# 1 推荐系统简介

## 一 概念

目的：达到长尾效应，成就多样性，（精细度）消除马太效应

基于信任度基础之上产生的推荐才是有效的

基于人口统计学：用户画像标签

基于内容推荐：资料画像标签

基于协同过滤推荐（Collaborative Filtering）

：

1 基于近邻的协同过滤

{  
 基于用户

基于资料

}

2 基于模型的协同过滤（机器学习）

混合推荐：

加权混合、切换混合、分区混合、分层混合

## 二 评测

### 1 方法

1. 离线实验：训练数据和测试数据进行测试
2. 用户调查：调查问卷
3. 在线实验：AB测试

### 2 评测指标

准确率：正确分类的百分比占比

精确率：召回数据中用户满意数据所占的比例

召回率：所有召回的占比满意的数据（应该检索到的数据）的比例

一般也用精确度/召回率表示算法的精度

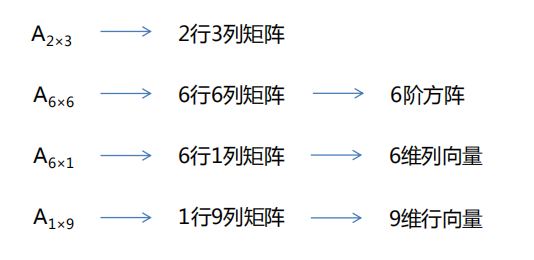
覆盖率：能够发掘长尾能力（所有商品都被推荐给用户）

混淆矩阵：



# 二 数学基础

## 1 矩阵



只有一行的矩阵A1\*n称为行矩阵，也叫行向量

只有一列的矩阵称为列矩阵，也叫列向量

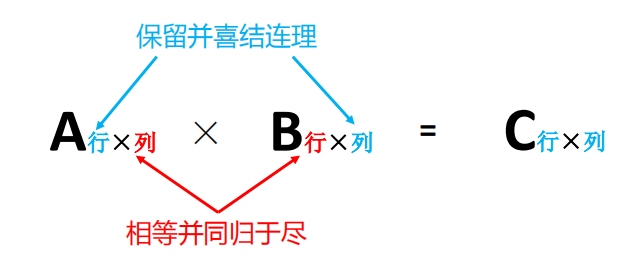
主对角线：左上角到右下角的直线

### 特殊矩阵：

1. 零矩阵：矩阵的元素全部全都为0的（O表示）。
2. 单位矩阵：主对角线上的元素为1，其余元素为0 用E或者I表示
3. 对角矩阵：只有在主对角线上的元素不为0，其余元素都为0的矩阵

### 矩阵的乘法：

矩阵与矩阵的相乘：一行乘一列，行定列移动，列尽下一行



### 矩阵的转置

把矩阵中的行按照相同的序列转成列，得到一个新的矩阵，AT

### 逆矩阵

AB=E(单位矩阵)

# 二 微积分

### 导数

导数是指在一元函数中，函数在某一点沿x轴正方向的变化率。

### 偏导数

偏导数指的是多元函数中，函数在某一点沿着某一坐标轴正方向的变化率

### 方向导数

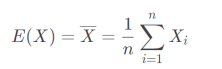
函数在某一点在某一趋近方向上的导数值（函数在除坐标轴正方向外，其他特定方向上的变化率）

### 梯度函数

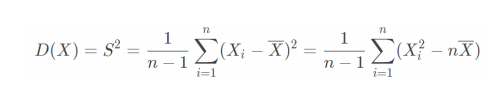
函数在某一点的梯度是这样一个向量，他的方向与取得最大方向导数的方向一致，而他的模为方向导数的最大值。（梯度函数是一个向量既有方向、又有大小）

