LUP分解求行列式

by zyb

2018 年 12 月 24 日

行列式

- 设矩阵$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} \end{bmatrix}$

行列式

- 设矩阵$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} \end{bmatrix}$
- $|A| = \sum_{(p_1,p_2,\ldots,p_n)} (-1)^{\tau(p_1,p_2,\ldots,p_n)} a_{(1,p_1)} a_{(2,p_2)} \cdots a_{(n,p_n)}$
  $p_1, p_2, \ldots, p_n$为一个排列

性质

- 若$A \xrightarrow{r_i \leftrightarrow r_j} B, (i \neq j)$，则$|A| = -|B|$。

性质

- 若$A \xrightarrow{r_i \leftrightarrow r_j} B, (i \neq j)$，则$|A| = -|B|$。
- 若$A \xrightarrow{k \times r_i + r_j \to r_j} B, (i \neq j)$，则$|A| = |B|$。

性质

- 若$A \xrightarrow{r_i \leftrightarrow r_j} B,(i \neq j)$，则$|A| = -|B|$。
- 若$A \xrightarrow{k \times r_i + r_j \rightarrow r_j} B,(i \neq j)$，则$|A| = |B|$。
- 若A为上三角或下三角矩阵，则$|A| = \prod_{i=1}^{n} a_{i,i}$

性质

- 若$A \xrightarrow{r_i \leftrightarrow r_j} B, (i \neq j)$，则$|A| = -|B|$。
- 若$A \xrightarrow{k \times r_i + r_j \to r_j} B, (i \neq j)$，则$|A| = |B|$。
- 若A为上三角或下三角矩阵，则$|A| = \prod_{i=1}^{n} a_{i,i}$
- $|AB| = |A| \times |B|$

LUP分解

- 对于矩阵A，它的一个LUP分解即

LUP分解

- 对于矩阵A，它的一个LUP分解即
- 一个置换矩阵P，一个单位下三角矩阵L，一个上三角矩阵U

LUP分解

- 对于矩阵A，它的一个LUP分解即
- 一个置换矩阵P，一个单位下三角矩阵L，一个上三角矩阵U
- 满足$PA = LU$

伪代码

```
LUP-DECOMPOSITION(A)
n = A.rows
let π[1..n] be a new array
for i = 1 to n
    π[i] = i
for k = 1 to n
    p = 0
    for i = k to n
        if |a_ik| > p
            p = |a_ik|
            k' = i
    if p == 0
        error "singular matrix"
    exchange π[k] with π[k']
    for i = 1 to n
        exchange a_ki with a_k'i
    for i = k + 1 to n
        a_ik = a_ik / a_kk
        for j = k + 1 to n
            a_ij = a_ij - a_ik a_kj
```

主要思想

- 根据行列式的性质，用初等变换把矩阵变成上三角矩阵

主要思想

- 根据行列式的性质，用初等变换把矩阵变成上三角矩阵
- 上三角矩阵的行列式就等于对角线元素的乘积

代码

```
inline double gauss() {
    double ans=1;
    for (int k=1;k<=n;k++) {
        double p=0;
        for (int i=k;i<=n;i++)
            if (abs(a[i][k])>p) p=abs(a[i][k]),k1=i;
        if(p==0) return 0;
        if (k1!=k) {
            for (int i=1;i<=n;i++) swap(a[k][i],a[k1][i]);
            ans*=-1;
        }
        ans*=a[k][k];
        for (int i=k+1;i<=n;i++) {
            double tmp=a[i][k]/a[k][k];
            a[i][k]=0;
            for (int j=k+1;j<=n;j++) a[i][j]-=tmp*a[k][j];
        }
    }
    return ans;
}
```

主要思想

- 对于矩阵A，求出它的一个LUP分解。

主要思想

- 对于矩阵A，求出它的一个LUP分解。
- $|A| = \frac{|L| \times |U|}{|P|}$

主要思想

- 对于矩阵A，求出它的一个LUP分解。
- $|A| = \frac{|L| \times |U|}{|P|}$
- 因为L是单位下三角矩阵，所以$|L| = 1$

主要思想

- 对于矩阵A，求出它的一个LUP分解。
- $|A| = \frac{|L| \times |U|}{|P|}$
- 因为L是单位下三角矩阵，所以$|L| = 1$
- 在LUP分解过程中，如果交换次数为偶数次，则$|P| = |I| = 1$，否则$|P| = -|I| = -1$

代码

```
inline double LUP() {
    double ans=1;
    for (int i=1;i<=n;i++) pi[i]=i;
    for (int k=1;k<=n;k++) {
        double p=0;
        for (int i=k;i<=n;i++)
            if (abs(a[i][k])>p) p=abs(a[i][k]),k1=i;
        if(p==0) return 0;
        if (k1!=k) {
            for (int i=1;i<=n;i++) swap(a[k][i],a[k1][i]);
            ans*=-1;
            swap(pi[k],pi[k1]);
        }
        ans*=a[k][k];
        for (int i=k+1;i<=n;i++) {
            double tmp=a[i][k]/a[k][k];
            a[i][k]=tmp;
            for (int j=k+1;j<=n;j++) a[i][j]-=tmp*a[k][j];
        }
    }
    return ans;
}
```

