机器学习之
——线性回归

主讲：刘丽珏

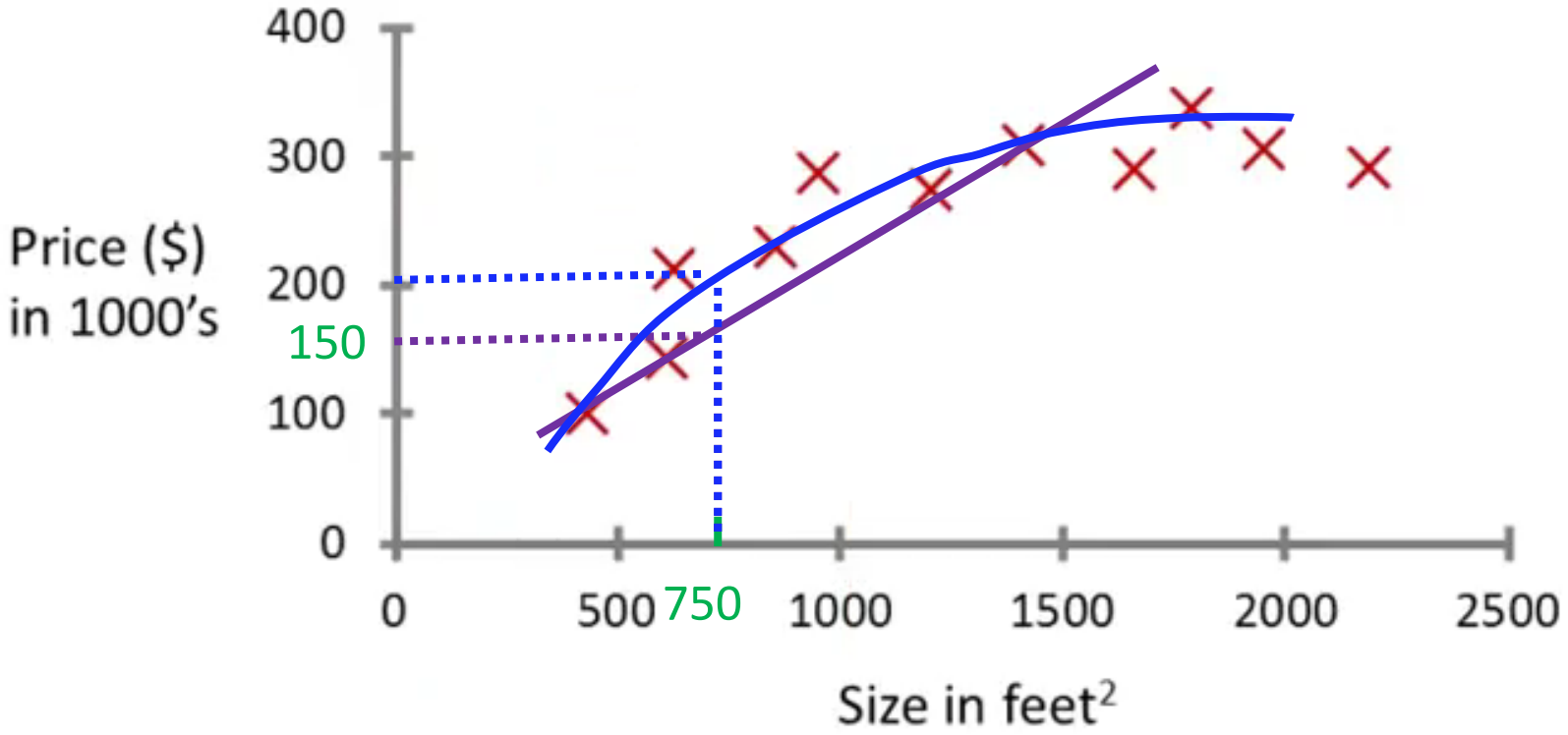
主要内容

- 引言
- 普通线性回归
- 基于梯度下降的线性回归
- 正则化与岭回归

引言

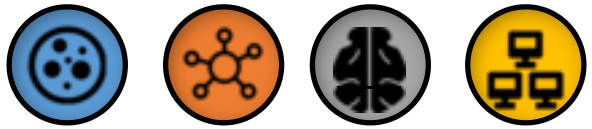
Regression:
预测连续的输出量“价格”

Housing price prediction.



引言

- 有监督学习
  - 从给定输入和输出的训练数据集中学习输入和输出之间的映射函数，然后利用该映射函数预测出测试样本的输出值，其中训练集中的每个样本都由输入和对应的输出(也称之为label)组成(labeled data)
    - 回归(Regression)——预测的目标值是数值型连续变量
    - 分类(Classification)——预测的目标值是离散的



01 线性回归简介

(Linear Regression Overview)

回归分析

- 用方程来表达感兴趣变量（因变量）和一系列相关变量（自变量）之间关系的分析过程
- 研究对象
  - 具有相关关系的变量，即不能用函数刻画，但有一定趋势性关系的变量
- 回归分析模型

$$y = f(x) + \varepsilon$$

其中x是自变量，y是因变量，ε为随机误差

- 通常设随机误差的期望 $E(\varepsilon) = 0$，则称

$$E(y|x) = f(x)$$

为回归函数

回归与线性回归

- 回归分析的目的
  - 根据自变量$x_1,x_2...x_n$以及因变量y的观测样本，去估计两者之间近似的函数关系f
    - 通常，假设函数f的数学形式是已知的，但其中的若干个参数未知
    - 首要问题：通过自变量和因变量的观测样本去估计未知的参数值——**参数回归**
- 线性回归
  - 函数f是线性函数
  - 线性模型是机器学习中最基本的模型

线性回归

- 线性回归方程的一般形式

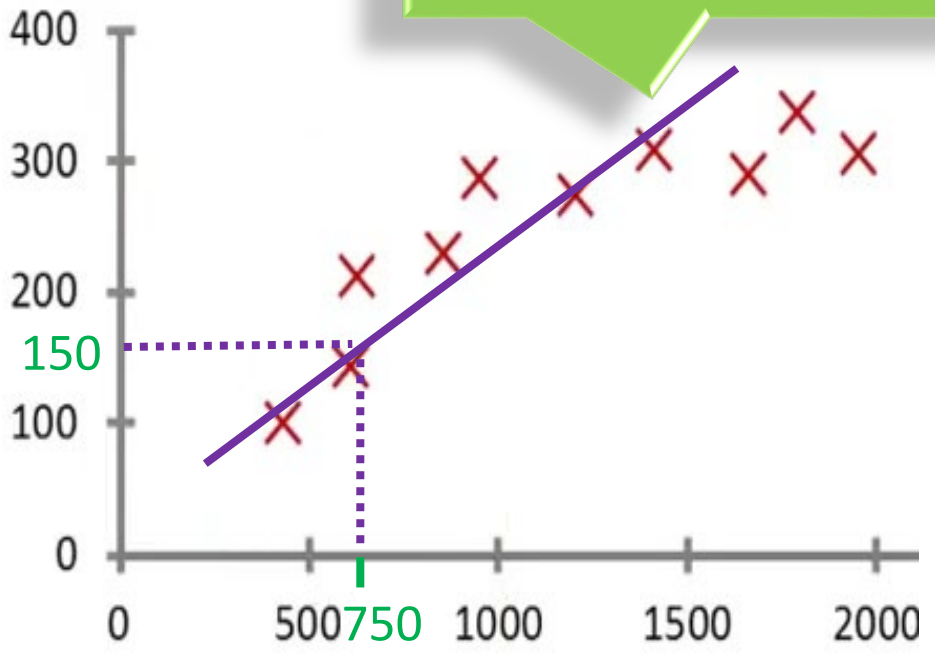
  $\hat{y} = f(x) = w_0 + \sum_{i=1}^{n} x_i w_i$,

  - $x^T = [x_1, x_2, \ldots, x_n]$，$w_i$为回归系数，若$w_i$>0，称$x_i$与$y$正相关，若$w_i$<0，称$x_i$与$y$负相关
  - 通常增广x为$[x_0=1, x_1, x_2, \ldots, x_n]$，则$f(x) = \sum_{i=0}^{n} x_i w_i$
  - x_i的取值
    - 定量输入
    - 定量输入的转换，如log, 平方根，平方等
    - 基础扩展变量，如$x_2=x_1^2$, $x_3=x_1^3$ 等，即多项式表示
    - 表示等级的数值数据或哑元，如用一个5维二进制向量x_i表示一个具有5个等级的输入G，若$G=j$，则$x_{ij} = 1$，其它为0
    - 变量之间的相互作用，如$x_3= x_1 * x_2$

线性回归

- 一元回归
  - 一个自变量，$n=1$
- 多元回归
  - 有两个或两个以上的自变量，$n>1$
- 回归
  - 求回归系数的过程
- 例——房价预测
  - 目标值Price
    - 一元回归问题
  - 计算公式
    - Price=0.2*Size
      - 回归方程
      - 0.2称为回归系数

- 一元线性模型为一条直线
- 多元模型为一个超平面



- 在一元回归分析中，通常会作如上图的散点图，即在以自变量和因变量为x，y坐标的坐标系中绘制(x,y)点对，以根据经验选定回归方程的类型
- 多元回归中也可以用散点图对某两个自变量或某个自变量与因变量进行分析

