xgboost的原理没你想像的那么难

milter (/u/511ba5d71aef) ＋关注
2017.09.02 18:04* 字数 6194 阅读 1878 评论 14 喜欢 46
(/u/511ba5d71aef)

> xgboost 已然火爆机器学习圈，相信不少朋友都使用过。要想彻底掌握xgboost，就必须搞懂其内部的模型原理。这样才能将各个参数对应到模型内部，进而理解参数的含义，根据需要进行调参。本文的目的就是让大家尽可能轻松地理解其内部原理。主要参考文献是陈天奇的这篇文章introduction to xgboost (https://xgboost.readthedocs.io/en/latest/model.html)。在我看来，这篇文章是介绍xgboost**最好的，没有之一**。英语好的同学建议直接看英文，若有不是很理解的地方，再来参考本文。

1、你需要提前掌握的几个知识点

1、监督学习

监督学习就是训练数据有标签的学习。比如说，我有10万条数据，每个数据有100个特征，还有一个标签。标签的内容取决于学习的问题，如果数据是病人进行癌症诊断做的各项检查的结果，标签就是病人是否得癌症。是为1，不是为0.

监督学习就是要从这10万条数据中学习到根据检查结果诊断病人是否得癌症的知识。由于学习的范围限定在这10万条数据中，也就是说，学习的知识必须是从这10万条数据中提炼出来。形象地理解，就是在这10万条带标签数据的“监督”下进行学习。因此称为监督学习。

2、监督学习的成果

监督学习学习到的知识如何表示，又是如何被我们人类使用呢？简单讲，学习到的知识用一个模型来表示，我们人类就用这个模型来使用学习到的知识。

那么，模型是什么东西？
模型就是一个数学表达式。最简单的一个模型就是线性模型，它长这个样子：
y^i=∑jθjxij。用我们上面的例子讲，xi就是我们10万条数据中的第i条，xij就是第i条数据中的第j个检查结果。y^i就是模型对这条数据的预测结果，这个值越大，表明病人得癌症的概率也大。通常，我们还需将y^i处理成0到1的值，以更清晰地表明这是一个概率预测，处理的方法一般是用sigmoid函数，不熟悉的朋友可参考其他资料。θj就是第j个检查结果对病人是否得癌症的“贡献度”，它是我们模型的参数，也就是我们从10条数据中学习到的知识。

可见，所谓监督学习，就是两步，一是定出模型确定参数，二是根据训练数据找出最佳的参数值，所谓最佳，从应用角度看，就是最大程度地吸收了10万条训练数据中的知识，但从我们寻找参数的过程来看，却有另一番解释，下文会详细解释，找到最佳参数后，我们就得出一个参数都是已知的模型，此时，知识就在其中，我们可以自由使用。

3、如何找出最佳参数
以上面的线性模型为例，病人有100个检查结果，那么就有100个参数θj（j从1到100）。每个参数可取值都是实数，100个参数的组合显然有无穷多个，我们怎么评判一组参数是不是最佳的呢？
此时，我们需要另外一个函数来帮助我们来确定参数是否是最佳的，这就是目标函数(object function)。

目标函数如何确定呢？用我们上面的例子来讲，我们要判断病人是否得癌症，假设我们对上面的线性模型的值y^i进行了处理，将它规约到了0和1之间。我们的10万条训练数据中，得癌症的病人标签为1，没得的标签为0.那么显然，最佳的参数一定就是能够将得癌症的病人全预测为1，没得癌症的病人全部预测为0的参数。这几乎就是完美的参数！
因此，我们的目标函数可以设为MSE函数：obj = ∑i(sigmoid(∑jθjxij) - yi)^2

上面的函数的意思就是对第i条数据，将模型预测的值规约到0和1，然后与该条数据的真是标签值（0和1）做差，再求平方。这个平方值越大，表明预测的越不准，就是模型的预测误差，最后，我们将模型对10万条数据的预测误差求和。就得出了一组具体的参数的预测好坏的度量值。

