系统工程导论

开课单位：清华大学自动化系
主讲教师：胡坚明 副教授

模块一：系统建模

定量建模方法

黑箱建模

进入今天的课程

内容和重点

■本章内容

- 引言
- 回归分析方法
- 病态线性回归问题
- 黑箱建模方法在交通流量预测方面的应用
- 基于深度学习的建模及其在交通中的应用

黑箱建模

■基本概念

（1）什么是黑箱?

- 不清楚物理结构，或结构过于复杂
- 不了解机理规律，或机理过于复杂

$f = ?$ $f(x|\theta), \theta = ?$

（2）黑箱建模

- 根据观测的输入输出数据，寻找规律，建立数学模型
- 系统工程的基础性问题

Tsinghua University

■基本概念

（3）黑箱建模的数学描述

$$x = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}^T \rightarrow \boxed{f = ?} \rightarrow y$$

（4）黑箱建模（曲面拟合、回归）方法

- 选择由待定参数决定的一类函数 $f(x|\theta)$
- 获取样本数据 $x(t), y(t), t = 1, 2, \cdots$
- 拟合样本数据 $\min\limits_{\theta} \sum\limits_{t} \left(y(t) - f(x(t)|\theta) \right)^2$
- 获得经验模型 $y \approx f(x|\hat{\theta})$

■基本概念

（5）关键问题——选择什么函数类$f(x|\theta)$

- 单值？线性？多项式？指数？对数……
- 是否符合实际情况？

比如：能否有好的模型预测股票走势、NBA赛事？

- 基本指标：

（1） 能够表示或逼近哪些函数（逼近能力）？

（2） 是否便于确定模型参数？

（3） 所得模型是否好用？

黑箱建模——方法

■多项式逼近（Polynomial Approximation）

采用多项式函数逼近 $f(x|\theta)$

（1）数学形式

$$
\begin{aligned}
f(x|\theta) = a_0 &+ a_1 x_1 + \cdots + a_n x_n \\
&+ b_{11} x_1{}^2 + b_{12} x_1 x_2 + \cdots + b_{nn} x_n{}^2 \\
&+ c_{111} x_1{}^3 + c_{112} x_1{}^2 x_2 + \cdots + c_{nnn} x_n{}^3 \\
&+ \cdots
\end{aligned}
$$

$$\theta = \left[a_0, a_1, \cdots, a_n, b_{11}, b_{12}, \cdots, b_{nn}, c_{111}, c_{112}, \cdots, c_{nnn}, \cdots\right]^T$$

■多项式逼近（Polynomial Approximation）

(2) 逼近能力

- Weierstrass定理

设$f(x)$在$a \le x \le b$上连续及$\varepsilon > 0$，则一定存在一个多项式$P(x)$使得$|f(x) - P(x)| \le \varepsilon, a \le x \le b.$

- Tayler展开

设$f(x)$ 在 $|x - x_0| < R$ 内处处可导，则当 $|x - x_0| < R$ 时，有

$$f(x) = \sum_{n=0}^{\infty} a_n (x - x_0)^n,$$

$$其中: a_n = \frac{f^{(n)}(x_0)}{n!}, n = 0, 1, 2, \cdots.$$

黑箱建模——方法

■多项式逼近（Polynomial Approximation）

(2) 逼近能力

- 对任意阶可导函数，由泰勒定理保证
- 对连续函数，由魏尔斯特拉斯定理保证

【结论】对任何连续函数，存在可以和其任意靠近的多项式函数序列

黑箱建模——方法

■多项式逼近（Polynomial Approximation）

(3) 容易确定模型参数

- 采用最小二乘方法（Least Square）估计参数

$$\hat{\theta}=\left(\Phi\Phi^{T}\right)^{-1}\Phi Y^{T}$$

- 若估计残差是白噪声（零均值、正态分布），则LS估计是无偏、有效、具最小方差的估计。

- LS应用极为广泛。

黑箱建模——方法

■多项式逼近（Polynomial Approximation）

（3）容易确定模型参数（估计方法证明）

$$x=\left[x_1\ x_2\cdots x_n\right]^T \rightarrow\ f=?\ \rightarrow\ y$$

观测值 $x_1(t), x_2(t), \cdots, x_n(t), y(t)$

逼近多项式
$$
\begin{aligned}
f(x\mid\theta) &= a_0 + a_1x_1 + \cdots + a_nx_n \\
&+ b_{11}x_1{}^2 + b_{12}x_1x_2 + \cdots + b_{nn}x_n{}^2 \\
&+ c_{111}x_1{}^3 + c_{112}x_1{}^2x_2 + \cdots + c_{nnn}x_n{}^3 \\
&+ \cdots
\end{aligned}
$$

待确定参数 $\theta=\left[a_0,a_1,\cdots,a_n,b_{11},b_{12},\cdots,b_{nn},c_{111},c_{112},\cdots,c_{nnn},\cdots\right]^T_{n\times 1}$

向量表示 $\phi(t)=\left[1\ x_1(t)\cdots x_n(t)\ x_1(t)^2\ x_1(t)x_2(t)\cdots\right]^T_{n\times 1}$

$f\left(x(t)\mid\theta\right)=\theta^T\phi(t)$ – 某一个观测值， t=1, 2, . . . N

黑箱建模——方法

- 多项式逼近（Polynomial Approximation）
  - （3）容易确定模型参数（估计方法证明）

记 $\Phi = [\phi(1)\ \phi(2)\ \phi(3) \cdots]_{n \times N}$ ——对应N组观测值

$Y = [y(1)\ y(2)\ y(3) \cdots]_{1 \times N}$ ——对应N个观测值

逼近误差

$$\sum_t \left(y(t) - f(x(t) \mid \hat{\theta}) \right)^2 = \left(Y - \hat{\theta}^T \Phi \right)\left(Y - \hat{\theta}^T \Phi \right)^T$$

使逼近误差最小，即

$$\min \sum_t \left(y(t) - f(x(t) \mid \hat{\theta}) \right)^2 = \min \left(Y - \hat{\theta}^T \Phi \right)\left(Y - \hat{\theta}^T \Phi \right)^T$$

■多项式逼近（Polynomial Approximation）

(3) 容易确定模型参数（估计方法证明）

对参数 θ 求导，得到

$$\frac{\partial\left(Y-\hat{\theta}^{T}\Phi\right)\left(Y-\hat{\theta}^{T}\Phi\right)^{T}}{\partial\hat{\theta}}=0$$

$$-2\Phi Y^{T}+2\Phi\Phi^{T}\hat{\theta}=0$$

求得模型参数的最小二乘估计：

$$\hat{\theta}=\left(\Phi\Phi^{T}\right)^{-1}\Phi Y^{T}$$

证明

■多项式逼近（Polynomial Approximation）

（4）多项式逼近的不足

模型好用性和逼近能力有严重矛盾!

- 增加逼近能力需要增加多项式阶次;
- 但是，当阶次较高时，对样本点的高精度拟合，几乎完全不能保证对非样本点的高精度预测!

【结论】设法探寻其他形式的逼近函数，兼有良好逼近能力和模型好用性

黑箱建模——方法

■多项式逼近（Polynomial Approximation）

直观上（看一维函数在原点处的泰勒展开）

$$g(x) \approx g(0)$$

$$g(x) \approx g(0) + g'(0)x$$

$$g(x) \approx g(0) + g'(0)x + \frac{1}{2}g''(0)x^2$$

逼近范围增加

多项式逼近的本质：

只在一点研究问题，通过不断分析该点各阶变化趋势逼近任意远处的函数值。

每个基函数 $x, x^2, \cdots$ 在任何区域都起作用，拟合任何样本点都会影响对其他样本点的拟合情况！

■多项式逼近（Polynomial Approximation）

理论上

假定最好的模型是 $f(x|\theta^*)=\sum_{k=0}^{M}\theta_k^* x^k$

拟合样本获得的模型是 $f(x|\hat{\theta})=\sum_{k=0}^{M}\hat{\theta}_k x^k$

参数估计误差 $\Delta\theta_k=\theta_k^*-\hat{\theta}_k \ ,0\le k\le M$

两个函数间的偏差 $\Delta f(x)=\sum_{k=0}^{M}\Delta\theta_k x^k$

在某些点的偏差可能被大幅度放大！

■多项式逼近（Polynomial Approximation）

(5) 解决问题的途径

限制基函数起作用的区域，用局部基函数代替全局基函数。

$$f(x\mid\theta)=\sum_{k}\alpha_{k}\rho_{k}\left(x\mid\beta(k)\right)$$

- 用一类函数做基函数
- 其中每个基函数只在局部区域起作用！

Tsinghua University

■常用基函数

(1) 辐射基函数(RBF: Radial Basis Function)

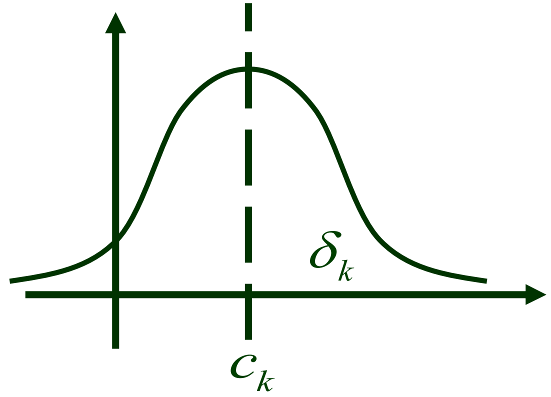
$$\rho\left(x \mid \beta(k)\right) = \eta\left(\frac{x - c_k}{\delta_k}\right)$$

例：高斯RBF

$$\rho\left(x \mid \beta(k)\right) = \exp\left(\frac{-\left\|x - c_k\right\|^2}{\delta_k^{\ 2}}\right)$$

距 c_k 较远处基函数不起作用

δ_k 控制基函数能起作用的范围



黑箱建模——方法

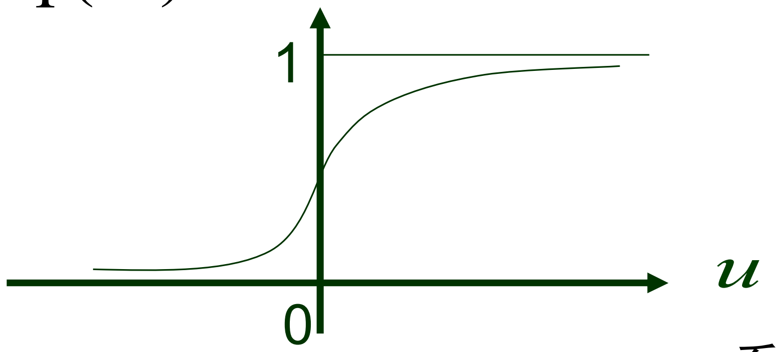
■基函数

（2）岭函数（Ridge Function）

$$\rho\left(x \mid \beta(k)\right)=\sigma\left(w_k^T x+d_k\right)$$

例：Sigmoid 函数

$$\sigma(u)=\frac{1}{1+\exp(-u)} \qquad \lim_{u\to-\infty}\sigma(u)=0, \lim_{u\to\infty}\sigma(u)=1$$



■基函数

(3) 两类常用人工神经网络

高斯Radial Basis 神经网络

$$\sum_k \alpha_k \exp\left(\frac{-\left\|x-c_k\right\|^2}{\delta_k{}^2}\right)$$

Sigmoid 神经网络

$$\sum_k \alpha_k \frac{1}{1+\exp\left(w_k^T x+d_k\right)}$$

■基函数

(4) 局部基函数性能

- 逼近能力?

  都能逼近任意连续函数

- 模型参数?

  可用基于导数的非线性规划算法，部分参数可用最小二乘公式

- 是否好用?

  有效克服多项式函数的缺陷

黑箱建模——方法

■基函数

(5) 逼近机理

① 划分连续小区间，限制基函数作用范围

考虑在区域（n维超矩形）

$$D=\left\{ x \middle| 0 \le x_i < 1,\ i=1,2,\cdots n \right\}$$

逼近连续函数

$$g(x): D \mapsto R$$

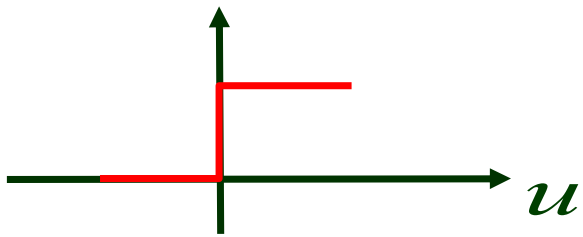
当n=1对应一维线段
当n=2对应二维正方形（边长为1）
当n=3对应三维立方体（边长为1）

黑箱建模——方法

用 I 代表 D 中左闭右开的一个立方体

例如 $I=\left\{ x \mid 0\le a_i \le x_i < b_i \le 1, i=1,2,\cdots n \right\}$

定义 $step(u)=\begin{cases} 1 & u\ge 0 \\ 0 & u<0 \end{cases}$



用 $step(x)$ 表示函数$hstep_I(x)$

$$hstep_I(x)=step\left(\sum_{i=1}^{n}\left(step\left(x_i-a_i\right)-step\left(x_i-b_i\right)\right)-n\right)$$

