线性回归

机器学习研究室

计算机科学与技术学院

吉林大学

大纲

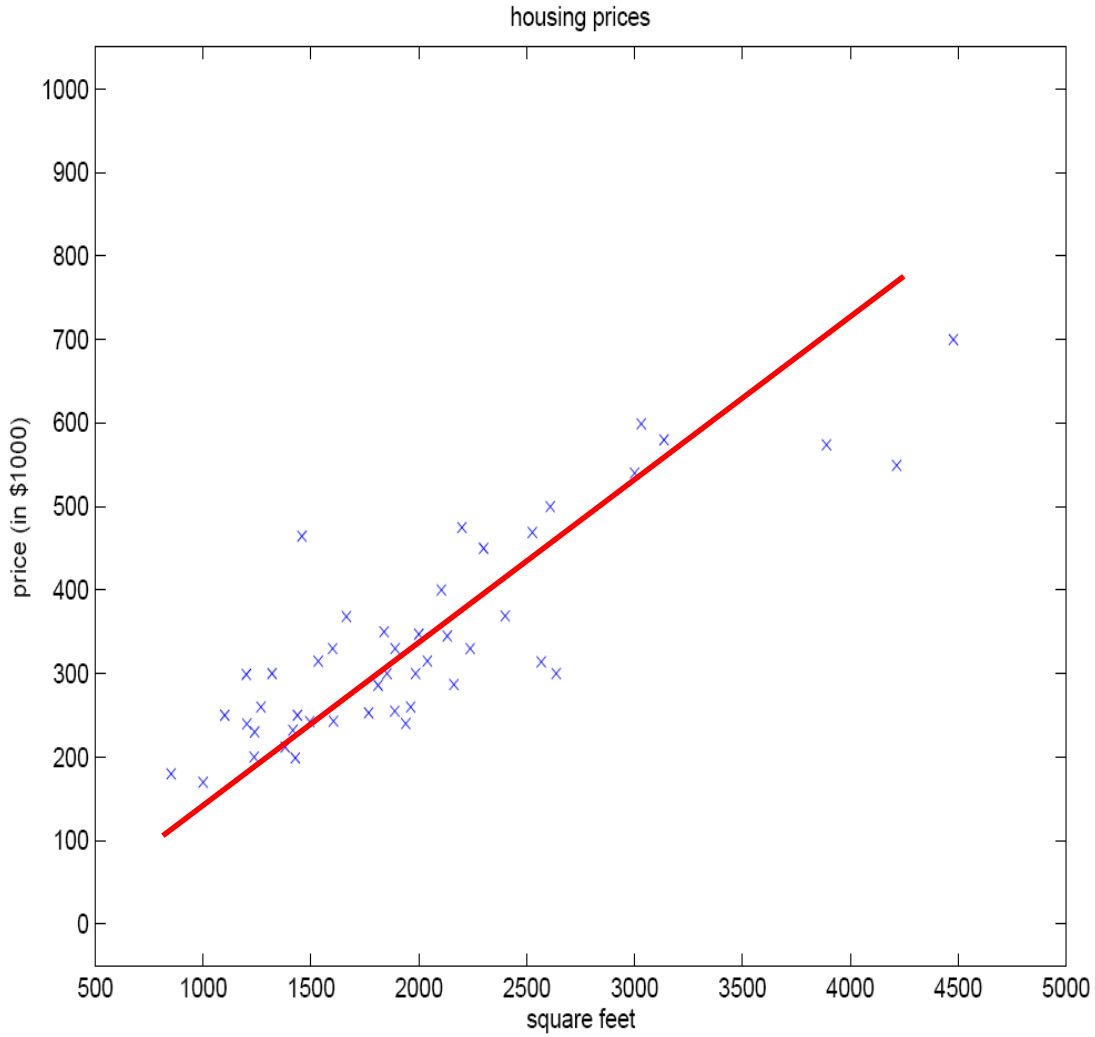
- 监督学习概述
- 线性回归
- 最小二乘目标函数
- 梯度下降法
- 最小二乘法的正规方程
- 局部加权线性回归
- 非线性模型的多元回归
- 机器学习中的若干讨论

监督学习概述

监督学习概述

Living area (feet^2)	Price (1000$s)
2104	400
1600	330
2400	360
1416	232
3000	540
……	……

监督学习概述

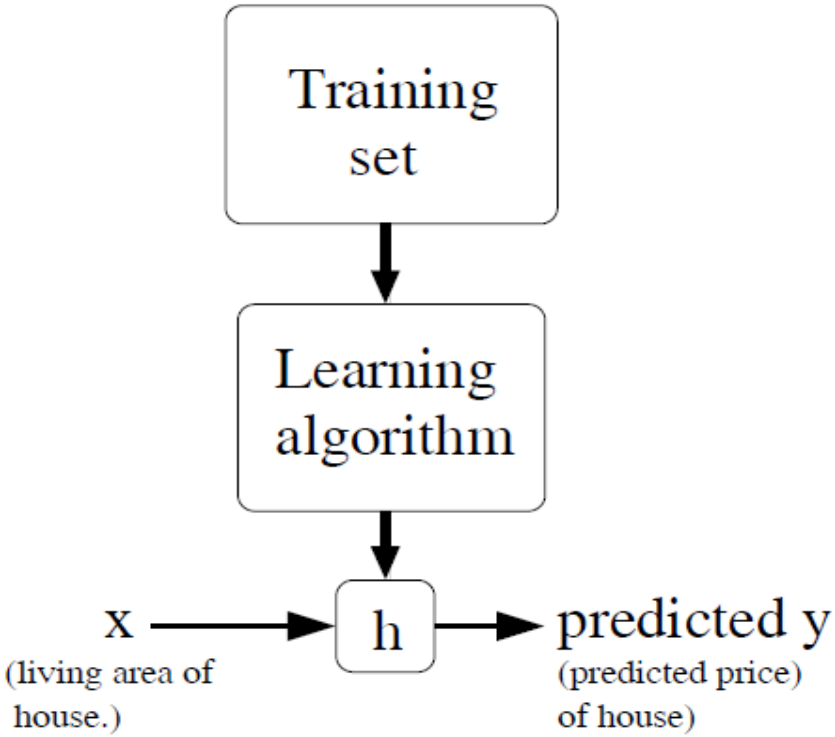


- 关键词：特征、目标、训练样本、训练集

监督学习概述

- 特征 $x^{(i)}$
- 目标 $y^{(i)}$
- 训练样本 $(x^{(i)}, y^{(i)})$
- 训练集

$$\{(x^{(i)}, y^{(i)}); i = 1, \ldots, m\}$$



监督学习的一般范式

- 假设 $h(x)$
  - ✓y为实数，回归问题
  - ✓ y={1，2，3，…,C}，分类问题

线性回归

线性回归

Living area (feet^2)	#bedrooms	Price (1000$s)
2104	3	400
1600	3	330
2400	3	360
1416	2	232
3000	4	540
……	……	……

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

$$h(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x,$$

线性回归

- 如何确定模型$h_\theta(x)$的参数θ

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

- 需要一种参数调整的目标——代价函数(cost function)

$$\min\ J(\theta) = \sum_{i=1}^{m} \left| h_\theta(x^{(i)}) - y^{(i)} \right|$$

cost function

线性回归

- 绝对误差 vs. 平方和误差

绝对误差

$$J(\theta) = \sum_{i=1}^{m} \left| h_\theta(x^{(i)}) - y^{(i)} \right|$$

平方和误差

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2.$$

LMS ?

线性回归

- 线性模型特点
  - ✓ 形式简单、易于建模
  - ✓ 可解释性
  - ✓ 非线性模型的基础
  - ✓ 引入层级结构或高维映射
- 系数反映了每个特征对结果的影响强弱

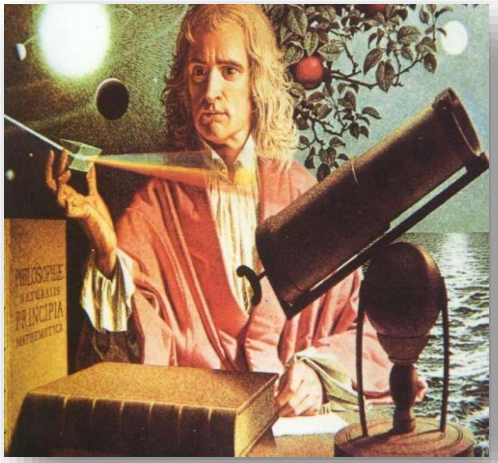
线性回归

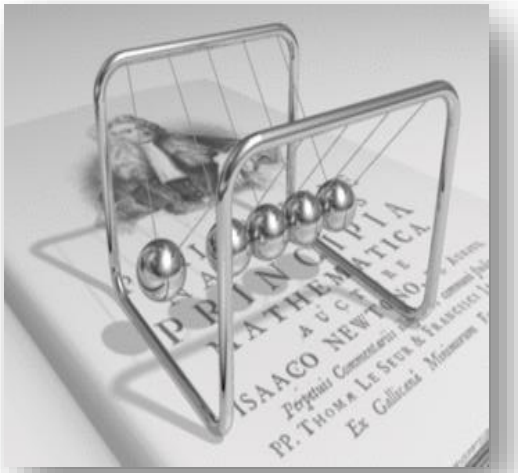
- **近代科学标志：　牛顿　线性化**
  - 17世纪~19世纪末

线性化

复杂系统 → **“理想方程”**

“理想模型”和“理想环境”