果真这样就完美了吗？
不是的。上面的目标函数仅仅评测了参数对训练数据来说的好坏，并没有评测我们使用模型做预测时，这组参数表现好坏。也就是说，对训练数据来说是好的参数，未必在预测时就是好的。为什么？
一是10万条数据中有错误存在
二是10万条数据未必涵盖了所有种类的样本，举个极端的例子，假如10万条数据全是60岁以上老人的检查结果，我们用学习到的模型取预测一个10岁的小孩，很可能是不准的。

那么，怎么评测一组参数是否对预测是好的呢？
答案是测了才知道！

这不是废话吗。

事实就是这样。真实的预测是最权威的评判。但我们还是可以有所作为的，那就是正则化。

所谓正则化就是对参数施加一定的控制，防止参数走向极端。以上面的例子来说，假如10万条数据中，得癌症的病人都是60岁以上老人，没得癌症的病人都是30岁以下年轻人，检查结果中有一项是骨质密度，通常，老人骨质密度低，年轻人骨质密度高。那么我们学习到的模型很可能是这样的，对骨质密度这项对应的参数θj设的非常大，其他的参数都非常小，简单讲，模型倾向于就用这一项检查结果去判断病人是否得癌症，因为这样会让目标函数最小。

明眼人一看便知，这样的参数做预测肯定是不好的。

正则化可以帮助我们规避这样的问题。

常用的正则化就是L2正则，也就是所有参数的平方和。我们希望这个和尽可能小的同时，模型对训练数据有尽可能好的预测。

最后，我们将L2正则项加到最初的目标函数上，就得出了最终的目标函数：
obj = ∑_i(sigmoid(∑_jθjxij) - yi)^2 + ∑j(θj^2)

能使这个函数值最小的那组参数就是我们要找的最佳参数。这个obj包含的两项分别称为**损失函数和正则项**。
这里的正则项，本质上是用来控制模型的复杂度。

Notes:
上面，我们为了尽可能简单地说明问题，有意忽略了一些重要的方面。比如，我们的例子是分类，但使用的损失函数却是MSE，通常是不这样用的。
对于回归问题，我们常用的损失函数是MSE，即：

$$L(\theta) = \sum_i (y_i - \hat{y}_i)^2$$

回归.PNG

对于分类问题，我们常用的损失函数是对数损失函数：

$$L(\theta) = \sum_i [y_i \ln(1 + e^{-\hat{y}_i}) + (1 - y_i) \ln(1 + e^{\hat{y}_i})]$$

分类.PNG

乍一看，这个损失函数怪怪的，我们不免要问，为什么这个函数就是能评判一组参数对训练数据的好坏呢？

我们用上面的例子来说明，假如有一条样本，它的标签是1，也就是yi = 1，那么关于这条样本的损失函数中就只剩下了左边那一部分，由于yi = 1，最终的形式就是这样的：

