山东大学 软件 学院

众智科学与网络化产业 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：201800301249 | 姓名： 王帅 | | 班级： 计软18.4 |
| 实验题目：计算聚集系数和邻里重叠度 | | | |
| 实验学时：4 | | 实验日期： 2020.3.6 | |
| 实验目的：  熟悉聚集系数、邻里重叠度 | | | |
| 硬件环境：  Pc机 | | | |
| 软件环境：  Dev C++ | | | |
| 实验步骤与内容：  输入：任意图的邻接矩阵  输出：  1）每个节点的聚集系数  2）每个节点对的邻里重叠度  代码如下：  #include<iostream>  using namespace std;  int main()  {  int n; //顶点数  int i,j,k;  cout<<"请输入图的顶点数：";  cin>>n;  int g[n][n]; //邻接矩阵  cout<<"请输入邻接矩阵："<<endl;  for(i=0;i<n;i++)  {  for(j=0;j<n;j++)  {  cin>>g[i][j];  }  }  //聚集系数：节点A的聚集系数 = A的任意两个朋友之间也是朋友的概率（即邻居间朋友对的个数除以总对数）  cout<<endl<<"每个结点的聚集系数："<<endl;  int benext[n]; //储存与当前节点相邻的节点  float groupnum; //相邻节点共有groupnum组  float friendnum; //相邻节点共有friendnum对朋友  int t=0; //相邻节点数  float convergencefactor; //聚集系数  for(i=0;i<n;i++)  {  for(j=0;j<n;j++){benext[j]=0;} //重置相邻节点数组  t=0;  for(j=0;j<n;j++)  {  if(g[i][j]==1) //所有相邻节点写入数组benext，一共有t个  {  benext[t]=j;  t++;  }  }  if(t>1)  {  groupnum = t\*(t-1)/2;  friendnum = 0;  for(j=0;j<t;j++)  {  for(k=0;k<t;k++)  {  if(g[benext[j]][benext[k]]==1) friendnum++;  }  }  friendnum = friendnum/2;  convergencefactor = friendnum/groupnum;  cout<<"顶点 "<<i<<" 的聚集系数："<<convergencefactor<<endl;  }  if(t==0||t==1){cout<<"顶点 "<<i<<" 的聚集系数：0"<<endl;}  }  //邻里重叠度：与A、B均为邻居的节点数/ 与节点A、B中至少一个为邻居的节点数  cout<<endl<<"每个结点对的邻里重叠度："<<endl;  float allneighbor; //都为邻居  float leastneighbor; //至少一个为邻居  float NeighborhoodOverlap; //邻里重叠度  for(i=0;i<n;i++)  {  for(j=i+1;j<n;j++)  {  allneighbor=0;  leastneighbor=0;  for(k=0;k<n;k++)  {  if(k!=i&&k!=j&&(g[i][k]==1&&g[j][k]==1)) allneighbor++;  if(k!=i&&k!=j&&(g[i][k]==1||g[j][k]==1)) leastneighbor++;  }  if(leastneighbor!=0)  {  NeighborhoodOverlap = allneighbor/leastneighbor;  cout<<"顶点 "<<i<<" 和顶点 "<<j<<" 的邻里重叠度为："<<NeighborhoodOverlap<<endl;  }  if(leastneighbor==0)  {  cout<<"顶点 "<<i<<" 和顶点 "<<j<<" 除了对方没有其他邻居"<<endl;  }  }  }  return 0;  }  运行结果：    输入一个邻接矩阵后，输出了每个节点的聚集系数和邻里重叠度。 | | | |
| 结论分析与体会：  本次实验让我上课所学到的知识与熟悉的软件编程结合起来，即加强了我的编程能力，又加深了我对聚集系数和邻里重叠度这两个概念的理解，更让我感受到了众智科学与网络化产业这门课程的实用性。 | | | |