行列式在图论中的应用

- 求二分图完美匹配个数的奇偶性

行列式在图论中的应用

- 求二分图完美匹配个数的奇偶性
- 等价于求邻接矩阵行列式的奇偶性

矩阵树定理

- 对于无向图G，求它的生成树个数。

矩阵树定理

- 对于无向图G，求它的生成树个数。
- 对于无向图G，定义矩阵C为它的度数矩阵—邻接矩阵，即对于一条边(u,v), $C_{u,u}++$, $C_{v,v}++$, $C_{u,v}--$, $C_{v,u}--$

矩阵树定理

- 对于无向图G，求它的生成树个数。
- 对于无向图G，定义矩阵C为它的度数矩阵—邻接矩阵，即对于一条边(u, v), $C_{u,u} + +, C_{v,v} + +, C_{u,v} - -, C_{v,u} - -$
- 生成树个数等于C的任意一个余子式的绝对值。

判断图是否存在完美匹配

- 对于图G，定义矩阵$A(G)$，对于G的每一条无向边(u, v)，则$A(G)_{u,v} = x_{u,v} A(G)_{v,u} = -x_{u,v}$

判断图是否存在完美匹配

- 对于图G，定义矩阵$A(G)$，对于G的每一条无向边(u,v)，则$A(G)_{u,v} = x_{u,v} A(G)_{v,u} = -x_{u,v}$
- $x_{u,v}$是只与边(u,v)有关的未知数。

判断图是否存在完美匹配

- 对于图G，定义矩阵$A(G)$，对于G的每一条无向边(u,v)，则$A(G)_{u,v}=x_{u,v}A(G)_{v,u}=-x_{u,v}$
- $x_{u,v}$是只与边(u,v)有关的未知数。
- 如果$|A(G)|\equiv 0$，则G不存在完美匹配。

证明

- 图G的一个环覆盖是用若干个环去覆盖G的所有节点，且每个节点恰好属于一个环

证明

- 图G的一个环覆盖是用若干个环去覆盖G的所有节点，且每个节点恰好属于一个环
- 图G的一个偶环覆盖是用若干个长度为偶数的环去覆盖G的所有节点，且每个节点恰好属于一个环

证明

- 图G的一个环覆盖是用若干个环去覆盖G的所有节点，且每个节点恰好属于一个环
- 图G的一个偶环覆盖是用若干个长度为偶数的环去覆盖G的所有节点，且每个节点恰好属于一个环
- G存在一个完美匹配，当且仅当G存在偶环覆盖：

证明

- 图G的一个环覆盖是用若干个环去覆盖G的所有节点，且每个节点恰好属于一个环
- 图G的一个偶环覆盖是用若干个长度为偶数的环去覆盖G的所有节点，且每个节点恰好属于一个环
- G存在一个完美匹配，当且仅当G存在偶环覆盖：
- => 一个完美匹配相当于用若干个长度为2的偶环覆盖了所有节点

证明

- 图G的一个环覆盖是用若干个环去覆盖G的所有节点，且每个节点恰好属于一个环
- 图G的一个偶环覆盖是用若干个长度为偶数的环去覆盖G的所有节点，且每个节点恰好属于一个环
- G存在一个完美匹配，当且仅当G存在偶环覆盖：
- $=>$ 一个完美匹配相当于用若干个长度为2的偶环覆盖了所有节点
- $<=$ 如果存在一个偶环覆盖，对于每一个偶环，取出第$1, 3, \ldots, 2k-1$条边，这些边构成了一个完美匹配。

证明

- 设点数为n，n的任意一个排列P，都相当于一个G的一个环覆盖

证明

- 设点数为n，n的任意一个排列P，都相当于一个G的一个环覆盖
- 如果(i, P_i)这条边不存在，则对行列式的贡献为0

证明

- 设点数为n，n的任意一个排列P，都相当于一个G的一个环覆盖
- 如果(i, P_i)这条边不存在，则对行列式的贡献为0
- 如果这个环覆盖中存在一个奇环，则对行列式的贡献为0

证明

- 设点数为n，n的任意一个排列P，都相当于一个G的一个环覆盖
- 如果(i, P_i)这条边不存在，则对行列式的贡献为0
- 如果这个环覆盖中存在一个奇环，则对行列式的贡献为0
- 如果$|A(G)| \equiv 0$，则说明不存在偶环覆盖，即不存在完美匹配

做法

- $|A(G)|$是一个多项式，取值与所有的$x_{u,v}$有关

做法

- $|A(G)|$是一个多项式，取值与所有的$x_{u,v}$有关
- 考虑随机化，对所有的未知数随机赋值，在一个大质数模域下判断行列式是否等于0

做法

- $|A(G)|$是一个多项式，取值与所有的$x_{u,v}$有关
- 考虑随机化，对所有的未知数随机赋值，在一个大质数模域下判断行列式是否等于0
- 若每次随机，得到的行列式均为0，则可以认为$|A(G)| \equiv 0$