## 三 概率统计基础知识

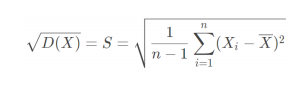
### 样本均值



### 无偏样本方差



### 样本标准差

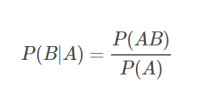


### 常见概率分布

均匀分布、正态分布（高斯分布）、指数分布

### 重要概率公式

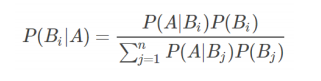
条件概率公式：



全概率公式：



贝叶斯公式：



# 2 机器学习

## 一 分类

1、有监督学习：提供数据并提高数据对应结果的机器学习过程。分类和回归

2、无监督学习：提供数据冰并且不提供数据对应结果的的机器学习过程。（聚类分析、密度估计）---谷歌新闻分类推荐

3、强化学习：通过与环境交互并获取延迟返回进而改进行为的学习过程

4 监督学习的三要素

模型：总结数据的内在规律，用数学函数描述的系统

策略：选取最优模型的评价准则

算法：选取最优模型的具体方法

**5 损失函数（loss function） L(y,f(x))**

损失函数用来衡量模型预测误差的大小,损失函数越小，模型就越好

一般常用平方损失函数、除了这些还有绝对值损失函数、对数损失函数、0-1损失函数

6 经验风险

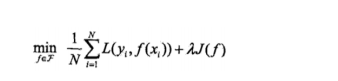
经验风险是指训练数据集的平均损失风险。

**经验风险最小化：ERM（Empirical Risk Minimization）**

7 过拟合和欠拟合（训练误差和测试误差）

欠拟合：学习的不够（泛华能力好）

过拟合：泛华能力差

**8 结构风险最小化：在erm 经验风险最小化的基础之上防止过拟合而提出的策略。**

## 二 数据验证

1 数据验证：

数据量足够多的话：分为三部分，训练集、验证集、测试集

2 验证方法

离线验证、用户调查、在线实验

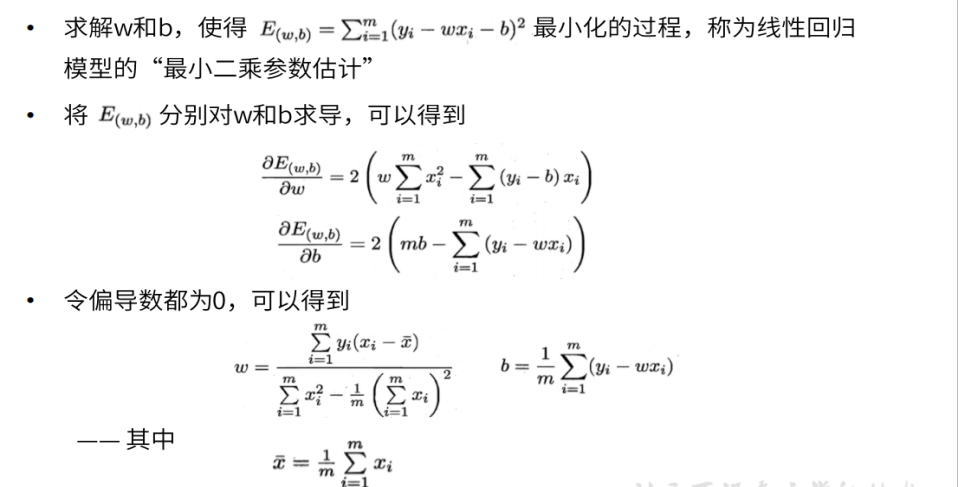
## 三 机器学习模型

### 1 线性回归

#### 最小二乘法参数估计（least square method）

求解w和b，使得损失函数的平方最小化的过程，成为线性回归模型的最小二乘法参数估计。

求解过程： 对不同维度的参数模型进行求偏导，然后联立偏导为0,求解模型参数



#### 多元线性回归（梯度下降法）

相同点：

本质和目标相同，两种方法都是经典的学习算法，在给定已知的数据的前提下利用求导算出一个数据模型，使得损失函数最小，然后对给定的新数据进行估算预测。

不同点：

损失函数：梯度下降可以选取其他损失函数，最小二乘只能是平方损失函数。

## 四 算法之协同过滤

1 基于用户的协同过滤

训练集：【【u,i】】

1. 基于余弦相似度计算两个用户的相似度

伪代码：

相同用户直接跳过，两个用户决定一个key

w[u][v] = len(train[u]&train[v])

W[u][v] /= math.sqrt(len(train[u])\* len(train[v]\*1.0))

策略：由于很多用户之间没有相同的商品行为，多数都浪费在计算|两个交集为0|的数据上了，可以先进行数据过滤筛选出有数据交集的，然后再除分母进行计算。

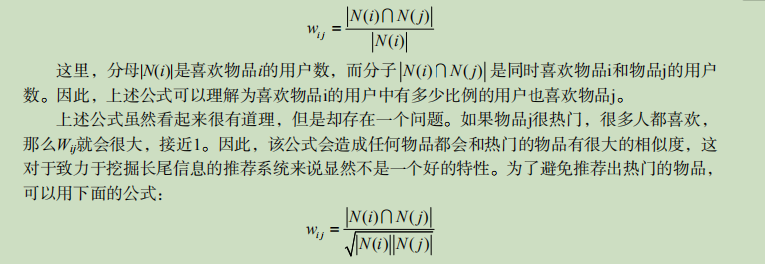
先建立物品到用户的倒排表，对于每个物品都保存对该物品产生行为的用户列表。

假设用户u和用户v同时属于倒排表中K个物品对应的用户列表，就有C[u][v]=K。从而，可以扫描倒排表中每个物品对应的用户列表，将用户列表中的两两用户对应的C[u][v]加1，最终就可以得到所有用户之间不为0的C[u][v]。

2 基于物品的协同过滤（ItemCF）

1 计算物品之间的相似度（同时下载的用户越多，相似度越高），共同被很多用户喜欢

2 根据物品的相似度和用户的历史行为给用户生成推荐列表



被更多的人同时喜欢的商品的相似度高，第二个公式很好的避免了一直推送热门商品。

喜欢i和喜欢j物品人的交集/喜欢i的人乘以喜欢j的人，然后再开方

K（最相似的k个物品）选值：最相似的k个物品 ，k增加会降低覆盖率，在一定程度内随着k的增加流行度会增加，到达一定程度会不再随之增长