线性回归

- 多项式回归可转换为多元线性回归
- 例
  - 一元m次多项式回归方程

$y = w_0 + w_1 x + w_2 x^2 + \cdots + w_m x^m$

可转换为m元线性回归方程 $y=w_0+w_1x_1+w_2x_2+\ldots+w_mx_m$

其中$x_i = x^i$

  - 二元二次多项式回归方程

$y = a_0 + a_1 x_1^2 + a_2 x_2^2 + a_3 x_1 x_2 + a_4 x_1 + a_5 x_2$

可转换为5元线性回归方程 $y=w_0+w_1z_1+w_2z_2+w_3z_3+w_4z_4+w_5z_5$

其中$z_1 = x_1^2$，$z_2 = x_2^2$，$z_3 = x_1 x_2$，$z_4 = x_1$，$z_5 = x_2$

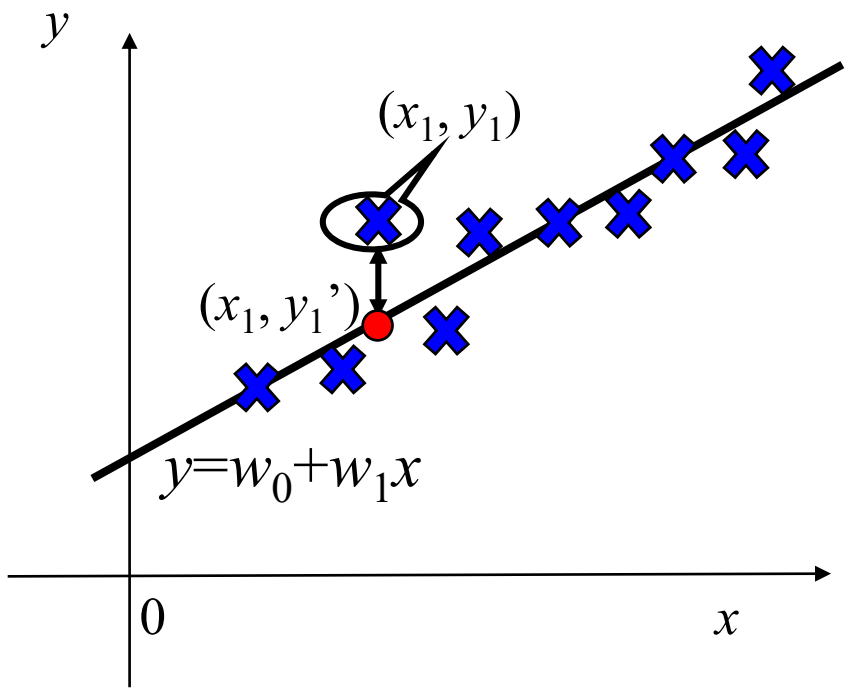
    - 非线性模型

线性回归的一般方法

- 收集数据，采集训练样本，每个训练样本包括特征向量及其对应的期望输出
  - 假设有n个已标记的样本数据，X_i为第i个样本，y_i是该样本给定的期望输出
    - 其中$X_i=[x_0=1, x_1, x_2, \ldots, x_m]$
    - 回归系数$W=[w_0, w_1, \ldots, w_m]^T$
    - 预测值 $\hat{y_i} = X_i W$
  - 设计训练算法找到回归系数
    - 显然，直观的方法，希望输出的期望值与预测值误差最小
- 测试，分析模型的效果
- 使用算法，对给定输入预测一个输出

线性回归的损失函数(Cost function)

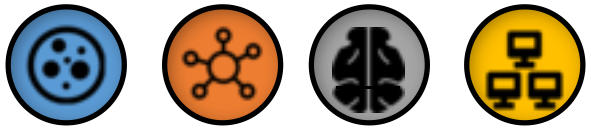
- 以一元回归为例



损失函数：(*Least Squares*,最小二乘）

$$L(W) = \sum_{i=1}^{n}(y_i - \hat{y_i})^2 \quad = \sum_{i=1}^{n}(y_i - X_1W)^2$$

n为样本的数量



02 普通线性回归

(OLR，Ordinary Linear Regression)

正则方程(Normal Equation)

- 对第i个训练样本，期望输出y_i，预测输出$\hat{y}_i$
- 构造损失函数

$$L(W)=\sum_{i=1}^{n}(y_i-\hat{y}_i)^2=\sum_{i=1}^{n}(y_i-X_iW)^2$$

  - 目的：$\min f(W)$
    - 最小二乘法
      - 又称最小平方法，是一种数学优化技术
      - 通过最小化误差的平方和寻找数据的最佳函数匹配
    - 如何求函数的最小值
      - 对W求导（即对每一个w_i求偏导）
      - 当函数为凸函数时，令导数等于0，可求出W的最优估计

正则方程(Normal Equation)

- 损失函数的矩阵表示

$$L(W) = (Y - XW)^T(Y - XW)$$
$$= Y^TY - Y^TXW - W^TX^TY + W^TX^TXW$$

其中$Y = [y_1 \quad y_2 \quad \dots y_n]^T$,

$W = [w_0 \quad w_1 \quad \dots w_m]^T$

$X = [X_1 \quad X_2 \quad \dots X_n]^T$,

$X_i = [x_{i0} \quad x_{i1} \quad \dots x_{im}]$

即$X = \begin{bmatrix} x_{10} & x_{11} & \dots x_{1m} \\ x_{20} & x_{21} & \dots x_{2m} \\ \dots & & \\ x_{n0} & x_{n1} & \dots x_{nm} \end{bmatrix}$

$W^TX^TY = (Y^TXW)^T$
为一个一元矩阵
$W^TX^TY = Y^TXW$

正则方程(Normal Equation)

$$L(W) = Y^TY - Y^TXW - W^TX^TY + W^TX^TXW$$
$$= Y^TY - 2W^TX^TY + W^TX^TXW$$

$$\frac{\partial W^TX^TXW}{\partial W} = (X^TX + (X^TX)^T)W = 2X^TXW$$

- 对W求导

$L(W)' = 2X^TXW - 2X^TY$

令$X^TXW - X^TY = 0$，求出

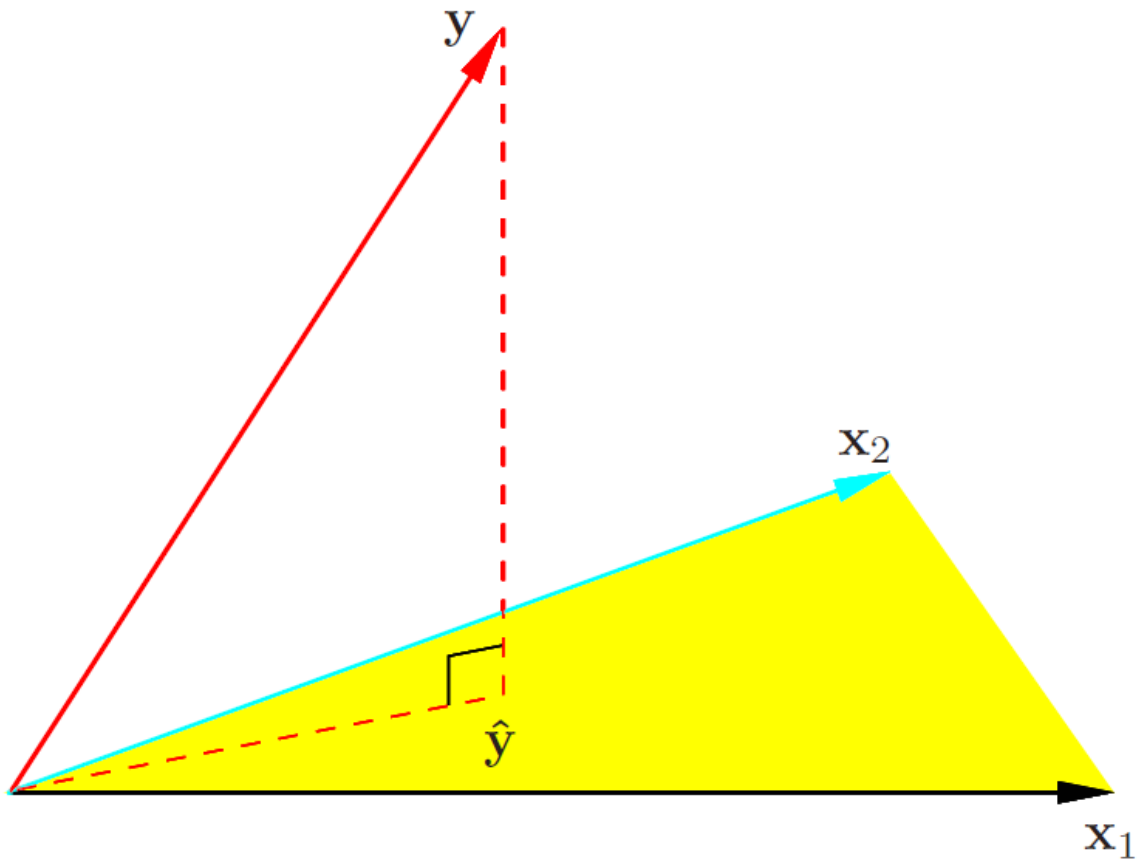
$\hat{W} = (X^TX)^{-1}X^TY$　公式（1）**正则方程**

