

Practice of AI

从决策树到XGBoost

Jim Xie

2020/3/6

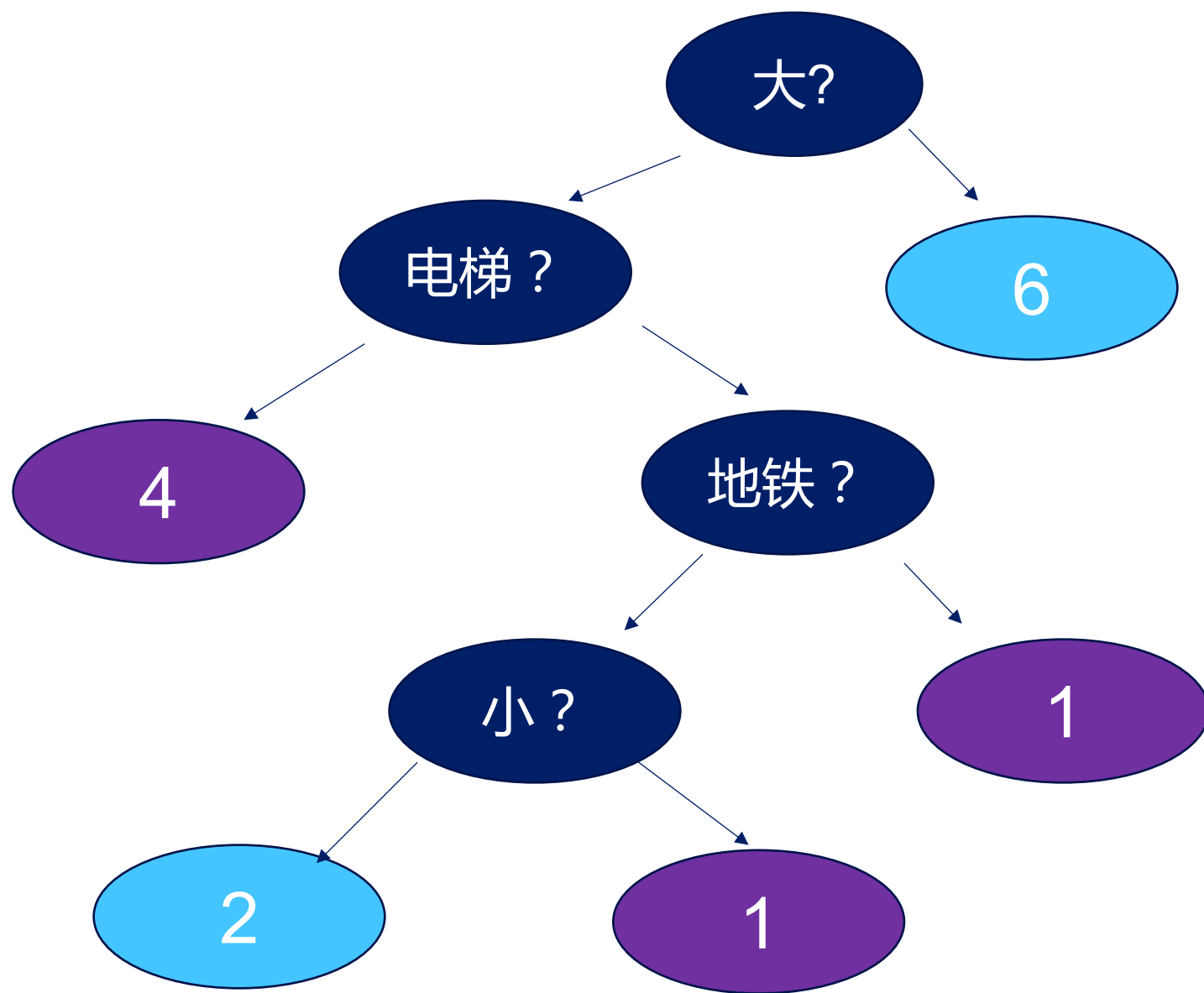
决策树



根据历史的买房的数据，
预测房子会卖给哪一类人：

- A：医生
- B：教师

样本	近地铁	有电梯	房子大小	候选人A或B
1	是	否	中	B
2	否	是	小	B
3	否	是	大	A
4	否	是	中	A
5	否	否	大	A
6	否	是	中	A
7	否	是	大	A
8	否	否	小	B
9	否	是	大	A
10	是	否	小	B
11	是	否	中	B
12	否	否	大	A
13	是	是	大	A
14	是	是	中	B

决策树



A 
B 



- 如何分裂?
- 如何停止分裂?

停止分裂

- **完全分类**
- **max_depth**
 - 最大深度（推荐5-20之间）
- **min_samples_split**
 - 结点下样本最少样本
- **max_leaf_nodes**
 - 最大叶子节数
- **min_impurity_split**
 - 最小节点的不纯度（如：信息增益）

熵

举例

- 猜硬币，下面两种情况，哪种更好猜？



正反面出现概率都是0.5

$$H(X) = -(0.5 \log_2 0.5 + 0.5 \log_2 0.5) = 1$$



正面出现概率0.2，反面出现概率0.8

$$H(X) = -(0.2 \log_2 0.2 + 0.8 \log_2 0.8) = 0.729$$

