**Practical 8: Implementation of Naïve Bayes Classification algorithm**

package prob4;

import java.io.\*;

import java.sql.\*;

import java.math.\*;

class Bayesian

{

public static void main(String[] args)

{

String name,egender,gender,temph,classn;

classn = null;name=egender=null;

float height=0;int eid=0;

System.out.println("Program for Bayesian Clasification");

System.out.println("Enter The Data : Id,Name,Gender,Height");

try

{

BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

temph = in.readLine();

eid = Integer.parseInt(temph);

name = in.readLine();

egender = in.readLine();

temph = in.readLine();

height = Float.parseFloat(temph);

}catch (Exception e ){}

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

Connection connect = DriverManager.getConnection("jdbc:odbc:person");

Statement sest = connect.createStatement();

float countms,countmm,countmt,countfs,countfm,countft;

countms=countmm=countmt=countfs=countfm=countft=0;

gender= "male";

ResultSet rs = sest.executeQuery("SELECT \* FROM personal where gender like '"+gender+"'");

while(rs.next())

{

float temp = rs.getFloat(4);

if(temp<=1.8)

countms++;

else

if(temp>1.99)

countmt++;

else

countmm++;

}

gender="female";

ResultSet rs1 = sest.executeQuery("SELECT \* FROM personal where gender like '"+gender+"'");

while(rs1.next())

{

float temp = rs1.getFloat(4);

if(temp<=1.71)

countfs++;

else

if(temp>1.99)

countft++;

else

countfm++;

}

float probms,probmm,probmt,probfs,probfm,probft;

probms=probmm=probmt=probfs=probfm=probft=0;

probms = countms/(countms+countfs);// use this

probmm = countmm/(countmm+countfm);

probmt = countmt/(countmt+countft);

probfs = countfs/(countms+countfs);

probfm = countfm/(countmm+countfm);

probft = countft/(countmt+countft);

ResultSet rs2 = sest.executeQuery("SELECT \* FROM personal ");

float s1,m1,t1,s2,m2,t2,s3,m3,t3,s4,m4,t4,s5,m5,t5,s6,m6,t6;

s1=m1=t1=s2=m2=t2=s3=m3=t3=s4=m4=t4=s5=m5=t5=s6=m6=t6=0;

while(rs2.next())

{

float temp1=rs2.getFloat(4);

if(temp1<=1.61)

s1++;

else

if(temp1>1.61&&temp1<=1.71)

s2++;

else

if(temp1>1.71&&temp1<=1.81)

m3++;

else

if(temp1>1.81&&temp1<=1.91)

m4++;

else

if(temp1>1.91&&temp1<=1.96)

m5++;

else

if(temp1>1.96&&temp1<=2.0)

t5++;

else

if(temp1>2.0)

t6++;

}

float ps1,pm1,pt1,ps2,pm2,pt2,ps3,pm3,pt3,ps4,pm4,pt4,ps6,pm6,pt6,ps5,pm5,pt5;

ps1=pm1=pt1=ps2=pm2=pt2=ps3=pm3=pt3=ps4=pm4=pt4=ps6=pm6=pt6=ps5=pm5=pt5=0;

ps1 = s1/(s1+s2+s3+s4+s5+s6);

ps2 = s2/(s1+s2+s3+s4+s5+s6);

ps3 = s3/(s1+s2+s3+s4+s5+s6);

ps4 = s4/(s1+s2+s3+s4+s5+s6);

ps5 = s5/(s1+s2+s3+s4+s5+s6);

ps6 = s6/(s1+s2+s3+s4+s5+s6);

pm1 = m1/(m1+m2+m3+m4+m5+m6);

pm2 = m2/(m1+m2+m3+m4+m5+m6);

pm3 = m3/(m1+m2+m3+m4+m5+m6);

pm4 = m4/(m1+m2+m3+m4+m5+m6);

pm5 = m5/(m1+m2+m3+m4+m5+m6);

pm6 = m6/(m1+m2+m3+m4+m5+m6);

pt1 = t1/(t1+t2+t3+t4+t5+t6);

pt2 = t2/(t1+t2+t3+t4+t5+t6);

pt3 = t3/(t1+t2+t3+t4+t5+t6);

pt4 = t4/(t1+t2+t3+t4+t5+t6);

pt5 = t5/(t1+t2+t3+t4+t5+t6);

pt6 = t6/(t1+t2+t3+t4+t5+t6);

float pshort,ptall,pmedium;

pshort=pmedium=ptall=0;

gender = "short";

ResultSet rs4 = sest.executeQuery("SELECT \* FROM personal where class like '"+gender+"'");

while(rs4.next()){

pshort++;

}

gender = "medium";

rs4 = sest.executeQuery("SELECT \* FROM personal where class like '"+gender+"'");

while(rs4.next()){

pmedium++;