$$\ln(1 + e^{-\hat{y}_i})$$

对数1.PNG

头上带一个小尖帽的yi就是我们模型的预测值，显然这个值越大，则上面的函数越倾向于0，yi趋向于无穷大时，损失值为0。这符合我们的要求。

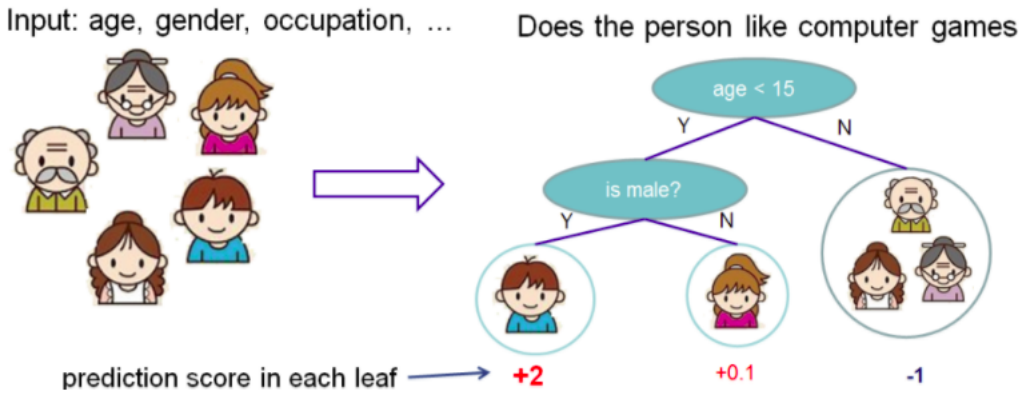
同理，对于yi=0的样本也可以做出类似的分析。

至于这个损失函数是怎么推导出来的，有两个办法，一个是用LR，一个是用最大熵。具体的推导过程请参阅其他资料。

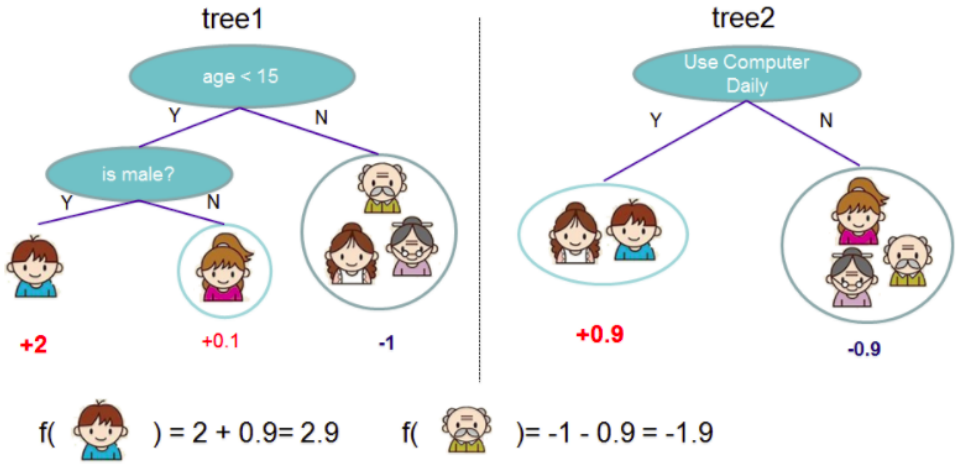
2、xgboost

既然xgboost就是一个监督模型，那么我们的第一个问题就是：xgboost对应的模型是什么？
答案就是一堆CART树。
此时，可能我们又有疑问了，CART树是什么？这个问题请查阅其他资料，我的博客中也有相关文章涉及过。然后，一堆树如何做预测呢？答案非常简单，就是将每棵树的预测值加到一起作为最终的预测值，可谓简单粗暴。

下图就是CART树和一堆CART树的示例，用来判断一个人是否会喜欢计算机游戏：



predict1.PNG



predict2.PNG

第二图的底部说明了如何用一堆CART树做预测，就是简单将各个树的预测分数相加。

xgboost为什么使用CART树而不是用普通的决策树呢？
简单讲，对于分类问题，由于CART树的叶子节点对应的值是一个实际的分数，而非一个确定的类别，这将有利于实现高效的优化算法。xgboost出名的原因一是准，二是快，之所以快，其中就有选用CART树的一份功劳。

知道了xgboost的模型，我们需要用数学来准确地表示这个模型，如下所示：

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in \mathcal{F}$$

predict3.PNG

这里的K就是树的棵数，F表示所有可能的CART树，f表示一棵具体的CART树。这个模型由K棵CART树组成。模型表示出来后，我们自然而然就想问，这个模型的参数是什么？因为我们知道，“知识”蕴含在参数之中。第二，用来优化这些参数的目标函数又是什么？