$$H(X) = -\sum_{i=1}^n P(x_i) \log_b P(x_i)$$

熵变小了，不确定性变小了，更好猜了

ID3

以房子大为优先分裂点，因为带来的熵减（信息增益）为0.5216最多

特征点	父节点熵	左子节点熵	右子节点熵	左右子节点的加权平均熵	信息增益
地铁？	0.9852	0.7642	0.7219	$0.7490 * 9/14 + 0.7219 * 5/14 = 0.7491$	0.2361
电梯？	0.9852	0.9183	0.8113	$0.9183 * 6/14 + 0.8113 * 8/14 = 0.8571$	0.128
大	0.9852	0.8113	0	$0.8113 * 8 /14 + 0.0 * 6/14 = 0.4636$	0.5216
小	0.9852	0.8454	0	$0.8454 * 11/14 + 0.0 * 3/14 = 0.6642$	0.321
中	0.9852	0.9183	0.971	$0.9183 * 9/14 + 0.9710 * 5/14 = 0.9371$	0.0481

ID3

继续分裂，指定完全分类或达到设定条件（如树深度）

样本	近地铁	有电梯	房子大小	候选A或B
1	是	否	中	B
2	否	是	小	B
4	否	是	中	A
6	否	是	中	A
8	否	否	小	B
10	是	否	小	B
11	是	否	中	B
14	是	是	中	B

左子树
用同样的方法进行划分

样本	近地铁	有电梯	房子大小	候选A或B
3	否	是	大	A
5	否	否	大	A
7	否	是	大	A
9	否	是	大	A
12	否	否	大	A
13	是	是	大	A

