

# 机器学习4

## 1 Normal Equation Noninvertibility

When implementing the normal equation in octave we want to use the 'pinv' function rather than 'inv.' The 'pinv' function will give you a value of  $\theta$  even if  $X^T X$  is not invertible.

If  $X^T X$  is **noninvertible**, the common causes might be having :

- Redundant features, where two features are very closely related (i.e. they are linearly dependent)
- Too many features (e.g.  $m \leq n$ ). In this case, delete some features or use "regularization" (to be explained in a later lesson).

Solutions to the above problems include deleting a feature that is linearly dependent with another or deleting one or more features when there are too many features.

## 2 Octave/Matlab Tutorial

### 2.1 Basic Operations

`xor(x,y)`:x和y中有且仅有一个值为真，则返回真。

`PS1(' >> ');`取消前面的提示

如果你想分配一个变量，但是不希望在屏幕上显示结果，你可以在命令后加一个分号。

```
>>a=pi;  
>>disp(sprintf('2 decimals:%0.2f',a))  
#sprintf命令生成一个字符串，'%0.2f'意味着将a放在这里并显示a值的小数点后两位数字。
```

format long:默认情况下是字符串。显示出的小数点位有点多。

format short:是默认的输出格式，只是打印小数位数的第一位。

A=[1 2;3 4;5 6]会产生一个三行两列的矩阵A，；的作用就是换行到下一行。

v=1:0.1:2

可以得到一个行向量，从1开始，增量为0.1，增加到2

```
>>ones(2,3)
#也可以用来生成矩阵，结果为一个两行三列的矩阵，不过矩阵中的所有元素都为1
```

```
ans = 1 1 1 1 1 1
```

```
>>c=2*ones(2,3)
#当我想生成一个元素为2的两行三列的矩阵
>>w=randn(1,3)
#则生成一行三列的元素均为随机值的行向量
>>hist(w)
#绘制直方图的命令
>>eye(4)
#生成一个4行4列单位阵
```

## 2.2 Moving Data Around

```
>>size(A)
#打印出矩阵A的行列
>>size(A,1)
#打印出矩阵A的行数
>>size(A,2)
#打印出矩阵A的列数
>>length(A)
#打印出A最大维度的大小
>>pwd
#得到octave的安装位置
>>cd 'C:\Users\ang\Desktop'
#把路径改为
>>load featuresX.dat
#我将加载了featuresX文件
```

```

>>who
#显示出在当前空间的所有变量
>>featuresX
#显示出所有数据
>>size(featuresX)
#代表是一个几行几列的矩阵
>>whos
#列出了变量名字还列出了变量的维度，还显示出需要占用多少内存空间以及数据类型是什么。
>>clear featuresX
#清除featuresX
>>v=priceY(1:10)
#表示将向量v的前十个元素存入v中
>>save hello.mat v;
#会将变量v存成一个叫hello.mat的文件
>>save hello.txt v -ascii
#存成一个文本文档或者将数据的ascii码存成文本文档
>>A(3,2)
#打印出A的三行两列的元素
>>A(2,:)
#打印出第二行的所有元素，: 表示该行或该列的所有元素
>>A(:,2)
#打印出A矩阵第二列的所有元素
>>A([1 3],:)
#取A矩阵第一个索引值为1或3的元素
>>A(:,2)
#返回第二列
>>A(:,2)=[10;11;12]
#取A矩阵第二列并给辅助10,11,12
>>A=[A,[100; 101; 102]];
#在A矩阵的右边附加了一个新的列矩阵
>>A(:)
#把A中的所有元素放入一个单独的列向量
>>C=[A; B]
#把分号后面的东西放在下面
\subsection{Computing on Data}
\paragraph{}
\begin{minted}{octave}
>>A*C
>>A.*B
#对应位元素相乘
>>A.^2
#对矩阵A中的每一个元素平方
>>1./V
#得到v中每一个元素的倒数
>>log(V)
#对每一个元素进行求对数运算

```

```

>>exp(V)
#以e为底，以V中元素为幂的运算。
>>abs(V)
#对V中的每一个元素求绝对值
>>V+ones(length(V).1)
#将V的各元素都加上这些1.
>>V+1
#把V中的每一个元素都加上1
>>B=A'
#求A的转置
>>val=max(a)
#返回a中的最大值
>>[val,ind]=max(a)
#将返回a矩阵中的最大值存入val，以及对该值对应的索引，对应的索引存入ind
>>B=max(A)
#A是一个矩阵的话，这样做就是对每一列求最大值
>>a=[1 15 2 0.5]
>>a<3
#将进行逐元素的运算，所以元素小于3的返回1，否则返回0
>>find(a<3)
#将告诉我a中的哪些元素是小于3的
>>A=majic(3)
#majic将返回一个矩阵，成为魔方阵或幻方，它们所有的行和列和对角线加起来都等于相同的值。
>>[r,c]=find(A>=7)
#将找出所有A矩阵中大于等于7的元素

```