机器学习实战

机器学习通常有以下几个步骤：

1. 收集数据
2. 准备输入数据
3. 分析输入数据
4. 训练算法（可使用交叉验证）
5. 测试算法（一般是监督学习算法需要此步骤）
6. 使用算法

本书算法代码使用的是Python语言，本书使用了Python库NumPy/Beautiful Soup/Mrjob/Vote Smart/Python-Twitter。

## 第一章 K-近邻算法

### 1 算法原理及实现

#### 1.1 原理

在训练集中选取离输入的数据点最近的k个邻居，根据k个邻居中出现最多的类别（最大表决规则），作为该数据点的类别。

#### 1.2 实现过程

1. 计算所有点与当前的距离

根据距离值进行排序

1. 选取与当前点距离最短的K个点
2. 确定K个点所在的类别
3. K个点中频率最高的类别可当作当前点的类别

### 2 算法优缺点

K值的选取对kNN学习模型有很大的影响。若K值过小，预测结果会对噪音样本点显得异常敏感。

优点：精度高、对异常值不敏感、无数据输入假定。

缺点：计算复杂度度高，空间复杂度高。

适用数据范围：数值型与标称型。

### 3 算法应用

1. 文本分类--使用K-近邻算法改进约会网站的匹配效果

### 识别手写数字

## 第二章 决策树

### ID3算法原理及实现

#### 1.1 原理

决策树是一个树结构（可以是二叉树或者非二叉树）。其每个非叶节点表示一个特征属性上的测试，每个分支代表这个特征属性分类结果。通过训练数据可以构造出一个决策树，测试数据从根节点输入，从叶子节点输出，该子节点的分类就是测试点的分类。

#### 1.2 信息增益

信息熵表示的是不确定度，均匀分布时，不确定度最大，此时熵就最大。在划分数据集之前之后信息发生的变化称为信息增益。信息增益可以衡量某个特征对分类结果的影响大小。集合信息的度量方式称为香农熵。

熵定义为信息的期望值，在明晰这个概念之前，我们必须知道信息的定义。如果待分类的事物可能划分在多个分类之中，则符号的信息定义为



其中是选择该分类的概率。

为了计算熵，我们需要计算所有类别所有可能值包含的信息期望值，通过下面的公式得到：，其中n是分类的数目。

#### 1.3 算法实现

1、计算数据香农熵

2、划分数据并选择最好的数据集划分方式

3、直到所有数据属于同一类，划分结束

4、创建树并保存

### 2 算法优缺点

优点：决策过程非常直观，容易被人理解；计算复杂度不高，且决策树可存储；对中间值的确定不敏感，可以处理不相关特征数据。

缺点：可能会产生过度匹配的问题。

### 3 算法应用

1. 使用决策树预测隐形眼镜类型

## 第三章 朴素贝叶斯

### 1 算法原理及实现

#### 1.1 原理

条件概率定义：设A,B是两个事件，且P(A)>0称P(B|A)=P(AB)/P(A)为在条件A下发生的条件事件B发生的条件概率。

贝叶斯公式

 

从而得到X属于各类的概率，最高的就是X的类别。

#### 1.2 算法实现

1. 首先需要构建词向量
2. 在多分类中，对于某一门类，用一个向量PVect记录所有关于本类的词向量的向量加，另一个标量PNum记录所有词的总量,则Pvect/Pnum就是概率P(A|B)。
3. 测试数据，针对数据词向量中的词计算出就每个类别的概率。
4. 比较概率，得出结果

### 算法优缺点

优点:可识别多类别问题，在数据较少的情况仍有效。

缺点：对输入数据的准备方式较为敏感。

### 3 算法应用

1. 文档分类
2. 垃圾邮件检测

垃圾邮件一般是宁愿多一点垃圾邮箱，也不想误删非垃圾邮件。

## 第四章 Logistic回归

### 1算法原理及实现

#### 1.1 算法原理

Logistic回归其实是围绕一个Logistic函数展开的。

Sigmoid函数 

因此，为了实现Logistic回归分类器，我们可以在每个特征乘以一个回归系数，结果代入Sigmoid函数，输出大于0.5 被分为1类，小于0.5被分为0类。



#### 1.2 梯度上升法

思想：要想找到某函数的最大值，最好的方法就是沿着该函数的梯度方向探寻。

函数f(x,y)的梯度的公式为

梯度的迭代公式如下：



该公式一直迭代下去，直到w满足条件方可终止迭代。

#### 1.3 梯度上升算法流程

1、每次回归系数初始化为1

2、计算真实类别与预测类别的差值

3、根据差值来调整回归系数

4、当完成一定的调整次数，返回最后回归系数

随机梯度上升算法与之不同的是每次调整回归系数的样本数据只是整个大样本的其中一个元数据，而梯度上升算法每次调整回归系数的样本数据是全部样本数据。此外随即上升梯度算法是一个在线算法。

### 2算法优缺点

优点：计算代价不高

缺点：容易欠拟合，分类精度可能不高

### 算法应用

1. 从疝气病症预测病马的死亡率

对于数据中缺失值的问题，一般采取以下几种做法：

1. 使用特征的平均值来填补
2. 使用特殊值填补，如-1
3. 忽略有缺失值的样本
4. 使用相似样本的均值填补
5. 使用另外的机器学习算法预测缺失值

## 第六章 支持向量机（SVM）

### 1 算法原理及实现

#### 1.1 基于最大间隔分隔数据

在一堆训练数据中，将数据分隔开来的直线称为分隔超平面。我们希望找到离分隔超平面最近的点，确保他们离分隔面的距离尽可能远。这里点到分隔面的距离称为间隔。我们希望间隔尽可能地大，这是因为如果我们犯错或者在有限数据上训练分类器的话，我们希望分类器更加健壮。

**支持向量（support vector）就是离分隔超平面最近的那些点。**

要计算点A到分隔超平面的距离，就必须给出点到分隔面的法线或垂线的长度，该值为 。

现在的目标就是找到分类器定义中的w和b。为此，需要找到具有最小间隔的数据点，而这些数据点就是支持向量。一旦找到具有最小间隔的数据点，我们就需要对该间隔最大化。这就写作：



对上式求解还是相当困难，我们可以将其转换成为另一种更容易求解的方式。如果令所有支持向量的都为1，那就可以通过求 的最大值来得到最终解。

这里的约束条件就是。对于这类优化问题，有一个非常著名的求解方法，即拉格朗日乘子法。于是，优化目标函数最后可以写成：



其约束条件为：

 ，和

至此一切都很完美，但是这里有个假设：数据必须100%线性可分。最后程序主要就是求解一个最优。

#### 1.2 SMO（序列最小优化）高效算法

SMO算法的工作原理是：每次循环中选择两个alpha进行优化处理。一旦找到一对合适的alpha，那么就增大其中一个同时减小另一个。这里所谓的“合适”就是指两个alpha必须要符合一定的条件，条件之一就是这两个alpha必须要在间隔边界之外，另一个条件就是这两个alpha还没有进行国区间化处理或者不在边界上。

### 2 算法优缺点

### 3 算法应用