右子树
全都是A不需要再划分

决策树扩展

□ ID4.5

把信息增益换成信息增益率

□ CART

把信息增益换成基尼系数

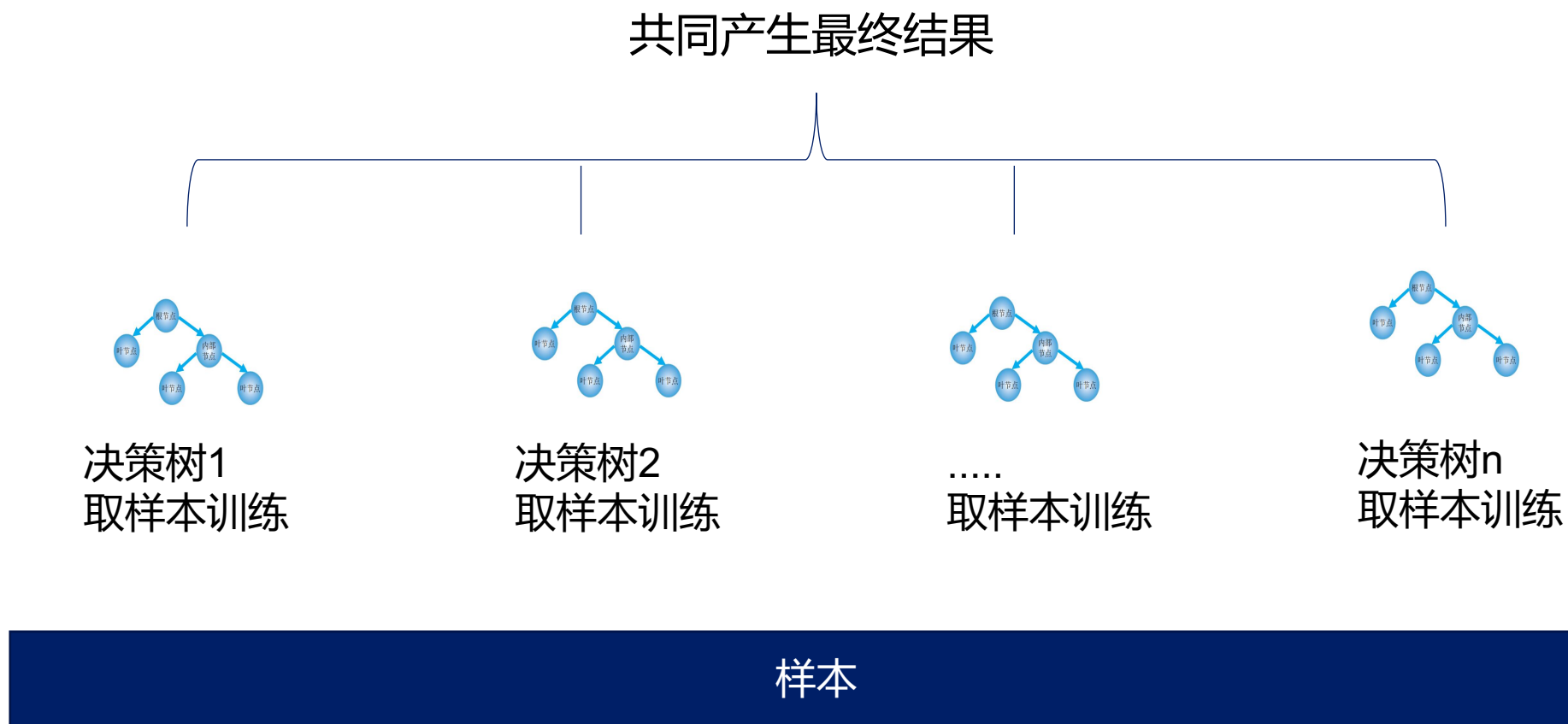
□ 分类树

返回叶子结点的类别

□ 回归树

返回叶子结点的平均数

随机森林



并行方式

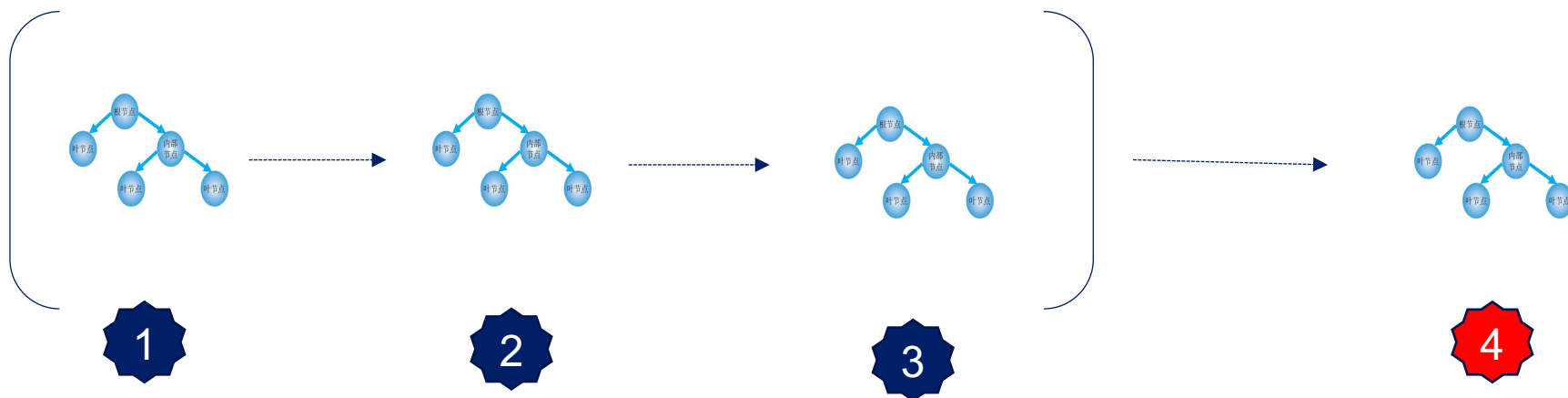
Bagging

1. 有放回随机取样，得到K个子样本
2. 分别训练K个模型
3. 民主投票，少数服从多数得到最终结果

Adaboost

1. 样本加权，增加错误分类的样本权重
2. 分布训练K个模型
3. 弱分类器加权，加权求和和得到预测结果

串行结构

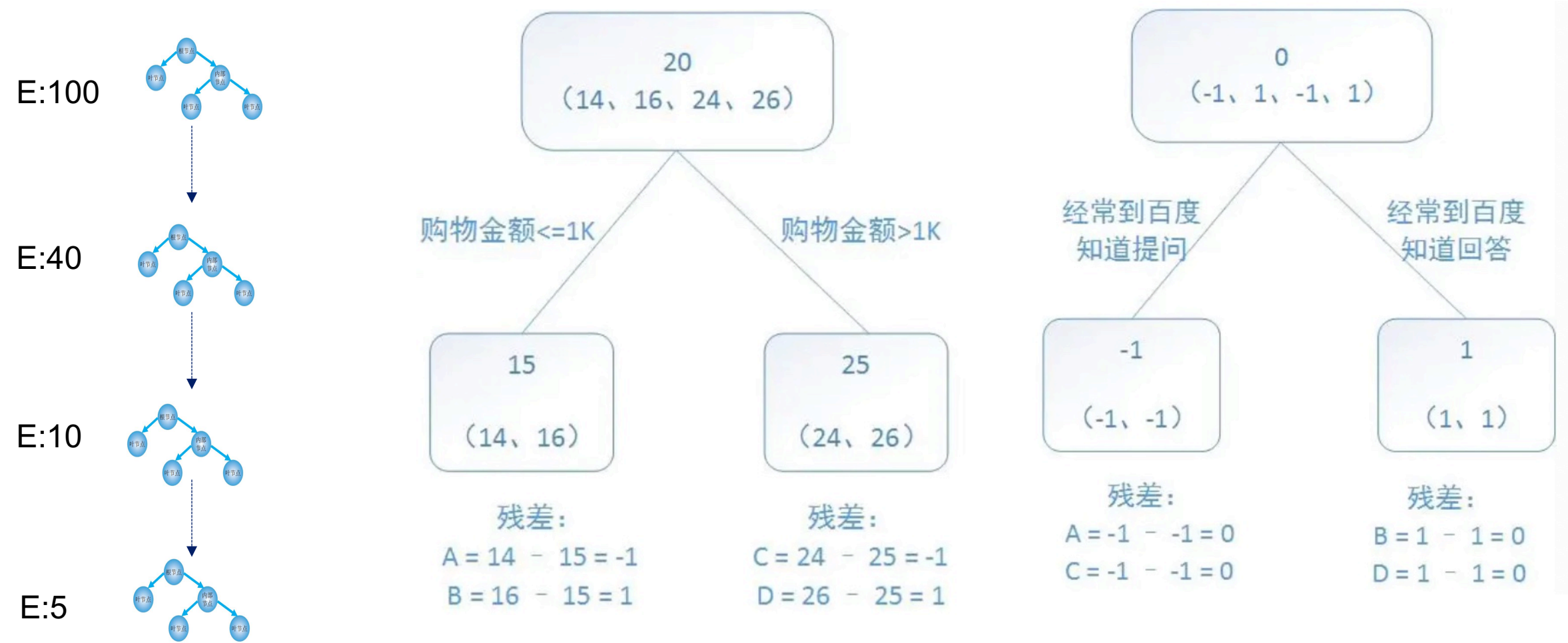


GBDT

新建第4颗时，利用到前3颗树学到的知识

GBDT (传统)

树串行组织，盯着残差学习，反向累积后产生结果



XGBoost

