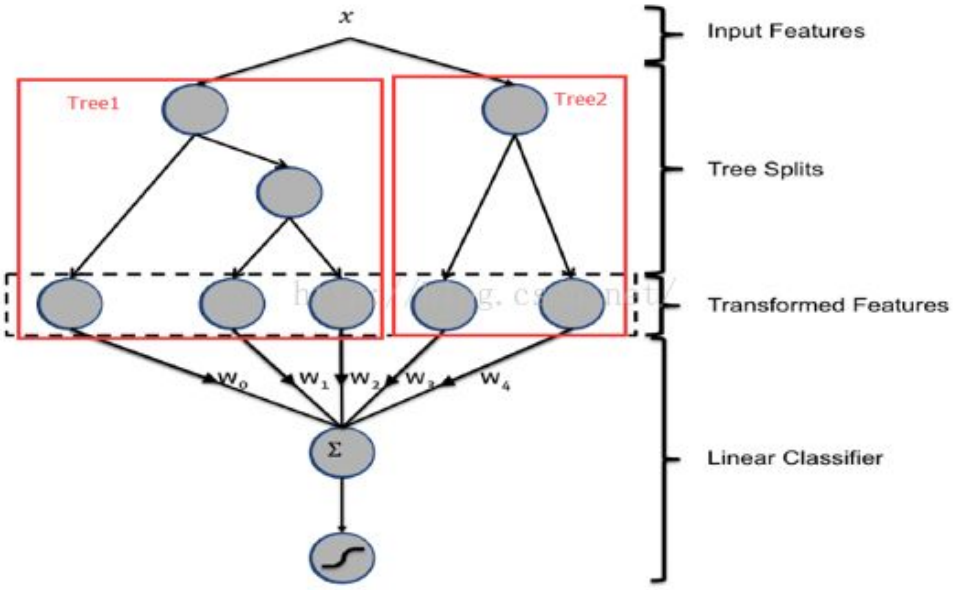
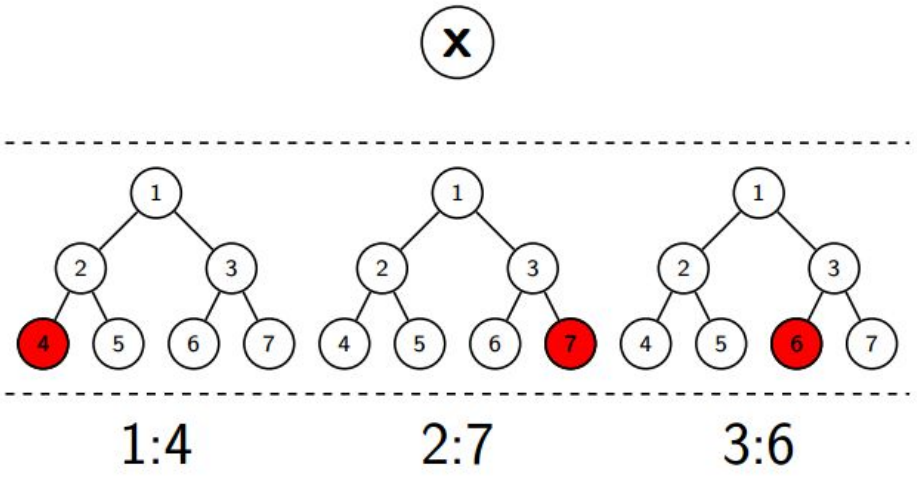
Feature transformations with ensembles of trees

GBDT/XGBoost/RF/RT+LR 融合

1.主要思想

GBDT 与 LR 的融合方式，Facebook 的 paper（见附件链接）实例如第一张图所示，图中 Tree1、Tree2 为通过 GBDT 模型学出来的两颗树，x 为一条输入样本，遍历两棵树后，x 样本分别落到两颗树的叶子节点上，每个叶子节点对应 LR 一维特征，那么通过遍历树，就得到了该样本对应的所有 LR 特征。由于树的每条路径，是通过最小化均方差等方法最终分割出来的有区分性路径，根据该路径得到的特征、特征组合都相对有区分性，效果理论上不会亚于人工经验的处理方式。