则有 $hstep_I(x)=\begin{cases} 1 & x\in I \\ 0 & x\notin I \end{cases}$

可选用 $hstep_I(x)$ 作基函数，它只在 I 区间内起作用，在其他区间不起作用，是局部基函数。

黑箱建模——方法

将 D 的每一个边 m 等分，可以把它分成

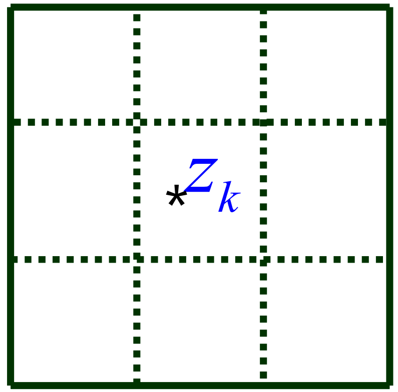
$M = m^n$ 个左闭右开的立方体的并集

例如 $n=2, m=3, M=9$

记这些小超立方体为 $I_1, I_2, \cdots, I_M$

记它们的中心点为 $z_1, z_2, \cdots z_M$

定义超钟型函数: $\hat{g}_m(x) = \sum_{k=1}^{M} g(z_k) hstep_{I_k}(x)$



则 $g(x) - \hat{g}_m(x) = g(x) - g(z_k), \forall x \in I_k$

利用连续性 $\lim_{m \to \infty} |g(x) - \hat{g}_m(x)| = 0, \forall x \in D$

说明：可以用超钟形函数逼近任意形式原函数！

黑箱建模——方法

② 辐射基函数类神经网络

$$\sum_k \alpha_k \eta\left(\frac{x-c_k}{\delta_k}\right) \qquad 如：\sum_k \alpha_k \exp\left(\frac{-\|x-c_k\|^2}{\delta_k{}^2}\right)$$

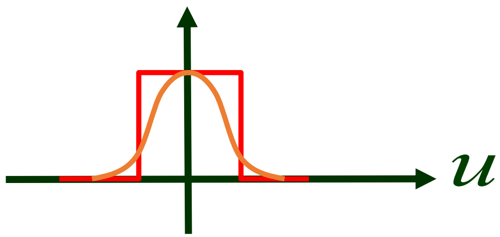
其中: 距 c_k 较远处基函数不起作用

δ_k 控制基函数能起作用的范围

与超钟形函数 $\hat{g}_m(x)=\sum_{k=1}^{M} g(z_k)hstep_{I_k}(x)$ 比较

选择 $c_k = z_k$

可以近似 $\alpha_k \approx g(z_k)$ $\eta\left(\frac{x-c_k}{\delta_k}\right) \approx hstep_{I_k}(x)$



$$\boxed{\hat{g}_m(x) \approx g(x)}$$

用辐射基函数近似hstep函数！

③ 岭函数类神经网络

$$\sum_k \alpha_k \sigma\left(w_k^T x + d_k\right)$$

其中 $\lim_{u\to-\infty}\sigma(u)=0, \lim_{u\to\infty}\sigma(u)=1$

只考虑$n=1$的情况 $\boxed{hstep_I(x)=step\left(\sum_{i=1}^{n}\left(step\left(x_i-a_i\right)-step\left(x_i-b_i\right)\right)-n\right)}$

$$hstep_{I_k}(x) = step\left(step\left(x-a_k\right)-step\left(x-b_k\right)-1\right)$$

$$= step\left(x-a_k\right)-step\left(x-b_k\right)$$

$$\hat{g}_m(x) = \sum_{k=1}^{M} g(z_k) hstep_{I_k}(x)$$

$$= \sum_{k=1}^{m} g(z_k)\left(step(x-a_k)-step(x-b_k)\right)$$

黑箱建模——方法

利用 $a_1 = 0, b_k = a_{k+1}, b_m = a_{m+1} = 1$

$$
\begin{aligned}
\hat{g}_m(x) &= \sum_{k=1}^{m} g(z_k)\big(step(x-a_k) - step(x-a_{k+1})\big) \\
&= g(z_1)\big(step(x-a_1) - step(x-a_2)\big) \\
&+ g(z_2)\big(step(x-a_2) - step(x-a_3)\big) \\
&+ g(z_3)\big(step(x-a_3) - step(x-a_4)\big) + \cdots \\
&= g(z_1) + \big(g(z_2) - g(z_1)\big) step(x-a_2) \\
&+ \big(g(z_3) - g(z_2)\big) step(x-a_3) + \cdots \\
&= g(z_1) + \sum_{k=2}^{m} \big(g(z_k) - g(z_{k-1})\big) step(x-a_k)
\end{aligned}
$$

黑箱建模——方法

$$\hat{g}_m(x) = g(z_1) + \sum_{k=2}^{m} \left(g(z_k) - g(z_{k-1}) \right) step(x - a_k)$$

注意对比两个函数

再利用 $\lim\limits_{u \to -\infty} \sigma(u) = 0, \lim\limits_{u \to \infty} \sigma(u) = 1$

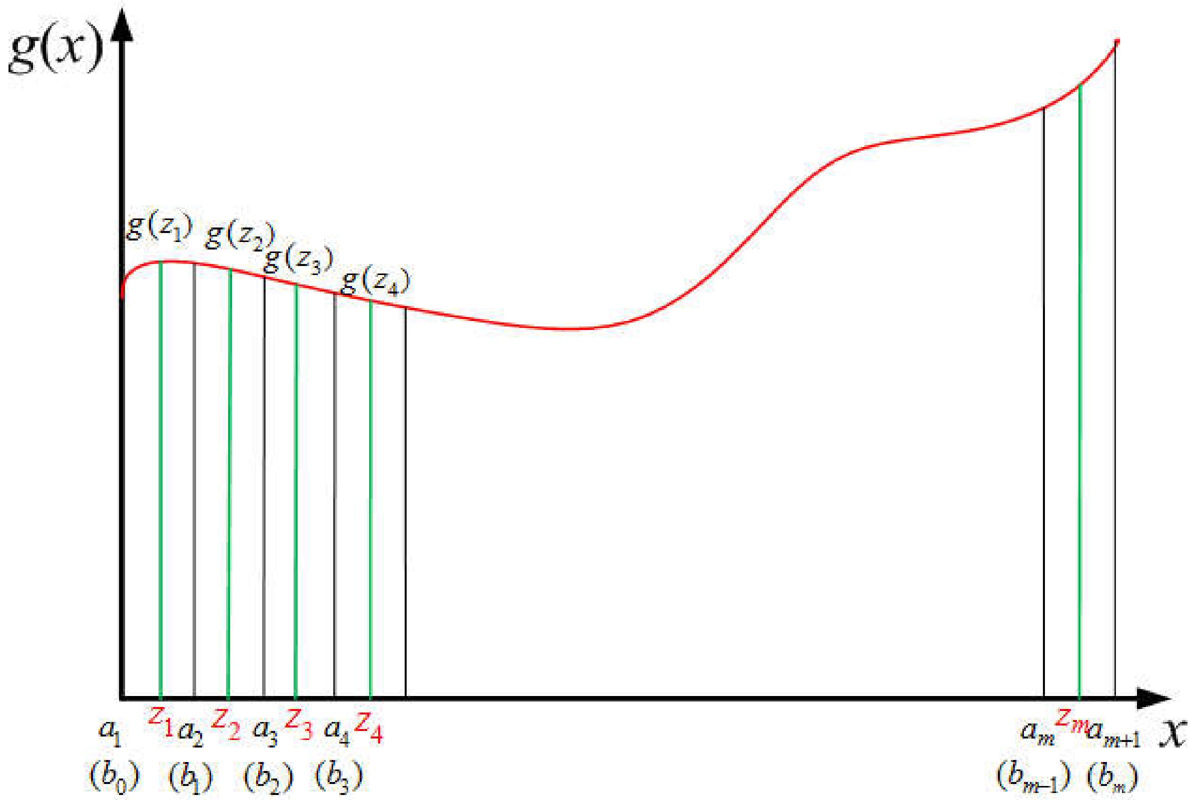
用岭函数（连续函数）代替阶跃函数

而且有 $\lim\limits_{w \to \infty} \sigma\left(w(x - a_k) \right) = step(x - a_k), \forall x \neq a_k$

可得 $\hat{g}_m(x) = \lim\limits_{w \to \infty} \left(g(z_1) + \sum_{k=2}^{m} \left(g(z_k) - g(z_{k-1}) \right) \sigma\left(w(x - a_k) \right) \right) \quad \forall x \neq a_k$

黑箱建模——方法

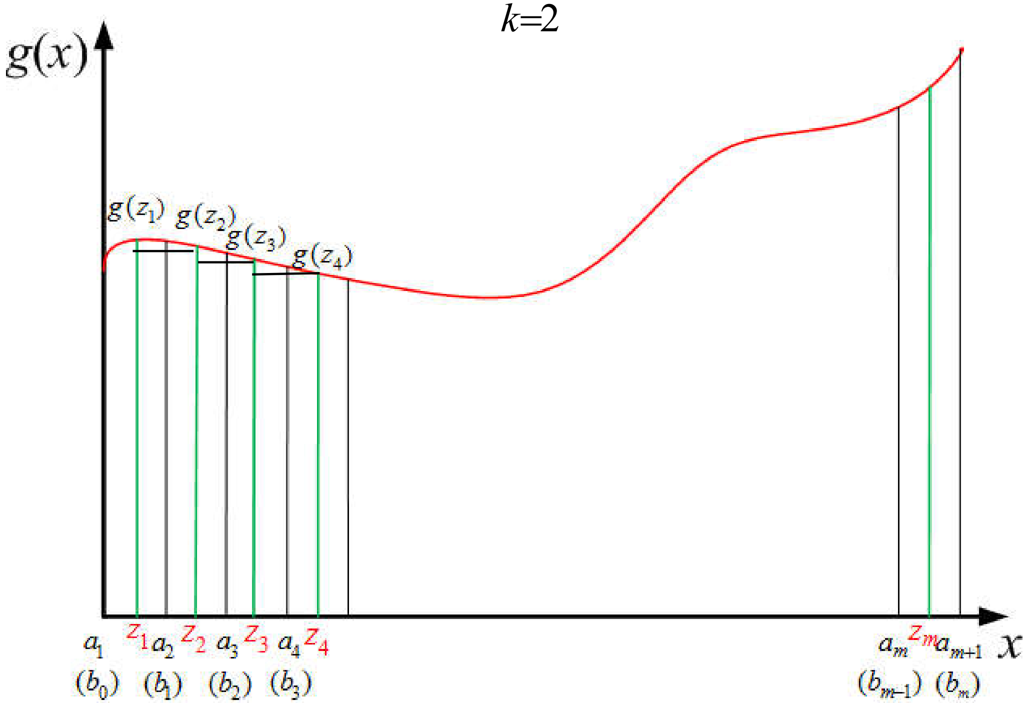
只考虑 $n=1$



黑箱建模——方法

当 w 很大，且 x 不等于任何 a_k 时

$$\hat{g}_m(x) \approx g(z_1) + \sum_{k=2}^{m} \left(g(z_k) - g(z_{k-1}) \right) \sigma\left(w(x - a_k) \right)$$



可以逼近原函数！

黑箱建模——方法

当 x 等于某个 a_k 时，例如 $x=a_3$

岭函数类神经网络

$$\hat{g}_m(x)\approx g(z_1)+\sum_{k=2}^{m}\left(g(z_k)-g(z_{k-1})\right)\sigma\left(w(x-a_k)\right)$$

右边近似等于：$g(z_1)+\left(g(z_2)-g(z_1)\right)+\left(g(z_3)-g(z_2)\right)\sigma(0)$

$$=g(z_2)+\left(g(z_3)-g(z_2)\right)\sigma(0)$$

$$\boxed{\sigma(0)=\frac{1}{2}}$$

则，$\lim\limits_{m\to\infty}\left(g(z_2)+\left(g(z_3)-g(z_2)\right)\sigma(0)\right)=g(a_3)$

仍然能逼近！

■基函数

④ 结论

- 辐射基函数类神经网络

  (用基函数显著大于0的部分)

$$\sum_k \alpha_k \eta\left(\frac{x-c_k}{\delta_k}\right) \approx \sum_k g(z_k) hstep_{I_k}(x)$$

- 岭函数类神经网络

  (用基函数接近1的部分)，所需要的节点数目 $M = m^n$

$$\sum_k \alpha_k \sigma\left(w_k x + d_k\right)$$

$$\approx g(z_1) + \sum_k \left(g(z_k) - g(z_{k-1})\right) \sigma\left(w_k x + d_k\right)$$

黑箱建模——回归分析

■背景

回归分析是一种常用的系统预测方法

黑箱建模

① 从一组数据出发，确定因变量和自变量之间的关系式（一元/多元、线性/非线性）

② 对关系式中的参数进行估计，并进行统计检验（假设检验）

③ 筛选自变量，即从大量自变量中找出影响显著的，剔除不显著的（抓主要矛盾）

④ 用求得的回归模型做预测

⑤ 对预测结果进行分析、评价

黑箱建模——回归分析

■线性回归问题

（1）问题描述

① 假定被解释变量（因变量）和回归变量（自变量）间近似

存在线性关系 $y \approx \theta^T x = \sum_{i=1}^{n} \theta_i x_i$

① 已有样本数据 $y(t), x(t), t = 1, 2, \cdots, N$

② 在最小二乘意义下，求解下列优化问题

$$\min_{\theta \in R^n} \sum_{t=1}^{N} \left(y(t) - \theta^T x(t) \right)^2$$

③ 从而确定参数的估计值

线性假定是否通用?