我们先来看第二个问题，模型的目标函数，如下所示：

$$\text{obj}(\theta) = \sum_{i}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

predict4.PNG

这个目标函数同样包含两部分，第一部分就是损失函数，第二部分就是正则项，这里的正则化项由K棵树的正则化项相加而来，你可能会好奇，一棵树的正则化项是什么？可暂时保持住你的好奇心，后面会有答案。现在看来，它们都还比较抽象，不要着急，后面会逐一将它们具体化。

3、训练xgboost

上面，我们获取了xgboost模型和它的目标函数，那么训练的任务就是通过最小化目标函数来找到最佳的参数组。

问题是参数在哪里？

我们很自然地想到，xgboost模型由CART树组成，参数自然存在于每棵CART树之中。
那么，就单一的 CART树而言，它的参数是什么呢？
根据上面对CART树的介绍，我们知道，确定一棵CART树需要确定两部分，第一部分就是树的结构，这个结构负责将一个样本映射到一个确定的叶子节点上，其本质上就是一个函数。第二部分就是各个叶子节点上的分数。

似乎遇到麻烦了，你要说叶子节点的分数作为参数，还是没问题的，但树的结构如何作为参数呢？而且我们还不是一棵树，而是K棵树!

让我们想像一下，如果K棵树的结构都已经确定，那么整个模型剩下的就是所有K棵树的叶子节点的值，模型的正则化项也可以设为各个叶子节点的值的平方和。此时，整个目标函数其实就是一个K棵树的所有叶子节点的值的函数，我们就可以使用梯度下降或者随机梯度下降来优化目标函数。现在这个办法不灵了，必须另外寻找办法。

4、加法训练

所谓加法训练，本质上是一个元算法，适用于所有的加法模型，它是一种启发式算法。关于这个算法，我的另一篇讲GBDT的文章中有详细的介绍，这里不再重复，不熟悉的朋友，可以看一下。运用加法训练，我们的目标不再是直接优化整个目标函数，这已经被我们证明是行不通的。而是分步骤优化目标函数，首先优化第一棵树，完了之后再优化第二棵树，直至优化完K棵树。整个过程如下图所示：

$$
\begin{aligned}
\hat{y}_i^{(0)} &= 0 \\
\hat{y}_i^{(1)} &= f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i) \\
\hat{y}_i^{(2)} &= f_1(x_i) + f_2(x_i) = \hat{y}_i^{(1)} + f_2(x_i) \\
&\cdots \\
\hat{y}_i^{(t)} &= \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i)
\end{aligned}
$$

predict6.PNG

在第t步时，我们添加了一棵最优的CART树ft，这棵最优的CART树ft是怎么得来的呢？非常简单，就是在现有的t-1棵树的基础上，使得目标函数最小的那棵CART树，如下图所示：

$$
\begin{aligned}
\text{obj}^{(t)} &= \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^{t} \Omega(f_i) \\
&= \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) + constant
\end{aligned}
$$

10.PNG

假如我们使用的损失函数时MSE，那么上述表达式会变成这个样子：

$$\begin{aligned}\text{obj}^{(t)} &= \sum_{i=1}^{n}(y_i - (\hat{y}_i^{(t-1)} + f_t(x_i)))^2 + \sum_{i=1}^{t}\Omega(f_i) \\ &= \sum_{i=1}^{n}[2(\hat{y}_i^{(t-1)} - y_i)f_t(x_i) + f_t(x_i)^2] + \Omega(f_t) + constant\end{aligned}$$

11.PNG

这个式子非常漂亮，因为它含有ft(xi)的一次式和二次式，而且一次式项的系数是残差。你可能好奇，为什么有一次式和二次式就漂亮，因为它会对我们后续的优化提供很多方便，继续前进你就明白了。

注意：ft(xi)是什么？它其实就是ft的某个叶子节点的值。之前我们提到过，叶子节点的值是可以作为模型的参数的。

但是对于其他的损失函数，我们未必能得出如此漂亮的式子，所以，对于一般的损失函数，我们需要将其作泰勒二阶展开，如下所示：

$$\text{obj}^{(t)} = \sum_{i=1}^{n}[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2}h_i f_t^2(x_i)] + \Omega(f_t) + constant$$