牛顿研究范式的核心-线性化，非线性、随机

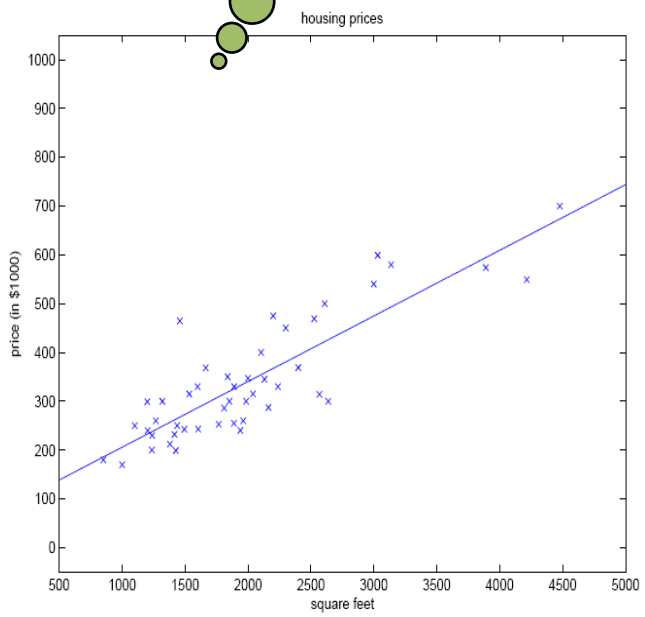
第四次人工智能危机与突破-郑志明 院士，**2019年7月**

线性回归-示例

- 单变量线性回归

Living area (feet^2)	Price (1000$s)
2104	400
1600	330
2400	360
1416	232
3000	540
……	……

By batch gradient descent



$\theta_0 = 71.27,$
$\theta_1 = 0.1345.$

线性回归-示例

- 多变量线性回归

Living area (feet^2)	#bedrooms	Price (1000$s)
2104	3	400
1600	3	330
2400	3	360
1416	2	232
3000	4	540
……	……	……

$$\theta_0 = 89.60, \theta_1 = 0.1392, \theta_2 = -8.738$$

机器学习中的若干讨论

特征规范化

特征规范化

- 各个特征变量的范围要保持相近
  - 考虑房价预测：房屋尺寸和房间数目
  - 如果这两个变量分别对应横纵坐标，则代价函数的等高线图非常扁，梯度下降算法将需要更多迭代
  - 特征的尺度缩放到-1到1之间

（维度特征-均值）/标准差

$x_1=(x_1-\text{mean}(x_1))/\text{std}(x_1)$

Size (feet2)	Number of bedrooms	Number of floors	Age of home (years)	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
...	...	...	...	...

最小二乘目标函数

最小二乘目标函数

- 最小二乘方目标函数

$$\min J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

为了方便使用梯度下降法求解模型参数

某些问题中具有物理意义

梯度下降法

梯度下降法

- 最小化优化问题

$$\min J(\theta)$$

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta).$$

梯度下降法

- 最大化优化问题

$$max\, J(\theta)$$

$$\theta_j := \theta_j + \alpha \frac{\partial}{\partial \theta_j} J(\theta).$$

梯度上升法
Or 爬山法

梯度下降法

- 梯度下降法——以线性回归为例

$$\min J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

- 初始化 θ
- 逐一更新全部的θ

关键因子

$$\theta_j := \theta_j - \alpha\frac{\partial}{\partial\theta_j}J(\theta).$$

α称为学习率
(learning rate)

梯度下降法

- 关键因子的计算
  - 如果只有一个训练样本

$$
\begin{aligned}
\frac{\partial}{\partial \theta_j} J(\theta) &= \frac{\partial}{\partial \theta_j} \frac{1}{2} (h_\theta(x) - y)^2 \\
&= 2 \cdot \frac{1}{2} (h_\theta(x) - y) \cdot \frac{\partial}{\partial \theta_j} (h_\theta(x) - y) \\
&= (h_\theta(x) - y) \cdot \frac{\partial}{\partial \theta_j} \left(\sum_{i=0}^{n} \theta_i x_i - y \right) \\
&= (h_\theta(x) - y) \, x_j
\end{aligned}
$$

梯度下降法

- 如果只有一个样本，则

$$\theta_j = \theta_j - \alpha(h_\theta(x) - y)x_j$$

其中，$0 \le j \le n$

LMS update rule OR,
Widrow-Hoff learning rule

- Widrow-Hoff 学习规则的性质
  - 参数更新量的模长正比于回归误差，即 $|h_\theta(x) - y|$

多个样本训练
怎么办？

梯度下降法

- Widrow-Hoff 学习规则向样本集的推广
  - Batch gradient descent (BGD)

{

$$\theta_j := \theta_j + \alpha \sum_{i=1}^{m} \left(y^{(i)} - h_\theta(x^{(i)})\right) x_j^{(i)} \qquad \text{(for every } j\text{)}.$$

}

  - Stochastic gradient descent (SGD)

{ for i=1 to m, {

$$\theta_j := \theta_j + \alpha \left(y^{(i)} - h_\theta(x^{(i)})\right) x_j^{(i)} \qquad \text{(for every } j\text{)}$$

} }

Or 增量梯度下降法

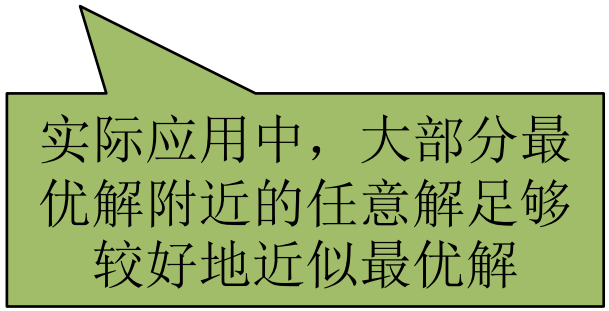
梯度下降法

- 有关梯度下降法的讨论
  - 学习率α

  向代价函数下降程度最大的方向迈出一小步
  - 学习率的选择
    - 学习率α过小，达到收敛所需的迭代次数高；
    - 学习率α过大，每次迭代可能会越过极小值，导致无法收敛。
- 应对措施
  - Try on different learning rates
  - 随机优化

梯度下降法

- Batch VS. stochastic gradient descent (BGD vs. SGD)
  - BGD 扫描整个训练集后再更新参数
  - SGD 遇到一个样本后立即更新参数
  - 对于大样本问题，BGD收敛较慢
  - 但SGD有可能发生震荡，而无法收敛到极小值

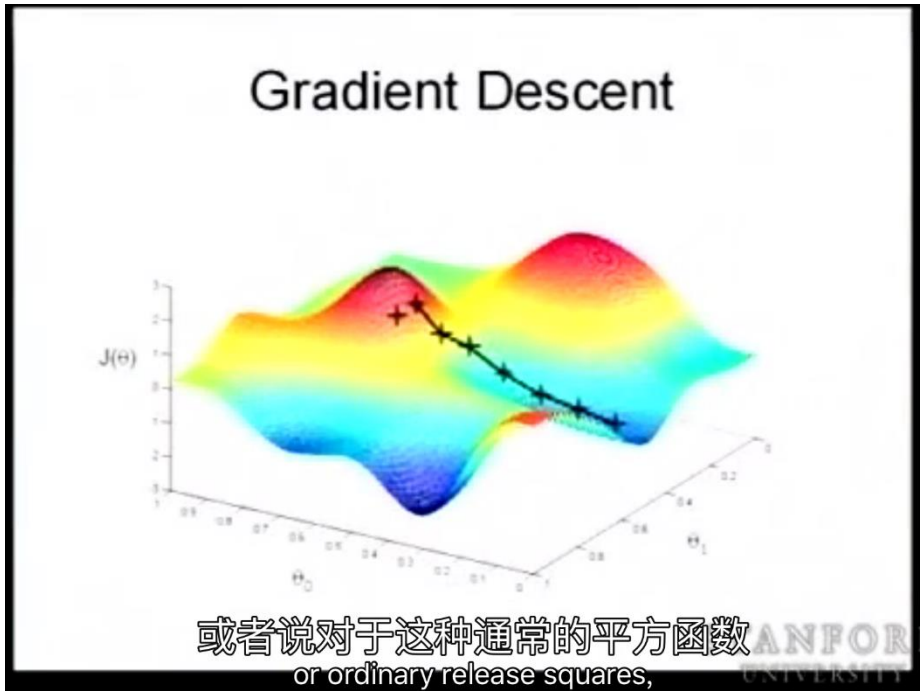


梯度下降法

- 全局极小值 vs. 局部极小值

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2.$$

$$\theta_j := \theta_j - \alpha\frac{\partial}{\partial\theta_j}J(\theta).$$



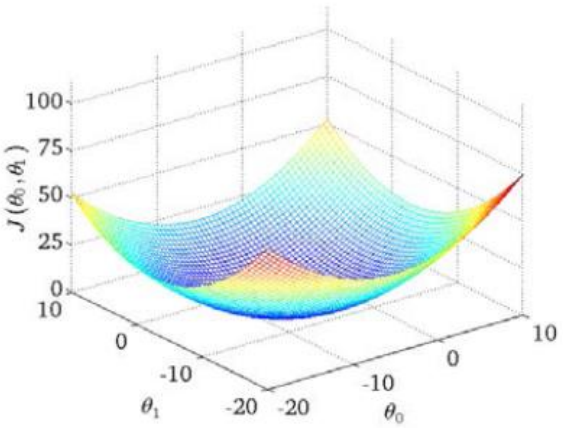
梯度下降法

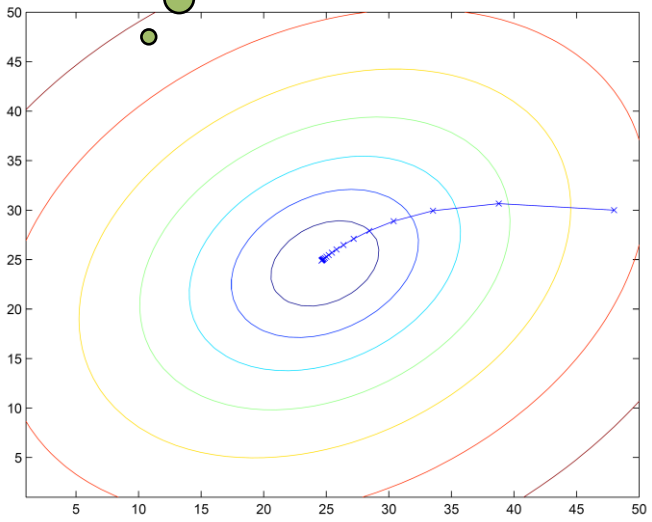
- 线性回归的全局极小值保证

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2.$$

$$h(x) = \sum_{i=0}^{n}\theta_i x_i = \theta^T x,$$

Convex quadratic function





linear regression： one global optimal solution

最小二乘法的正规方程

最小二乘法的正规方程

- Find closed-form (闭式) the value of θ that minimizes $J(\theta)$.
- X：$m \times (n+1)$： m 个样本，n个特征