黑箱建模——回归分析

■线性回归问题

（2）可以从两种角度考虑线性模型是否合适

- 黑箱建模的观点

局部区域可用线性函数逼近一般函数

基于某些常识和经验，例如

$$\text{需电量} \approx f\,(\text{钢产量,铁产量, 煤产量...})$$

- 经验

一定的生产条件下，生产单位钢、铁、 煤等的耗电量基本不变

- 所以：需电量可近似为钢、铁、煤等主要产品产量的线性函数

黑箱建模——回归分析

■线性回归问题

（3）线性最小二乘回归的解

对优化问题 $\min\limits_{\theta\in R^n}\sum\limits_{t=1}^{N}\left(y(t)-\theta^T x(t)\right)^2$

记 $X=\left[x(1)\,x(2)\cdots x(N)\right]_{n\times N}$

$Y=\left[y(1)\,y(2)\cdots y(N)\right]_{1\times N}$

如果存在 $\left(XX^T\right)^{-1}$

最优解为 $\hat{\theta}=\left(XX^T\right)^{-1}XY^T$

■一元线性回归

（1）一元线性回归原理

一元线性回归处理两个系统变量 x 和 y 之间的线性因果关系，也就是拟合直线问题。

假设已经得到了x 和y 的若干数据对 x_i 和y_i，i=1,2, …,N称为样本点。如果x 和y 存在某种线性关系，则x 和y 可用

$$y = a + bx + \epsilon$$

表示，a 和 b 是待定系数， ϵ是随机变量。该模型为一元回归模型。 a 和 b 为回归系数。

将样本点带入，理论上可以通过 N个方程估计a 和 b 的值，记作$\hat{a}$ 和$\hat{b}$ 。

黑箱建模——回归分析

■一元线性回归

（1）一元线性回归原理

由最小二乘原理，估计目标是使“误差平方和”最小：

$$\sum_{i=1}^{N}\varepsilon_i^2=\sum_{i=1}^{N}(y_i-\hat{a}-\hat{b}x_i)^2=\min_{a,b}\sum_{i=1}^{N}(y_i-a-bx_i)^2$$

称$\hat{a}$和$\hat{b}$分别是a 和 b的最小二乘估计。

而$y=\hat{a}+\hat{b}x$称为一元回归方程。

其中 $\hat{a}=\bar{y}-\hat{b}\bar{x}$， $\hat{b}=\dfrac{\sum X_iY_i}{\sum X_i^2}=\dfrac{L_{xy}}{L_{xx}}$

■一元线性回归

（1）一元线性回归原理

为求取最小二乘估计$\hat{a}$ 和 $\hat{b}$，对 $Q=\sum_{i=1}^{N}(y_i-a-bx_i)^2$ 求一阶偏导数并令其为零。有：

$$\frac{\partial(\sum e_i^2)}{\partial a}=-2\sum(y_i-a-bx_i)=0$$

$$\frac{\partial(\sum e_i^2)}{\partial b}=-2\sum x_i(y_i-a-bx_i)=0$$

称为：正则方程。（canonical equation）

记 $X_i=[x_1-\bar{x},x_2-\bar{x},\cdots,x_N-\bar{x}]$ $\bar{x}=\frac{1}{N}\sum_{i=1}^{N}x_i$

$Y_i=[y_1-\bar{y},y_2-\bar{y},\cdots,y_N-\bar{y}]$, $\bar{y}=\frac{1}{N}\sum_{i=1}^{N}y_i$

解正则方程，有：

则 $\hat{a}=\bar{y}-\hat{b}\bar{x}$, $\hat{b}=\frac{\sum X_iY_i}{\sum X_i^2}=\frac{L_{xy}}{L_{xx}}$

从而可得到回归方程：

$$y=\hat{a}+\hat{b}x$$

Tsinghua University

■一元线性回归

（2）回归方程检验（验证回归模型有效性）

- 相关系数分解法（平方和分解）

因为 $L_{yy}=\sum_{i=1}^{N}(y_i-\bar{y})^2=\sum_{i=1}^{N}y_i^2-N\bar{y}^2=\sum_{i=1}^{N}(\hat{y}_i-\bar{y})^2+\sum_{i=1}^{N}(y_i-\hat{y}_i)^2$

记作 （总平方和TSS） （解释平方和ESS） （剩余平方和RSS）

即 TSS=ESS+RSS——平方和分解

定义

$$r^2=\frac{ESS}{TSS}=\frac{\sum_{i=1}^{N}(\hat{y}_i-\bar{y})^2}{\sum_{i=1}^{N}(y_i-\bar{y})^2}$$

表示总平方和——TSS中由 x 解释了或者说是回归解释了的部分，最小二乘法将使这个部分达到最大。

■一元线性回归

（2）回归方程检验（验证回归模型有效性）

- 相关系数分解法（平方和分解）

$$r^2 = \frac{ESS}{TSS} = \frac{\sum_{i=1}^{N}(\hat{y}_i - \overline{y})^2}{\sum_{i=1}^{N}(y_i - \overline{y})^2}$$

$r = \pm\sqrt{r^2}$ 称为相关系数，r 符号与 b 相同。

相关系数的大小能反映出 x 与 y 之间的线性密切程度。其值介于-1和1之间。当 r 接近于 0，说明 x 与 y 线性关系很弱，回归方程无实际意义。

Tsinghua University

■一元线性回归

(2）回归方程检验（验证回归模型有效性）

- 相关系数分解法（平方和分解）

也可以运用假设检验方法刻画回归方程的线性因果关系，构造统计量：

$$t=\frac{\sqrt{N-2r}}{\sqrt{1-r^{2}}}$$

定理4-1：设 r 是总体（x, y）的相关系数，当假设H_0: r =0成立时，统计量 t 服从自由度(degree of freedom)为 N-2 的 t 分布。

■一元线性回归

（2）回归方程检验（验证回归模型有效性）

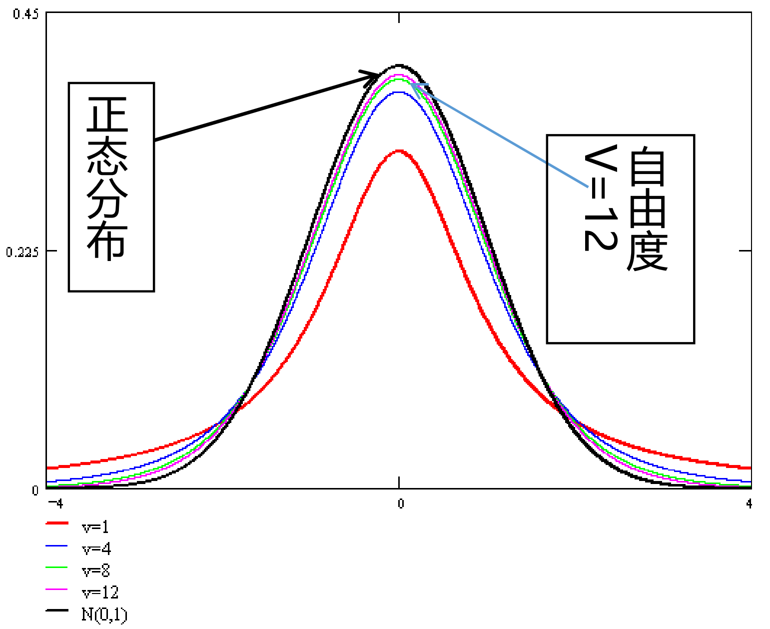
• 相关系数分解法（平方和分解）

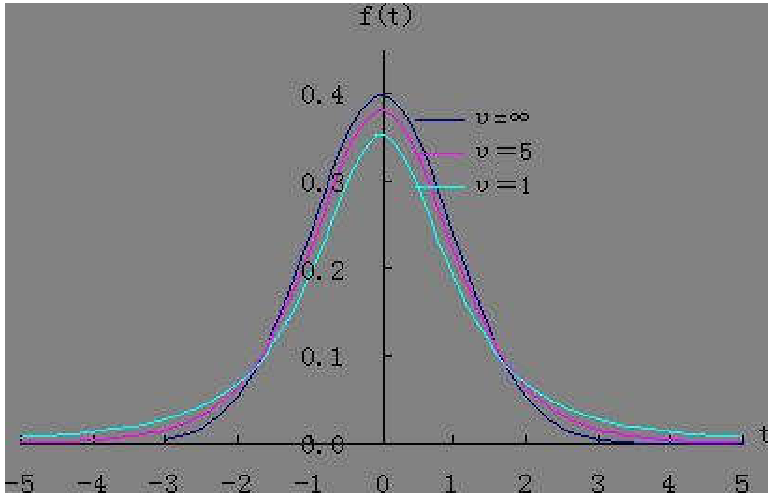
对于给定的显著性水平(significance level) $\alpha(0 \le \alpha \le 1)$，(称1-α 为置信度(degree of confidence)或置信水平）以及自由度（一般用 f 表示）（N-2)，查 t 分布表，得到相应的临界值 t_α ，从而对H0进行假设检验，即：

当 $t > t_\alpha$ 时，否定原假设(null hypothesis)（接受备则假设(alternative hypothesis)），认为 x 与 y 存在线性关系；

反之，当 $t \le t_\alpha$ 时，接收原假设，认为 x 与 y 不存在线性关系。

黑箱建模——回归分析





t分布是一簇曲线，其形态变化与n（确切地说与自由度ν）大小有关。自由度ν越小，t分布曲线越低平；自由度ν越大，t分布曲线越接近标准正态分布（normal distribution）曲线

Tsinghua University

■一元线性回归

（2）回归方程检验（验证回归模型有效性）

- F 检验法

定理4-2: 在假设H0: b=0成立时，TSS, ESS, RSS分别是自由度为$f_T = N-1, f_E = 1, f_r = N-2$的$\chi^2$变量，并且RSS与ESS相互独立，于是统计量

$$F = \frac{ESS/f_E}{RSS/f_R} = \frac{(N-2)ESS}{RSS}$$

服从自由度为(f_E, f_R)的 F 分布。

若n个相互独立的随机变量ξ1，ξ2，…，ξn，均服从标准正态分布（也称独立同分布（independent identically distributed, iid）于标准正态分布），则这n个服从标准正态分布的随机变量的平方和构成一新的随机变量，其分布规律称为 $\chi^2(n)$ 分布（Chi Square Distribution），其中参数 n 称为自由度。

Tsinghua University

■一元线性回归

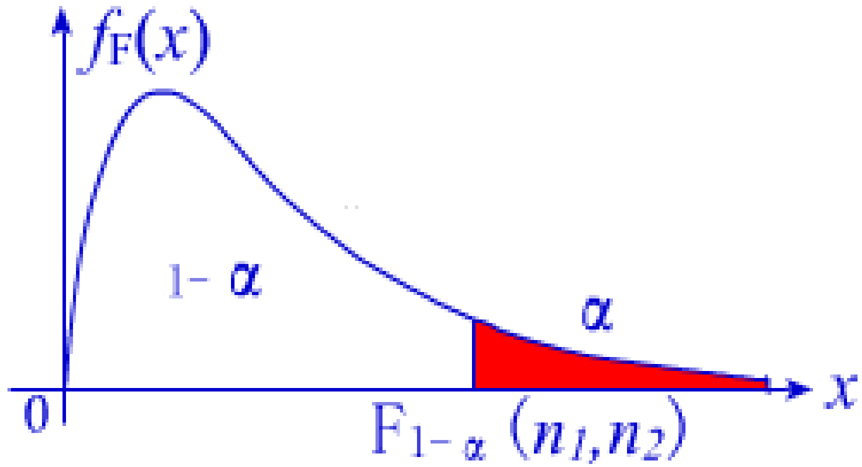
（2）回归方程检验（验证回归模型有效性）

- F 检验法

对于给定的显著性水平$\alpha(0 \le \alpha \le 1)$，以及自由度 (1，N-2)，查F分布表，得到相应的临界值F_α，从而对H_0进行假设检验，即：

当$F > F_\alpha$时，否定原假设，认为 x 与 y 存在线性关系；
当$F \le F_\alpha$时，接收原假设，认为 x 与 y 不存在线性关系。

黑箱建模——回归分析



F分布定义为:设X、Y为两个独立的随机变量，X服从自由度为m的 χ^2 分布，Y服从自由度为n的 χ^2 分布，这2 个独立的卡方分布分别除以各自的自由度以后的比率，即：F=(x/m)/(y/n)服从自由度为 (m, n) 的F-分布，其中第一自由度为m，第二自由度为n的F分布

黑箱建模——回归分析

■一元线性回归

（3）精度分析

设 S_δ 为 y 的剩余均方差，它表示变量 y 偏离回归直线的误差

$$S_\sigma = \sqrt{\frac{\sum_{i=1}^{N}(y_i-\hat{y})^2}{N-2}} = \sqrt{\frac{(1-r^2)L_{yy}}{N-2}}$$

给定显著性水平α，对某一x_0，相应的y_0将以 $(1-\alpha)$ 的概率落在下述区间（称为置信区间）

$$(\hat{y}_0 - Z_{\alpha/2}S_\delta,\ \hat{y}_0 + Z_{\alpha/2}S_\delta)$$

式中，$\hat{y}_0$ 是对应于 x_0 的 y_0 的预测值，$Z_{\alpha/2}$ 是标准正态分布上 $\alpha/2$ 百分位点的值。

黑箱建模——回归分析

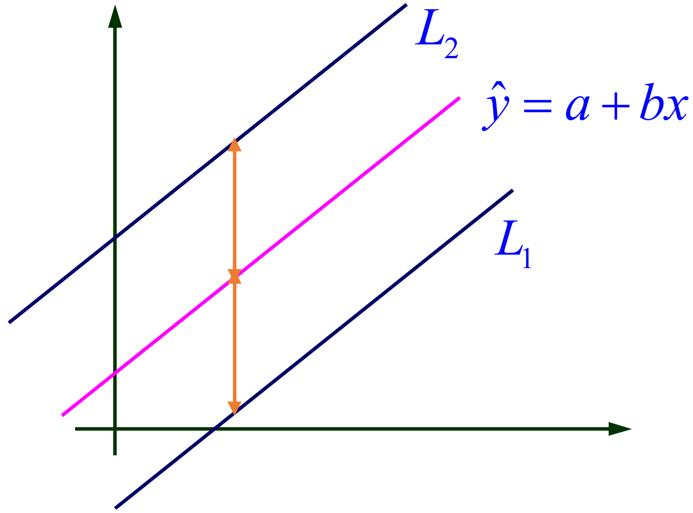
■一元线性回归

(3) 精度分析

以线性回归预测方程 $\hat{y}=a+bx$ 为中心线，上下做两条平行直线如下图所示

$$L_1: y_1 = a + bx - Z_{\alpha/2}S_\delta$$

$$L_2: y_2 = a + bx + Z_{\alpha/2}S_\delta$$



这表明，在全部可能出现的观察值 y_i 中，大约有 $100(1-\alpha)\%$ 的点落在 L_1 和 L_2 之间的范围内。显然，L_1 和 L_2 越靠近，预测精度越高。