逆矩阵存在的条件$|X^TX| <> 0$，行列式不等于0

*Hat*矩阵

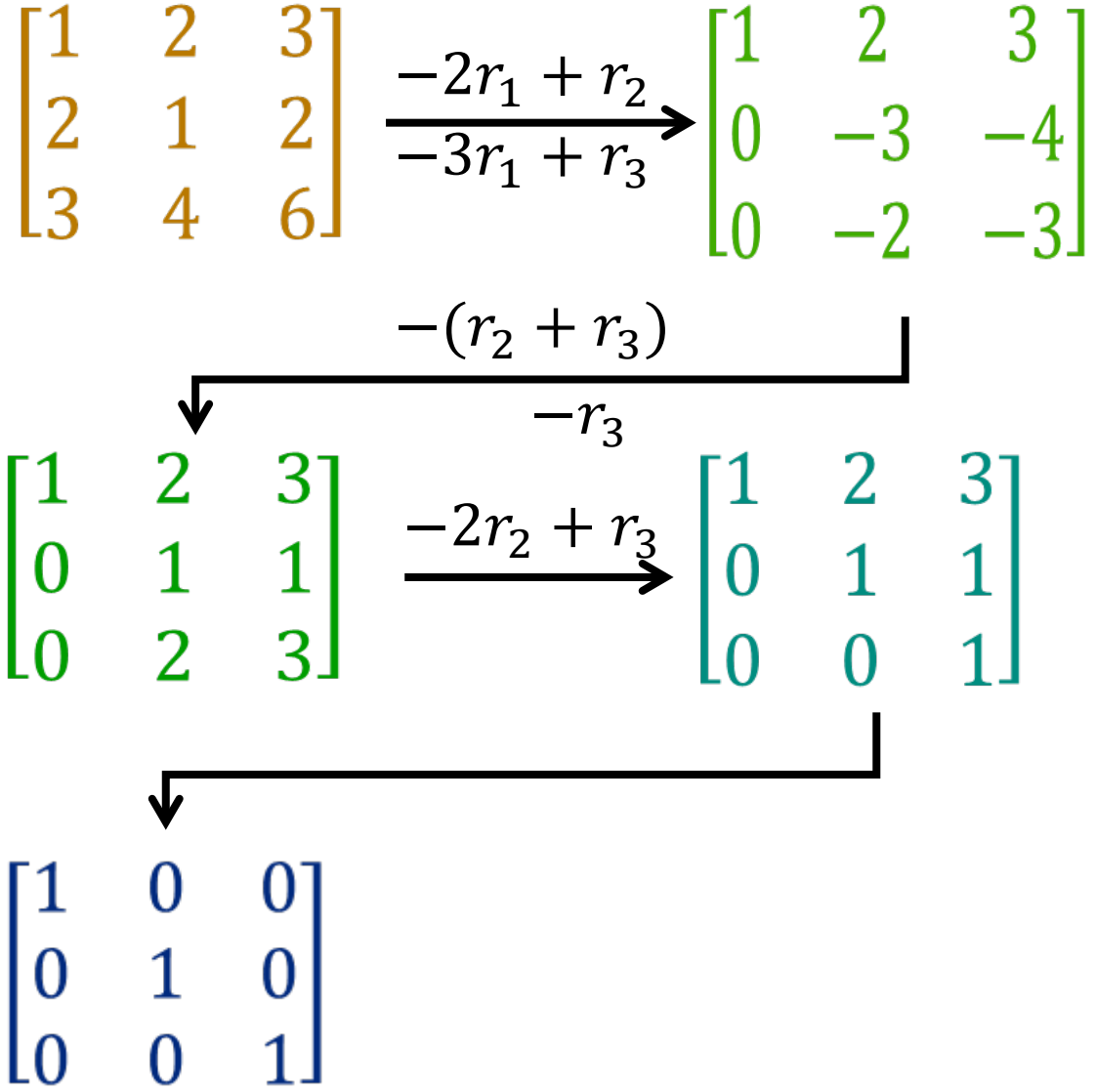
- 根据上述求解的结果，有 $\hat{Y} = X\hat{W} = X(X^TX)^{-1}X^TY$
- X的各列张成(span)一个R^n的子空间(X的列空间)
  - X的列向量的所有线性组合的集合构成的子空间
- 其中$H = X(X^TX)^{-1}X^T$称为*Hat*矩阵
  - Because it puts the *hat* on Y ($\hat{Y}$)
  - *Hat*矩阵是Y到X的列空间的投影矩阵

由$L(W)' = 2X^TXW - 2X^TY = 0$
得$X^T(XW - Y) = 0$
即残差$\hat{Y} - Y$与X^T正交



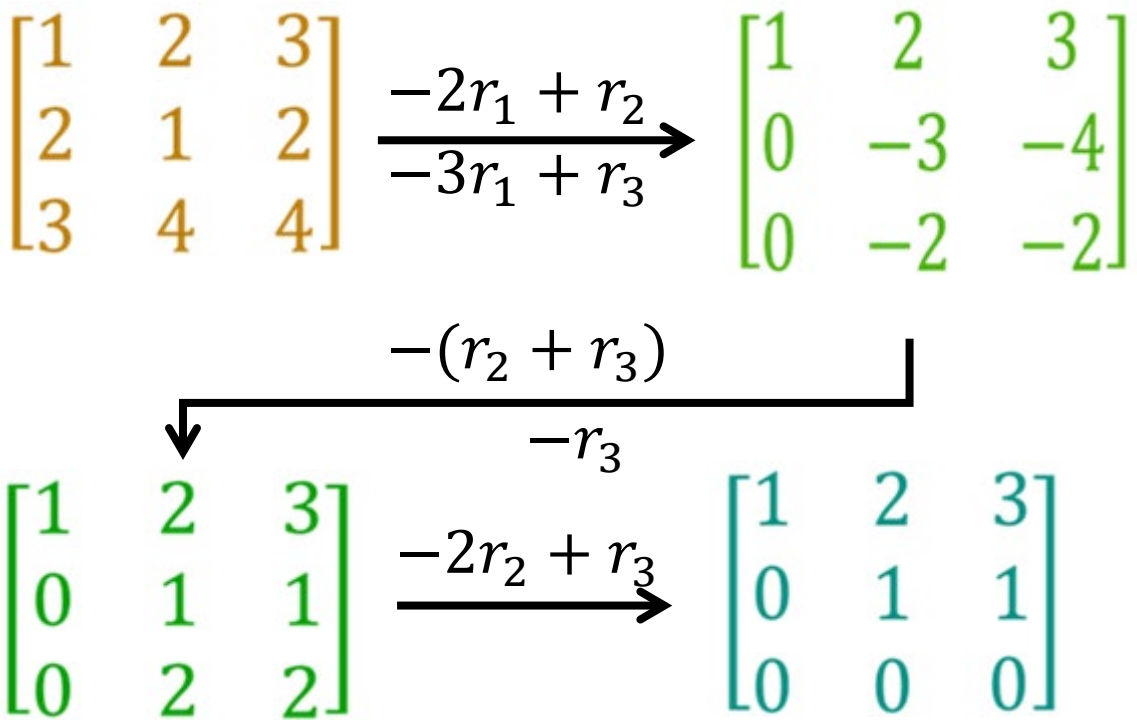
正则方程的问题

- 正则方程中要求X^TX 的逆，即要求X^TX是满秩的（非奇异）
  - 秩——用初等行变换将矩阵A化为阶梯形矩阵，则矩阵中非零行的个数就定义为这个矩阵的秩
  - 初等行变换
    - 某一行，乘以一个非零倍数
    - 某一行，乘以一个非零倍数加到另一行
    - 某两行，互换



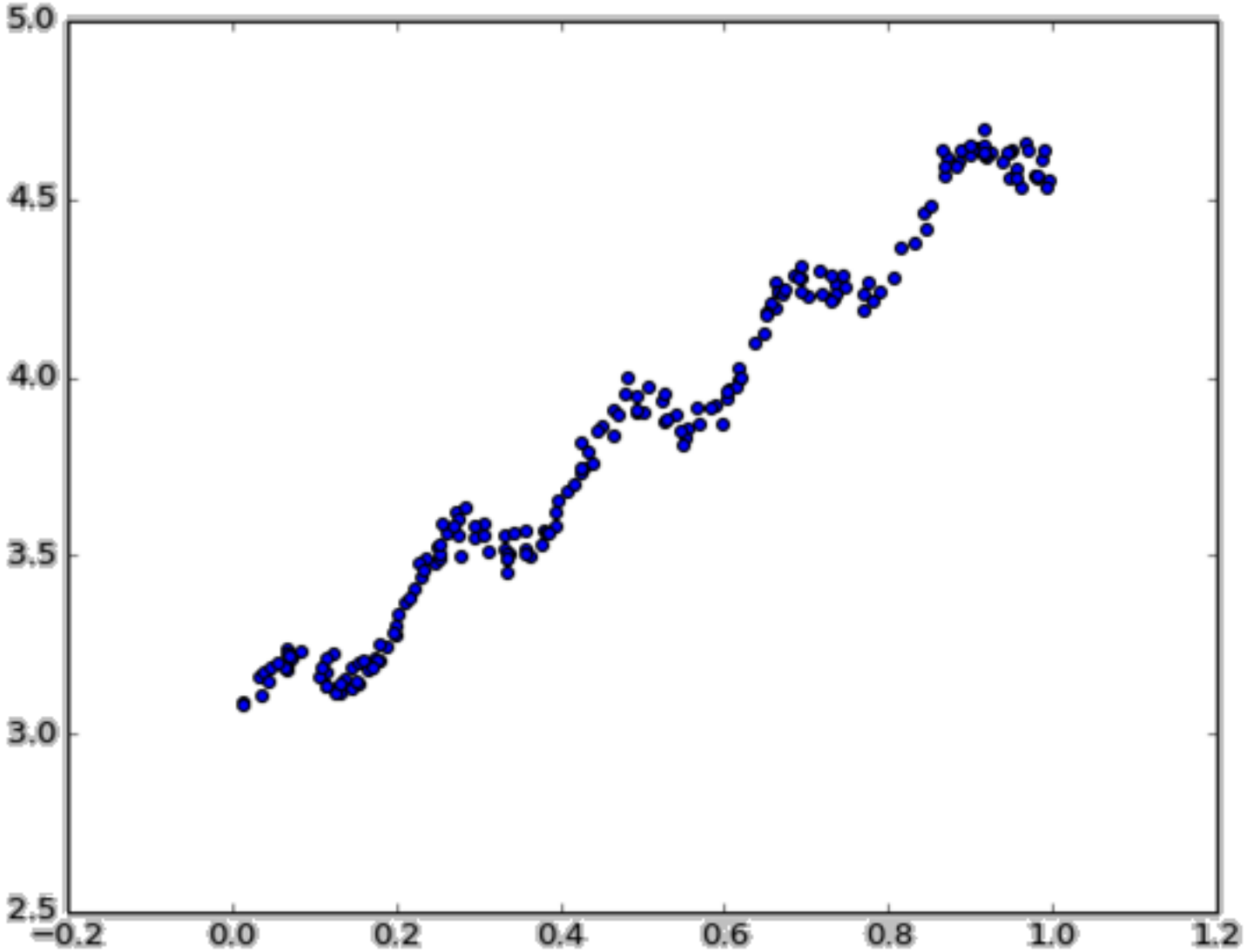
正则方程的问题

- 奇异矩阵意味着什么？
  - 矩阵中存在某些行线性相关
  - 如：$x_2 = 3x_1$
- 这种情况导致$\hat{W}$的解不唯一
- 通常做法
  - 重新编码和/或删除X中的冗余列
- 在信号和图像分析中也可能发生秩亏
  - 输入的特征数量p可能超过训练样本的数量N
    - 通过过滤来减少特征
    - 通过正则化来控制拟合
    - 迭代寻优的方法——梯度下降