$$X = \begin{bmatrix} — & (x^{(1)})^T & — \\ — & (x^{(2)})^T & — \\ & \vdots & \\ — & (x^{(m)})^T & — \end{bmatrix}. \quad \vec{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}.$$

$$h_\theta(x^{(i)}) = (x^{(i)})^T\theta$$

最小二乘法的正规方程

- inner product

$$x, y \in \mathbb{R}^n \qquad x^T y \in \mathbb{R} = \sum_{i=1}^{n} x_i y_i$$

- outer product

$$x \in \mathbb{R}^m,\ y \in \mathbb{R}^n$$

$$xy^T \in \mathbb{R}^{m \times n} = \begin{bmatrix} x_1 y_1 & x_1 y_2 & \cdots & x_1 y_n \\ x_2 y_1 & x_2 y_2 & \cdots & x_2 y_n \\ \vdots & \vdots & \ddots & \vdots \\ x_m y_1 & x_m y_2 & \cdots & x_m y_n \end{bmatrix}$$

最小二乘法的正规方程

$$
\begin{aligned}
X\theta - \vec{y} &= \begin{bmatrix} (x^{(1)})^T\theta \\ \vdots \\ (x^{(m)})^T\theta \end{bmatrix} - \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix} \\
&= \begin{bmatrix} h_\theta(x^{(1)}) - y^{(1)} \\ \vdots \\ h_\theta(x^{(m)}) - y^{(m)} \end{bmatrix}.
\end{aligned}
$$

最小二乘法的正规方程

$$\begin{aligned}\frac{1}{2}(X\theta-\vec{y})^T(X\theta-\vec{y}) &= \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)})-y^{(i)})^2 \\ &= J(\theta)\end{aligned}$$

$$\begin{aligned}\nabla_\theta J(\theta) &= \nabla_\theta \frac{1}{2}(X\theta-\vec{y})^T(X\theta-\vec{y}) \\ &= X^TX\theta - X^T\vec{y}\end{aligned}$$

参考：矩阵导数@斯坦福大学机器学习课程讲义汇编

https://daiwk.github.io/assets/matrix+vector+derivatives+for+machine+learning.pdf

最小二乘法的正规方程

- 最小化 $J(\theta)$，令$J(\theta)$关于θ 的偏导数为0，得到θ的闭式解。

$$\theta = (X^T X)^{-1} X^T \vec{y}.$$

- 前提：$X^T X$ 满秩或者正定

最小二乘法的正规方程

	梯度下降法	正规方程法
学习速率 α	需要设置	不需要
计算次数	需要多次迭代	不需要迭代
特征归一化	需要	不需要
时间复杂度	O(kn2)	O(n3)，需要计算 X^TX 的逆
特征数量	即使 n 很大也可工作	如果 n 很大计算速度慢

可见特征方程✦得到的是解析解，无需迭代，也没有设置学习速率的繁琐，需要特征归一化，但是求解正规方程需要求矩阵的逆，然而不是所有的矩阵都可逆，而且有些可逆矩阵✦的求逆极其耗费时间，所以特征方程法看似简单，其实使用场景并不多。只有当特征值比较小的时候，可以考虑使用特征方程法。

正规方程形式回归求解-示例

- 正规方程形式回归求解示例

Size (feet2)	Number of bedrooms	Number of floors	Age of home (years)	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

正规方程形式回归求解-示例

- 样本矩阵

X =

2104	5	1	45
1416	3	2	40
1534	3	2	30
852	2	1	36

- 增广矩阵

X =

1	2104	5	1	45
1	1416	3	2	40
1	1534	3	2	30
1	852	2	1	36

(即，X=[1 2104 5 1 45; 1 1416 3 2 40; 1 1534 3 2 30;1 852 2 1 36])

正规方程形式回归求解-示例

- (X'*X)^(-1)*X'

-0.5005 -3.3477 2.3643 2.5059
-0.0030 -0.0064 -0.0014 -0.0048
4.6270 4.4531 2.9180 2.7891
1.2573 2.6055 1.0205 0.5059
-0.0028 0.0811 -0.0872 0.0093

1.0e+03 *

- y=[460 232 315 178]'
- (X'*X)^(-1)*X'*y

0.1839
-0.0042
4.5771
1.5943
-0.0083

正规方程形式回归求解-示例

- XN(:,1)=X(:,1),for i=2:5 x1=X(:,i), x1=(x1-mean(x1))./std(x1) , XN(:,i)=x1, end

(XN'*XN)^(-1)*XN'*y

296.2500
63.6070
92.3072
-28.4804
-35.4597

MATLAB中的回归示例

Matlab中的回归示例

- carsmall.mat： Measurements of cars, 1970, 1976, 1982

load carsmall

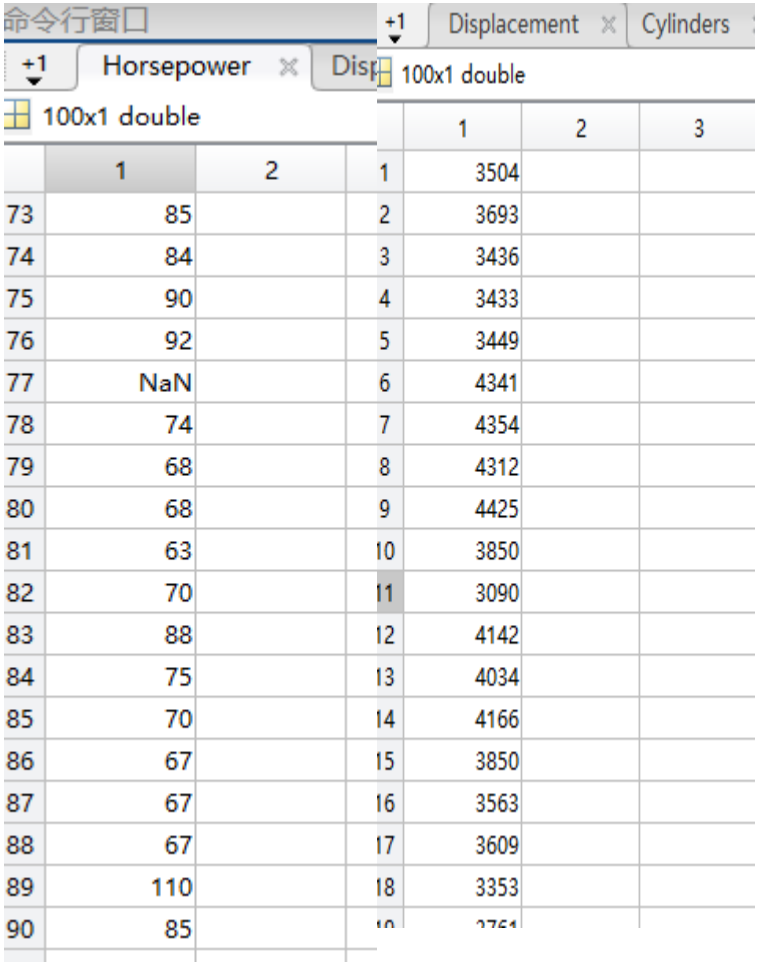
table(Weight,Origin,MPG,Model_Year,Model,Horsepower,Displacement,Cylinders,Acceleration)

线性模型回归

x1 = Weight;
x2 = Horsepower;
% Contains NaN data
y = MPG;

- find(isnan(y)==1)

ans =
11
12
13
14
15
18



	1	2
73	85	
74	84	
75	90	
76	92	
77	NaN	
78	74	
79	68	
80	68	
81	63	
82	70	
83	88	
84	75	
85	70	
86	67	
87	67	
88	67	
89	110	
90	85	

	1	2	3
1	3504		
2	3693		
3	3436		
4	3433		
5	3449		
6	4341		
7	4354		
8	4312		
9	4425		
10	3850		
11	3090		
12	4142		
13	4034		
14	4166		
15	3850		
16	3563		
17	3609		
18	3353		

