install.packages("plyr")

install.packages("reshape2")

library(plyr)

library(reshape2)

#数据整合和重塑

#1、根据训练集创建朴素贝叶斯分类器

#1.1、生成类别的概率

## 计算训练集合D中类别出现的概率，即P{c\_i}

## 输入：trainData 训练集，为数据框

## strClassName 指明训练集中名称为strClassName列为分类结果

## 输出：数据框，P{c\_i}的集合，类别名称|概率（列名为prob）

class\_prob <- function(trainData, strClassName){

length.train <- nrow(trainData)

dTemp <- ddply(trainData, strClassName, "nrow")

#ddply用于对trainData进行分组统计

#统计taste的频数

dTemp <- ddply(dTemp, strClassName, mutate, prob = nrow/length.train)

#继上一步计算频率

dTemp[,-2]

}

##1.2、生成每个类别下，特征取不同值的概率

feature\_class\_prob <- function(trainData, strClassName){

#横表转成纵表

data.melt <- melt(trainData, id = c(strClassName))

#统计频数

aa <- ddply(data.melt, c(strClassName, "variable", "value"), "nrow")

#计算每一种特征对应的分类的频数

bb <- ddply(aa, c(strClassName, "variable"), mutate, sum = sum(nrow), prob = nrow/sum)

#增加列名

colnames(bb) <- c("class.name",

"feature.name",

"feature.value",

"feature.nrow",

"feature.sum",

"prob")

# 返回结果

bb[, c(1,2,3,6)]

}

pre\_class <- function(oneObs, pc, pfc){

colnames(oneObs) <- c("feature.name", "feature.value")

colnames(pc) <-c("class.name", "prob")

colnames(pfc) <- c("class.name", "feature.name", "feature.value", "prob")

#取出特征的取值的条件概率

feature.all <- join(oneObs, pfc, by = c("feature.name", "feature.value"), type="inner")

#取出特征取值的条件概率的连乘

feature.prob <- ddply(feature.all, .(class.name), summarize, prob\_fea = prod(prob))

#prod是连乘函数

#取出类别的概率

class.all <- join(feature.prob, pc, by = "class.name", type = "inner")

#输出结果

ddply(class.all, .(class.name), mutate, pre\_prob = prob\_fea \* prob)[,c(1,4)]

}

trainData <-data.frame(

color=c("green","black","black","white","black","white"),

tuber=c("crouch","crouch","crouch","little","little","crouch"),

lines=c("clear","clear","clear","Slightly","Slightly","blur"),

taste=c("good","good","bad","bad","bad","bad")

)

length.train <- nrow(trainData)

head(trainData)

#待预测样本

oneObs<-data.frame(

feature.name =c("color", "tuber", " lines"),

feature.value =c("green","crouch","Slightly")

)

#预测分类

pc <- class\_prob(trainData,"taste")

pfc <- feature\_class\_prob(trainData,"taste")

pre\_class(oneObs, pc,pfc)

