# 实验二：SVM, 决策树方法的实现

任务：

1. 掌握决策树进行回归和分类任务
2. 掌握用随机森林方法进行分类， 进而熟悉GBDT方法；
3. 掌握SVM的分类应用。

## Ⅰ 方法原理介绍

* **决策树**

可以用一个树状结构表示多级分类器的决策过程，这个树状结构就是决策树。决策树有一个根节点，一组中间节点和一组叶节点及节点间相连接的分支组成，根节点和中间节点用分支连接他们的子节点。一个决策树对应一个决策方案，决策树中，每一个非叶节点都有一个相应的决策规则，从而给出模式空间的一个划分，因即使具有某个或某些相同的属性，也可以是不同类的样本，故还要根据其他决策规则进行划分，这在决策树中体现为分支，最后分化将特征空间分化为若干子区域，每个子区域中，只含一类或含几类但某一类占绝大多数；而中间划分将特征子空间分成较少的区域，每个区域混杂着几类样本，没有哪个样本占绝对优势。

因此，决策树的任意节点都对应一个区域，从而任意节点都对应类别，中间节点对应几个类别，叶节点对应一个类别。树的各节点的分支节点是互斥的，且各叶节点覆盖该节点决策规则所设计的属性空间。决策树的节点用类别符号来标记，不同的叶节点可能有相同的类别标记，及几个叶节点都对应一个类别。父节点的分支表示该节点对应的类别向子节点的指派细分。一个节点向下的分支数称为分支系数或分支率(branching ratio)。

涉及决策树要解决的问题：

a) 选择一个适当的树结构；包括节点处分数的确定，某节点知否是叶节点。

b) 确定在每一个非叶节点处要使用的特征；

c) 确定每一个非叶节点处的决策规则；

树分类器的设计原则是所选用的特征和所作的决策应使后续的子节点对应的区域或模式集尽可能“纯”，当不考虑误判所带来的损失代价时，为了量化“纯”这个指标，可以定义“不纯度”指标。节点n 处的不纯度*I(n)*应满足，当节点对应的模式集为同一类时，*I(n) = 0*，而当各类模式在该节点处均匀分布或数量相等时，*I(n)*最大，最常用的不纯度指标时“熵不纯度”(entropy impurity)，其定义为

C:\Users\meyyj\AppData\Local\Temp\1509426610(1).png

其中，*P (wj)*是节点*n* 处属于*wj*类的模式数相对总数的频率或频数。由熵的性质知道，若所有模式都属于一类，则不纯度*I(n) = 0*，否则是大于零的正数，当各模式以相同的频率或频数出现时，不纯度*I (n)*取最大。

如果不考虑效率等，那么样本所有特征的判断级联起来终会将某一个样本分到一个类终止块上。实际上，样本所有特征中有一些特征在分类时起到决定性作用，决策树的构造过程就是找到这些具有决定性作用的特征，根据其决定性程度来构造一个倒立的树--决定性作用最大的那个特征作为根节点，然后递归找到各分支下子数据集中次大的决定性特征，直至子数据集中所有数据都属于同一类。所以，构造决策树的过程本质上就是根据数据特征将数据集分类的递归过程，我们需要解决的第一个问题就是，当前数据集上哪个特征在划分数据分类时起决定性作用。

为了找到决定性的特征、划分出最好的结果，我们必须评估数据集中蕴含的每个特征，寻找分类数据集的最好特征。完成评估之后，原始数据集就被划分为几个数据子集。这些数据子集会分布在第一个决策点的所有分支上。如果某个分支下的数据属于同一类型，则该分支处理完成，称为一个叶子节点，即确定了分类。如果数据子集内的数据不属于同一类型，则需要重复划分数据子集的过程。如何划分数据子集的算法和划分原始数据集的方法相同，直到所有具有相同类型的数据均在一个数据子集内（叶子节点）。如下图就是一个决策树实例（目标是两类--见或者不见，每个样本有年龄、长相、收入、是否公务员四个特征）：



**决策树的学习过程:**

一棵决策树的生成过程主要分为以下3个部分:

**特征选择**：特征选择是指从训练数据中众多的特征中选择一个特征作为当前节点的分类标准，如何选择特征有着很多不同量化评估标准，从而衍生出不同的决策树算法。

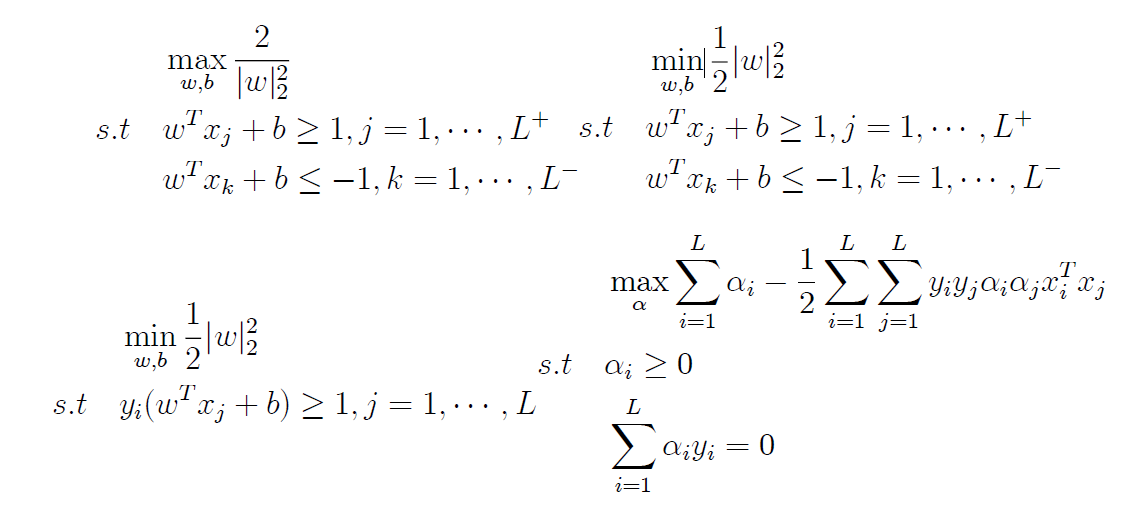
**决策树生成**： 根据选择的特征评估标准，从上至下递归地生成子节点，直到数据集不可分则决策树停止生长。 树结构来说，递归结构是最容易理解的方式。

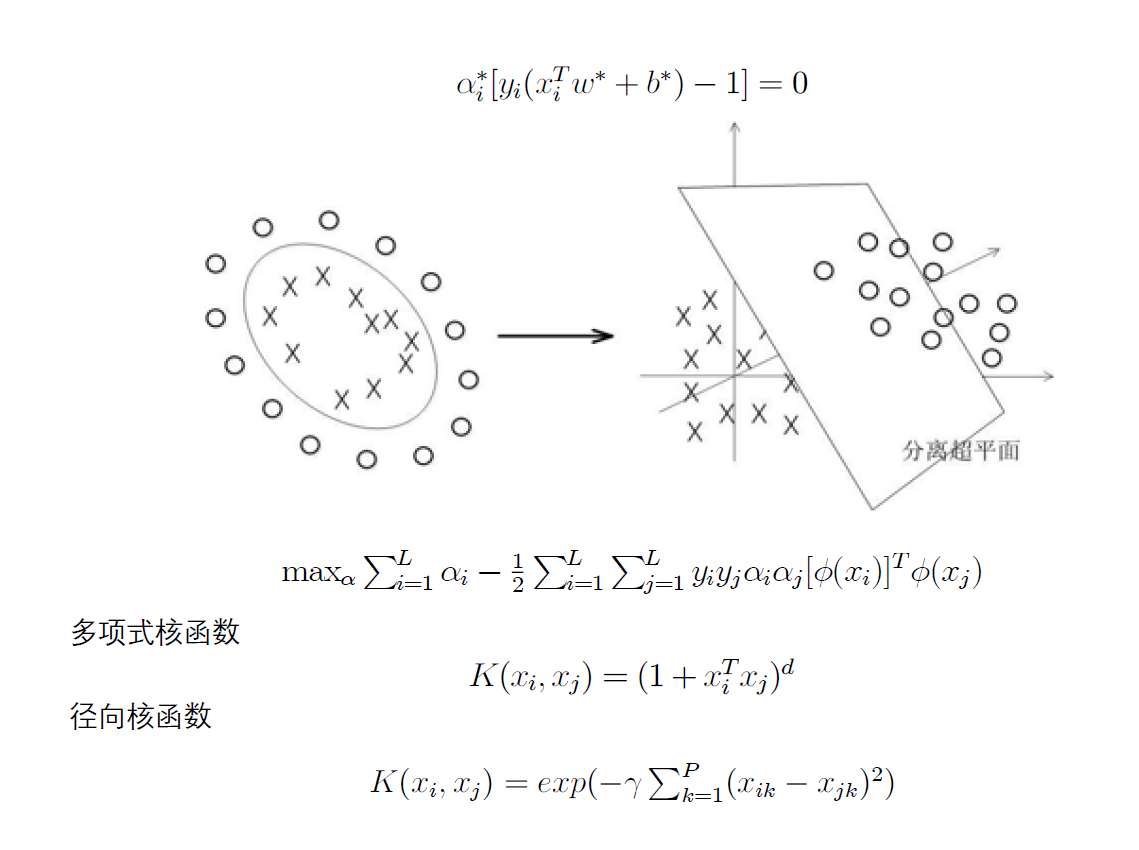
**剪枝**：决策树容易过拟合，一般来需要剪枝，缩小树结构规模、缓解过拟合。剪枝技术有预剪枝和后剪枝两种。