■一元线性回归

（4）一元线性回归的步骤

① 数据平移和归一化压缩变换，得到新变量

$$x' = \frac{x - c_1}{d_1}, y' = \frac{y - c_2}{d_2}$$

② 计算新变量的 $\overline{x}', \overline{y}', L_{x'y'}, L_{x'x'}, L_{y'y'}, \hat{b}' = L_{x'y'} / L_{x'x'}, \hat{a}' = \overline{y}' - \hat{b}'\overline{x}'$

③ 带回原变量，得到 x 和 y 的回归方程 $\frac{y - c_2}{d_2} = \hat{a}' + \hat{b}' \frac{x - c_1}{d_1}$

④ 按下述公式进行统计检验

$$ESS' = \hat{b}'^2 L_{x'x'} = \hat{b}' L_{x'y'}$$

$$RSS' = L_{y'y'} - ESS'$$

$$F' = \frac{(N-2)ESS}{RSS}$$

⑤ 求置信区间，对回归直线进行预测。$S'_\delta = \sqrt{\frac{RSS'}{N-2}}, \quad S_\delta = d_2 S'_\delta$

黑箱建模——回归分析

■一元非线性回归

（1）函数变换线性化方法

$$y = a_0 b_0^{\ x}, \frac{1}{y} = a + \frac{b}{x}$$

① 根据机理或测点图形判断，选择一种易于线性化的非线性函数；

② 将自变量和因变量通过适当的数学变换得到线性关系；

③ 对变换后的自变量和因变量做线性回归分析；

④ 将得到的线性回归分析方程通过逆变换得到自变量和因变量的非线性关系。

常用的、易于线性化的非线性函数有幂函数、指数函数、双曲函数等。如幂函数 $y = a_0 x^b$ 可以等号两端取对数，并令 $Y = \ln y, a = \ln a_0, X = \ln x$，则有 $Y = a + bX$

黑箱建模——回归分析

■一元非线性回归

(2) 多项式变换线性化方法

当已知变量间存在某种非线性函数关系，但这种非线性函数关系又无法确定时，运用非线性函数的函数逼近论原理，可以采用多项式作为近似模型。

$$y = c_0 + c_1 x + c_2 x^2 + \cdots + c_n x^n$$

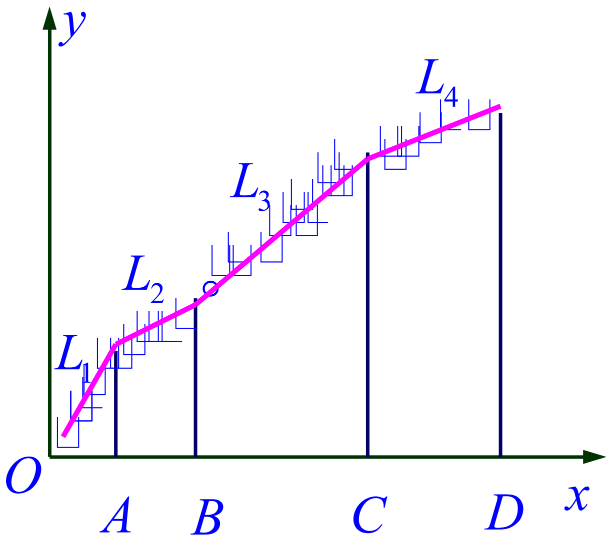
令 $t_1 = x, t_2 = x^2, \cdots, t_n = x^n$

则上式化为 $y = c_0 + c_1 t_1 + c_2 t_2 + \cdots + c_n t_n$

即 y 关于 n 个变量 $t_1, t_2, \cdots, t_n$ 多元线性回归模型。

■一元非线性回归

（3）分段线性化方法



$$L_1 : O - A : y = a_1 + b_1 x, \quad x \in OA$$
$$L_2 : A - B : y = a_2 + b_2 x, \quad x \in AB$$
$$L_3 : B - C : y = a_3 + b_3 x, \quad x \in BC$$
$$L_4 : C - D : y = a_4 + b_4 x, \quad x \in CD$$

对上述四个模型在各自的区域内分别进行求解，即可得到总体非线性回归模型。

Tsinghua University

■一元非线性回归

(4) 直接非线性回归分析方法

对于一般的非线性对应关系，可采用直接非线性回归分析。设变量 x 和 y 服从如下的非线性函数模型，即：

$$y = f(x)$$

则基于非线性回归模型的因变量预测值 $\hat{y}_i = f(x_i, \hat{\boldsymbol{\theta}})$

其中 $\hat{\boldsymbol{\theta}} = [\hat{\theta}_1 \ \hat{\theta}_2 \ \cdots \ \hat{\theta}_k]^T$ 是非线性回归模型中 k 个未知参数的估计值。则预测误差为

$$\varepsilon_i = y_i - \hat{y}_i = y_i - f(x_i, \hat{\boldsymbol{\theta}})$$

式中 x_i 和 y_i 是样本观测值。非线性回归就是找出 $\hat{\boldsymbol{\theta}}$ 的最优估计值 $\hat{\boldsymbol{\theta}}^*$。

■一元非线性回归

（4）直接非线性回归分析方法

最佳估计准则——最小二乘准则

预测误差的加权平方和为：

$$J(\hat{\boldsymbol{\theta}}) = \sum_i w_i \varepsilon_i^2 = \sum_i w_i [y_i - f(x_i, \hat{\boldsymbol{\theta}})]^2$$

式中， $w_i \in [0,1]$ 为回归权系数，表示第 i 个样本值的重要程度，w_i 愈大则表明样本值愈重要。（当无需区分各样本值重要性时，可取 $w_i = 1, i = 1, 2, \ldots$ 。）

于是，上述非线性回归问题转化为非线性规划问题：

$$\min_{\hat{\boldsymbol{\theta}}} J(\hat{\boldsymbol{\theta}}) = \min_{\hat{\boldsymbol{\theta}}} \sum_i w_i [y_i - f(x_i, \hat{\boldsymbol{\theta}})]^2$$

求解上述非线性规划问题，可得到最佳估计值 $\hat{\boldsymbol{\theta}}$ 。

举例

设有一元非线性函数 $f(x)=\dfrac{\theta_1 x}{x+\theta_2}$

当无加权时，有 $J(\hat{\boldsymbol{\theta}})=\sum_i\left[y_i-\dfrac{\theta_1 x}{x+\theta_2}\right]^2$

$$=\sum_i y_i^{\ 2}-2\hat{\theta}_1\sum_i\frac{x_i y_i}{x_i+\hat{\theta}_2}+\hat{\theta}_1^{\ 2}\sum_i\frac{x_i^{\ 2}}{(x_i+\hat{\theta}_2)^2}$$

则有 $\dfrac{\partial J(\hat{\boldsymbol{\theta}})}{\partial\hat{\theta}_1}=-2\hat{\theta}_1\sum_i\dfrac{x_i y_i}{x_i+\hat{\theta}_2}+2\hat{\theta}_1\sum_i\dfrac{x_i^{\ 2}}{(x_i+\hat{\theta}_2)^2}=0$

$$\frac{\partial J(\hat{\boldsymbol{\theta}})}{\partial\hat{\theta}_2}=2\hat{\theta}_1\sum_i\frac{x_i y_i}{(x_i+\hat{\theta}_2)^2}-2\hat{\theta}_1^{\ 2}\sum_i\frac{x_i^{\ 2}}{(x_i+\hat{\theta}_2)^3}=0$$

求解上述方程组，即可得到最佳回归参数 $\hat{\theta}_1^*,\hat{\theta}_2^*$ 。

(可采用数值分析方法或求解无约束非线性规划方法求解上述方程组)

黑箱建模——回归分析

■多元线性回归分析预测

（1）原理

假设有n个自变量 $x_1, x_2, \cdots x_n$ 和一个因变量 y，经过 N次试验，获得 N 个样本值为：

$$(y_i, x_{i1}, x_{i2}, \cdots x_{in}), i = 1, 2, \cdots, N$$

如果 y 与 $x_1, x_2, \ldots, x_n$ 均存在线性关系，则其回归模型可表示为：

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

式中，$\beta_0, \beta_1, \beta_2, \cdots, \beta_n$是未知的 n+1 个回归参数，我们的目的是运用上面 N 个样本值估计出这 n+1 个回归参数。

黑箱建模——回归分析

清華大學 Tsinghua University

将 N 个 样本值分别代入回归模型，则对应 N 个样本的回归预测值为：

$$
\begin{aligned}
\hat{y}_1 &= \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \cdots + \beta_n x_{1n} \\
\hat{y}_2 &= \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \cdots + \beta_n x_{2n} \\
&\vdots \\
\hat{y}_N &= \beta_0 + \beta_1 x_{N1} + \beta_2 x_{N2} + \cdots + \beta_n x_{Nn}
\end{aligned}
$$

回归预测误差为：

$$
\begin{aligned}
\varepsilon_1 &= y_1 - \hat{y}_1 \\
\varepsilon_2 &= y_2 - \hat{y}_2 \\
&\vdots \\
\varepsilon_N &= y_N - \hat{y}_N
\end{aligned}
$$

黑箱建模——回归分析

Tsinghua University

定义以下向量和矩阵：

$$\mathbf{Y}=\begin{bmatrix} y_1 & y_2 & \cdots & y_N \end{bmatrix}$$

$$\mathbf{X}=\begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_{11} & x_{21} & \cdots & x_{N1} \\ x_{12} & x_{22} & \cdots & x_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1n} & x_{2n} & \cdots & x_{Nn} \end{bmatrix}$$

$$\boldsymbol{\beta}=\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}$$

$$\boldsymbol{\varepsilon}=\begin{bmatrix} \varepsilon_1 & \varepsilon_2 & \cdots & \varepsilon_N \end{bmatrix}$$

则回归方程及回归预测误差可表示为：

$$\mathbf{Y}=\boldsymbol{\beta}^{\mathrm{T}}X+\boldsymbol{\varepsilon} \longrightarrow \hat{\boldsymbol{\beta}}=(\mathbf{X}\mathbf{X}^T)^{-1}(\mathbf{X}\mathbf{Y}^T)$$

黑箱建模——回归分析

为求取β 的最佳估计$\hat{\beta}^*$，运用最小二乘原理，回归预测误差的平方和为最小，即：

$$\min_{\hat{\boldsymbol{\beta}}} J(\hat{\boldsymbol{\beta}}) = \min_{\hat{\boldsymbol{\beta}}} \sum_{i=1}^{N} \varepsilon_i^2 = \min_{\hat{\boldsymbol{\beta}}} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

式中，$\hat{\boldsymbol{\beta}}$ 为$\boldsymbol{\beta}$ 的某一个估计。

将$J(\hat{\boldsymbol{\beta}})$ 对 $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \cdots, \hat{\beta}_n$ 求偏导数，令诸偏导数为0，并将 n 个样本值代入，可得方程

$$\mathbf{A}\hat{\boldsymbol{\beta}}\,|_{\hat{\boldsymbol{\beta}}=\hat{\boldsymbol{\beta}}^*} = \mathbf{B}$$

黑箱建模——回归分析

$$\mathbf{A}\hat{\boldsymbol{\beta}}\,|_{\hat{\boldsymbol{\beta}}=\hat{\boldsymbol{\beta}}^*} = \mathbf{B} \qquad (*)$$

其中:

$$\mathbf{A} = \begin{bmatrix} N & \sum_{i=1}^{N} x_{i1} & \sum_{i=1}^{N} x_{i2} & \cdots & \sum_{i=1}^{N} x_{in} \\ \sum_{i=1}^{N} x_{i1} & \sum_{i=1}^{N} x_{i1}^2 & \sum_{i=1}^{N} x_{i1}x_{i2} & \cdots & \sum_{i=1}^{N} x_{i1}x_{in} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{N} x_{in} & \sum_{i=1}^{N} x_{i1}x_{in} & \sum_{i=1}^{N} x_{i2}x_{in} & \cdots & \sum_{i=1}^{N} x_{in}^2 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \sum_{i=1}^{N} y_i \\ \sum_{i=1}^{N} x_{i1}y_i \\ \vdots \\ \sum_{i=1}^{N} x_{in}y_i \end{bmatrix} \quad \hat{\boldsymbol{\beta}} = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_n \end{bmatrix}$$

又有

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_{11} & x_{21} & \cdots & x_{N1} \\ x_{12} & x_{22} & \cdots & x_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1n} & x_{2n} & \cdots & x_{Nn} \end{bmatrix}$$

于是有

$$\mathbf{A} = \mathbf{X}\mathbf{X}^T$$

$$\mathbf{B} = \mathbf{X}\mathbf{Y}^T$$

于是 $\mathbf{A}\hat{\boldsymbol{\beta}}|_{\hat{\boldsymbol{\beta}}=\hat{\boldsymbol{\beta}}^*}=\mathbf{B}$ （*）式可以写成：

$$\boxed{(\mathbf{X}\mathbf{X}^T)\hat{\boldsymbol{\beta}}=(\mathbf{X}\mathbf{Y}^T)}$$