线性模型回归

- find(isnan(x1)==1)

ans =

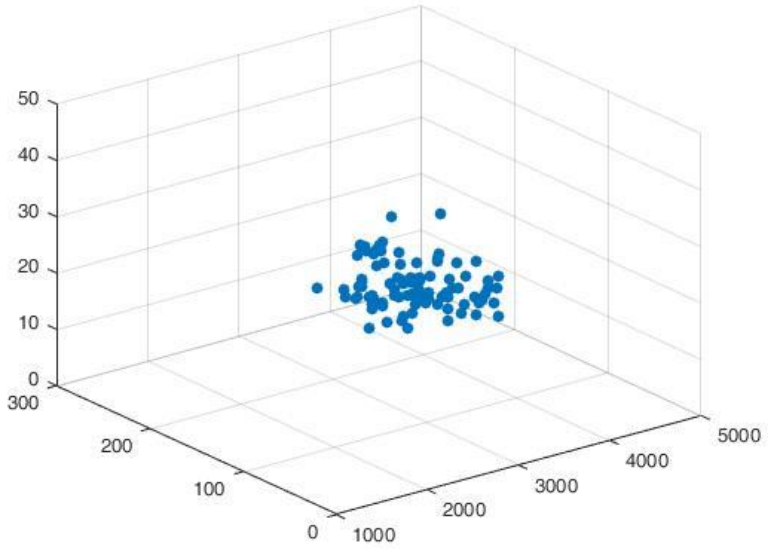
空矩阵: 0×1

- find(isnan(x2)==1)

ans = 77

线性模型回归

```
scatter3(x1,x2,y,'filled')
hold on
```

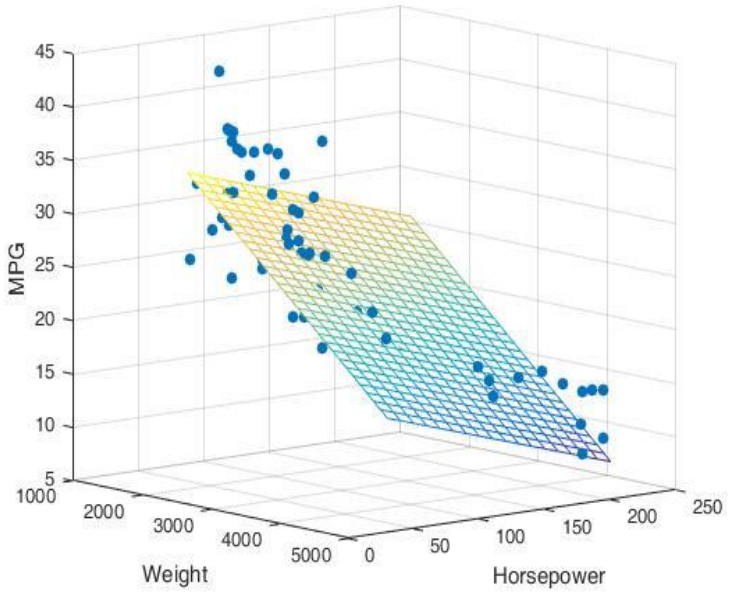



```
X = [ones(size(x1)) x1 x2];
b = regress(y,X) % Removes NaN data
```

线性模型回归

```
x1fit = min(x1):100:max(x1);
x2fit = min(x2):10:max(x2);
[X1FIT,X2FIT] = meshgrid(x1fit,x2fit);
YFIT = b(1) + b(2)*X1FIT + b(3)*X2FIT
```

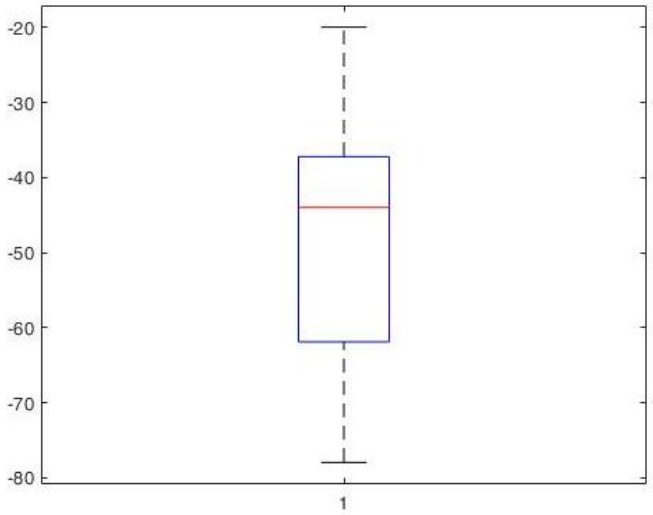
```
mesh(X1FIT,X2FIT,YFIT)
xlabel('Weight')
ylabel('Horsepower')
zlabel('MPG')
view(50,10)
```



线性模型回归

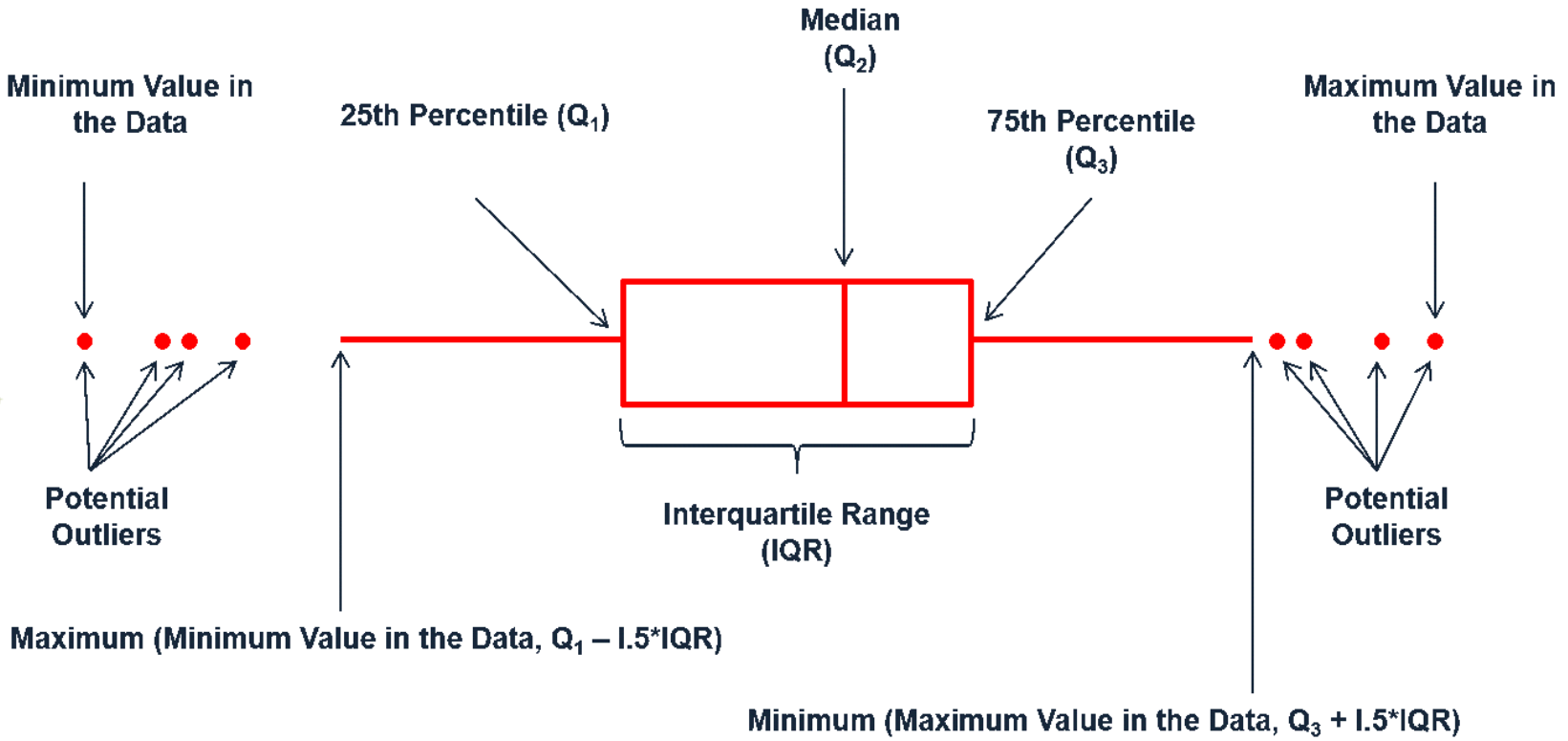
- 箱线图误差显示

```
hold off
err=y-b(1) + b(2)*x1 + b(3)*x2;
 pp=find(isnan(err)==0);
 err=err(pp); // get valid err
boxplot(err)


norm(err)
```