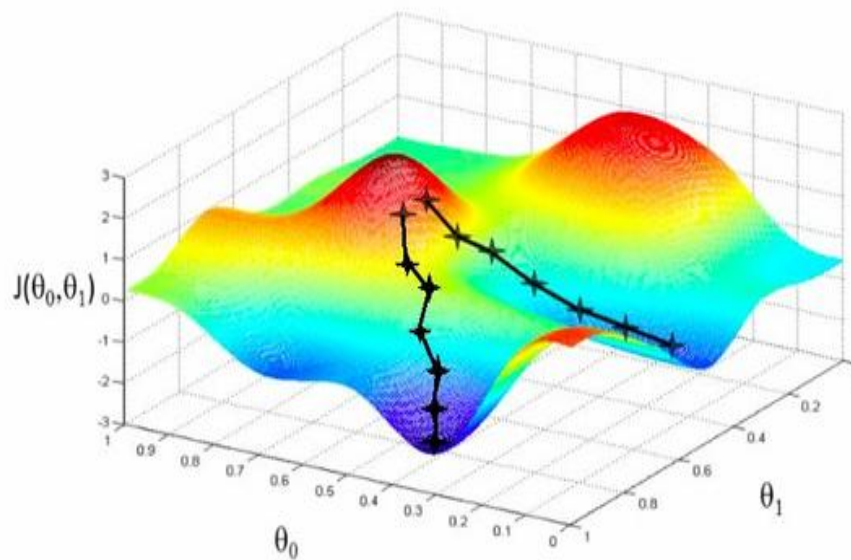
改进GBDT的损失函数，增加二阶展开，增加正则项

$$Obj^{(t)} = \sum_{i=1}^n l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t)$$

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

$$f(x + \Delta x) \simeq f(x) + f'(x)\Delta x + \frac{1}{2}f''(x)\Delta x^2$$

把决策树看成 x , $f(x)$ 为树的输出, $\Delta x \rightarrow$ 待求的决策树



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

$$Obj = -\frac{1}{2} \sum_{j=1}^T \frac{G_j^2}{H_j + \lambda} + \gamma T$$

采样方法

X1	X2	X3	X4	X5
----	----	----	----	----	-------

- ❑ 掷骰子：[0,9] 随机一个数，> 7 放到测试集，否则训练集
- ❑ 将顺序打乱,取前30%至测试集，其他为训练集

```
test_size=0.20 #测试样本所占比例
```

```
shuffle=False #是否要乱序
```

```
random_state=1 #随机种子
```

```
from sklearn.model_selection import train_test_split
X, y = np.arange(100).reshape((10, 10)), range(10)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=1, shuffle=True)
```

不平衡样本



正常样本 (比重高) ●-----> 异常样本 (比重低)

静态采样

- 平衡样本



- 不平衡样本