12.PNG

其中：

$$\begin{aligned}g_i &= \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)}) \\ h_i &= \partial^2_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})\end{aligned}$$

13.PNG

这里有必要再明确一下，gi和hi的含义。gi怎么理解呢？现有t-1棵树是不是？这t-1棵树组成的模型对第i个训练样本有一个预测值y^i是不是？这个y^i与第i个样本的真实标签yi肯定有差距是不是？这个差距可以用l(yi,y^i)这个损失函数来衡量是不是？现在gi和hi的含义你已经清楚了是不是？

来，答一个小问题，在优化第t棵树时，有多少个gi和hi要计算？嗯，没错就是各有N个，N是训练样本的数量。如果有10万样本，在优化第t棵树时，就需要计算出个10万个gi和hi。感觉好像很麻烦是不是？但是你再想一想，这10万个gi之间是不是没有啥关系？是不是可以并行计算呢？聪明的你想必再一次感受到了，为什么xgboost会辣么快！

好，现在我们来审视下这个式子，哪些是常量，哪些是变量。式子最后有一个constant项，聪明如你，肯定猜到了，它就是前t-1棵树的正则化项。l(yi, yi^t-1)也是常数项。剩下的三个变量项分别是第t棵CART树的一次式，二次式，和整棵树的正则化项。再次提醒，这里所谓的树的一次式，二次式，其实都是某个叶子节点的值的一次式，二次式。

我们的目标是让这个目标函数最小化，常数项显然没有什么用，我们把它们去掉，就变成了下面这样：

$$\sum_{i=1}^{n}[g_i f_t(x_i) + \frac{1}{2}h_i f_t^2(x_i)] + \Omega(f_t)$$

14.PNG

好，现在我们可以回答之前的一个问题了，为什么一次式和二次式显得那么漂亮。因为这些一次式和二次式的系数是gi和hi，而gi和hi可以并行地求出来。而且，gi和hi是不依赖于损失函数的形式的，只要这个损失函数二次可微就可以了。这有什么好处呢？好处就是xgboost可以支持自定义损失函数，只需满足二次可微即可。强大了我的哥是不是？

5、模型正则化项

上面的式子已然很漂亮，但是，后面的Ω(ft)仍然是云遮雾罩，不清不楚。现在我们就来定义如何衡量一棵树的正则化项。这个事儿并没有一个客观的标准，可以见仁见智。为此，我们先对CART树作另一番定义，如下所示：

$$f_t(x) = w_{q(x)}, w \in R^T, q : R^d \to \{1, 2, \cdots, T\}.$$

16.PNG

需要解释下这个定义，首先，一棵树有T个叶子节点，这T个叶子节点的值组成了一个T维向量w，q(x)是一个映射，用来将样本映射成1到T的某个值，也就是把它分到某个叶子节点，q(x)其实就代表了CART树的结构。w_q(x)自然就是这棵树对样本x的预测值了。

有了这个定义，xgboost就使用了如下的正则化项：

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

17.PNG

注意：这里出现了γ和λ，这是xgboost自己定义的，在使用xgboost时，你可以设定它们的值，显然，γ越大，表示越希望获得结构简单的树，因为此时对较多叶子节点的树的惩罚越大。λ越大也是越希望获得结构简单的树。

为什么xgboost要选择这样的正则化项？很简单，好使！效果好才是真的好。

6、见证奇迹的时刻

至此，我们关于第t棵树的优化目标已然很清晰，下面我们对它做如下变形，请睁大双眼，集中精力：

$$Obj^{(t)} \approx \sum_{i=1}^{n}[g_i w_{q(x_i)} + \frac{1}{2}h_i w^2_{q(x_i)}] + \gamma T + \frac{1}{2}\lambda\sum_{j=1}^{T} w_j^2$$
$$= \sum_{j=1}^{T}[(\sum_{i\in I_j} g_i)w_j + \frac{1}{2}(\sum_{i\in I_j} h_i + \lambda)w_j^2] + \gamma T$$