线性模型回归

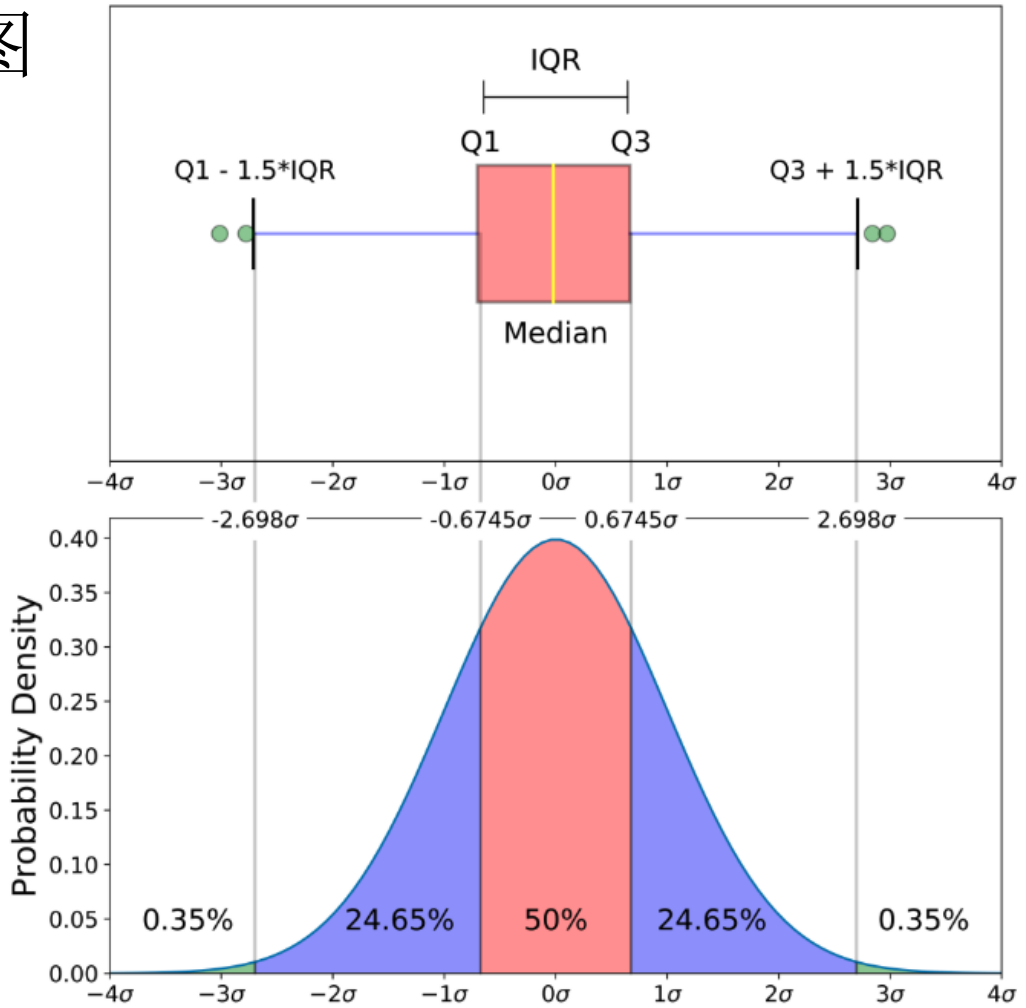
- 箱线图的解释



中位数，统计学中的专业名词，代表一个样本、种群或概率分布中的一个数值，其可将数值集合划分为数量相等的上下两部分。

线性模型回归

正态分布的箱线图

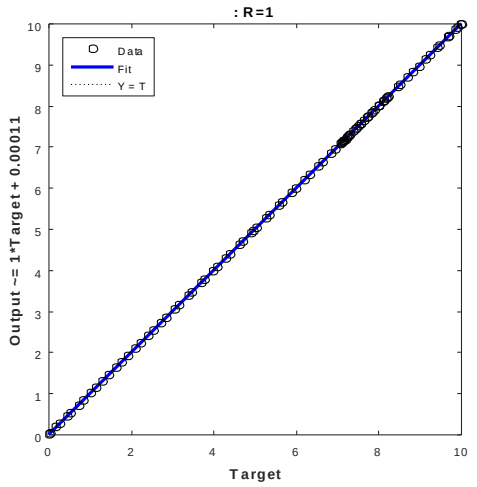


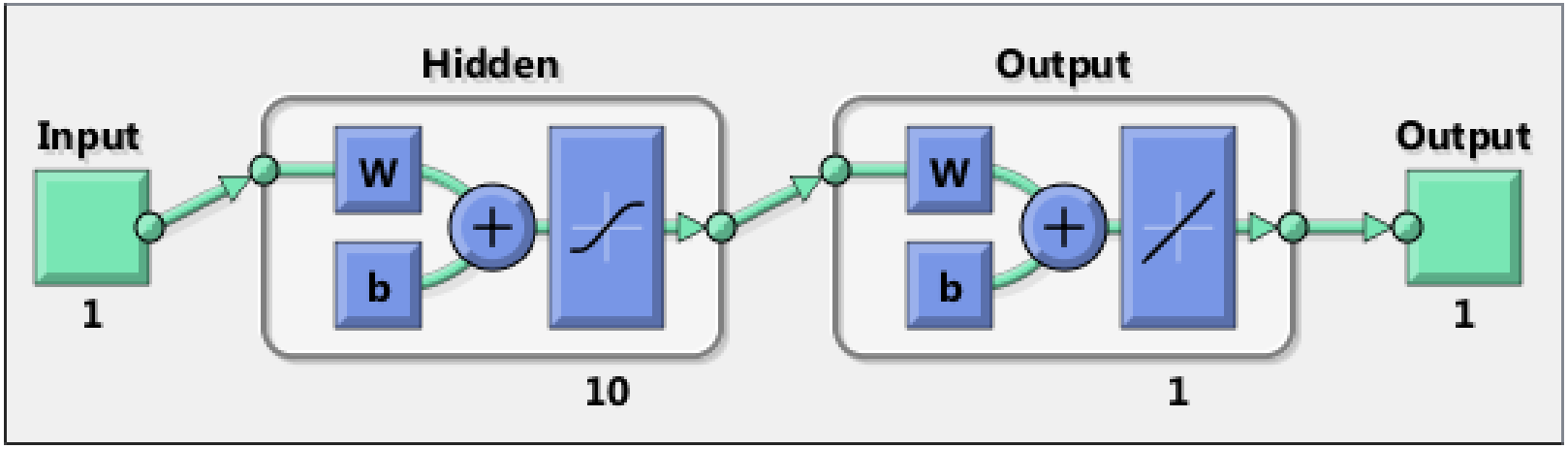
Comparison of a boxplot of a nearly normal distribution and a probability density function (pdf) for a normal distribution

前馈神经网络回归方法

- 前馈神经网络回归方法

```
[x,t] = simplefit_dataset;
net = feedforwardnet(20);
net = train(net,x,t); y = net(x);
[r,m,b] = regression(t,y);
plotregression(t,y)
```





局部加权线性回归

局部加权线性回归

- 线性回归方法的步骤

1. Fit θ to minimize $\sum_i (y^{(i)} - \theta^T x^{(i)})^2$.
2. Output $\theta^T x$.

局部加权线性回归

- 局部加权线性回归的步骤(LWLR)

1. Fit θ to minimize $\sum_i w^{(i)}(y^{(i)} - \theta^T x^{(i)})^2$.

2. Output $\theta^T x$.

- 权值的作用
  - 放大邻近点的贡献
  - 缩小甚至忽略远距离点的贡献

局部加权线性回归

- 权值形式的选择
  - 如果x是标量

$$w^{(i)} = \exp\left(-\frac{(x^{(i)} - x)^2}{2\tau^2}\right)$$

  - 如果x是向量

$$w^{(i)} = \exp(-(x^{(i)} - x)^T(x^{(i)} - x)/(2\tau^2))$$

或者

τ和Σ待定，称为带宽

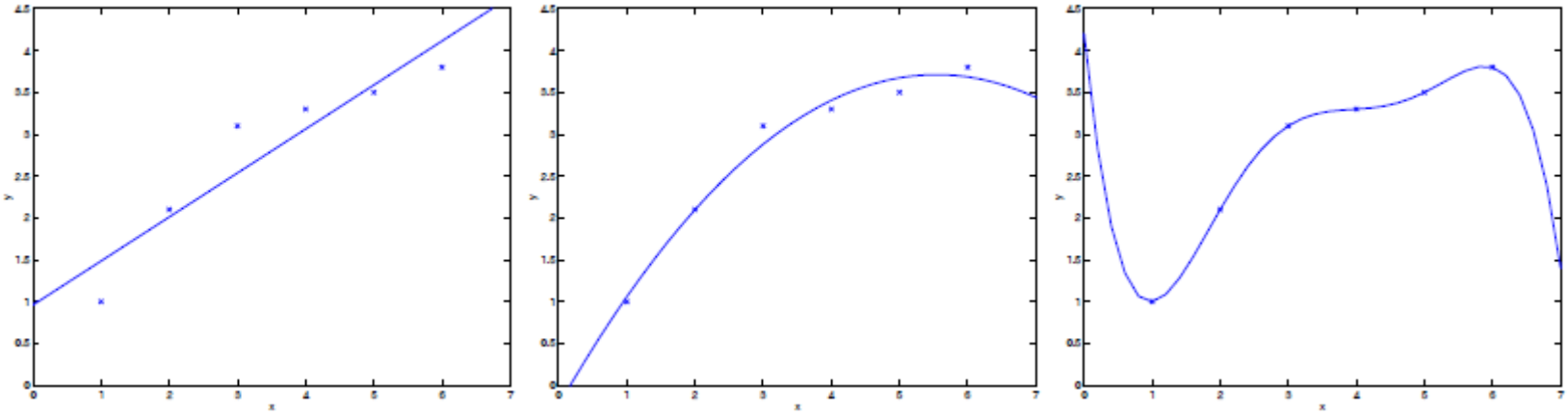
$$w^{(i)} = \exp(-(x^{(i)} - x)^T\Sigma^{-1}(x^{(i)} - x)/2)$$

非线性模型的多元回归

非线性模型的多元回归

- 实际数据可能并不适用线性预测模型

$\theta_0 + \theta_1 x$　　$\theta_0 + \theta_1 x + \theta_2 x^2$　　$\sum_{j=0}^{5} \theta_j x^j$



- 换个角度考虑：引入新特征 x^2

$\theta_0 + \theta_1 x_1 + \theta_2 x_2$

单一指标的
不同形式！

非线性模型的多元回归

- $\phi_j = x^j$, for $j = 0, 1, \cdots, n$
- $\phi_j = \frac{(x-\mu_j)^2}{2\sigma_j^2}$, for $j = 0, 1, \cdots, n$
- $\phi_j = \frac{1}{1+\exp(-s_j x)}$, for $j = 0, 1, \cdots, n$

$$\sum_{j=0}^{n} \theta_j \phi_j(x)$$

非线性模型的多元回归

- 多元回归对应的目标函数及偏导数
  - $J(\theta) = \sum_i (\sum_j \theta_j \phi_j(x^i) - y^i)^2$
  - $J(\theta) = \sum_i (\theta^T \phi(x^i) - y^i)^2$
  - $\frac{\partial y}{\partial \theta_j} \sum_i (\theta^T \phi(x^i) - y^i)^2 = \sum_i (\theta^T \phi(x^i) - y^i) \phi_j(x^i)$