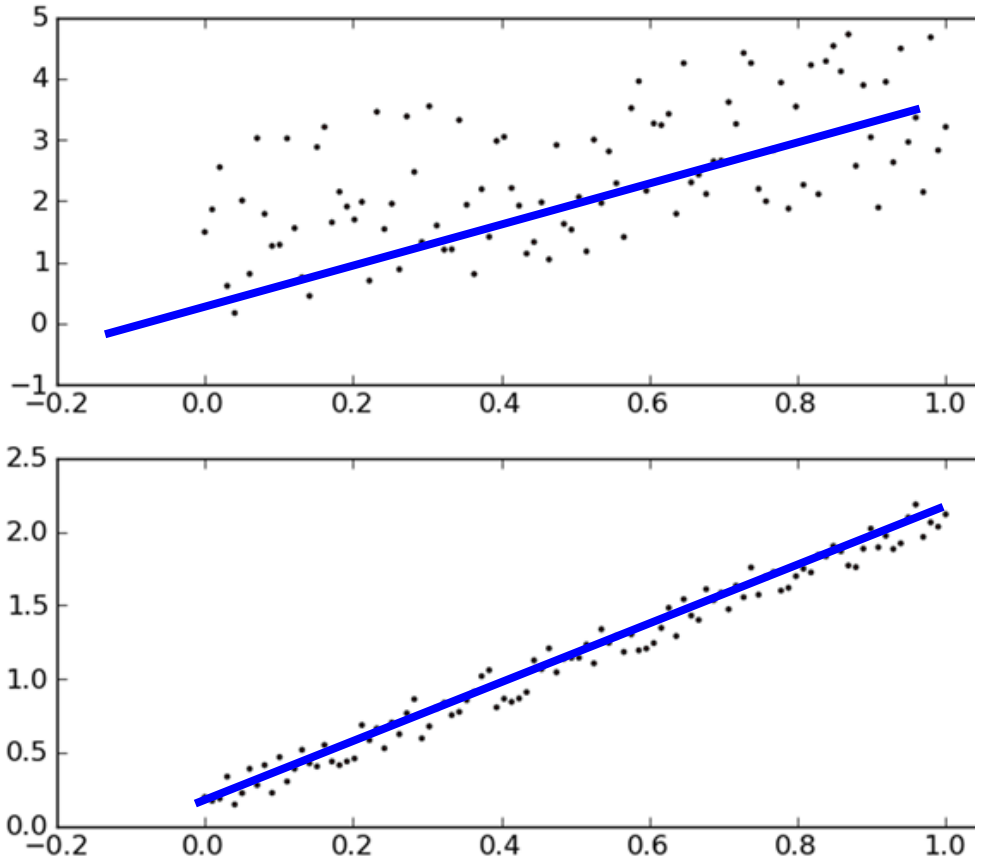
练习

- 对右边的散点图给出最佳拟合直线
  - 对应数据文件——ex0.txt
  - 编程思想
    - 读入数据文件中的数据
    - 建立输入、输出矩阵
    - 根据公式（1）计算回归系数



回归分析的评价指标

- 如何判断模型的优劣？
  - 例：右边两组数据集得到完全相同的回归系数(0, 2.0)
  - 常用检验指标
    - MAE（Mean Absolute Error）平均绝对差值
    - MSE(Mean Square Error)均方误差
    - RMSE(Root Mean Square error)均方根误差
    - 相关系数
    - 拟合优度R^2和调整R^2



回归分析的评价指标

- 设n为样本量，$\hat{y}_i$为预测值，y_i为真实值，则
  - MAE（Mean Absolute Error）平均绝对差值

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i|$$

  - MSE(Mean Square Error)均方误差

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

  - RMSE(Root Mean Square error)均方根误差

$$RMSE = \sqrt{MSE}$$

回归分析的评价指标

- 皮尔逊相关系数R (Pearson's Correlation Coefficient)

$$R(X,Y)=\frac{cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}}$$

其中$cov(X,Y)$是协方差，$\sqrt{D(X)}$是方差

- 显然$R(X,X)=1$，取值范围[-1,1]
- 通过计算$\hat{y}$与y之间的相关系数来衡量优劣
- R=0时，两变量无关系
- 当X的值增大（减小），Y值增大（减小），两个变量为正相关，相关系数在0.00与1.00之间
- 当X的值增大（减小），Y值减小（增大），两个变量为负相关，相关系数在-1.00与0.00之间

R的绝对值	相关程度
0.8-1.0	极强相关
0.6-0.8	强相关
0.4-0.6	中等程度相关
0.2-0.4	弱相关
0.0-0.2	极弱相关或无相关

- 注意：皮尔逊相关系数只能判断线性相关性
- 如：x,y的取值为(-1,-1) (-1, 1), (1,-1), (1,1)，显然$x^2+y^2=2$，但$r(x,y)=0$
- 还可用于判断自变量与因变量的关系，以确定该自变量有没有纳入回归方程的必要

回归分析的评价指标

- 拟合优度
  - 又称判定系数、可决系数，反映了回归方程所能解释的因变量总变异的比例
  - 一元回归中的可决系数

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

其中SSR为回归平方和，SSE为残差平方和，SST为总离差平方和,且

$$SSR = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$$

$$SSE = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2$$

$$SST = SSR + SSE = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

  - 注意：要使$SST = SSR + SSE$成立，回归方程必须包含截距项（$w_0 x_0$，x_0=1）
  - R^2越接近1，说明SSR占总离差的比例越高，即因变量的变化主要由自变量的不同取值造成，回归方程对样本点拟合得越好

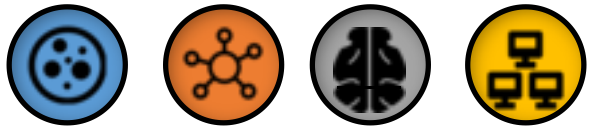
回归分析的评价指标

- 拟合优度
  - 多元回归中当增加自变量时，即使这个自变量在统计上并不显著，SSE也会减少，从而R^2变大
  - 为避免增加自变量而高估R^2，多元回归中用调整R^2来评价拟合优度

$$\text{调整}R^2 = 1 - \frac{n-1}{n-k-1}\frac{SSE}{SST} = 1 - (1-R^2)\frac{n-1}{n-k-1}$$

其中k为自变量的个数

  - 调整后的R方永远小于R方，并且调整R方的值不会由于回归中自变量个数的增加而越来越接近1
  - 0.5为调整R方的临界值
    - 如果调整R方小于0.5，要分析所采用和未采用的自变量
    - 如果调整R方与R方存在明显差异，意味着所用的自变量不能很好的测算因变量的变化
    - 调整R方与R方差距越大，模型的拟合越差



03 基于梯度下降的线性回归

正则方程的问题

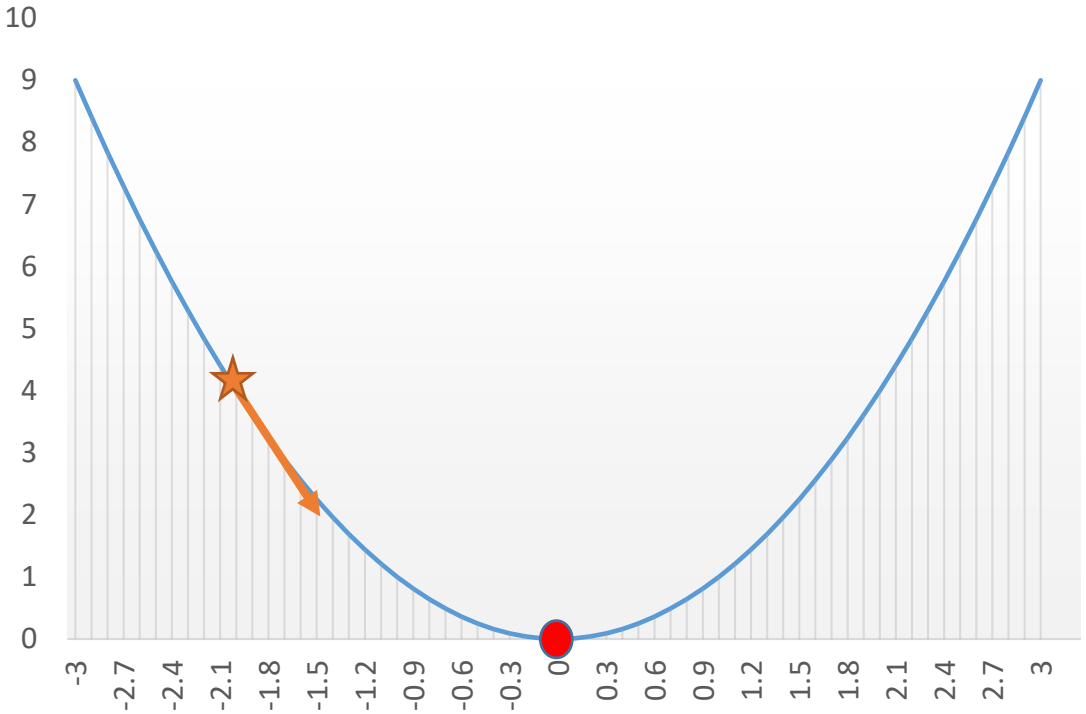
- OLR使用正则方程来求解回归系数

$$\hat{W} = (X^T X)^{-1} X^T Y$$

- 若$X^T X$不可逆，则无法求解
  - 特征数量大于样本数量时也无法求解
- 且若参数维度非常大，解析解计算量过大，显然也不适用
  - 特征数量不超过100000时可用正则方程求解
- 另一种求解回归系数的方法
  - 梯度下降