18.PNG

这里需要停一停，认真体会下。Ij代表什么？它代表一个集合，集合中每个值代表一个训练样本的序号，整个集合就是被第t棵CART树分到了第j个叶子节点上的训练样本。理解了这一点，再看这步转换，其实就是内外求和顺序的改变。如果感觉还有困难，欢迎评论留言。

进一步，我们可以做如下简化：

We could further compress the expression by defining $G_j = \sum_{i\in I_j} g_i$ and $H_j = \sum_{i\in I_j} h_i$:

$$\text{obj}^{(t)} = \sum_{j=1}^{T}[G_j w_j + \frac{1}{2}(H_j + \lambda)w_j^2] + \gamma T$$

19.PNG

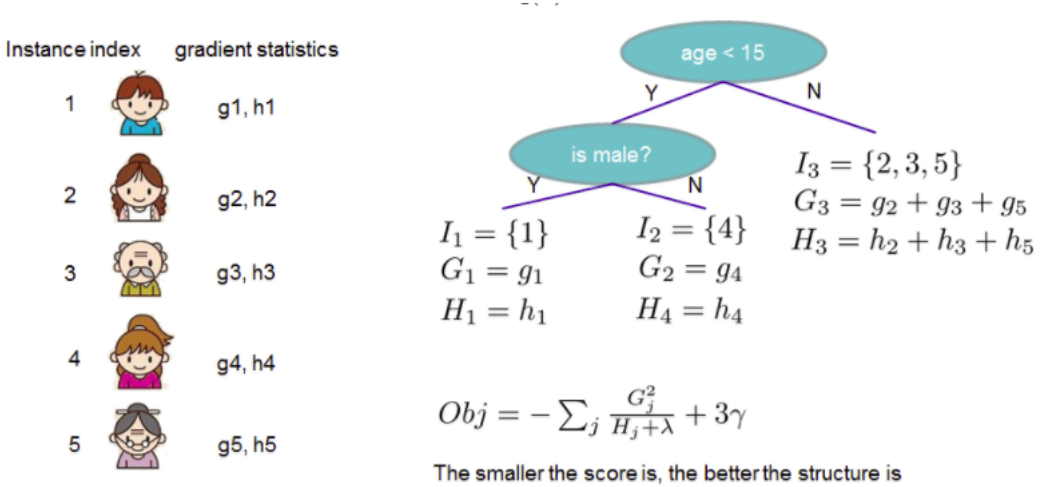
其中的Gj和Hj应当是不言自明了。

对于第t棵CART树的某一个确定的结构（可用q(x)表示），所有的Gj和Hj都是确定的。而且上式中各个叶子节点的值wj之间是互相独立的。上式其实就是一个简单的二次式，我们很容易求出各个叶子节点的最佳值以及此时目标函数的值。如下所示：

$$w_j^* = -\frac{G_j}{H_j + \lambda}$$
$$\text{obj}^* = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j + \lambda} + \gamma T$$

20.PNG

obj*代表了什么呢？
它表示了这棵树的结构有多好，值越小，代表这样结构越好！也就是说，它是衡量第t棵CART树的结构好坏的标准。注意~注意~注意~，这个值仅仅是用来衡量结构的好坏的，与叶子节点的值可是无关的。为什么？请再仔细看一下obj*的推导过程。obj*只和Gj和Hj和T有关，而它们又只和树的结构(q(x))有关，与叶子节点的值可是半毛关系没有。如下图所示：