非线性模型的多元回归

$$\Phi = \begin{pmatrix} \phi_0(x^1) & \phi_1(x^1) & \cdots & \phi_m(x^1) \\ \phi_0(x^2) & \phi_1(x^2) & \cdots & \phi_m(x^2) \\ \vdots & \vdots & \cdots & \vdots \\ \phi_0(x^n) & \phi_1(x^n) & \cdots & \phi_m(x^n) \end{pmatrix}$$

$$\theta = (\Phi^T \Phi)^{-1} \Phi^T \mathrm{y}$$

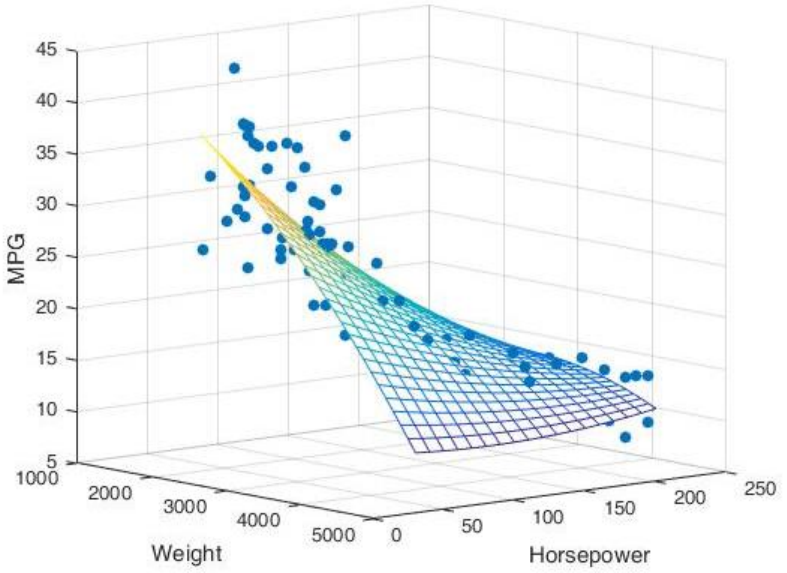
注：参考最小二乘的正则方程

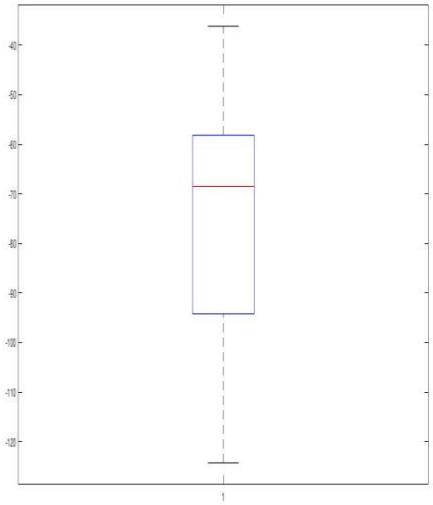
非线性多元回归的matlab示例

```
load carsmall
x1 = Weight;
x2 = Horsepower;
% Contains NaN data
y = MPG;



X = [ones(size(x1)) x1, x2, x1.*x1, x2.*x2, x1.*x2];
b = regress(y, X)

x1fit = min(x1):100:max(x1);
x2fit = min(x2):10:max(x2);
[X1FIT, X2FIT] = meshgrid(x1fit, x2fit);

YFIT = b(1) + b(2)*X1FIT +
    b(3)*X2FIT+b(4)*X1FIT.*X1FIT+b(5)*X2FIT.*X2FIT+b(6)*X1FIT.*X2FIT



scatter3(x1,x2,y,'filled')
hold on
```

非线性多元回归的matlab示例

```
mesh(X1FIT, X2FIT, YFIT)
xlabel('Weight')
ylabel('Horsepower')
zlabel('MPG')
```





Tips: mesh(X,Y,Z) creates a mesh plot, which is a three-dimensional surface that has solid edge colors and no face colors. The function plots the values in matrix Z as heights above a grid in the *x*-*y* plane defined by X and Y. The edge colors vary according to the heights specified by Z.

机器学习中的若干讨论

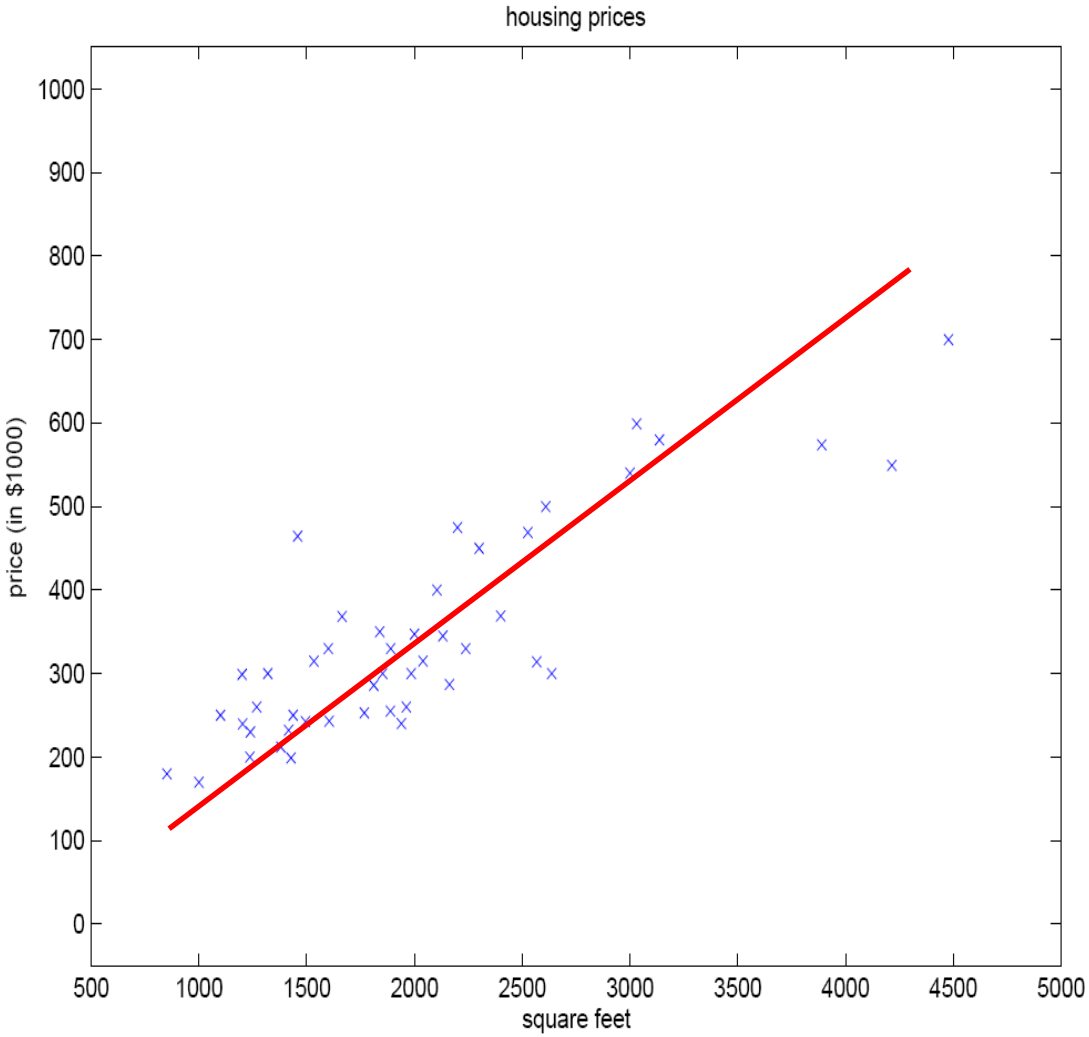
机器学习中的若干讨论

样本离群点

样本离群点

Living area (feet^2)	Price (1000$s)
2104	400
1600	330
2400	369
1416	232
3000	540
⋮	⋮