梯度下降

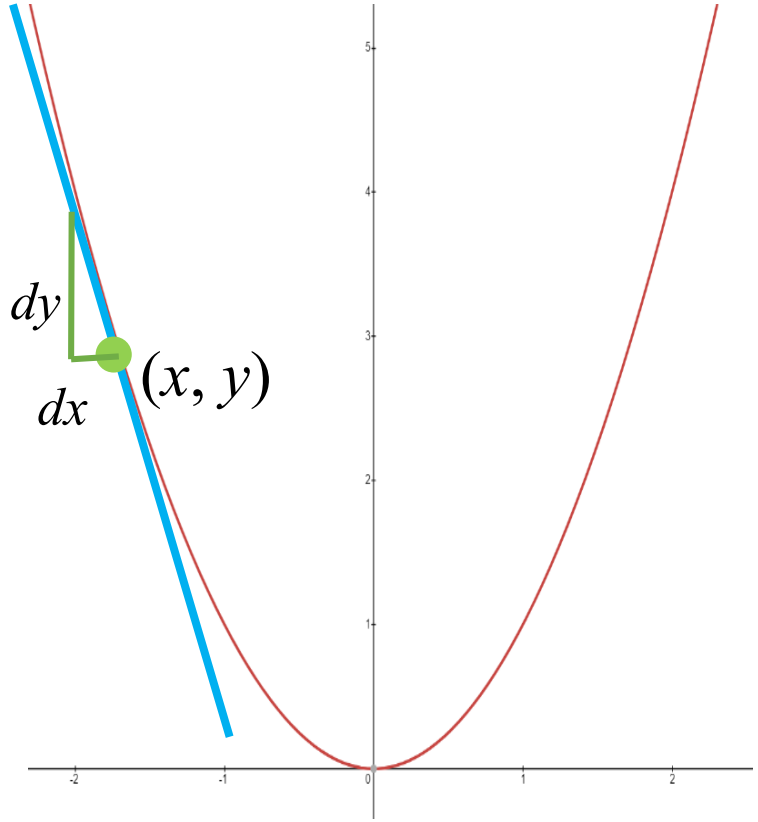
- 一种迭代寻优的算法
- 要找到某函数的最小值，总是沿着其下降最快的方向搜索
  - 梯度上升



梯度？

梯度与梯度下降

- 若损失函数图像如下



导数（梯度）——过(x, y)点的切线的斜率

三维空间中偏导数即函数在$x=0$, $y=0$, $z=0$这三个面上的投影曲线，其切线的斜率

高维空间类似

梯度下降

- 回到损失函数
  - $f(W) = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - X_i W\ )^2$
  - 稍加修改

$$f(W) = \frac{1}{2}\sum_{i=1}^{n}(y_i - X_i W\ )^2$$

  - 梯度

$$\Delta = \frac{\partial f(W)}{\partial W} = \frac{1}{2}\frac{\partial \sum_{i=1}^{n}(y_i - X_i W\ )^2}{\partial W}$$

$$= -\sum_{i=1}^{n} X_i(y_i - X_i W)$$

梯度下降

- 梯度确定搜索的方向
- 还需确定搜索的步长
  - 记为 α
  - 回归系数更新公式

$$\hat{W} = \hat{W} - \alpha\Delta = \hat{W} + \alpha \sum_{i=1}^{n} X_i(y_i - X_i W)$$

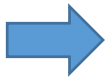
梯度下降的一些注意事项

- 确定步长 α
  - 取值取决于数据样本
  - 步长太大，会导致迭代过快，甚至有可能错过最优解
  - 步长太小，迭代速度太慢，收敛速度慢
- 终止条件算法
  - 最大终止代数
  - 误差小于给定的小值
- 参数的初始值选择
  - 初始值不同，获得的最小值也有可能不同，梯度下降求得的只是局部最小值
  - 如果损失函数是凸函数则一定是最优解

梯度下降的一些注意事项

- 数据归一化
  - 由于样本不同特征的取值范围不一样，可能导致迭代很慢，为了减少特征取值的影响，可以对特征数据归一化
  - 常用归一化公式
    - $x = \frac{x-u}{\sigma}$
      - 其中u为均值，σ为均方差，x为特征
    - $x = \frac{x-x_{min}}{x_{max}-x_{min}}$
    - 例：按第二种归一化方法

1	0.067732	3.176513
1	0.42781	3.816464
1	0.995731	4.550095



1	0	0
1	0.388016	0.465899
1	1	1

梯度下降算法

每个回归系数初始化为1

LOOP

计算整个数据集的梯度

$$\Delta = -\sum_{i=1}^{n} X_i(y_i - X_i W)$$

更新回归系数

$$\hat{W} = \hat{W} - \alpha\Delta$$

ENDLOOP

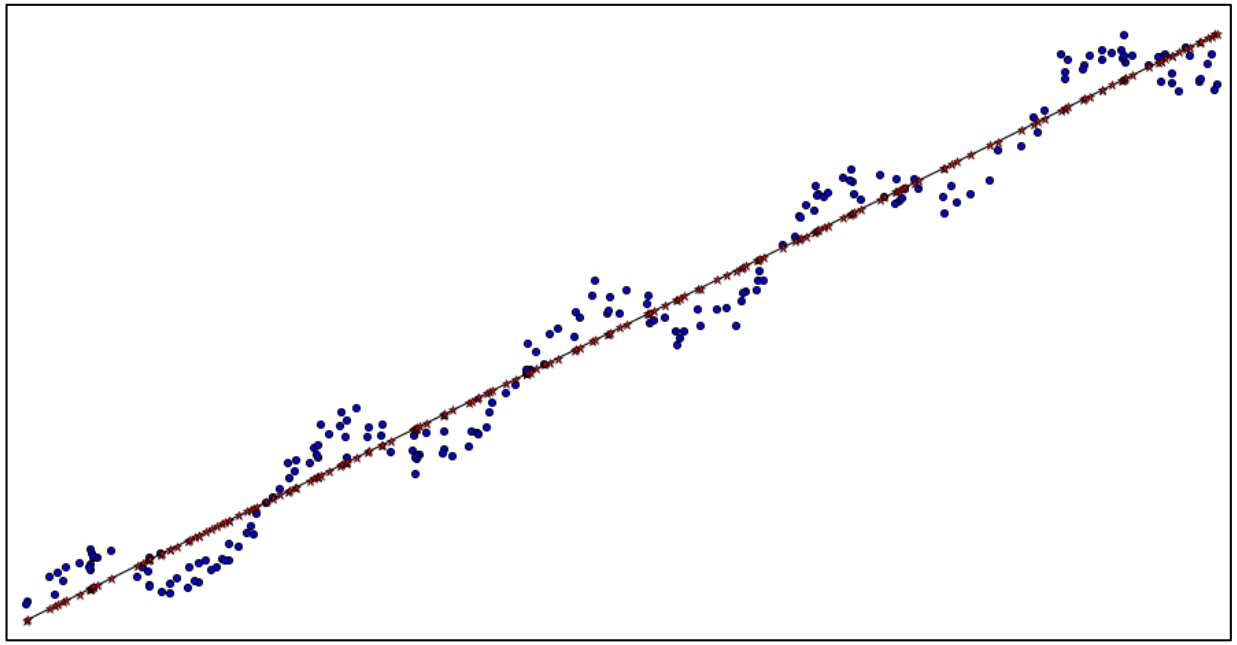
梯度下降实战

- 修改线性回归的代码，加入gradDscent(xArr, yArr)函数
- 结果相比较
  - 解析解

[[3.00774324] [1.69532264]]

  - 梯度下降

[[3.00758726] [1.69562035]]



梯度下降存在的问题

- 需要确定的参数比较多
- 每次更新回归系数时要遍历整个数据集
  - $\Delta = -\sum_{i=1}^{n} X_i(y_i - X_i W)$
- 数据集数据量大时计算代价太高
- 改进
  - 仅在新样本来时对回归系数进行增量式更新
  - 每次只用一个样本点来更新回归系数
  - 随机梯度下降
    - $\Delta = -X_i(y_i - X_i W)$

随机梯度下降(Stochastic Gradient Descent)

- 算法步骤

  BEGIN

  所有回归系数初始化为1

  对数据集中每个样本

  计算该样本的梯度

  更新回归系数值

  返回回归系数值

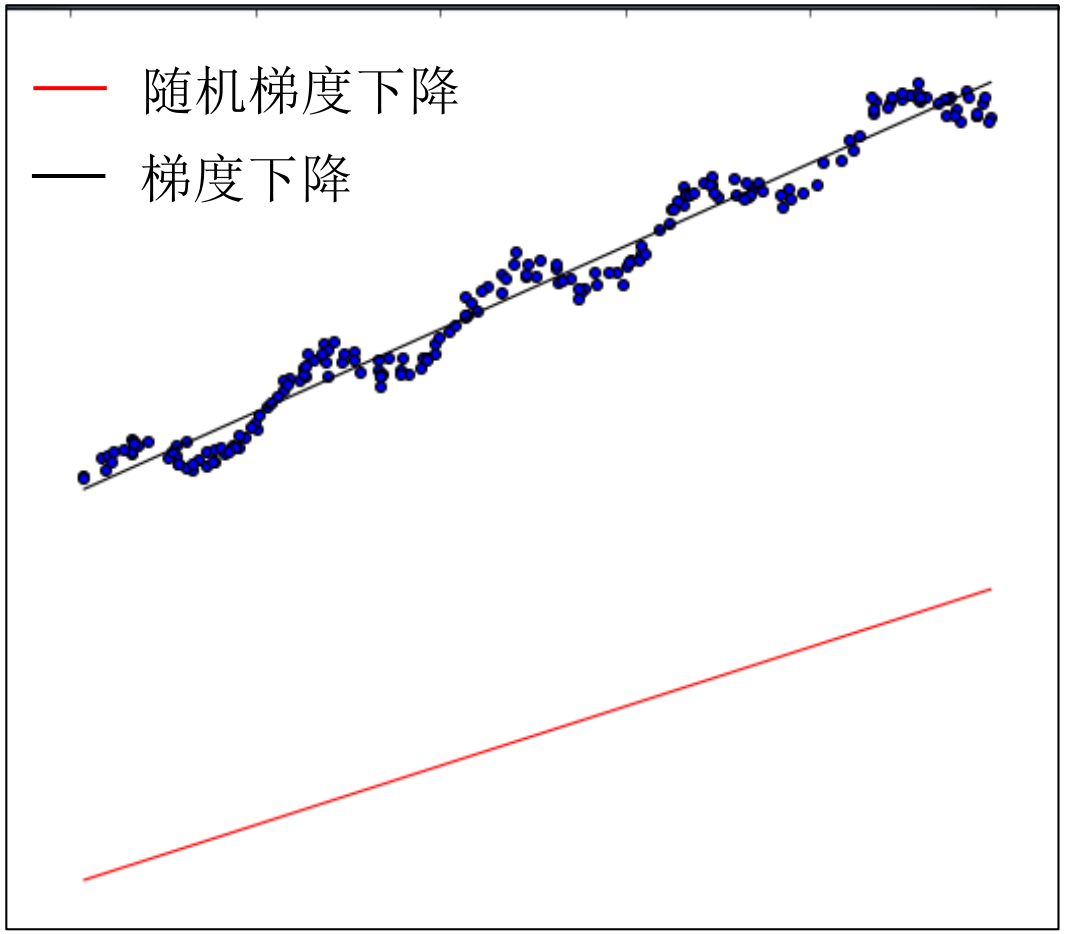
  END

- 修改代码
- 实验结果
  - 随机梯度下降的回归系数

    [1.4159566 1.21208609]
  - 梯度下降的回归系数

    [[3.00758726] [1.69562035]]