23.PNG

7、找出最优的树结构

好了，有了评判树的结构好坏的标准，我们就可以先求最佳的树结构，这个定出来后，最佳的叶子结点的值实际上在上面已经求出来了。

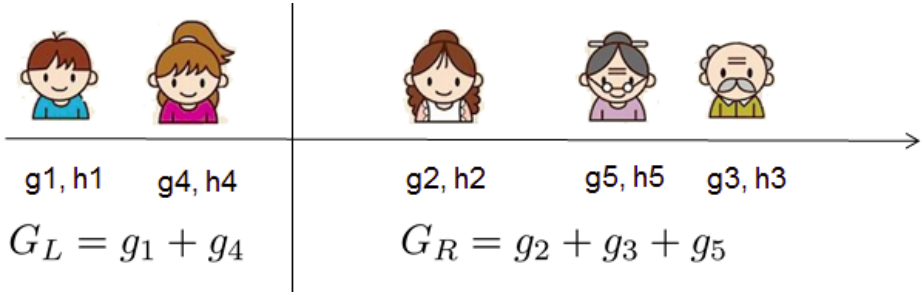
问题是：树的结构近乎无限多，一个一个去测算它们的好坏程度，然后再取最好的显然是不现实的。所以，我们仍然需要采取一点策略，这就是逐步学习出最佳的树结构。这与我们将K棵树的模型分解成一棵一棵树来学习是一个道理，只不过从一棵一棵树变成了一层一层节点而已。如果此时你还是有点蒙，没关系，下面我们就来看一下具体的学习过程。
我们以上文提到过的判断一个人是否喜欢计算机游戏为例子。最简单的树结构就是一个节点的树。我们可以算出这棵单节点的树的好坏程度obj*。假设我们现在想按照年龄将这

棵单节点树进行分叉，我们需要知道：

1、按照年龄分是否有效，也就是是否减少了obj的值

2、如果可分，那么以哪个年龄值来分。

为了回答上面两个问题，我们可以将这一家五口人按照年龄做个排序。如下图所示：



29.PNG

按照这个图从左至右扫描，我们就可以找出所有的切分点。对每一个确定的切分点，我们衡量切分好坏的标准如下：

$$Gain = \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma$$

27.PNG

这个Gain实际上就是单节点的obj*减去切分后的两个节点的树obj*，Gain如果是正的，并且值越大，表示切分后obj*越小于单节点的obj*，就越值得切分。同时，我们还可以观察到，Gain的左半部分如果小于右侧的γ，则Gain就是负的，表明切分后obj反而变大了。γ在这里实际上是一个临界值，它的值越大，表示我们对切分后obj下降幅度要求越严。这个值也是可以在xgboost中设定的。

扫描结束后，我们就可以确定是否切分，如果切分，对切分出来的两个节点，递归地调用这个切分过程，我们就能获得一个相对较好的树结构。

注意：xgboost的切分操作和普通的决策树切分过程是不一样的。普通的决策树在切分的时候并不考虑树的复杂度，而依赖后续的剪枝操作来控制。xgboost在切分的时候就已经考虑了树的复杂度，就是那个γ参数。所以，它不需要进行单独的剪枝操作。

8、大功告成

最优的树结构找到后，确定最优的叶子节点就很容易了。我们成功地找出了第t棵树！撒花！！！

机器学习与计算机视觉 (/c/ee1275bb82ca?utm_source=desktop&utm_medium=notes-included-collection)

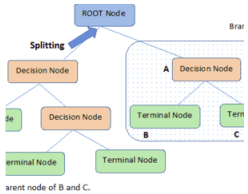
程序员 (/c/NEt52a?utm_source=desktop&utm_medium=notes-included-collection)

首页投稿 (/c/bDHhpK?utm_source=desktop&utm_medium=notes-included-collection)

程序员首页投稿 (/c/89995286335f?utm_source=desktop&utm_medium=notes-included-collection)

展开更多

(/p/ff9b7b031fed?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

关于基于树的建模的完整教程（从R＆Python）(/p/ff9b7b031fed?utm_ca…

翻译自analyticsvidhya 基于树的学习算法被认为是最好的和最常用的监督学习(supervised learning)方法之一。基于树的方法赋予预测模型高精度，稳定性和易于解释的能力。 与线性模型不同，它们非常好地映射…