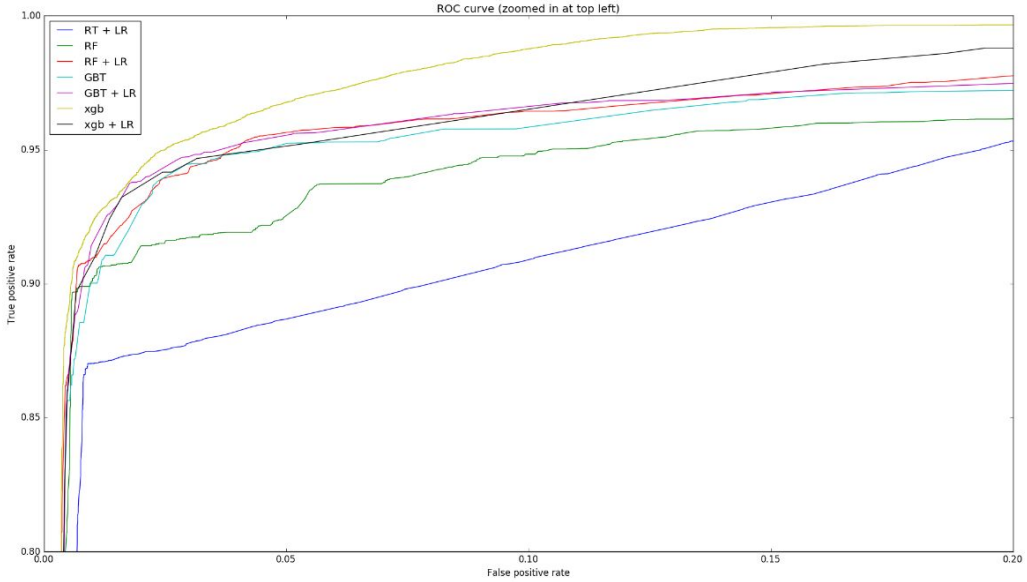


思想很简单，就是先用已有特征训练 GBDT 模型，然后利用 GBDT 模型学习到的树来构造新特征，最后把这些新特征加入原有特征一起训练模型。构造的新特征

向量是取值 0/1 的，向量的每个元素对应于 GBDT 模型中树的叶子结点。当一个样本点通过某棵树最终落在这棵树的一个叶子结点上，那么在新特征向量中这个叶子结点对应的元素值为 1，而这棵树的其他叶子结点对应的元素值为 0。新特征向量的长度等于 GBDT 模型里所有树包含的叶子结点数之和。
举例说明：上面第一张图中的两棵树是 GBDT 学习到的，第一棵树有 3 个叶子结点，而第二棵树有 2 个叶子节点。对于一个输入样本点 x，如果它在第一棵树最后落在其中的第二个叶子结点，而在第二棵树里最后落在其中的第一个叶子结点。那么通过 GBDT 获得的新特征向量为[0, 1, 0, 1, 0]，其中向量中的前三位对应第一棵树的 3 个叶子结点，后两位对应第二棵树的 2 个叶子结点。
那么，GBDT 中需要多少棵树能达到效果最好呢？具体数字依赖于应用以及数据量。一般数据量较少时，树太多会导致过拟合。在作者的应用中，大概 500 棵左右效果就基本不改进了。另外，作者在建 GBDT 时也会对每棵树的叶子结点数做约束——不多于 12 个叶子结点。

XGBoost 是 gbdt 的扩展，同样可以实现上述过程，原理是一样的。而且 xgboost 相比 gbdt 有更高的效率和精度，结果也会更好。
RF 是 bagging，并行化和泛化能力比较强，也可以实现上述融合。
RT 是完全随机树或成 Extra Tree；和 RF 很像，但有一些的不同，随机性更大，会降低些方差，但会增大偏差。

对于 xgboost 输出结果 index 是一个矩阵，第二层算法如果是 Logistic Regression 就 One-Hot Encoding，如果后面是 LibFFM，就直接用 index，这样 Variance 应该还会小一些。
下图是样品数量为 80000，根据 make_classification 函数随机生成的 n 类数据集对 xgb/gbdt/rf/rt+lr 进行实验的对比结果。



Facebook 论文地址：
https://pdfs.semanticscholar.org/daf9/ed5dc6c6bad5367d7fd8561527da30e9b8dd.pdf