样本离群点

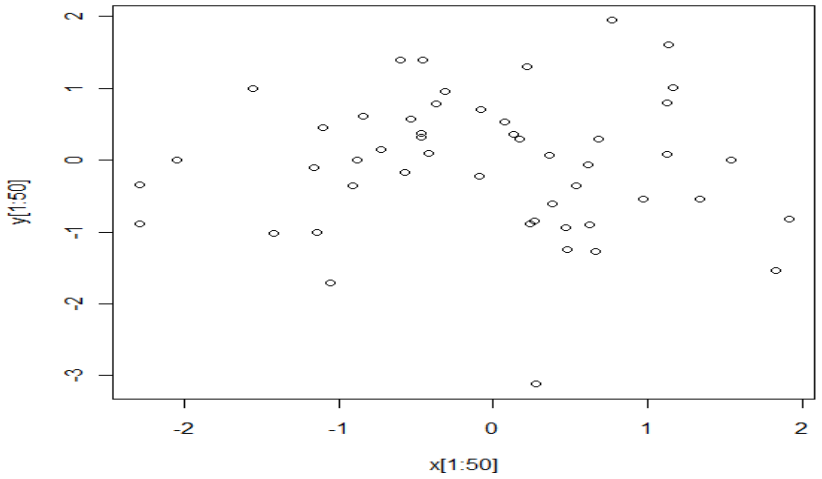


- 正常样本分布

样本离群点

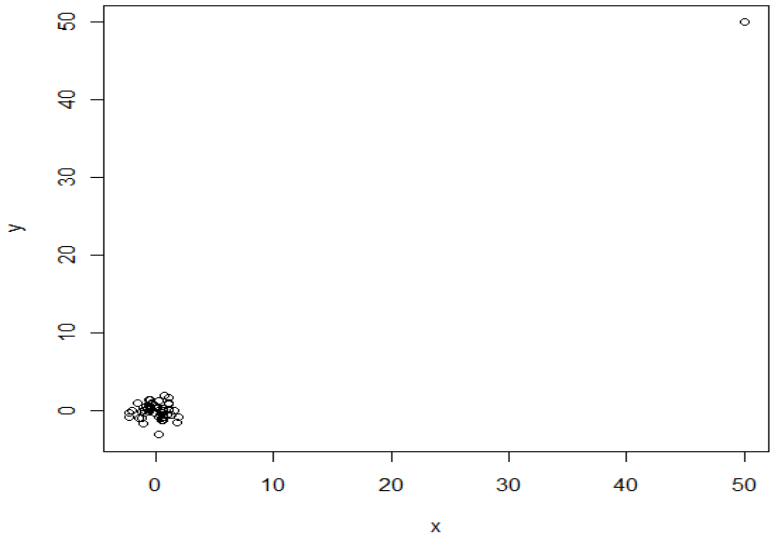
- Pearson sensitive for outliers
  - x=rnorm(50)
  - y=rnorm(50)
  - cor(x,y)
    - [1] -0.1545128
  - cor(x,y,method="spearman")
    - [1] -0.184922

// In R, rnorm(20) will generate 20 standard normal variates (mean=0 and sd=1 are the defaults)



样本离群点

- x[51]=50
- y[51]=50
- cor(x,y)
  - [1] 0.9775784
- cor(x,y,method="spearman")
  - [1] -0.1165611

