1. 初始化权重：初始时，每个样本的权重都是相等的，即 $w_i = \frac{1}{N}$，其中 N 是样本数量，每个样本的初始权重是 0.1。
2. 对于每一轮：
   - 选择阈值 v 最小化加权分类误差。
   - 计算加权误差率 $\epsilon = \sum_{\text{错误}_i} w_i$ 。
   - 计算弱分类器的权重$\alpha = \frac{1}{2}\ln(\frac{(1-\epsilon)}{\epsilon})$。
   - 更新样本权重：对于正确分类的样本，$w_i \leftarrow w_i \times e^{-a}$ ，对于错误分类的样本，$w_i \leftarrow w_i \times e^{a}$。然后归一化所有样本权重使它们总和为 1。(这一步 PPT 上没有，可以实施也可以不实施)
3. 组合弱分类器：最终的强分类器是所有弱分类器的加权组合，即 $\text{sign}(\sum_t a_t h_t(x))$，其中 $h_t(x)$是第 t 轮的弱分类器。

以第一轮为例：
在第一轮中，最佳阈值 v 是 2，方向是 "greater"（即 x> 2 ），所有 x 值大于 2 的被分类为 -1，否则为 1。弱分类器的加权误差率是 0.3。
弱分类器的权重$\alpha = 0.424$，然后更新训练样本的权重。这将为下一轮的弱分类器训练提供基础。

前五轮迭代：

$$\alpha \approx 0.424, v = 2, " > "$$
$$\alpha \approx 0.650, v = 8, " > "$$
$$\alpha \approx 0.752, v = 6, " < "$$
$$\alpha \approx 0.711, v = 2, " > "$$
$$\alpha \approx 0.726, v = 8, " > "$$

这些弱分类器组合成一个强分类器，强分类器的决策基于所有弱分类器的加权投票结果。使用这些弱分类器来对新数据进行分类以检验强分类器的性能。

**10. 试析随机森林为何比决策树 Bagging 集成的训练速度更快。**
答：决策树的生成过程中，最耗时的就是搜寻最优切分属性；随机森林在决策树训练过程中引入了随机属性选择，大大减少了此过程的计算量；因而随机森林比普通决策树 Bagging 训练速度要快。

**11. Gradient Boosting［Friedman，2001］是一种常用的 Boosting 算法，试析其与 AdaBoost 的异同。**
答：Gradient Boosting 和其它 Boosting 算法一样，通过将表现一般的数个模型（通常是深度固定的决策树）组合在一起来集成一个表现较好的模型。抽象地说，模型的训练过程是对一任意可导目标函数的优化过程。通过反复地选择一个指向负梯度方向的函数，该算法可被看做在函数空间里对目标函数进行优化。因此可以说 Gradient Boosting = Gradient Descent + Boosting。和 AdaBoost 一样，Gradient Boosting 也是重复选择一个表现一般的模型并且每次基于先前模型的表现进行调整。不同的是，AdaBoost 是通过提升错分数据点的权重来定位模型的不足而 Gradient Boosting 是通过算梯度（gradient）来定位模型的不足。因此相比 AdaBoost, Gradient Boosting 可以使用更多种类的目标函数。