在 $(\mathbf{X}\mathbf{X}^T)$ 满秩的条件下，存在逆矩阵 $(\mathbf{X}\mathbf{X}^T)^{-1}$ ，所以

$$\hat{\boldsymbol{\beta}}=(\mathbf{X}\mathbf{X}^T)^{-1}(\mathbf{X}\mathbf{Y}^T)$$

上式是多元线性回归预测模型式 $\mathbf{Y}=\boldsymbol{\beta}^{\mathrm{T}}X+\boldsymbol{\varepsilon}$ 中参数 $\boldsymbol{\beta}$ 的最小二乘估计。

Tsinghua University

例题

平炉炼钢过程中，铁水的含碳量指标y与所加的两种矿石的量x_1，x_2有关，也与冶炼时间 x_3 有关。现场操作进行 49 组分析试验，如下表所示。试求出y与x_1，x_2，x_3的三元线性回归预测模型。

样本号	y	x_1	x_2	x_3	样本号	y	x_1	x_2	x_3
1	4.3302	2	18	50	26	2.7066	9	6	39
2	3.6458	7	9	40	27	5.6314	12	5	51
3	4.4830	5	14	46	28	5.8152	6	13	41
4	5.5468	12	3	43	29	5.1302	12	7	47
…					…				
21	4.6805	9	0	40	46	4.7115	4	10	45
22	3.1272	4	6	32	47	4.5310	10	5	40
23	2.6104	0	17	47	48	5.3637	3	17	64
24	3.7174	9	0	44	49	6.0771	4	15	72
25	3.8946	2	6	39					

完整数据见本章阅读材料

黑箱建模——回归分析

解:

①为求出 y 与x_1，x_2，x_3的三元线性回归预测模型，对数据预处理

$$y_i = \mu_0 + \beta_1(x_{i1} - \overline{x}_1) + \beta_2(x_{i2} - \overline{x}_2) + \beta_3(x_{i3} - \overline{x}_3) + \varepsilon_i, \quad i = 1, 2, \cdots, 49$$

②计算各变量的总和、算术平均值、交叉乘积和等。

$$\sum_{i=1}^{49} y_i = 224.5196 \qquad \overline{y} = 4.582$$

$$\sum_{i=1}^{49} x_{i1} = 259 \qquad \overline{x}_1 = 5.286$$

$$\sum_{i=1}^{49} x_{i2} = 578 \qquad \overline{x}_2 = 11.796$$

$$\sum_{i=1}^{49} x_{i3} = 2411 \qquad \overline{x}_3 = 49.204$$

$$\sum_{i=1}^{49} x_{i1}^2 = 2031 \qquad \sum_{i=1}^{49} x_{i2}^2 = 8572$$

$$\sum_{i=1}^{49} x_{i3}^2 = 124879 \qquad \sum_{i=1}^{49} x_{i1}x_{i2} = 2137$$

$$\sum_{i=1}^{49} x_{i1}x_{i3} = 12355 \qquad \sum_{i=1}^{49} x_{i2}x_{i3} = 29216$$

$$\sum_{i=1}^{49} x_{i1}y_i = 1180.3 \qquad \sum_{i=1}^{49} x_{i2}y_i = 1180.3$$

$$\sum_{i=1}^{49} x_{i3}y_i = 11292.72$$

黑箱建模——回归分析

由此得

$$l_{11}=\sum_{i=1}^{49}x_{i1}^2-\frac{1}{49}(\sum_{i=1}^{49}x_{i1})^2=662.0$$

$$l_{22}=\sum_{i=1}^{49}x_{i2}^2-\frac{1}{49}(\sum_{i=1}^{49}x_{i2})^2=1753.959$$

$$l_{33}=\sum_{i=1}^{49}x_{i3}^2-\frac{1}{49}(\sum_{i=1}^{49}x_{i3})^2=6247.959$$

$$l_{21}=l_{12}=\sum_{i=1}^{49}x_{i1}x_{i2}-\frac{1}{49}(\sum_{i=1}^{49}x_{i1})(\sum_{i=1}^{49}x_{i2})=-918.143$$

$$l_{31}=l_{13}=\sum_{i=1}^{49}x_{i1}x_{i3}-\frac{1}{49}(\sum_{i=1}^{49}x_{i1})(\sum_{i=1}^{49}x_{i3})=-388.857$$

$$l_{23}=l_{32}=\sum_{i=1}^{49}x_{i3}x_{i2}-\frac{1}{49}(\sum_{i=1}^{49}x_{i3})(\sum_{i=1}^{49}x_{i2})=776.041$$

$$l_{1y}=\sum_{i=1}^{49}x_{i1}y_i-\frac{1}{49}(\sum_{i=1}^{49}x_{i1})(\sum_{i=1}^{49}y_i)=-6.433$$

$$l_{2y}=\sum_{i=1}^{49}x_{i2}y_i-\frac{1}{49}(\sum_{i=1}^{49}x_{i2})(\sum_{i=1}^{49}y_i)=69.13$$

$$l_{3y}=\sum_{i=1}^{49}x_{i3}y_i-\frac{1}{49}(\sum_{i=1}^{49}x_{i3})(\sum_{i=1}^{49}y_i)=245.571$$

Tsinghua University

③计算系数矩阵 $\mathbf{A}$ 及 $\mathbf{A}^{-1}$ ，常数矩阵 $\mathbf{B}$

$$\mathbf{A}=\begin{bmatrix} N & 0 \\ 0 & \mathbf{L} \end{bmatrix}=\begin{bmatrix} 49 & 0 & 0 & 0 \\ 0 & 662.0 & -918.143 & -388.857 \\ 0 & -918.143 & 1753.959 & 776.041 \\ 0 & -388.857 & 776.041 & 6247.959 \end{bmatrix}$$

$$\mathbf{A}^{-1}=\begin{bmatrix} \frac{1}{49} & 0 & 0 & 0 \\ 0 & 0.005515 & 0.002984 & -0.00001623 \\ 0 & 0.002984 & 0.002112 & -0.00008345 \\ 0 & -0.00001623 & -0.00008345 & 0.0001694 \end{bmatrix}$$

$$\mathbf{B}=\begin{bmatrix} 224.5196 \\ -6.433 \\ 69.13 \\ 245.571 \end{bmatrix}$$

④ 计算回归系数和回归方程

$$\begin{bmatrix} \hat{\mu}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\beta}_3 \end{bmatrix} = \mathbf{A}^{-1}\mathbf{B} = \begin{bmatrix} 4.582 \\ 0.1604 \\ 0.1706 \\ 0.0359 \end{bmatrix}$$

由此可得回归方程

$$\hat{y} = 4.582 + 0.1604(x_1 - 5.286) + 0.1706(x_2 - 11.769) + 0.0359(x_3 - 49.204)$$

即 $\hat{y} = 0.7014 + 0.1604x_1 + 0.1706x_2 + 0.0359x_3$

Tsinghua University

■多元线性回归分析预测

（2）显著性检验

对于 n 组给定的样本数据 $(y_i, x_{i1}, x_{i2}, \cdots, x_{in}), i = 1, 2, \cdots N$ ，总可以按最小二乘原理建立一个多元线性回归方程。为验证回归方程的有效性，还须进行统计检验。

同样有：TSS=ESS+RSS——平方和分解公式

应用 F 统计检验方法原理如下。

■多元线性回归分析预测

（2）显著性检验

在假设H0: $\beta_1=\beta_2=\cdots=\beta_n=0$成立时，TSS, ESS,RSS分别是自由度为 N-1, n和 N-n-1 的 χ^2变量，并且RSS与ESS相互独立，于是统计量

$$F=\frac{ESS/f_E}{RSS/f_R}=\frac{(N-n-1)\cdot ESS}{n\cdot RSS}$$

服从自由度为(n, N-n-1) 的 F 分布。

对于给定的显著性水平α，以及自由度(n, N-n-1)，查 F 分布表，得到相应的临界值 F_α，从而对H0进行假设检验，即：

当$F>F_\alpha$ 时，否定原假设，认为 x 与 y 存在线性关系；
当$F\le F_\alpha$ 时，接收原假设，认为 x 与 y 不存在线性关系。

对前面例题，可以计算

$$TSS=\sum_{i=1}^{n}(y_i-\bar{y})^2=44.905$$

$$ESS=\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2=15.221$$

$$RSS=\sum_{i=1}^{n}(y_i-\hat{y}_i)^2=29.684$$

$$f_T=N-1=48$$

$$f_E=n=3$$

$$f_R=N-n-1=45$$

$$ESS/f_E=5.074$$

$$RSS/f_R=0.66$$

$$F=\frac{5.074}{0.66}=7.69$$

若给定 $\alpha=0.01$，查 F 分布表 $F_{0.01}(3,45)=4.25$ ，因此

$$F=7.69>F_{0.01}(3,45)=4.25$$

检验结果表明 y 与 $x_1,x_2,\cdots,x_n$ 间存在线性关系。

■多元线性回归分析预测

（3）预测精度

对于多元线性回归的预测模型

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

仿照一元线性回归问题的处理方法，可以用 $S_\delta = \sqrt{\dfrac{RSS}{N-n-1}}$

近似地表示 y 偏离回归平面的误差，于是，可以预测在各变量 $x_1, x_2, \cdots, x_n$ 取固定的样本值时，预测值 $\hat{y}$ 将以 $(1-\alpha)$ 的概率落在下述区域内，即

$$(\hat{y}_0 - Z_{\alpha/2} S_\delta, \hat{y}_0 + Z_{\alpha/2} S_\delta)$$

式中，$\hat{y}_0$ 是采用 $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \cdots \hat{\beta}_n$ 作为最佳回归参数时的预测值，$Z_{\alpha/2}$是标准正态分布上 $\alpha/2$百分位点的值。

黑箱建模——病态线性回归

■严格病态线性回归问题概述

（1）产生原因

样本数据中回归变量间严格线性相关！

（2）具体描述

存在$r\ (1\le r<n)$个线性无关的 n 维向量

$$a(1),\ a(2),\cdots a(r)$$

使得$a^T(i)x(t)=0,\ \forall 1\le i\le r,\ 1\le t\le N$

证明过程

■严格病态线性回归问题概述

由此产生的后果：

存在$m\left(\le n-r\right)$个彼此正交的单位向量

$b(i)\in R^n$，$1\le i\le m$ 使得每个 $x(t)$ 都可以

表示为它们的线性组合，即

$$x(t)=B_m z(t),\ 1\le t\le N$$

$$B_m=\left[b(1)\,b(2)\cdots b(m)\right]_{n\times m},\ B_m^T B_m=I_m$$

$$z(t)=\left[z_1(t)\,z_2(t)\cdots z_m(t)\right]^T_{m\times N}\in R^m,$$

并且不存在更少数目的向量满足上式

黑箱建模——病态线性回归

■严格病态线性回归问题概述

（3）对线性回归问题的影响

记 $Z=\left[z(1)\ z(2)\cdots z(N)\right]$

于是 $X=B_mZ$

$$XX^T=B_mZZ^TB_m^T$$

B_m是 $n\times m$ 矩阵, 秩等于 $m(m\leq n-r<n)$

XX^T的秩不会大于B_m 的秩, 没有逆矩阵!

进而无法利用公式求取参数向量!

■严格病态线性回归问题概述

（4）严格病态线性回归问题的处理方法

回归变量间严格线性相关，并不能否定需电量可近似为钢、铁、煤等主要产品产量的线性函数。

只是在这种情况下不能由样本数据正确估计所有的回归系数

但是仍可能确定需电量关于所有回归变量的线性函数（此时这种函数不唯一）

黑箱建模——病态线性回归

先求最大无关向量，再变换得到结果！

如果能确定满足

$$B_m^T B_m = I_m \quad x(t) = B_m z(t), 1 \le t \le N$$

的 B_m , $z(t), 1 \le t \le N$ 和最小的 m

由于 $y(t) \approx c^T x(t) = c^T B_m z(t) = d^T z(t)$

先估计 $\hat{d} = \left(ZZ^T\right)^{-1} ZY^T$ （一定存在）

再利用 $z(t) = B_m^T B_m z(t) = B_m^T x(t)$, $\forall t$

可建立 $y \approx \hat{d}^T z = \hat{d}^T B_m^T x = \hat{c}^T x$

但是，实际数据中存在的问题是:

1）无法确定自变量是线性相关的

$$a^T(i)x = 0,\ 1 \le i \le r$$

2）无法确定满足

$$B_m^T B_m = I_m \quad x(t) = B_m z(t), 1 \le t \le N$$

的B_m , $z(t), 1 \le t \le N$ 和最小的 m

分析：可否从n递减寻找符合要求的m

1）对任何$n \geq \hat{m} > m$ 存在 $b(m+1), \cdots, b(\hat{m})$

使 $B_{\hat{m}} = \left[B_m \ b(m+1) \cdots b(\hat{m})\right]$ 满足 $B_{\hat{m}}^T B_{\hat{m}} = I_{\hat{m}}$

令 $\hat{z}(t) = \left[\, z^T(t) \ 0 \cdots 0 \,\right]^T$ 就有 $x(t) = B_{\hat{m}} \hat{z}(t)$