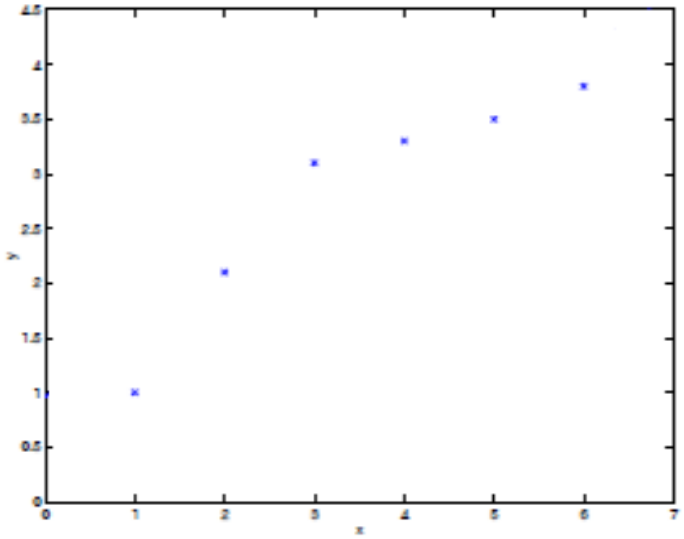


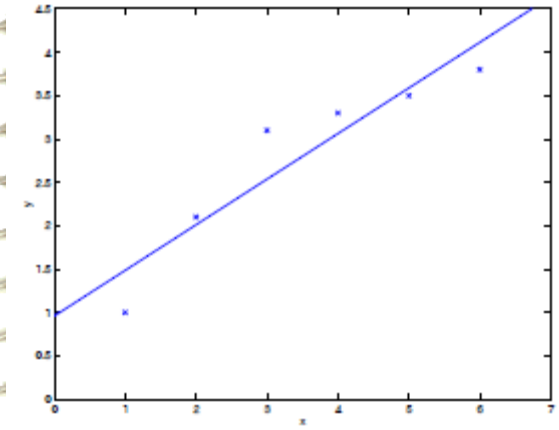
- 真的线性相关吗

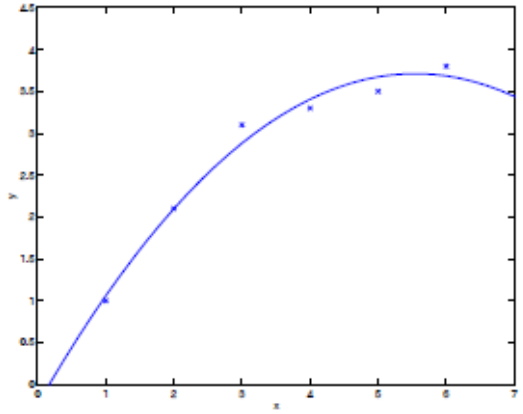
机器学习中的若干讨论

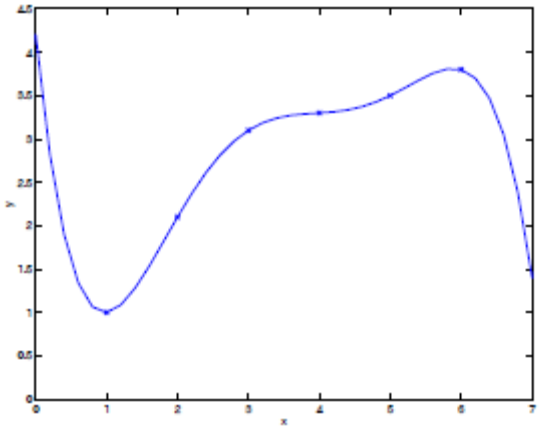
欠拟合与过拟合

欠拟合与过拟合

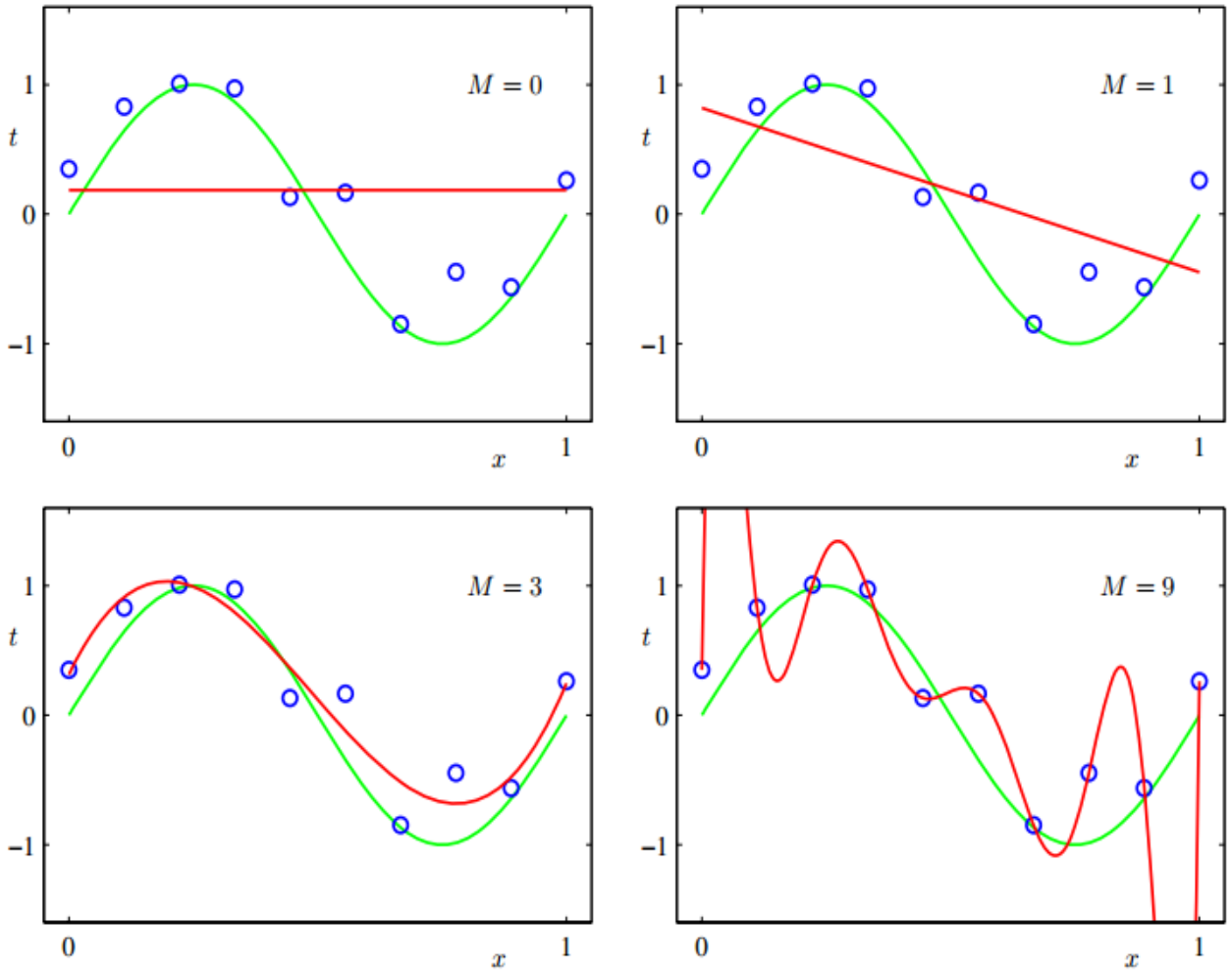








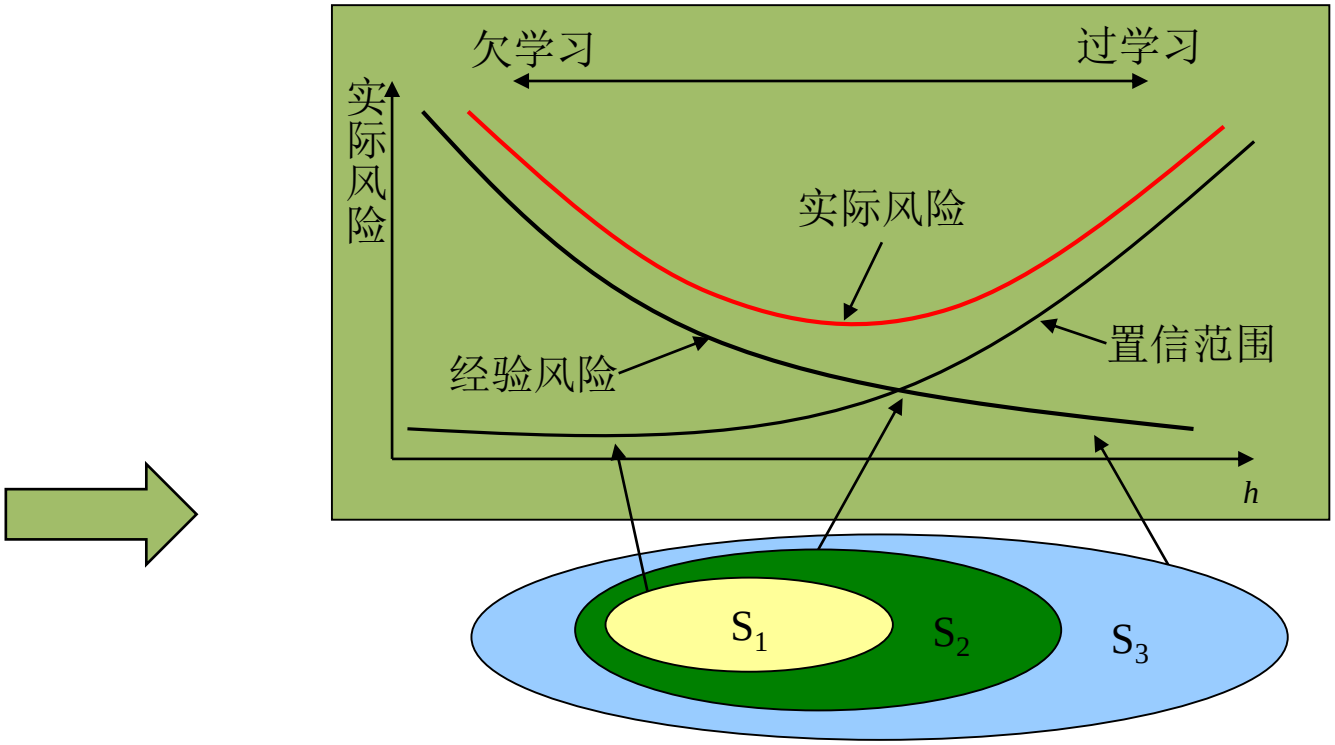
欠拟合与过拟合



欠拟合与过拟合

- 统计学习理论

结构风险最小化示意图



函数子集：$S_1 \subset S_2 \subset S_3$

VC维：$h_1 \le h_2 \le h_3$

欠拟合与过拟合

模型选择：

- 奥卡姆剃刀 (Occam‘s razor)法则：由14世纪方济会修士奥卡姆的威廉提出的逻辑学法则，他在《箴言书注》2卷15题说“切勿浪费多余功夫去做本可以较少功夫完成之事”。换言之，如果关于同一个问题有许多种理论，每一种都能作出同样准确的预言，那么应该挑选其中使用假定最少的。

-正则化

-特征选择

欠拟合与过拟合

- 正则化
  - 考虑最简单的线性回归模型，以平方误差为损失函数，并引入L_2范数正则化项防止过拟合，则有

岭回归

$$\min_{\boldsymbol{w}} \sum_{i=1}^{m} (y_i - \boldsymbol{w}^\top \boldsymbol{x}_i)^2 + \lambda \|\boldsymbol{w}\|_2^2$$

  - 将L_2范数替换为L_1范数，则有

LASSO回归

$$\min_{\boldsymbol{w}} \sum_{i=1}^{m} (y_i - \boldsymbol{w}^\top \boldsymbol{x}_i)^2 + \lambda \|\boldsymbol{w}\|_1$$

易获得稀疏解

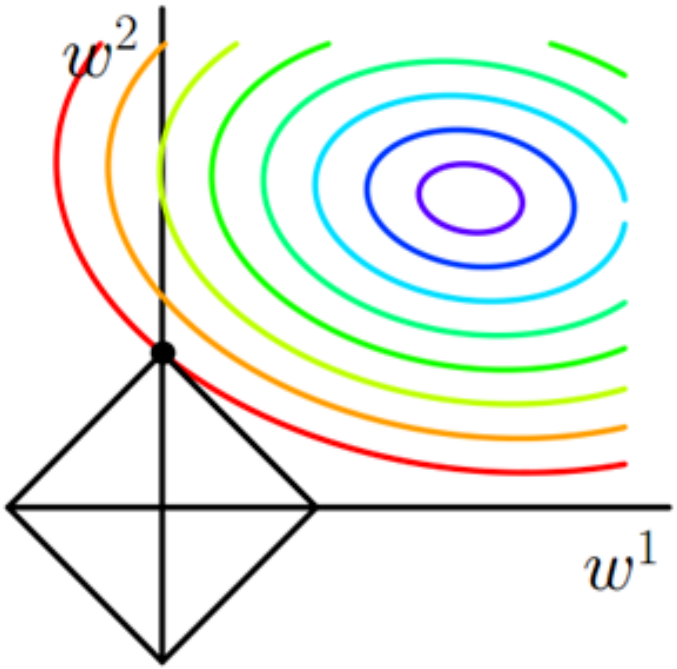
Ridge regression
[Tikhonov and Arsenin, 1977])

LASSO regression
[Tibshirani, 1996]

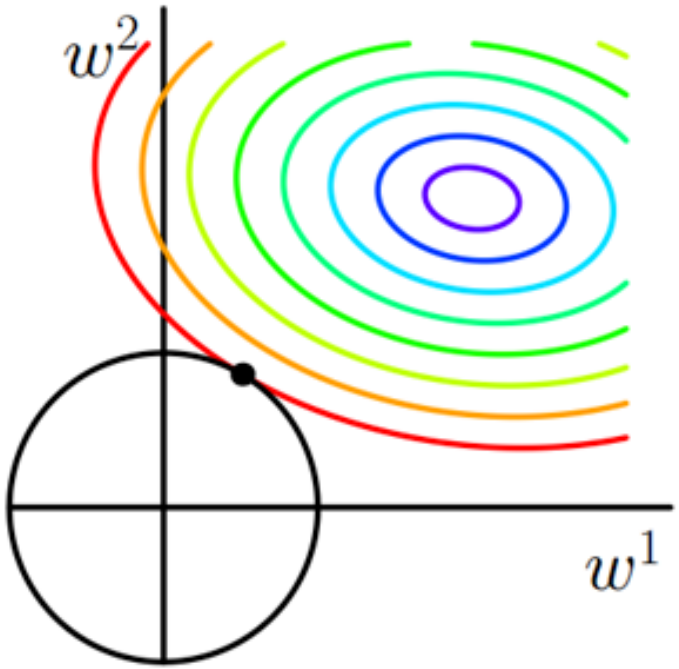
LASSO: Least absolute shrinkage and selection operator

https://www.cnblogs.com/shixisheng/p/7252816.html

欠拟合与过拟合



(a) ℓ_1-ball meets quadratic function. ℓ_1-ball has corners. It's very likely that the meet-point is at one of the corners.



(b) ℓ_2-ball meets quadratic function. ℓ_2-ball has no corner. It is very unlikely that the meet-point is on any of axes."

作业

- Retrieve the data named Boston Housing Data set from UCI Machine Learning Repository
  - https://archive.ics.uci.edu/ml/machine-learning-databases/housing/
- Build a regression model by BGD and SGD, respectively, with the retrieved housing data
  - Upload the snapshot of your screen and discuss the difference between BGD and SGD
- Have a try to build the regression model with any kind of artificial neural networks

作业

- 写出目标函数J(θ)的梯度推导过程

$$\begin{aligned}\frac{1}{2}(X\theta-\vec{y})^T(X\theta-\vec{y}) &= \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)})-y^{(i)})^2 \\ &= J(\theta)\end{aligned}$$

$$\begin{aligned}\nabla_\theta J(\theta) &= \nabla_\theta\frac{1}{2}(X\theta-\vec{y})^T(X\theta-\vec{y}) \\ &= X^TX\theta-X^T\vec{y}\end{aligned}$$

参考：矩阵导数@斯坦福大学机器学习课程讲义汇编

https://daiwk.github.io/assets/matrix+vector+derivatives+for+machine+learning.pdf

作业

针对某地区的房价预测，请你做如下工作：

1. 采用线性回归模型的梯度下降法，使用python或MATLAB进行求解（不能调用库里的模型，需包含训练、测试功能）：

（1）分别使用原始数据和归一化数据进行实验，以图或表的形式进行结果对比；

（2）训练过程分别采用批处理梯度下降BGD与随机梯度下降SGD方法实现，以图或表的形式进行结果对比。(注意：SGD实现需打乱数据顺序)

2. 调用库里模型分别用岭回归和LASSO回归模型实现房价预测，要求：

（1）以图或表的形式将线性模型和岭回归以及LASSO回归模型进行对比（归一化数据+SGD）；

（2）试对最终得到的三个模型的系数进行对比分析。

（提示：python实现时可以命令行输入pip install scikit-learn安装sklearn，from sklearn.linear_model import Ridge, Lasso 即可直接调用岭回归和Lasso回归模型；MATLAB实现时可以调用ridge函数与lasso函数直接获取岭回归和Lasso回归模型）