}

gender="tall";

rs4 = sest.executeQuery("SELECT \* FROM personal where class like '"+gender+"'");

while(rs4.next()){

ptall++;

}

float total;

total=pshort+pmedium+ptall;

pshort = pshort/total;pmedium = pmedium/total;ptall = ptall/total;

float ptgivens,ptgivenm,ptgivent;

ptgivens=ptgivenm=ptgivent=0;

if(egender.equals("male"))

{

if(height<=1.61)

{

ptgivens = probms\*ps1;

ptgivenm = probmm\*pm1;

ptgivent = probmt\*pt1;

}

else if(height>1.61&&height<=1.71)

{

ptgivens =probms\*ps2;

ptgivenm = probmm\*pm2;

ptgivent = probmt\*pt2;

}

else if(height>1.71&&height<=1.81)

{

ptgivens = probms\*ps3;

ptgivenm = probmm\*pm3;

ptgivent = probmt\*pt3;

}

else if(height>1.81&&height<=1.91)

{

ptgivens = probms\*ps4;

ptgivenm = probmm\*pm4;

ptgivent = probmt\*pt4;

}

else if(height>1.91&&height<=2.0)

{

ptgivens = probms\*ps5;

ptgivenm = probmm\*pm5;

ptgivent = probmt\*pt5;

}

else if(height>2.0)

{

ptgivens = probms\*ps6;

ptgivenm = probmm\*pm6;

ptgivent = probmt\*pt6;

}

float pls,plm,plt,ptotal;

pls=plm=plt=ptotal=0;

pls=ptgivens\*pshort;

plm=ptgivenm\*pmedium;

plt=ptgivent\*ptall;

ptotal= pls+plm+plt;

float psgivent,pmgivent,ptagivent;

psgivent=pmgivent=ptagivent=0;

psgivent = pls/ptotal;

pmgivent = plm/ptotal;

ptagivent= plt/ptotal;

if(psgivent>pmgivent&&psgivent>ptagivent)

classn="short";

else

if(pmgivent>psgivent&&pmgivent>ptagivent)

classn="medium";

else

classn="tall";

}

else if(egender.equals("female"))

{

if(height<=1.61)

{

ptgivens = probfs\*ps1;

ptgivenm = probfm\*pm1;

ptgivent = probft\*pt1;

}

else if(height>1.61&&height<=1.71)

{

ptgivens = probfs\*ps2;

ptgivenm = probfm\*pm2;

ptgivent = probft\*pt2;

}

else if(height>1.71&&height<=1.81)

{

ptgivens = probfs\*ps3;

ptgivenm = probfm\*pm3;

ptgivent = probft\*pt3;

}

else if(height>1.81&&height<=1.91)

{

ptgivens = probfs\*ps4;

ptgivenm = probfm\*pm4;

ptgivent = probft\*pt4;

}

else if(height>1.91&&height<=2.0)

{

ptgivens = probfs\*ps5;

ptgivenm = probfm\*pm5;

ptgivent = probft\*pt5;

}

else if(height>2.0)

{

ptgivens = probfs\*ps6;

ptgivenm = probfm\*pm6;

ptgivent = probft\*pt6;

}

float pls,plm,plt,ptotal;

pls=plm=plt=ptotal=0;

pls=ptgivens\*pshort;

plm=ptgivenm\*pmedium;

plt=ptgivent\*ptall;

ptotal= pls+plm+plt;

float psgivent,pmgivent,ptagivent;

psgivent=pmgivent=ptagivent=0;

psgivent = pls/ptotal;

pmgivent = plm/ptotal;

ptagivent= plt/ptotal;

if(psgivent>pmgivent&&psgivent>ptagivent)

classn="short";

else

if(pmgivent>psgivent&&pmgivent>ptagivent)

classn="medium";

else

classn="tall" ;

}

System.out.println("\nClass Of Entered Tuple Is "+classn);

Statement inst = connect.createStatement();

inst.executeUpdate("insert into personal values ("+eid+",'" + name + "','" + egender + "'," + height + ",'" + classn + "')");

connect.close();

}

catch (Exception e) { System.out.println(e); }

} }

Database for Naive-bayesian Classification

===========================================

Pid Name Gender Height class

1 A female 1.6 short

2 B male 2 tall

3 C female 1.9 medium

4 D female 1.88 medium

5 E female 1.7 short

6 F male 1.85 medium

7 G female 1.6 short

8 H male 1.7 short

9 I male 2.2 tall

10 J male 2.1 tall

11 K female 1.8 medium

12 L male 1.95 medium

13 M female 1.9 medium

14 N female 1.8 medium

15 O female 1.75 medium

**OUTPUT:**

Program for Bayesian Clasification

Enter The Data : Id,Name,Gender,Height

16

P

male

1.95

Class Of Entered Tuple Is tall.