GD VS. SGD



- 随机梯度下降的结果明显不如梯度下降
- 但直接比较两者的结果并不公平
  - GD针对整个数据集迭代了500次
  - SGD的迭代次数只是数据集中样本的个数

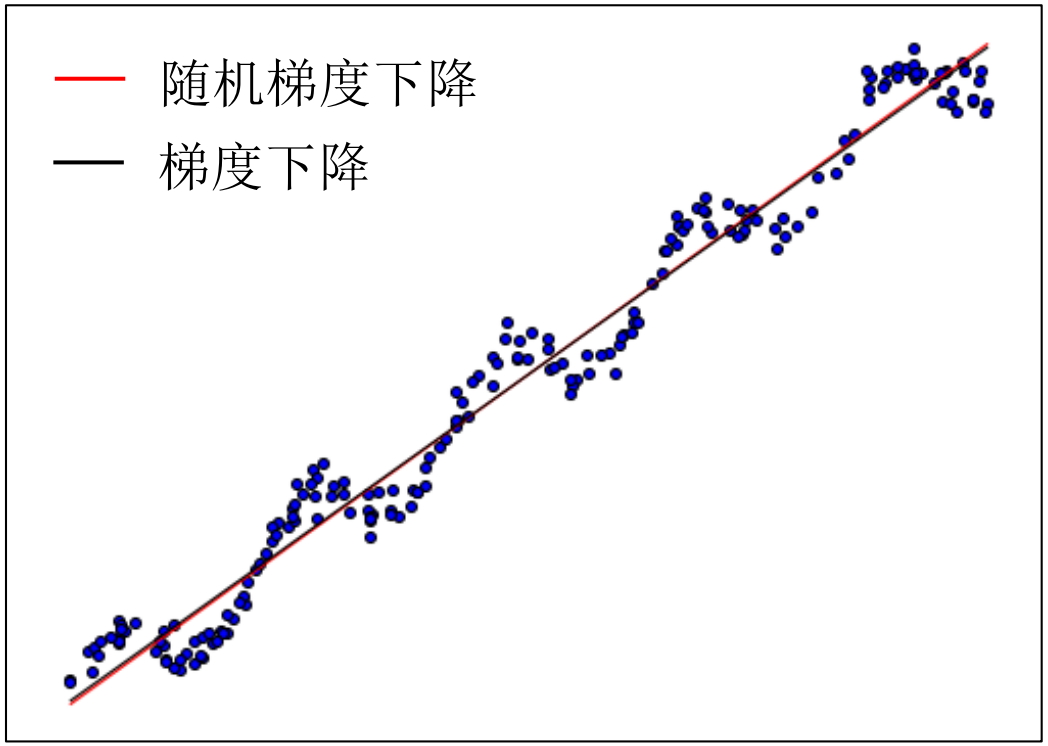
实验结果分析

- 修改代码
  - 增加步长，加快调整步伐
  - 让SGD针对所有样本迭代200次
- 第二种修改实验结果
  - SGD

[2.99841277 1.7135262]

  - GD

[[3.00758726] [1.69562035]]



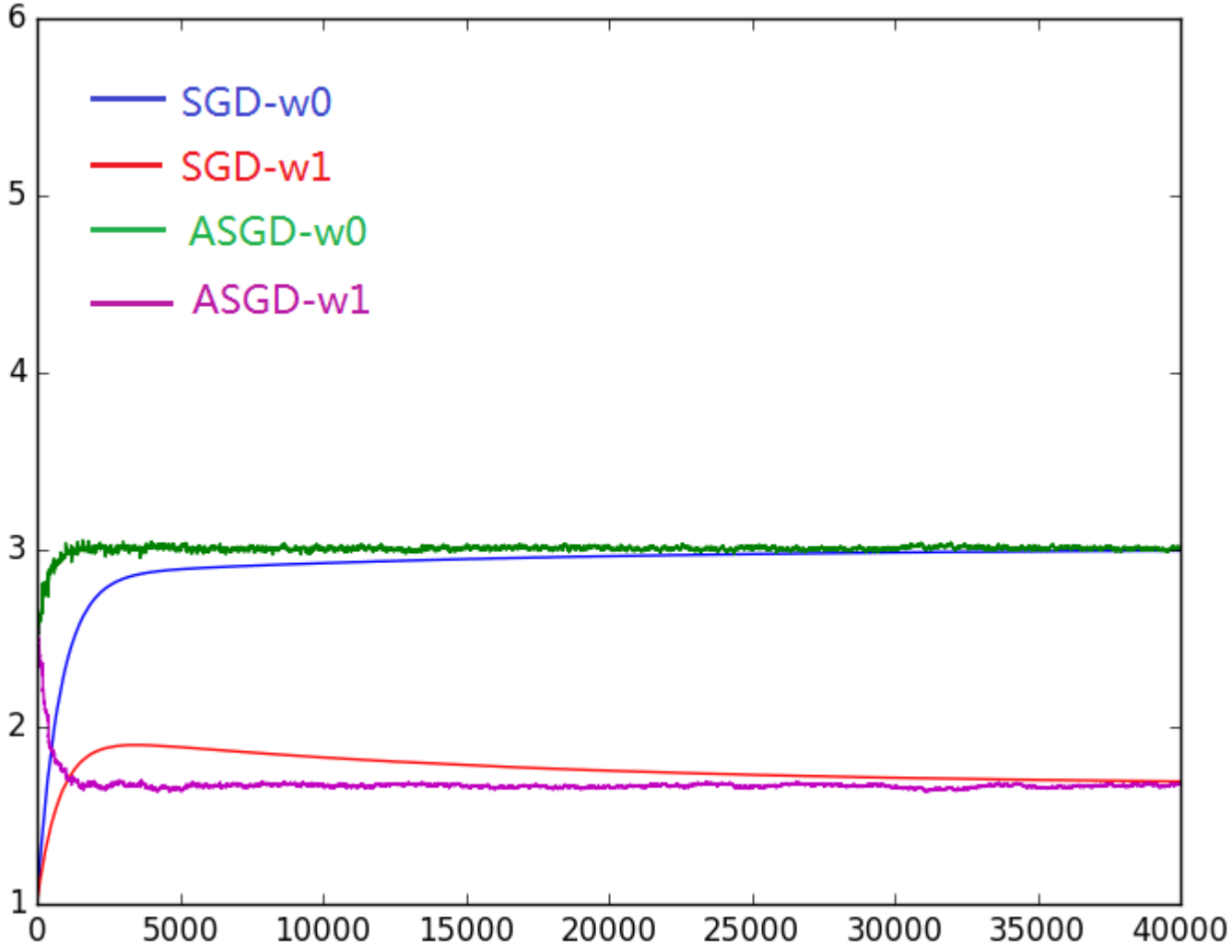
随机梯度下降的改进

- 步长决定了每一次回归系数调整的幅度
  - 大的步长能加快收敛速度，但有可能错过最优值
  - 小的步长收敛速度慢，但有助于找到最优值
- 改进
  - 步长 $\alpha=f(iter)$
    - $iter$为迭代的代数
  - 即令步长等于当前迭代数的函数
    - 一般为线性函数
    - 随$iter$增加减少

随机梯度下降的改进

- 数据集中的点不一定都能拟合到直线上
  - 原算法中依次取样本数据中的点，很可能造成回归系数的震荡
    - ex0.txt和ex1.txt的数据简单，没有这种现象
- 改进
  - 每次随机取一个样本点计算回归系数的增量

实验结果分析



- 针对ex1.txt数据集得到的回归系数收敛曲线
- 对于这个数据集显然顺序取点的效果比随机取点要好

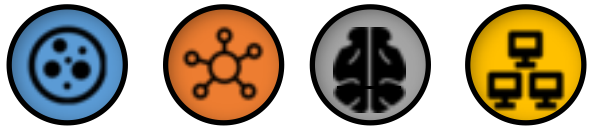
练习

- 分别采用梯度下降和随机梯度下降算法对ex0.txt，ex1.txt数据集进行分析
- 改进SGD，设计一个步长的调整函数，尝试不同的参数，与未改进的SGD进行比较
- 将GD，SGD，ASGD的结果画在一张图上，用不同颜色表示，并给出最后的回归系数
- 分析OLR，GD，SGD和改进的SGD的结果，计算相关系数，并自行设计表格，列表比较

练习——房价预测

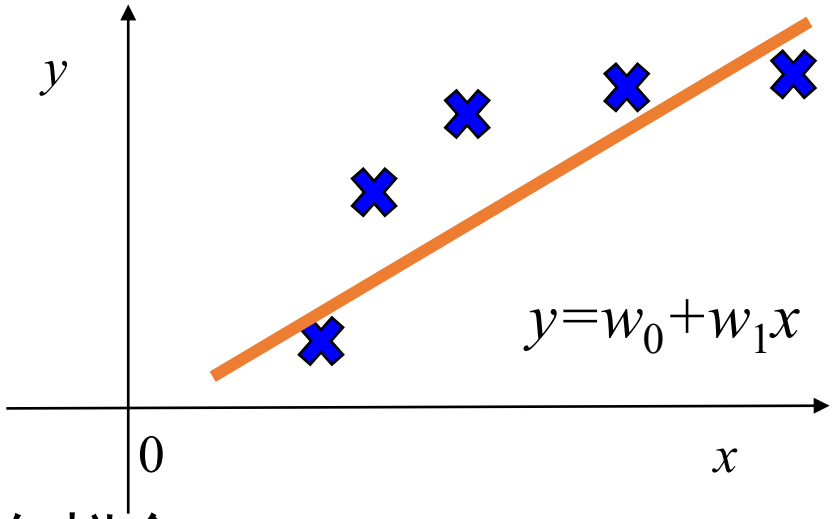
- 数据文件：ex1data1.txt， ex1data2.txt

文件名	特征数	特征名	类型	样本数量
ex1data1.txt	1	房屋面积	连续型	97
		售价（标签）	连续型	
ex1data2.txt	2	房屋面积	连续型	47
		房间数量	整型	
		售价（标签）	连续型	