珞珈村下山 (/u/cc62f0e70e83?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

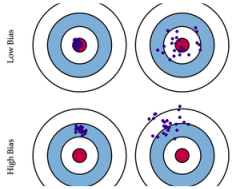
机器学习算法小结与收割offer遇到的问题 (/p/ace5051d0023?utm_campai…

机器学习是做NLP和计算机视觉这类应用算法的基础，虽然现在深度学习模型大行其道，但是懂一些传统算法的原理和它们之间的区别还是很有必要的。可以帮助我们做一些模型选择。本篇博文就总结一下各种机器…

在河之简 (/u/5ff1acaa6334?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/3e9e1c5c22e3?utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

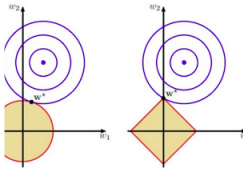
GBDT总结笔记 (/p/3e9e1c5c22e3?utm_campaign=m…



[TOC] Supervised learning 监督学习的要点:数据:对于输入数据 $$$x_i \in R^d$$$,训练数据里的第i个样本。模型:如何对于给定的 $$$x_i$$$预测…

跑得比谁都慢 (/u/b317a4550a9b?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/0d70bf2510b7?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

面试 (/p/0d70bf2510b7?utm_campaign=maleskine&utm_content=note&…

ML & DM 集成学习 模型融合 ensemble http://wakemeup.space/?p=109 EM EM算法的目标是找出有隐性变量的概率模型的最大可能性解，它分为两个过程E-step和M-step，E-step通过最初假设或上一步得出的模型…

章鱼哥呀 (/u/19777d5480c1?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

集成树模型（Ensemble）(/p/a69194bdab9e?utm_campaign=maleskine…

博客园：梯度提升树(GBDT)原理小结博客园：一步一步理解GB、GBDT、xgboost知乎：机器学习算法中GBDT和XGBOOST的区别有哪些？ 介绍下rf，adaboost，gbdt，xgboost的算法原理？（注意adaboost，…

闫阿佳 (/u/c5b5b010d1b3?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

你在害怕什么 (/p/f45061089f37?utm_campaign=maleskine&utm_conten…

我这样问我自己。 近来老是想起在大学的时光，实际上并没有过去多久，不过才两三个月。 可是毕业就像是一条鸿沟，把我的前二十二年生生地隔开，从此以后，就没有退路了，以前各种借口逃避的事情，工作啊，…

戚风儿 (/u/1ae79b343fe4?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

《慢思考》读后感 (/p/58acd046ab22?utm_campaign=maleskine&utm_c…

我从未像现在这样渴望提升自己，看各种类型的公众号，学英语，参加培训班，把每天的时间安排的没有一丝空隙。是的，我给自己营造了一个忙到没有时间去思考的假象。直到突然有一天，想回头望望自己走过的…

位一一 (/u/7e318ae0d6df?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/55cd9a223a43?

utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

第一封情书 (/p/55cd9a223a43?utm_campaign=maleskine&utm_content…

林: 我今天和朋友说起自贡灯会，发现最想一起去看的人是你。 想让你和我看到的是同一片风景。 所以忽然觉得好孤独。 这样的风景， 所有的风景， 我都只能一个人看。

戀物念一樣 (/u/6fdb4f67b0ea?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

恶作剧之吻 (/p/ab85be706aca?utm_campaign=maleskine&utm_content…

不知道为什么 已经24岁的我最近疯狂迷恋恶作剧之吻 看着看着眼泪狂飙 无法理解自己

Masuier (/u/32556f8f5f5e?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

木风恒简慢素生活纪：第二五九天纪录 (/p/0870efefd7b1?utm_campaign=…

实话生活 体悟人生 今天是2017年1月11日 天气阴雨 温度10-14度 今天早上五点多起床 起来了以后 先按柔腹部九下 然后就去上厕所 在上厕所的过程中 用十指梳头 以及按柔耳朵 上完以后 接着就手指沾食盐刷牙和洗…

木风恒 (/u/8b81c65944e2?utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)