所以下述优化问题的最优目标值为零

$$\min \sum_{t=1}^{N} \left(x(t) - Lv(t)\right)^T \left(x(t) - Lv(t)\right)$$

$$s.t. \ L^T L = I_{\hat{m}}, v(t) \in R^{\hat{m}}$$

最优解能满足 $x(t) = Lv(t), 1 \leq t \leq N$

实际上 $v(t)$ 是 $x(t)$ 向 $\hat{m}$ 子空间上的投影

2）一旦$\hat{m} < m$，下述优化问题的最优目标值一定大于零。

$$\min \sum_{t=1}^{N} \left(x(t) - Lv(t) \right)^T \left(x(t) - Lv(t) \right)$$

$$s.t.\ L^T L = I_{\hat{m}},\ v(t) \in R^{\hat{m}}$$

■严格病态线性回归问题概述

（4）严格病态线性回归问题的处理方法

从$\hat{m}=n-1$开始逐渐减少$\hat{m}$，依次求解

$$\min \sum_{t=1}^{N}\left(x(t)-Lv(t)\right)^{T}\left(x(t)-Lv(t)\right)$$

$$s.t.\ \ L^{T}L=I_{\hat{m}},\ v(t)\in R^{\hat{m}}$$

找到使最优目标值等于零的最小m

其对应的最优解就可用作所需要的

$$B_{m} \text{ 和 } z(t), 1\le t\le N$$

黑箱建模——病态线性回归

■实际病态线性回归问题及处理方法

（1）问题描述

更接近实际情况的病态线性回归问题

存在 $r\ (1 \le r < n)$个线性无关的 n 维向量

$$a(1), a(2), \cdots a(r)$$

使得 $a^T(i)x(t) \approx 0,\ \forall 1 \le i \le r,\ 1 \le t \le N$

近似线性, 称为“病态线性”

■实际病态线性回归问题及处理方法

（2）问题后果

存在$m(\leq n-r)$个彼此正交的单位向量

$b(i)\in R^n$，$1\leq i\leq m$ 使得每个$x(t)$ 都可以

近似地表示为它们的线性组合，即

$$x(t)\approx B_m z(t),\ 1\leq t\leq N$$

$$B_m=\left[b(1)\,b(2)\cdots b(m)\right],\ B_m^T B_m=I_m$$

并且不存在更少数目的向量满足上式

黑箱建模——病态线性回归

■实际病态线性回归问题及处理方法

（2）问题后果

对线性回归问题的影响：

由于 $XX^T \approx BZZ^T B^T$ 接近奇异

其逆矩阵即使存在，参数估计值

$$\hat{c} = \left(XX^T\right)^{-1} XY^T$$

也很不可靠！　（为什么？）

■实际病态线性回归问题及处理方法

（3）理由

用 $c_* \in R^n$ 表示真正的(或最好的)模型参数

最小的样本误差为 $\eta = Y - c_*^T X$

于是 $Y^T = X^T c_* + \eta^T$

从而可得参数估计误差

$$c_* - \hat{c} = c_* - \left(XX^T\right)^{-1} XY^T$$

$$= c_* - \left(XX^T\right)^{-1} X(X^T c_* + \eta^T)$$

$$= -\left(XX^T\right)^{-1} X\eta^T$$

黑箱建模——病态线性回归

清華大學 Tsinghua University

Review: 实对称矩阵性质

A为 n 阶实对称矩阵，则：

1）A的特征根皆为实数。

2） R^n 中属于A的不同特征值的特征向量必正交。

3）存在正交矩阵C，使得 $C^{-1}AC$ 为对角阵。

来看实对称矩阵的正交对角化：

$$Q = \left[q(1)\ q(2) \cdots q(n) \right] \in R^{n \times n}$$

存在 $\lambda_i \in R,\ 1 \le i \le n$

满足 $Q^T Q = I_n \qquad \lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_n \ge 0,$

$$\left(XX^T \right) q(k) = \lambda_k q(k), 1 \le k \le n$$

等价于 $XX^T = Q \Lambda Q^T$

$$\Lambda = diag\left\{ \lambda_1, \lambda_2, \cdots, \lambda_n \right\}$$

黑箱建模——病态线性回归

其实

1）$\lambda_1,\lambda_2,\cdots,\lambda_n$ 是矩阵 XX^T
依从大到小的顺序排列的特征值

2）$q(1),q(2),\cdots,q(n)$
分别是对应的特征向量

黑箱建模——病态线性回归

因为 $Q^TQ=I_n \quad Q\in R^{n\times n}$

所以 $Q^{-1}=Q^T \quad \left(Q^T\right)^{-1}=Q$

$$\left(XX^T\right)^{-1}=\left(Q\Lambda Q^T\right)^{-1}=\left(Q^T\right)^{-1}\Lambda^{-1}Q^{-1}=Q\Lambda^{-1}Q^T$$

$$c_*-\hat{c}=-Q\Lambda^{-1}Q^TX\eta^T \qquad \boxed{c_*-\hat{c}=-\left(XX^T\right)^{-1}X\eta^T}$$

XX^T 接近奇异，本质上就是某些特征根远比其他特征根小

由于 $\Lambda^{-1}=diag\left\{\lambda_1^{-1},\lambda_2^{-1},\cdots,\lambda_n^{-1}\right\}$

参数估计误差可能被严重放大!

黑箱建模——病态线性回归

Tsinghua University

举例：

对需电量例子的特征根数据：

k	λ_k
1	1180.923
2	46.791
3	9.804
4	3.453
5	2.133
6	1.068
7	0.518
8	0.469
9	0.121
10	0.093
11	0.004

黑箱建模——病态线性回归

■实际病态线性回归问题及处理方法

（4）可采取的办法

类似严格病态问题的处理方法：

从 $\hat{m}=n-1$ 开始逐渐减少 $\hat{m}$ ，求解

$$\min \sum_{t=1}^{N}\left(x(t)-Lv(t)\right)^{T}\left(x(t)-Lv(t)\right)$$

$$s.t.\ \ L^{T}L=I_{\hat{m}},\ v(t)\in R^{\hat{m}}$$

当最优目标值显著增加时停止，可得满足

$$B_m^T B_m = I_m \quad x(t)\approx B_m z(t), 1\le t\le N$$

的 B_m , $z(t), 1\le t\le N$ 和最小的 m

再借助 $Z=\left[z(1)\ z(2)\cdots\ z(N)\right]$ 建模

黑箱建模——病态线性回归

上述优化问题的解析解：

对向量逼近的已知结论

对任何 m, $Q_m=\left[q(1)\,q(2)\cdots q(m)\right]$

和 $z(t)=Q_m^T x(t)$, $1\le t\le N$

是下述优化问题的最优解

$$\min \sum_{t=1}^{N}\left(x(t)-Lv(t)\right)^T\left(x(t)-Lv(t)\right)$$

$$s.t.\ \ L^T L=I_m,\ v(t)\in R^m$$

将最优解 $z(t)=Q_m^T x(t)$ 代入目标函数，得

$$\sum_{t=1}^{N}\left(x(t)-Q_m Q_m^T x(t)\right)^T\left(x(t)-Q_m Q_m^T x(t)\right)$$

$$=\sum_{t=1}^{N} x^T(t)\left(I_n-Q_m Q_m^T\right)\left(I_n-Q_m Q_m^T\right)x(t)$$

$$=\sum_{t=1}^{N} x^T(t)x(t)-\sum_{t=1}^{N} x^T(t)Q_m Q_m^T x(t)$$

$$\boxed{=\sum_{i=1}^{n}\lambda_i-\sum_{i=1}^{m}\lambda_i=\sum_{i=m+1}^{n}\lambda_i}$$

根据特征值和特征向量性质（迹），有

$$\sum_{t=1}^{N} x^T(t)Q_m Q_m^T x(t)=\sum_{i=1}^{m}\lambda_i,\ \sum_{t=1}^{N} x^T(t)x(t)=\sum_{i=1}^{n}\lambda_i$$

所以上述优化问题的最优目标值为 $\sum_{i=m+1}^{n}\lambda_i$

黑箱建模——病态线性回归

■实际病态线性回归问题及处理方法

所以，处理病态线性回归问题的基本方法是

找出使逼近误差 $\sum_{i=m+1}^{n} \lambda_i$

可以接受的最小正整数 m

确定 $Q_m = [q(1)\ q(2) \cdots q(m)]$

$$Z = Q_m^T X$$

$$\hat{d} = \left(ZZ^T\right)^{-1} ZY^T$$

$$\hat{c} = Q_m \hat{d}$$

得到 $y \approx \hat{d}^T z = \hat{d}^T Q_m^T x = \hat{c}^T x$

黑箱建模——病态线性回归

■**实际病态线性回归问题及处理方法**

用 $d_* \in R^m$ 表示 $y(t) \approx d^T z(t)$ 的最好参数

最小样本误差为 $\mu = Y - d_*^T Z$ 即 $Y^T = \mu^T + Z^T d_*$

参数估计误差（$Q = [Q_m \; q(m+1) \cdots q(n)]$）

$$d_* - \hat{d} = d_* - \left(ZZ^T\right)^{-1} ZY^T = d_* - \left(ZZ^T\right)^{-1} Z\left(\mu^T + Z^T d_*\right)$$

$$= -\left(ZZ^T\right)^{-1} Z\mu^T = -\left(Q_m^T XX^T Q_m\right)^{-1} Z\mu^T$$

$$= -\left(Q_m^T Q \Lambda Q^T Q_m\right)^{-1} Z\mu^T = -\left(\begin{bmatrix} I_m & 0 \end{bmatrix} \Lambda \begin{bmatrix} I_m \\ 0 \end{bmatrix}\right)^{-1} Z\mu^T$$

$$= -\Lambda_m^{-1} Z\mu^T$$

其中 $\Lambda_m^{-1} = diag\left\{\lambda_1^{-1}, \lambda_2^{-1}, \cdots, \lambda_m^{-1}\right\}$

■**实际病态线性回归问题及处理方法**

上述解法的含义

- 对病态线性回归问题，XX^T接近奇异，本质上就是某些特征根远比其他特征根小，而且非常接近于0，这将导致其倒数趋近无穷大。
- 即由于$\Lambda^{-1}=diag\left\{\lambda_1^{-1},\lambda_2^{-1},\cdots,\lambda_n^{-1}\right\}$使得线性回归误差被严重放大.

■实际病态线性回归问题及处理方法

处理病态线性回归问题的思路：

- 确定可以接收的最大逼近误差，做为截断依据，舍弃较小的特征值。
- 保留剩余较大的特征值，求得相应的规范特征向量。
- 经过矩阵变换，可以求得降维的回归方程，较好地消除了病态带来的结果敏感性。

黑箱建模——病态线性回归

■实际病态线性回归问题及处理方法

(5) 实际使用中必要的规范化措施

解决实际病态回归问题还必须考虑

1）变量单位的对建模结果有何影响，如何处理？

2）如何描述逼近误差是否可接受？

■**实际病态线性回归问题及处理方法**

（5）实际使用中必要的规范化措施

问题一　优化问题的目标函数为

$$\sum_{t=1}^{N}\left(x(t)-Lv(t)\right)^{T}\left(x(t)-Lv(t)\right)$$

如果令 $u(t)=Lv(t)$

可写成 $\sum_{t=1}^{N}\sum_{i=1}^{n}\left(x_i(t)-u_i(t)\right)^2$

变量采用不同单位，其样本数值会改变，对不同变量的逼近效果会随之改变，从而建模结果也会改变！

黑箱建模——病态线性回归

■**实际病态线性回归问题及处理方法**

（5）实际使用中必要的规范化措施

样本数据规范化

$$\bar{x}_i(t)=\frac{x_i(t)-e(x_i)}{\sqrt{\delta^2(x_i)}} \quad \forall i\,,t$$

其中 $e(x_i)=\frac{1}{N}\sum_{t=1}^{N}x_i(t)$ 是样本均值

$$\delta^2(x_i)=\frac{1}{N-1}\sum_{t=1}^{N}\left(x_i(t)-e(x_i)\right)^2$$ 是样本方差

目的 $e\left(\bar{x}_i\right)=0,\ \delta^2(\bar{x}_i)=1$

消除变量单位的影响！

黑箱建模——病态线性回归

■**实际病态线性回归问题及处理方法**

（5）实际使用中必要的规范化措施

令 $\bar{y}(t)=\dfrac{y(t)-e(y)}{\sqrt{\delta^2(y)}} \quad \forall t$

由于 $y(t)\approx\sum_{i=1}^{n}c_i x_i(t) \quad \forall t$

所以 $e(y)\approx\sum_{i=1}^{n}c_i e(x_i)$

$$\bar{y}(t)\approx\left(\delta^2(y)\right)^{-\frac{1}{2}}\sum_{i=1}^{n}c_i\left(x_i(t)-e(x_i)\right)=\sum_{i=1}^{n}\bar{c}_i\bar{x}_i(t)$$

可以通过估计 $\bar{c}$ 获得 c 的估计！

黑箱建模——病态线性回归

为什么回归分析之前要对样本数据进行归一化?

