Math 6350 fall 2019 MSDS Robert Azencott

Homework HW4 is an Individual HomeWork

Due date Mon Nov 18th at Midnight

there are only 2 parts for the full HW4

your grade for HW4 will have a coefficient 3 in the average HW grade

HW4 Part1 : SVM classification for Simulated Data

Question 1 : Generate a Data Set by Simulations

We seek to generate 5000 cases x^1 ... x^{5000} in R^4

each case x = [x1 x2 x3 x4] has 4 numerical features

Step 1

using random sampling of uniform distribution over the interval [-2, +2]

select 16 random numbers Aij with i= 1 2 3 4 and j = 1 2 3 4

select 4 random numbers Bi with i= 1 2 3 4

select 1 random number c

display the values of these random numbers

Define the polynomial of degree 2 in the 4 variables x1 x2 x3 x4 as follows

Pol(x) = $\sum_i \sum_j$ Aij xi xj + $\sum_i$ Bi xi + c/20

Step 2

using random sampling of uniform distribution over the interval [-2, 2]

select 10, 000 vectors x^1 ... $x^{10,000}$ in R^4

each such vector x^n has 4 randomly chosen coordinates with values in [-2, 2]]

for each selected x^n compute U(n) = Pol(x^n) and y(n) = sign[U(n)]

define two classes by

CL(1) = class1= set of all x^n such that y(n) = +1

CL(-1) = class1= set of all x^n such that y(n) = -1

keep only 2500 cases in CL(1) and 2500 cases in CL(-1), Center and Rescale this data set of size 5000 so that the standardized data set will have mean = 0 and dispersion =1

Then Split each class into a training set and a test set , using the proportions 80% and 20%

this defines a training set TRAIN and a test set TEST of resp. sizes 4000 and 1000

Question 2: SVM classification by linear kernel

(stated for R programming environment, but must have equivalent forms in Python)

Fix arbitrarily the "cost" parameter in the svm() function, for instance cost = 5

Select the kernel parameter kernel = "linear "

Run the svm() function on the set TRAIN

compute the number S of support vectors and the ratio s = S/4000

compute the percentages of correct prediction PredTrain and PredTest on the sets TRAIN and TEST

compute two confusion matrices (one for the set TRAIN and one for the test set

confusion matrices must be converted in terms of frequencies of correct predictions within each class

compute the errors of estimation on PredTRAIN, PredTEST, and on the terms of the confusion matrices

interpret your results

Question 3 : optimize the parameter "cost"

Select a list of 6 values for the "cost " parameter

Run the tuning function tune() for the linear svm() to identify the best value of "cost"

Evaluate the performance characteristics of the "best" linear svm as in question 2

Question 4: SVM classification by radial kernel

Fix the "cost" parameter in the svm() function to the best cost value identified in question 3

Select the kernel parameter kernel = "radial " which means that the kernel ks given by the formula

K(x,y) = exp(-gamma || x- y ||^2)

Select arbitrarily the gamma parameter "gamma" = 1

Run the svm() function on the set TRAIN

as in question 2 compute the number S and the ratio s = S/4000 the percentages of correct predictions PredTrain and PredTest and the two confusion matrices

interpret your results

Question 5 : optimize the parameter "cost"and "gamma"

Select a list of 5 values for the "cost " parameter and a list of 5 values for the parameter "gamma"

On the TRAIN set , run the tuning function tune() for the radial svm() to identify the best value of the pair ("cost", "gamma") among the 25 values you have listed

Evaluate the performance characteristics of the "best" radial svm as in question 2

Interpret your results

Question 6 : SVM classification using a polynomial kernel

Implement the steps of question 4 and 5 for the svm() fonction based on the polynomial kernel

K(x,y) = (a + <x,y>)^4

You will have to optimize the choice of the two parameters "a" >0 and ''cost"

