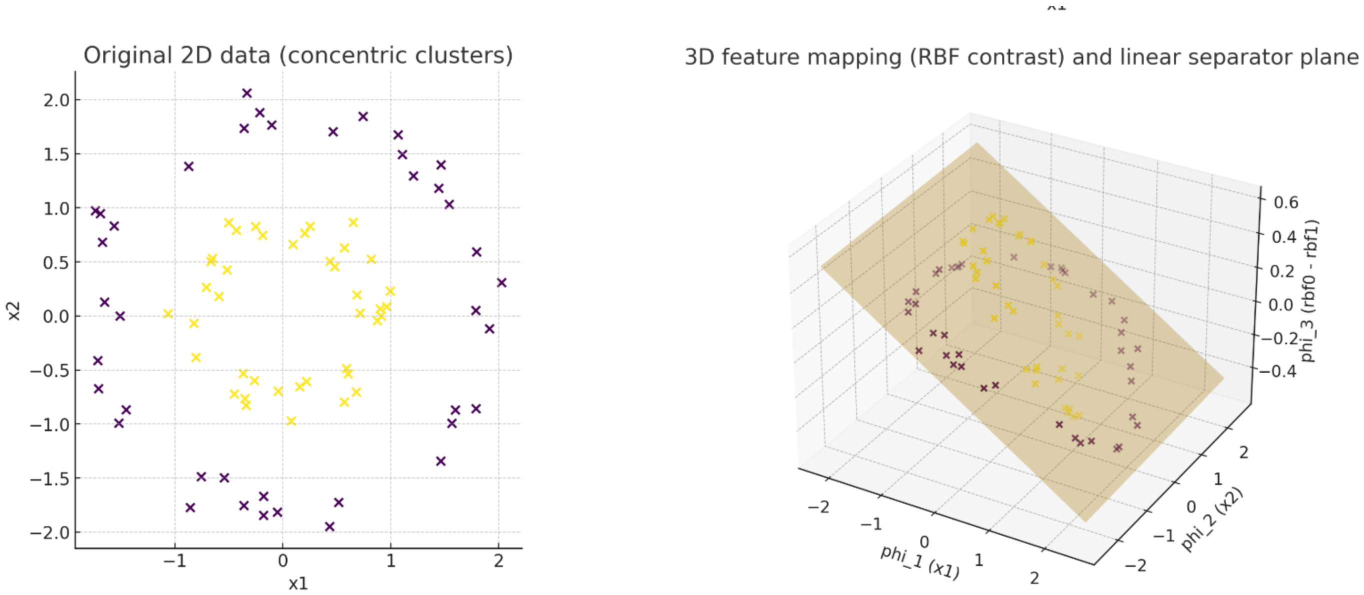
1. **学习更难**：单变量只需要比较所有候选阈值，多变量则需要优化权重向量 $\mathbf{w}$，这相当于在每个节点都要解一个小优化问题。
2. **可解释性下降**：单变量树很直观——“如果年龄>30 则……”。多变量树则变成——“如果$0.7 \times$ 身高 $+ 0.3 \times$ 体重 > 1.2”，这谁看得懂？

3. **计算成本更高**：每次分裂要做矩阵运算，不像单变量那样只排序一次。

所以，多变量决策树更像是研究性质的延伸，而不是工业界常用的“标配”。

核树

不管了，看到“线性的”就感觉头大，为什么不能天然表示非线性的呢？让我们带个核试试看这货到底长什么样？（如果你看到这里还不知道核到底是干嘛的，请观看 局部加权线性回归、逻辑回归、随机梯度与牛顿法 当中的核函数讲解）



可以看到当我们的决策树带上核之后，也拥有了 “画曲线” 决策边界的能力，但是代价呢？好吧，我个人认为这是用处不大的。在工业界，更多人选择 **随机森林 / GBDT + 特征工程** 或 **SVM（核方法）（这些都会在后边的章节进行讲解）**，而不是直接“核化决策树”。因为这些性能更稳定，可解释性与泛化性更好。