- 归一化的具体作用是统一样本的统计分布特性。无论是为了建模还是为了计算，首先基本度量单位要统一。
- 如果不进行归一化，一旦样本中某个变量的值远远大于其他变量，就会导致回归算法收敛速度严重下降，甚至可能进入饱和区从而导致无法得到正确的回归参数。
- 一般来说，线性回归不一定要做归一化，但要此时想对各回归系数之间进行比较就不那么方便了，因为各系数值受各变量单位的影响。为便于比较，需要求出标准化回归系数，消除仅由单位不同所带来的差别。

■**实际病态线性回归问题及处理方法**

（5）实际使用中必要的规范化措施

问题二　根据相对误差

$$\frac{\sum_{t=1}^{N}\left(x(t)-Q_mQ_m^Tx(t)\right)^T\left(x(t)-Q_mQ_m^Tx(t)\right)}{\sum_{t=1}^{N}x^T(t)x(t)}=\frac{\sum_{i=m+1}^{n}\lambda_i}{\sum_{i=1}^{n}\lambda_i}$$

是否小于给定阈值确定 m

Tsinghua University

例题: 用 $Y=\left[y(1)\cdots y(N)\right]$ 和 $X=\left[x(1)\cdots x(N)\right]$ 表示 $y\in R^1$ 和 $x\in R^4$ 的样本矩阵，已知 $XX^T=Q\Lambda Q^T$，Q 是正交矩阵，

$$Q=\begin{bmatrix} -.37 & -.81 & .07 & -.45 \\ -.29 & .45 & -.54 & -.64 \\ -.88 & .17 & .10 & .44 \\ -.05 & .34 & .83 & -.44 \end{bmatrix} \quad XY^T=\begin{bmatrix} .83 \\ .25 \\ 1.89 \\ .68 \end{bmatrix}$$

$$\Lambda=diag\{1.25, 0.03, 1.5, 0.02\}$$

试建立 y 对 x 的线性回归方程。

首先对特征值排序后看出

$$m=2 \qquad Q_m=\begin{bmatrix} .07 & -.37 \\ -.54 & -.29 \\ .10 & -.88 \\ .83 & -.05 \end{bmatrix}$$

再利用

$$ZY^T=Q_m^T XY^T$$

$$\left(ZZ^T\right)^{-1}=\Lambda_m^{-1}=\begin{pmatrix} 1/1.5 & 0 \\ 0 & 1/1.25 \end{pmatrix}=\begin{pmatrix} 0.6667 & 0 \\ 0 & 0.8000 \end{pmatrix}$$

黑箱建模——病态线性回归

可求得 $\hat{d}=\left(ZZ^T\right)^{-1}ZY^T=\Lambda_m^{-1}Q_m^T XY^T$

$$=\begin{pmatrix}0.6667 & 0\\ 0 & 0.8000\end{pmatrix}\begin{bmatrix}.07 & -.37\\ -.54 & -.29\\ .10 & -.88\\ .83 & -.05\end{bmatrix}^T\begin{bmatrix}.83\\ .25\\ 1.89\\ .68\end{bmatrix}$$

$$=\begin{bmatrix}0.4510\\ -1.6614\end{bmatrix}$$

于是：$\hat{c}=Q_m\hat{d}=[0.6463\quad 0.2383\quad 1.5072\quad 0.4574]^T$

则有：

$$y\approx\hat{c}^T x=0.6463x_1+0.2383x_2+1.5072x_3+0.4574x_4$$

黑箱建模——交通流量预测

□ 城市交通流的预测

即处理交通参数的时间序列。

在现实生活中存在各种各样的时间序列问题,人们对于时序的研究主要基于两种方法：一种是用随机过程的理论建立线性关系模型，如回归模型、ARIMA模型等；另外一种方法则是利用非线性动力学方法，研究低自由度的混沌系统。

□ 预测方法举例

时空自回归滑动平均求和模型----STARIMA（Autoregressive Integrated Moving Average Model)

多变量自适应回归样条模型----MARS （Multivariate Adaptive Regressive Splines)

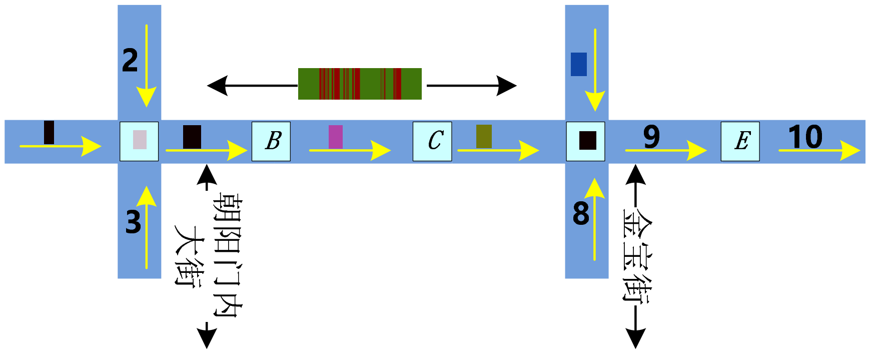
- 时空自回归滑动平均求和模型(STARIMA)

由Pfeifer在1980年提出，后来广泛应用于多个领域的预测问题。它的主要优点在于综合了数据本身的相关性，以及在某种区域内各位置相邻采集点所采集数据间的相关性。

- STARIMA在交通预测方面的应用

交通流作为实际交通路网中的数据，在时间上其自身有自相关性，而在空间位置上与其相邻的路段也具有相关性，因此在交通流的预测上希望同时考虑这两方面的因素。

黑箱建模——交通流量预测



第i阶空间权值矩阵表示了与下游路段相隔i条路段的上游路段的权值

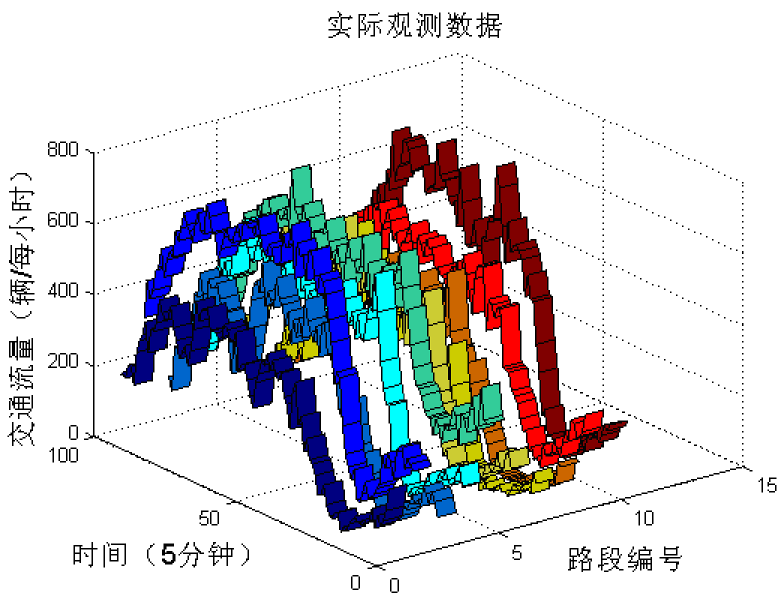
$$U_1=\begin{pmatrix}0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\1/3&1/3&1/3&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0&0\\0&0&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&1/3&1/3&1/3&0\\0&0&0&0&0&0&0&0&1&0\end{pmatrix}\qquad U_2=\begin{pmatrix}0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\1/3&1/3&1/3&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0\\0&0&0&0&1/3&0&0&0&0&0\\0&0&0&0&0&1/3&1/3&1/3&0&0\end{pmatrix}$$

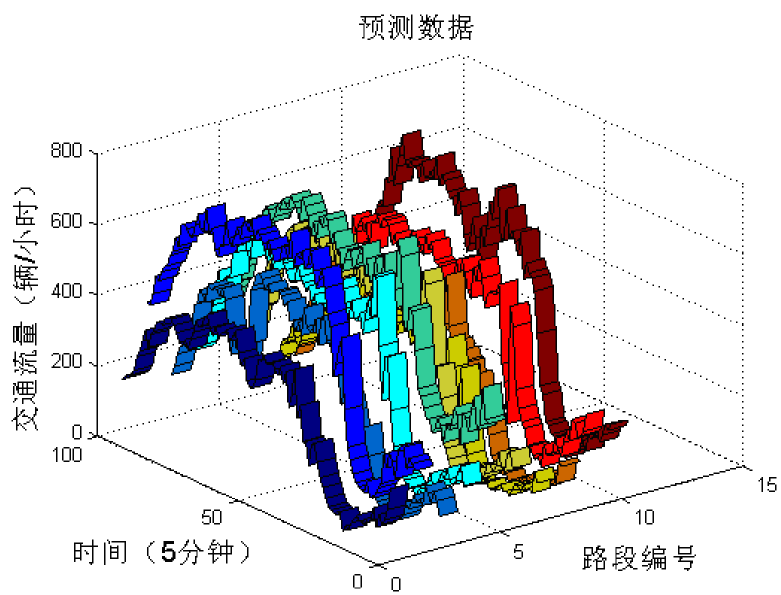
假设该时间序列由自身前3个时间点的数据、(t-1)时刻1阶邻居和2阶邻居采集点数据、以及模型误差项构成，则预测模型如下：

$$Z_t = \phi_{10} Z_{t-1} + \phi_{20} Z_{t-2} + \phi_{30} Z_{t-3} + \phi_{11} U_1 Z_{t-1} + \phi_{12} U_2 Z_{t-1} - \eta_{10} \varepsilon_{t-3} + \varepsilon_t$$

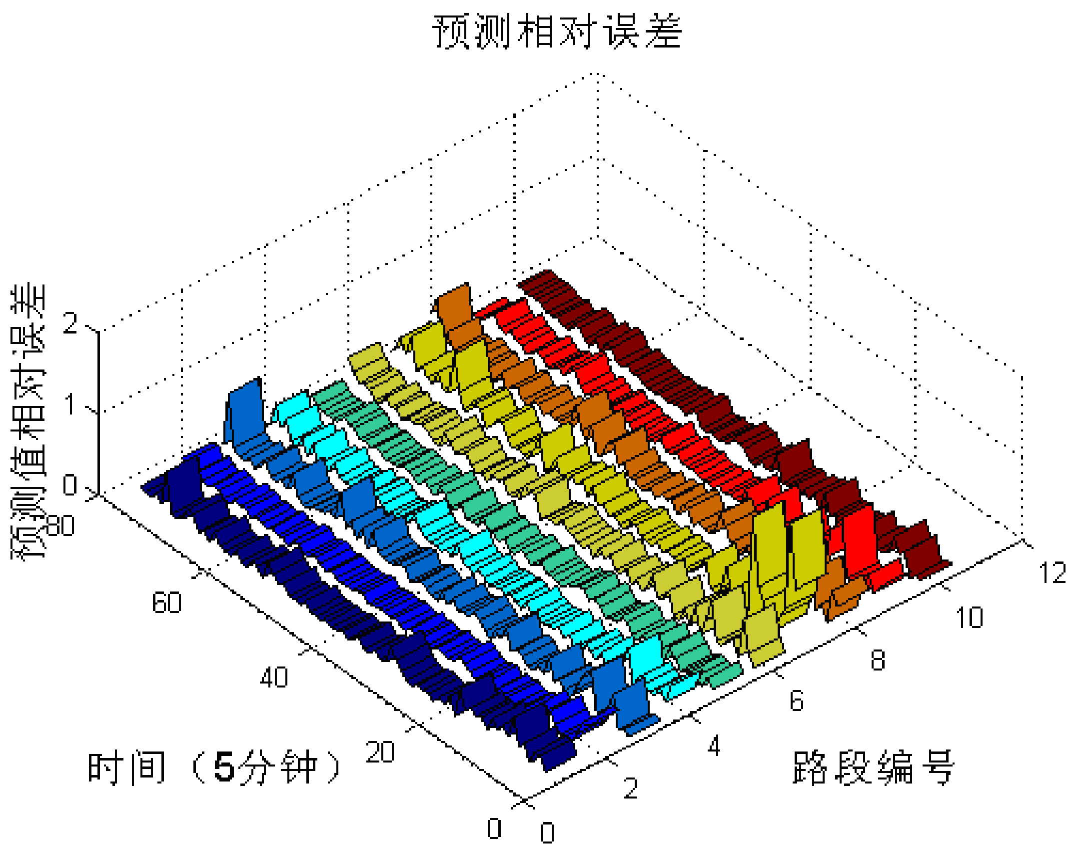
上式右边前三项代表了该序列的时间相关性，第4、5项代表了该序列的空间相关性。

STARIMA预测结果





黑箱建模——交通流量预测



黑箱建模——交通流量预测

◘ 多变量自适应回归样条模型 (MARS)

该算法在1991年由Stanford University的统计学家、物理学家Jerome Friedman首先提出。作为一种新的回归建模算法，MARS在对高维数据中的一些复杂结构，如非线性和变量之间的相关性分析等方面被证明是非常有效的。

◘ MARS在交通预测方面的应用

因为各相邻路口之间流量存在着一定的相关关系，利用MARS算法对已知的完整数据进行拟合，就可以得到某一时刻相邻路口之间流量的近似函数关系。

黑箱建模——交通流量预测

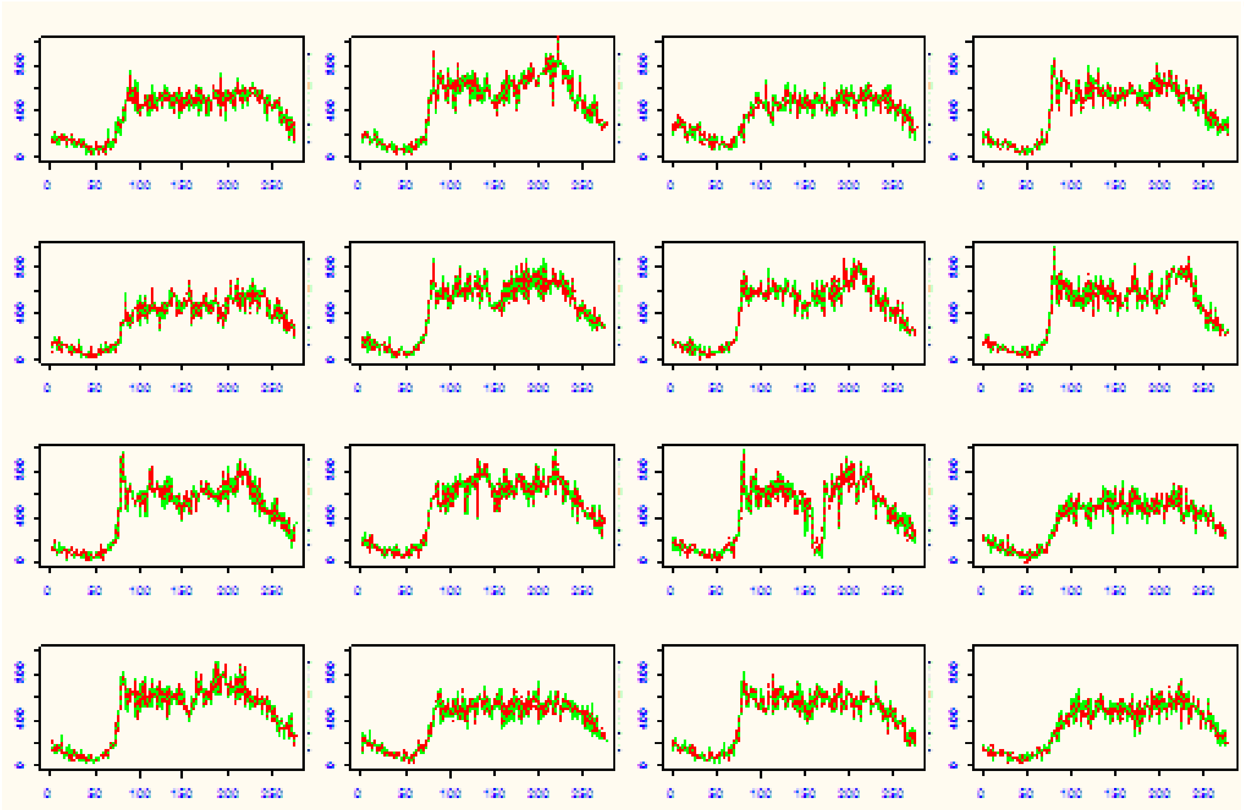
考虑到交通流数据是一个时间序列，流量与时间的关系是非常密切的，汽车从一个路口开到下一个路口中间会经过一段时间，因此有理由相信，相邻路口之间的流量之间存在着某种时间上的“滞后”关系。