HW4 Part2 : SVM classification for Real Data

Question 1 : Download and Describe your choice of a real data set DS

Step 1 Describe the original classification task for DS:

practical goal of the original classification task

# of cases

# of features per case

# of original classes and their names

list of features names and meaning of each feature

indicate if features are continuous or discrete

for discrete features indicate the number of distinct values

Step 2 Describe precisely the reduced data set RDS

# of cases in RDS

# of actually kept features per case

# of classes you kept and their names

STRONG SUGGESTION : KEEP ONLY THE THREE LARGEST CLASSES FOT THIS PROBLEM

MAKE SURE THAT YOU HAVE AT LEAST 5000 examples IN THE TRAIN SET

for each kept class indicate if you have used cloning to increase its size

give all the kept classes sizes (after eventual cloning if needed)

list names of the features you kept in RDS

for continuous features kept in RDS: compute and display their mean and standard deviation within each class

for original discrete features which you kept in RDS:

> compute and display the histograms of these values within each class
>
> describe the recoding these values (which may force you to create new binary fetures

Step3 : Center and Rescale the whole RDS so that each feature will then have global mean = 0 and global stand. dev. =1

Split each class into a training set and a test set , using the proportions 80% and 20%

this defines a training set TRAIN and a test set TEST

give the new sizes of the classes within TRAIN and within TEST

give the sizes of TRAIN and TEST

Question 2: SVM classification by radial kernel

Select the kernel parameter   kernel = "radial "   so that the kernel K is given by the formula

K(x,y) = exp(-gamma || x- y ||^2)

Step1 : optimize the parameters "cost"and "gamma"

select the 2 larest classes CL1 and CL2

for the classification CL1 vs CL2

Select a list of 4 values for the "cost " parameter and a list of 4 values for the parameter "gamma"

On the TRAIN set , run the tuning function tune() for the radial svm() to identify the best value of the pair ("cost", "gamma") among the 16 values you have listed

This function does not use your TEST set but only the TRAIN set with 10-fold cross validation to estimate the performance identify the "best " radial svm performance value provided by this tuning set of 16 runs

Step 2: Re-evaluation of tuning :

Pick another two classes CL1 vs CL3 for instance and re-launch the tuning as above to see which pair of parameters {gamma, cost} are selected

Fix your own best parameters gamma and cost based on questions 1 & 2

Question 3 : for the largest 3 classes CL1 CL2 CL3 , compute 3 SVMs

Use the best parameters previously identified to train 3 svms :

SVM1 to classify CL1 vs (not CL1)

SVM2 to classify CL2 vs (not CL2)

SVM3 to classify CL3 vs (not CL3)

for each one of these 3 SVMs compute their 2 confusion matrices (on the TRAIN set and on the TEST set) and convert them into frequencies per true class .Compute the confidence intervals on these frequencies. Copare the resukts between TEST and TRAIN tests

Compute also the percentages of support vectors for each SVM1 SVM2 SVM3. Evaluate how confident you are for each one of these 3 SVMs

Question 4 : for the largest 3 classes CL1 CL2 CL3 , combine the three SVMs to classify all cases

As detailed in class , describe precisely for each new case x in Rp , how you can combine the three classifications of x provided by SVM1 SVM2 SVM3 , taking account of their confusion matrices, in order to obtain a terminal classification of x into one of the 3 classes CL1 CL2 CL3

Implement and run this combined classification on all x in CL1 CL2 CL3 which belong to TRAIN

Do the same operation on all x in CL1 CL2 CL3 which belong to TEST

Compute the associated two confusion matrices (which will be 3x3 matrices) and compare them

Question 5

Repeat the whole preceding procedure using the polynomial kernel (K(x,y) = (1+<x,y>)^2 .

There will be only one parameter to estimate by tuning, namely the Cost parameter. Compare the final performance results to the approach using radial kernels

Question 6 : SVM classification using a polynomial kernel

Implement the steps of question 4 and 5 for the svm() fonction based on the polynomial kernel

K(x,y) = (a + <x,y>)^4

You will have to optimize the choice of the two parameters "a" >0 and ''cost"