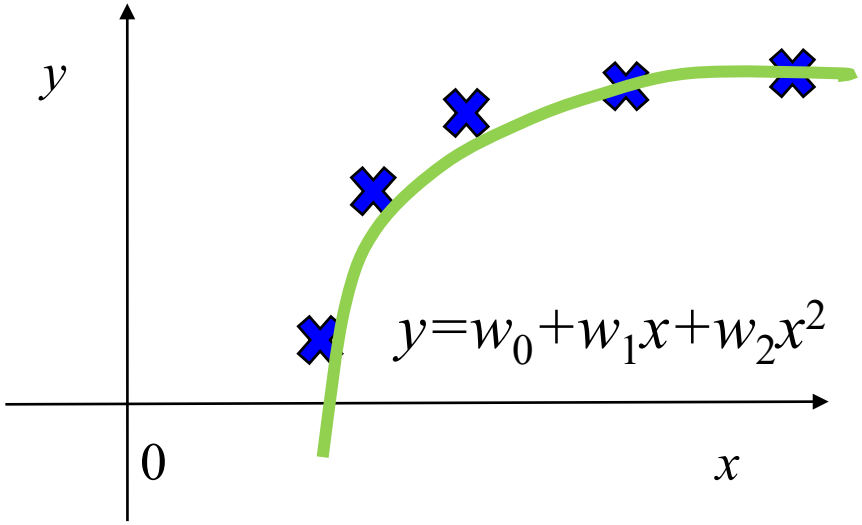
04 正则化

欠拟合与过拟合

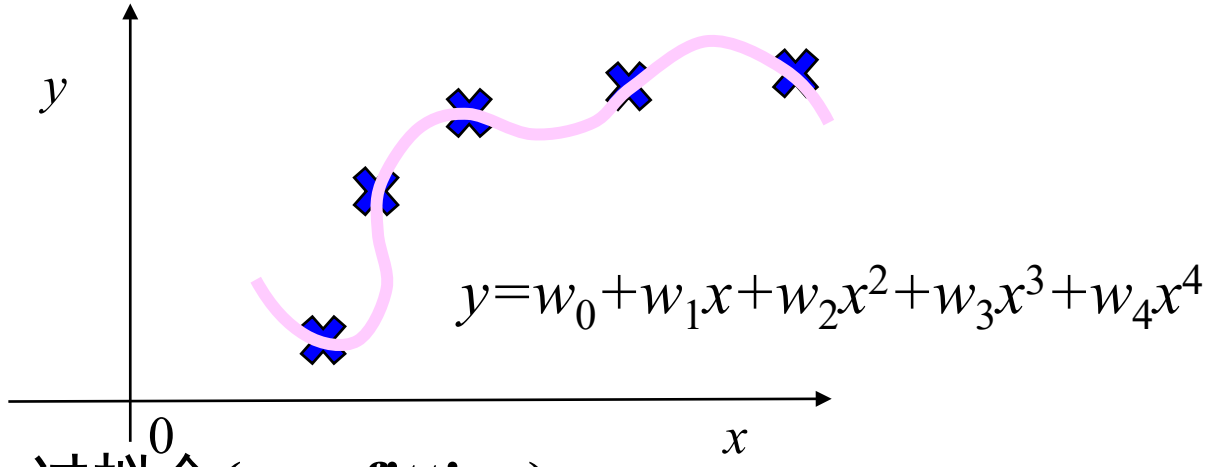


欠拟合(underfitting)

高偏差 (high bias)：与真实值偏差大



刚刚好

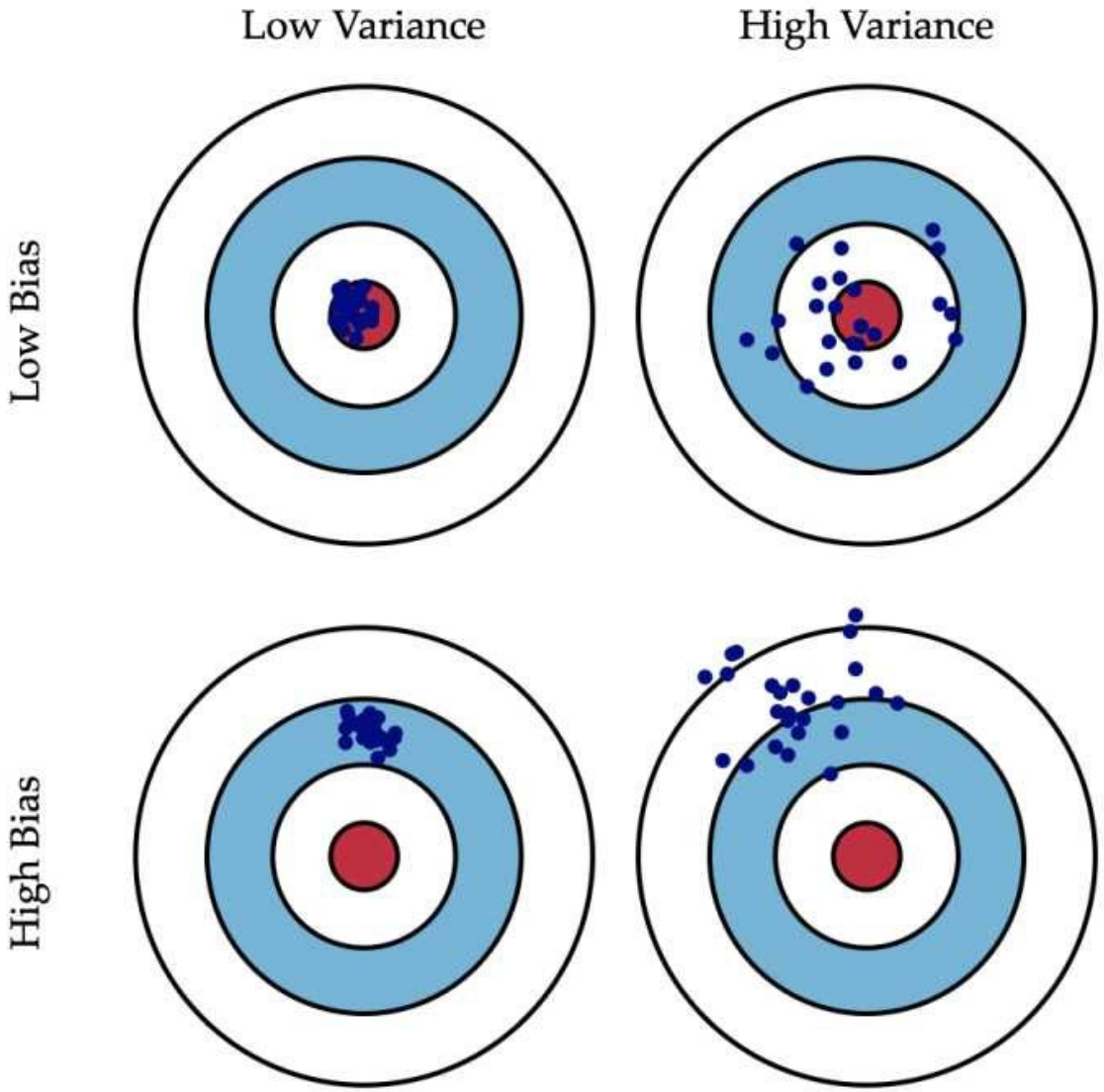


过拟合(overfitting)

高方差(high variance)：波动大，分散

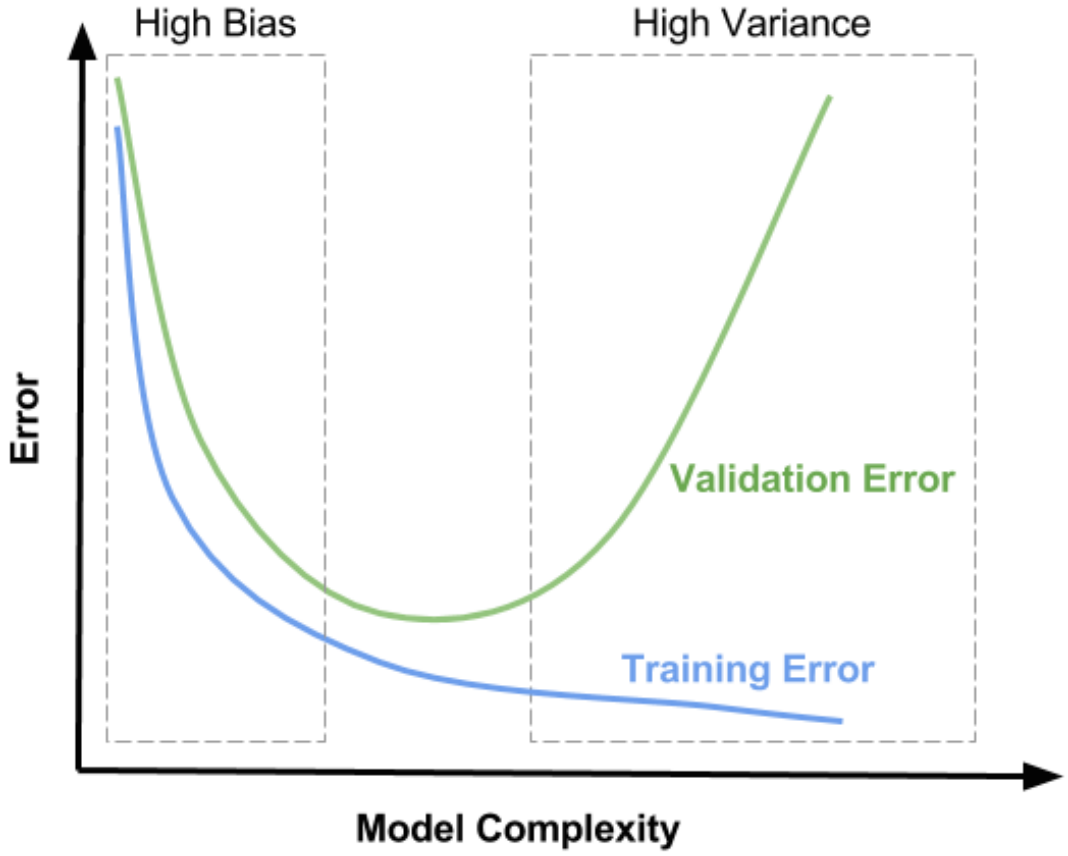
误差（Error）

- Error = Bias + Variance
- 偏差(Bias)反映的是模型在样本上的输出与真实值之间的误差，即模型本身的精准度
- 方差(Variance)反映的是模型每一次输出结果与模型输出期望之间的误差，即模型的稳定性