* **支持向量机（SVM）**

考虑二分类问题，设定*n* 维训练样本*x1; x2; … ; xN*，为处理方便，定义每个样本的类别属性值，*w1*类的训练样本*xi*，其属性值*yi =1，w2*类的训练样本*xj*，其属性值*yi = -1*，于是上述的各训练样本重新表示为*(x1; y1); (x2; y2); … ; (xN; yN)*.设这些样本是线性可分的，即存在线性界面将这些训练样本正确地分为两类，很容易知道，这样地分类界面通常有无数个，所以取得线性判别函数的权向量是权空间解中的一个，更进一步，希望从中求得的分类界面不但能将两类样本正确地分开，同时希望它与最近的样本点的距离都达到最大，就是说，线性分类界面方向的选取应使分别过两类样本中间分类界面最近的点且与界面平行的两平面之间距离最大。上述的分类界面称为最优分界面，相应的分类器称为最大余量分类器(maximum margin classifier)。两个平面间的距离称为分类间距，这两个类界面上的训练样本称为支持向量(support vector).







## Ⅱ 实验任务

### 决策树和集成学习

#### 学习并运行 http://scikit-learn.org/stable/modules/tree.html

1. 通过命令help(tree.DecisionTreeClassifier) ， 了解算法各个参数含义；
2. 导入iris数据集， from sklearn.datasets import load\_iris；

写一段程序 SMLprog1.py 能够打开该数据集，划分训练集和测试集，分别用gini准则和entropy准则训练决策树，并可视化树。 如果是缺省值得到的模型是多少个叶子节点？ 分别设置max\_leaf\_nodes 参数数目从9到3，看看树结构的变化，并说明之。（Tips： .dot文件在linux环境可用命令dot -Tpdf tree.dot -o iris.pdf 转化成pdf， 在Windows环境下可用graphviz打开）

变换 max\_depth 值， 讨论树的深度对模型影响。

1. 了解多输出决策树回归问题Multi-output Decision Tree Regression， <http://scikit-learn.org/stable/auto_examples/tree/plot_tree_regression_multioutput.html>
2. 用上述决策树解决人脸图像补全问题，参见：

<http://scikit-learn.org/stable/auto_examples/plot_multioutput_face_completion.html#example-plot-multioutput-face-completion-py>

改变决策树的参数，讨论其影响。

1. **采用random forests进行人脸补全，并给出与一般决策树方法的可视化对比图， 请参考**

http://scikit-learn.org/stable/auto\_examples/ensemble/plot\_random\_forest\_regression\_multioutput.html#sphx-glr-auto-examples-ensemble-plot-random-forest-regression-multioutput-py

### SVM分类和SVR回归

#### SVR回归

1. 了解核函数的影响：

<http://scikit-learn.org/stable/auto_examples/svm/plot_svm_regression.html#sphx-glr-auto-examples-svm-plot-svm-regression-py>

1. 写一段代码用SVR实现对波士顿房价数据的回归拟合结果，并评估之。

#### SVM回归

1）、学习scikit-learn中svm函数操作：

<http://scikit-learn.org/stable/modules/svm.html#svm>

2）、对于手写体数字图片，用不同核函数训练svm， 给出评估结果。