OCSVM-1test

Rodrigo

November 15, 2016

## Proposta

Queremos testar a eficiência de um classificador **one-class SVM**(OCLSVM) [1]. A ideia é utilizar um grande número de observações como exemplo da classe positiva e mostrar que, desse modo, é possível classificar com grande **precisão e acurácia** (isso é melhor discutido na conclusão) se uma levedura pertence ou não a uma determinada espécie.

### Saccharomyces, a classe positiva

A levedura *Saccharomyces cerevisiae* é uma das leveduras mais importantes no processo industrial.Por esse motivo, ela é a que possui a maior quantidade de observações no banco de dados do NYCY[2]. Utilizaremos essa levedura como objeto de estudo sobre a eficácia de um modelo OCSVM para reconhecer uma levedura através de seus padrões fisiológicos.

## Dados

Utilizamos scripts em Python e coletamos um grande número de observações entre 9 classes(espécies diferentes).

Temos portanto as seguintes dimensões de dados

dim(full\_data)

## [1] 1069 53

## O problema dos dados faltantes

Mesmo com um número tão expressivo de dados ainda temos o problema de dados faltantes. Os parâmetros utilizados dependem diretamente da facção ou não de experimentos de crescimento em determinadas condições, feitos em laboratório. Por isso, é comum encontrarmos observações muito completas, com quase todos os cenários testados, e encontrar outras muito incompletas e com quase todos os atributos faltantes.

Por isso resolvemos pegar um grande número de dados e remover a nosso critério aqueles com uma porcentagem muito grande de dados faltantes.

### Imputacao pela moda

Para poder rodar os algoritmos supervisionados durante o estudo foi preciso adotar alguma abordagem de correção dos dados faltantes. Para resolver um atributo faltante em uma observação utilizamos como argumento que a moda da **classe** desse atributo seria uma boa abordagem, visto que acreditamos em um alto grau de similaridade as observações da mesma espécie. Além disso, mesmo que muitas das observações tenham que ser futuramente descartadas, pela grande porcentagem de atributos faltantes, não queremos desperdiçar informação delas. Por isso, levamos em consideração **todo o conjunto original** para calcular a moda de cada atributo de cada classe.

Essa imputação foi feita através de um script em Python. Desse modo, temos dois arquivos CSV. Um original, com dados faltantes e um com todos os dados faltantes imputados. Carregando os dados imputados.

setwd('/home/rodrigo/Dropbox/UNICAMP/MC886/trabalho/ProjetoY/datasets')  
imputed\_full\_data = read.csv("YPDS\_merge\_imputado.csv", header=T)

### Removendo dados faltantes

Agora vamos retirar todos as observações com mais de 10% de dados faltantes.

pMiss = function(x){sum(is.na(x))/length(x)\*100}  
  
percent = 10  
  
new\_rows\_idx = which(apply(x\_data,1,pMiss) < 10)  
new\_rows\_idx\_length = length(new\_rows\_idx)

Vamos trabalhar então com 394 observações.

Agora que removemos o grosso das observações esparsas podemos remover as colunas com mais de 10% de dados faltantes.

pMiss = function(x){sum(is.na(x))/length(x)\*100}  
  
percent = 10  
  
new\_cols\_idx = which(apply(x\_data[new\_rows\_idx,],2,pMiss) < 10)  
new\_cols\_idx\_length = length(new\_cols\_idx)

Vamos trabalhar então com 47 atributos.

Então, vamos atualizar o nosso dataset para utilizar só observações e atributos com menos de 10% de dados faltantes.

# Selecionando o novo dataset no conjunto total de dados imputados  
x\_data = imputed\_full\_data[new\_rows\_idx,new\_cols\_idx]  
y\_data = imputed\_full\_data[new\_rows\_idx,1]

Analisando esse novo dataset...

length\_x\_data = dim(x\_data)[1]  
# 0 é o ID da sacchraromyces  
length\_sacchraromyces = length(which(y\_data == 0))

Podemos ver que, dessas 394 observações, 333 são da espécie Saccharomyces, classe positiva.

## Conjunto de treino e teste

Vamos utilizar agora, 250 das observações da espécie Saccharomyces para treino e o restante para compor o teste. O teste é composto por exemplos da classe positiva, Saccharomyces, e exemplos de outras classes negativas que serão classificadas como desconhecidas.

x\_saccharomyces\_train = x\_data[1:250,]  
  
x\_test = x\_data[250:nrow(x\_data),]  
y\_test = y\_data[250:length(y\_data)]

## Treinando o classificador one-class SVM

library(e1071)  
  
svm\_model = svm(  
 x = x\_saccharomyces\_train,   
 y = NULL,  
 scale = FALSE,   
 kernel="radial",   
 type="one-classification",  
 nu=0.01)  
   
svm.pred = predict(svm\_model, x\_test)  
table(Predicted=svm.pred,Reference=y\_test)

## Reference  
## Predicted 0 1 2 3 4 5 8 9  
## FALSE 1 20 4 5 13 13 2 4  
## TRUE 83 0 0 0 0 0 0 0

Isso nos dá uma acurácia de 0.9931034 e uma precisão de 0.9880952 para esse teste. É preciso destacar que a acurácia não é uma boa escolha para um problema tão desbalanceado, natural dos problemas OPEN-SET. Talvez fosse melhor avaliar com métricas como F-Measure [3].

Além disso, não fizemos nenhuma procura por hiperparametros no nosso modelo. Isso também é algo que deve ser avaliado.