误差

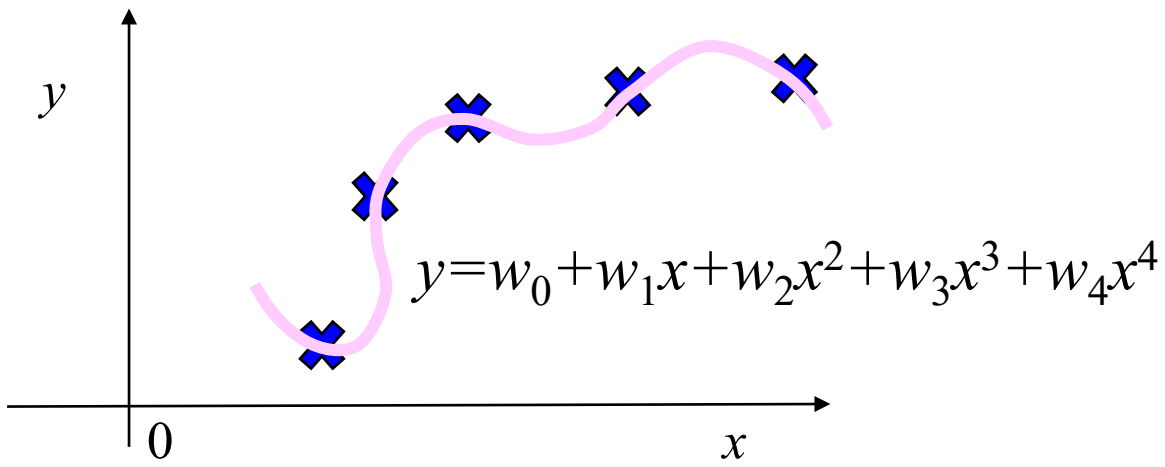
- 训练误差
  - 学习到的模型的预测值与真实值（标签）在训练数据集上的平均误差
  - 本质上不重要，仅对判断给定问题是否为容易学习的问题有意义
- 测试误差
  - 预测值与真实值在测试数据集上的平均误差
  - 反映了学习方法对未知测试数据的预测能力，是重要指标
  - 通常将学习方法对未知数据的预测能力称为泛化能力



过拟合

- 通常发生在变量（特征）过多的时候
- 训练数据上误差逐渐减小时，测试数据上误差反而增大，模型泛化能力弱
  - 泛化——一个假设模型能够应用到新样本的能力
- 原因
  - 数据量小
  - 数据噪音大
  - 模型任务本身很复杂
  - 模型搜索空间太大
    - 假设模型包括二项式、三项式…1024项式…，最后选择的最优模型是高次项的模型
- 防止过拟合的方法
  - 获取更多数据
  - 权值衰减
  - 减少特征的数量…
  - 正则化
    - 对模型参数添加先验知识，使得模型复杂度较小
    - 过拟合的时候，拟合函数的系数往往非常大，正则化减小参数数值的大小，从而减小特征变量的数量级

过拟合



- 拟合函数波动很大
- 在某些很小的区间里，函数值的变化剧烈
- 即函数在某些小区间里的导数绝对值非常大
  - 只有系数足够大，才能保证导数值很大
- 减小系数，即回归参数，可有效改善过拟合现象
  - 加入正则项，使模型具备稀疏、低秩、平滑等人想要的特性

给损失函数加入正则项

- 原来的损失函数只考虑了经验风险

$$f(W) = \sum_{i=1}^{n} (y_i - X_i W\ )^2 = (Y - XW)^T (Y - XW)$$

- 加入正则项共同构成结构风险

$$f(W) = (Y - XW)^T (Y - XW) + \lambda R(W)$$

其中λ是参数，不同的值有不同的正则化效果

L0，L1与L2范数

- L0范数
  - 向量中非0的元素的个数，用$\|W\|_0$表示
  - 例：W=[0.4 0 -0.7 0.3]，$\|W\|_0=3$
  - 使用L0范数的目的
    - 希望W的大部分元素都是0
    - 实现W的稀疏化
    - 减少特征
- L1范数
  - 向量中各个元素绝对值之和，也称“稀疏规则算子”，用$\|W\|_1$表示
  - 例：W=[0.4 0 -0.7 0.3]，$\|W\|_1=1.4$
  - 使用L1范数的目的
    - 与L0相同，两者在一定条件下等价
    - L0范数很难优化求解
    - L1具有更好的优化求解特性被广泛应用

L0，L1与L2范数

- L2范数
  - 向量各元素的平方和的平方根，用$\|W\|_2$表示
  - 例：W=[0.4 0 -0.7 0.3]，

$$\|W\|_2 = \sqrt{0.4^2 + 0^2 + (-0.7)^2 + 0.3^2}$$

  - 使用L2范数的目的
    - 使得W的每个元素都很小，都接近于0
    - 与L1范数不同，它不会让W的元素等于0，而是接近于0
    - 越小的参数说明模型越简单
    - 越简单的模型则越不容易产生过拟合现象

加入正则项的线性回归

- 岭回归（Ridge Regression）
  - 加入L2范数的线性回归
  - 正则项为L2范数的平方
    - 缩减（shrinkage）
  - 损失函数

$$f(W) = (Y - XW)^T(Y - XW) + \|W\|$$
$$(Y - XW)^T(Y - XW) + \lambda W^T W$$

则$f(W)' = 2X^T XW - 2X^T Y + 2\lambda W$

$$= (X^T X + \lambda I)W - 2X^T Y$$

令$f(W)' = 0$，则

$$W = (X^T X + \lambda I)^{-1} X^T Y$$

  - 若$X^T X$
    - 岭回归加入λI

加入正则项的线性回归

- Lasso回归（ Lasso Regression）
  - 加入L1范数的线性回归
  - 损失函数

$$f(W) = (Y - XW)^T(Y - XW) + \|W\|$$

则$f(W)' = 2X^TXW - 2X^TY + \lambda sgn(W)$

  - 计算复杂度高
  - 需使用二次规划算法

Scikit-learn中的线性回归

- Scikit-learn是一个用于机器学习的功能强大的python包
- 在数据量不是过大的情况下，可以解决大部分问题
- Scikit-learn中线性回归模块均包含在linear_model 中
  - 引入包：from sklearn import linear_model
  - 普通线性回归LinearRegression()
  - 岭回归Ridge()
  - Lasso回归Lasso（ ）

项目1——鲍鱼年龄预测

- 实验数据来自UCI数据集
  - 鲍鱼年龄可从鲍鱼壳上的年轮推算
  - 共4177条数据，8个特征，1个标签，无缺失数据

特征名	类型	单位	描述
性别（Sex）	标称型	--	M（雄），F（雌），I（婴儿）
长度（Length）	连续型	mm	外壳最长方向的长度
直径（Diameter）	连续型	mm	垂直于长度的尺寸
高度（Height）	连续型	mm	连壳带肉的厚度
总重（Whole weight ）	连续型	克	整个鲍鱼的重量
去壳重（Shucked weight ）	连续型	克	肉的重量
内脏重量（Viscera weight ）	连续型	克	放血后肠道重量
壳重（Shell weight ）	连续型	克	干燥后重量
轮数（Rings）	整型	--	标签，加1.5为鲍鱼的年龄

项目2——波士顿房价预测

- 波士顿房价数据集
  - 包含对房价的预测，以千美元计，给定的条件是房屋及其相邻房屋的详细信息
  - 共有 506 个样本，13 个特征和1个标签

特征名	描述	特征名	描述
CRIM	城镇人均犯罪率。	DIS	到波士顿五个中心区域的加权距离。
ZN	住宅用地超过 25000 sq.ft. 的比例。	RAD	辐射性公路的接近指数。
INDUS	城镇非零售商用土地的比例。	TAX	每 10000 美元的全值财产税率。
CHAS	查理斯河空变量（如果边界是河流，则为1；否则为0）。	PTRATIO	城镇师生比例。
NOX	一氧化氮浓度。	B	1000（Bk-0.63）^ 2，其中 Bk 指代城镇中黑人的比例。
RM	住宅平均房间数。	LSTAT	人口中地位低下者的比例。
AGE	1940 年之前建成的自用房屋比例。	MEDV	自住房的平均房价，以千美元计。

项目3——前列腺癌预测

- 数据来源于Stamey等1989年的研究项目
- 数据说明文件：prostate.info.txt
- 数据文件：prostate.data（文本格式）
- 特征说明
  - 红色为预测目标

特征名	描述
lcavol	log cancer volume
lweight	log prostate weight
age	
lbph	log of the amount of benign prostatic hyperplasia
svi	seminal vesicle invasion
lcp	log of capsular penetration
gleason	Gleason score
pgg45	percent of Gleason scores 4 or 5
lpsa	log of prostate-specific antigen
train	train/test indicator

实验要求

- 在代码中加入残差的计算，绘制残差图
  - 残差图是以$\hat{y}_i$为横坐标，以残差$\hat{y}_i - y_i$为纵坐标的散点图
- 从数据集中分出两部分，一部分作为训练集，一部分作为测试集
- 分别采用OLR，梯度下降和采用不同步长参数的随机梯度下降进行回归分析，得到预测模型
- 计算每个模型在训练集上的误差，列表比较
- 计算每个模型在测试集上的误差，列表比较
- 分析上述实验结果，你得到什么启发？
- 对所有数据进行实验，用所有数据的前面70%做训练集，剩下的做测试集进行交叉验证，可分工测试不同的参数，比比看谁的结果最优？