Machine Learning Homework 6- Huynh Hoang Trung Nghia

Huynh Hoang Trung Nghia

3/9/2020

Data loading and partition

```
#Loading and Clean data
data <- read.csv("train.csv")
data[c("Age")] <- imputeMissings::impute(data[c("Age")])
data$Embarked[c(62,830)] <- "S"
data$Partner <- apply(data[c("SibSp","Parch")],1,function(a) !(a[1] == 0 & a[2] == 0))
data$Below10 <- apply(data[c("Age")],1,function(a) a <= 10)
data <- select(data,Survived, Pclass, Sex,Partner,Below10, Fare, Embarked)
for (i in c("Survived", "Pclass", "Partner","Below10")) {
  data[,i] <- as.factor(data[,i])
}

#data partition
index <- sample(c(1,2),nrow(data), replace = T,prob = c(0.8,0.2))
train <- data[index == 1,]
test <- data[index == 2,]
```

NaiveBayes and test function

```
#NaiveBayes
NaiveBayes <- function(data, class, predictor = "all"){
  library(dplyr)
  if(predictor == "all"){
    pred <- select(data, -class)
  } else {
    pred <- select(data, predictor)
  }
  if(is.factor(data[,class]) == FALSE ) stop("the independent variable must be categorical variable")
  ycount <- xtabs(~ data[,class])
  mod <- list("Target" = class,
              "levels" = levels(data[,class]),
              "py"= table(data[,class])/nrow(data)
              )
  j <- 4
  for(i in names(pred)){
    if (is.factor(pred[,i])== TRUE) {
      px <- t(apply(xtabs(~setNames(pred[,i],i) + setNames(data[,class],class)),1,function(a) a/ycount))
      px <- cbind(px, c(table(data[,i])/nrow(data)))
    } else if(is.factor(pred[,i])== FALSE){
      px <- data %>% group_by_(.dots = class) %>% summarise_(
        setNames(paste0("mean(",i,")"),"mean"),
```

```
        setNames(paste0("sd(",i,")"),"sd"))
      px <- rbind(px,c("all", mean(pred[,i]),sd(pred[,i])))
    } else {
      stop("one of the predictors is not categorical or continuous variables")
    }
    mod[j] <- list(px)
    names(mod)[j] <- i
    j <- j+1
  }
  return(mod)
}

#Test function
test_Naive <- function(model, new_data, prob = FALSE, thread = 0.5){
  class <- model$Target
  prediction <- c()
  if(sum(is.na(test)) != 0) stop("There are missing values in The Test Set")
  test <- select(new_data, - class)
  first_lev <- model$levels[1]
  for(j in 1:nrow(new_data)){
    p.x_y <- 1
    p.x <- 1
    for(i in names(test)){
      x <- test[j,i]
      if (is.factor(test[,i])== TRUE) {
        p.x_i <- model[[i]][x,3]
        p.x_y_i <- model[[i]][x,first_lev]
      } else if(is.factor(test[,i])== FALSE){
        p.x_i <- dnorm(x, mean = as.numeric(model[[i]][is.na(model[[i]][class]),2]),
                       sd = as.numeric(model[[i]][is.na(model[[i]][class]),3]))
        p.x_y_i <- dnorm(x, mean = as.numeric(model[[i]][model[[i]][class] == first_lev,2][1,1]),
                         sd = as.numeric(model[[i]][model[[i]][class] == first_lev,3][1,1]))
      } else {
        stop("one of the predictors is not categorical or continuous variables")
      }
      p.x_y <- p.x_y*p.x_y_i
      p.x <- p.x*p.x_i
    }
    prediction[j] <- (p.x_y*model$py[first_lev])/p.x
  }
  result <- sapply(prediction, function(a) ifelse(a > thread, a <- first_lev, a <- model$levels[2]))
  ifelse(prob == TRUE,return(prediction ),return(result))
}
```

#fit and predict the data to the NaiveBayes model

```
#fit
NaiMod <- NaiveBayes(data = train,class = "Survived")
NaiMod
```

```
## $Target
## [1] "Survived"
##
```

```
## $levels
## [1] "0" "1"
##
## $py
##
##         0         1
## 0.6281337 0.3718663
##
## $Pclass
##           0         1
## 1 0.1485588 0.4269663 0.2520891
## 2 0.1751663 0.2359551 0.1977716
## 3 0.6762749 0.3370787 0.5501393
##
## $Sex
##                0         1
## female 0.1507761 0.6928839 0.3523677
## male   0.8492239 0.3071161 0.6476323
##
## $Partner
##              0         1
## FALSE 0.691796 0.4794007 0.6128134
## TRUE  0.308204 0.5205993 0.3871866
##
## $Below10
##                0         1
## FALSE 0.94900222 0.8876404 0.92618384
## TRUE  0.05099778 0.1123596 0.07381616
##
## $Fare
## # A tibble: 3 x 3
##   Survived `mean(Fare)`     `sd(Fare)`
##   <fct>    <chr>            <chr>
## 1 0        22.2461917960089 32.955793781415
## 2 1        51.1720194756554 70.9097693751963
## 3 <NA>     33.0027321727019 52.3714191652296
##
## $Embarked
##            0          1
##   0.00000000 0.00000000 0.00000000
## C 0.13303769 0.27340824 0.18523677
## Q 0.06651885 0.07865169 0.07103064
## S 0.80044346 0.64794007 0.74373259
```

```
#predition
result <- test_Naive(model = NaiMod, new_data = test)
result
```

```
##   [1] "1" "0" "0" "0" "0" "0" "1" "1" "1" "0" "1" "1" "0" "0" "0" "0" "0"
##  [18] "0" "0" "1" "0" "0" "0" "0" "0" "0" "1" "1" "0" "1" "0" "1" "0" "0"
##  [35] "0" "0" "1" "0" "0" "0" "0" "1" "0" "0" "0" "0" "1" "0" "0" "1" "0"
##  [52] "0" "0" "1" "0" "0" "0" "1" "1" "0" "0" "0" "0" "1" "0" "1" "0" "1"
##  [69] "0" "1" "1" "0" "0" "0" "0" "1" "0" "0" "0" "1" "1" "0" "0" "0" "0"
##  [86] "1" "0" "0" "0" "0" "1" "1" "0" "1" "0" "0" "0" "1" "0" "0" "0" "1"
```

```
## [103] "0" "0" "0" "1" "1" "0" "0" "0" "0" "1" "1" "0" "1" "0" "0" "1" "1"
## [120] "1" "1" "1" "0" "1" "1" "1" "0" "0" "0" "0" "0" "1" "0" "0" "1" "1"
## [137] "0" "1" "0" "0" "1" "0" "1" "1" "0" "0" "1" "1" "1" "1" "0" "0" "1"
## [154] "0" "0" "0" "0" "0" "0" "0" "1" "0" "0" "0" "0" "0" "1" "1" "0" "1"
## [171] "0" "1" "0"
```

```
#accuracy
sum(result == test$Survived,na.rm = T)/nrow(test)
```

```
## [1] 0.7514451
```