因为拟合用到的是相邻路段的数据，所以这种滞后关系不应当太长，谨慎起见在拟合的时候加入相邻路段前后25分钟的数据，即得到模型：

$$flow_{link_j}(t)=\phi\begin{pmatrix} flow_{link_{i_1}}(t-5), flow_{link_{i_1}}(t-4),\cdots, flow_{link_{i_1}}(t+4), flow_{link_{i_1}}(t+5), \\ flow_{link_{i_2}}(t-5), flow_{link_{i_2}}(t-4),\cdots, flow_{link_{i_2}}(t+4), flow_{link_{i_2}}(t+5), \\ \vdots \\ flow_{link_{i_k}}(t-5), flow_{link_{i_k}}(t-4),\cdots, flow_{link_{i_k}}(t+4), flow_{link_{i_k}}(t+5) \end{pmatrix}$$

黑箱建模——交通流量预测

MARS预测结果（红色为预测数据）



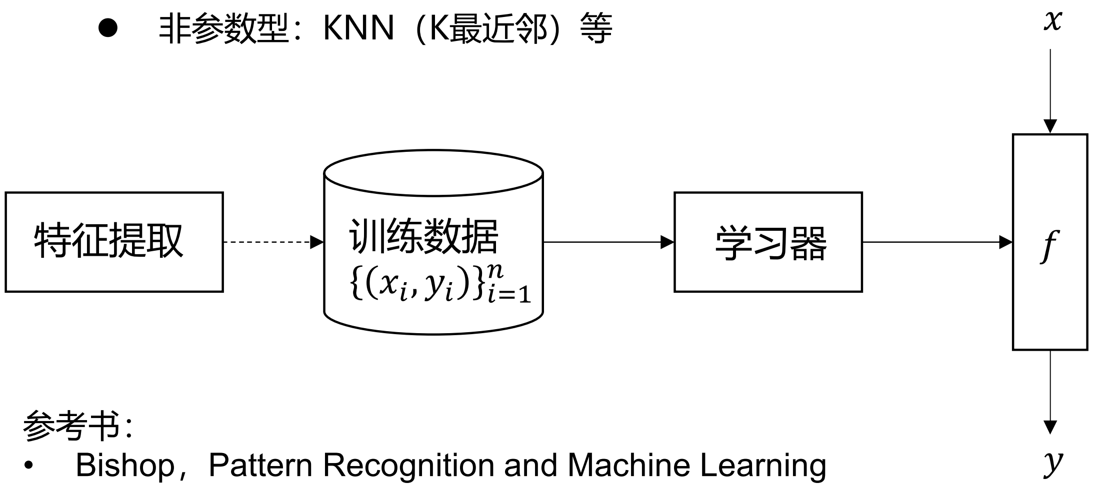
黑箱建模——交通流量预测

软件演示：《城市路网交通流预测和分析系统 》

黑箱建模——深度学习介绍

■机器学习方法与深度学习方法

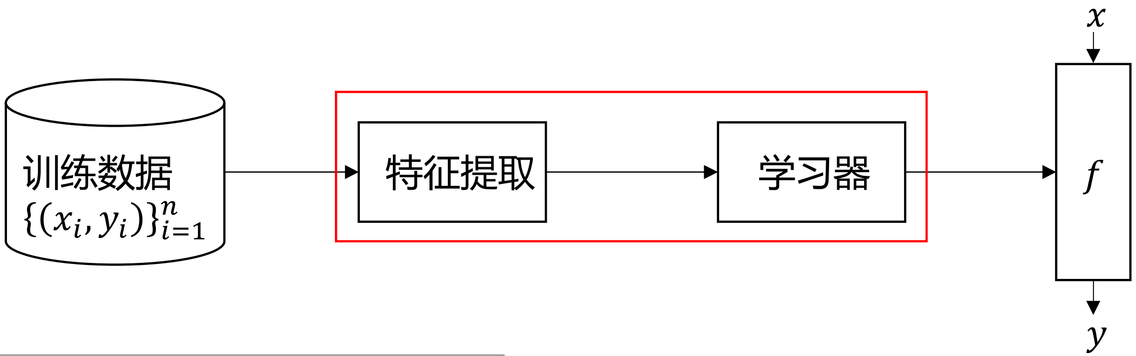
- 一般机器学习方法范式
  - 假设空间（Hypothesis Space）
  - 学习算法
    - 参数型：回归、决策树等
    - 非参数型：KNN（K最近邻）等

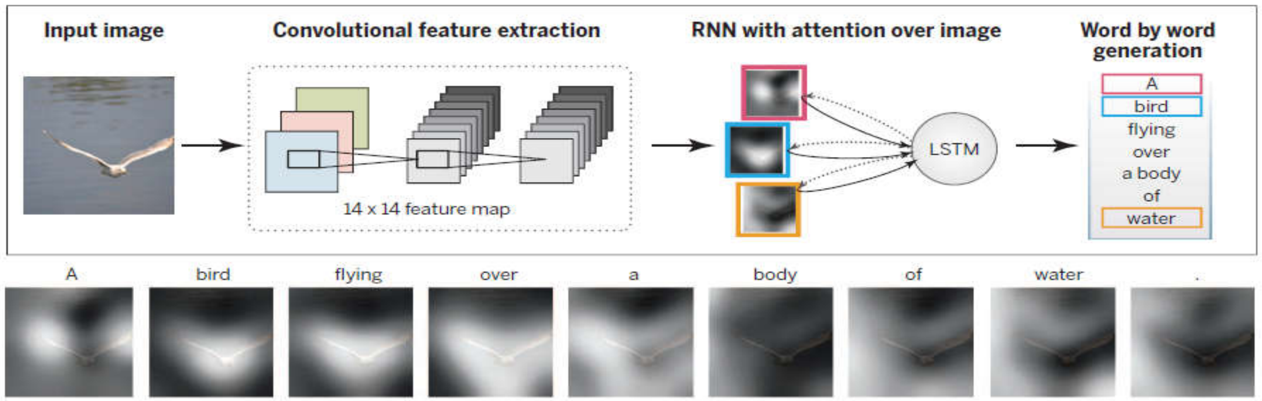


参考书：
- Bishop，Pattern Recognition and Machine Learning
- Goodfellow, Deep Learning

■机器学习方法与深度学习方法

- 深度学习方法
  - 机器学习方法的“子领域”





黑箱建模——深度学习介绍

■机器学习方法与深度学习方法

- 深度学习一般性模型
  - 最小化经验风险（Empirical Risk)

经验风险

$$argmin_\theta J(D_{train};\theta) = \boxed{\sum_{i=1}^{n} \underbrace{L(y_i, \underbrace{f_\theta(x_i)}_{\text{模型}})}_{\text{损失函数}}} + \underbrace{\Omega(\theta)}_{\text{参数先验}}$$

- 损失函数：度量输出值和实际值的差异
- 模型：假设真实模型可以写成一类特定的带参数的函数形式
- 参数先验：正则项，减少模型复杂度，避免过拟合（Overfitting)

正则项的作用

Tsinghua University

■机器学习方法与深度学习方法

- 深度学习一般性模型
  - 最小化经验风险（Empirical Risk)

经验风险

$$argmin_{\theta}\, J(D_{train};\theta) = \boxed{\sum_{i=1}^{n} L\big(y_i, f_{\theta}(x_i)\big)} + \Omega(\theta)$$

损失函数　模型　参数先验

```
for ( t = 1 to T) {
    doSomeThings();
    θ^{t+1} ← θ^{t} − η∇_θ J(D; θ^{t}); //Gradient-based Optimization Algorithms
    doOtherThings();
}
```

■机器学习方法与深度学习方法

- 深度学习一般性模型
  - 最小化经验风险（Empirical Risk）

经验风险

$$argmin_\theta J(D_{train};\theta)=\boxed{\underbrace{\sum_{i=1}^{n}L\big(y_i,\underbrace{f_\theta(x_i)}_{\text{模型}}\big)}_{\text{损失函数}}}+\underbrace{\Omega(\theta)}_{\text{参数先验}}$$

损失函数　模型　参数先验

损失函数：均方误差
模型：一维多元线性模型
参数先验：高斯分布

多元线性回归：$J(D;\theta)=\sum_{i=1}^{n}(y_i-\theta^T x_i)^2+\lambda||\theta||_2^2$

参数解析解：$\hat{\theta}=\left[\sum_{i=1}^{n}x_i x_i^T+\lambda I\right]^{-1}\left[\sum_{i=1}^{n}x_i y_i\right]$

正规方程
更一般的做法是用梯度下降进行更新，速度更快结果更稳定

黑箱建模——深度学习介绍

■机器学习方法与深度学习方法

- Why Deep Learning?

机器学习的弊端：局部不变性先验

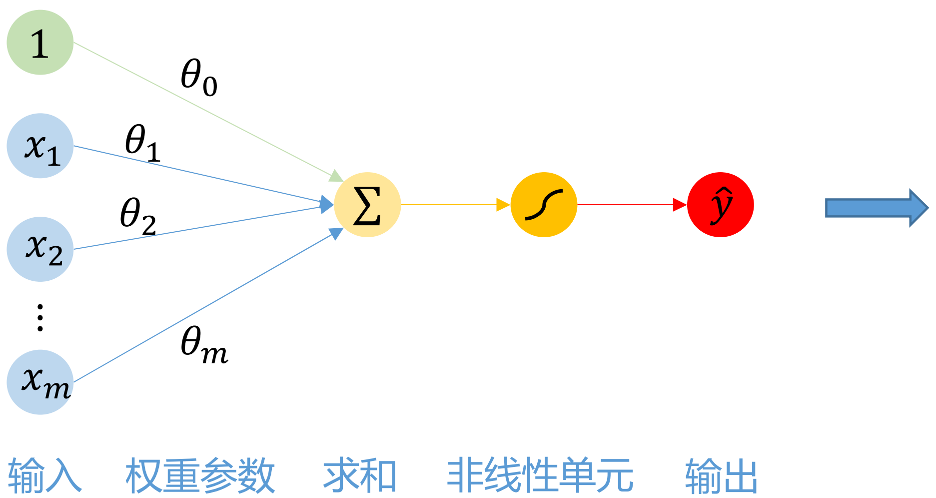
- K-近邻算法：根据周围K个样本的标签预测当前样本的标签
- 局部基函数：样本仅对周围一定距离的空间产生影响
- $f(x) \approx f(x+\epsilon)$

需要足够多的样本才能学到具有很强泛化性的模型!

- 高维数据下由于维度灾难，样本数量远远不够，因此传统的机器学习算法在高维数据下泛化性很差
- 需要更具适用性、更温和的数据假设
  - 数据由不同的特征组合而成，这些特征可以来自不同的层次结构
  - CNN：局部感受野（卷积核参数共享）等
  - RNN：数据/任务具有一定的周期性等

■前馈神经网络

- 感知机
  - 1958年，Rosenblatt仿照神经元结构提出感知机



$$\hat{y} = g\left(\theta_0 + \sum_{i=1}^{m} x_i\theta_i\right)$$
$$= g(\theta^T x)$$

$\boxed{\text{定义}x_0 = 1}$

Tsinghua University

■前馈神经网络

- 感知机
  - Rosenblatt的感知机

$$\hat{y} = g(\theta^T x) = sgn(\theta^T x - T)$$

$$= \begin{cases} 1, if\ \theta^T x - T \geq 0 \\ 0, \quad\quad otherwise \end{cases}$$

Rosenblatt： “感知机可以表示任何布尔函数！”

AND： $x \wedge y = sgn(x + y - 2)$

OR： $x \vee y = sgn(x + y - 1)$

NOT： $\bar{x} = sgn(-x)$

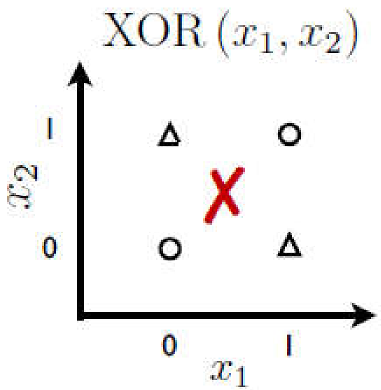
XOR： ？

好像不能表示XOR?

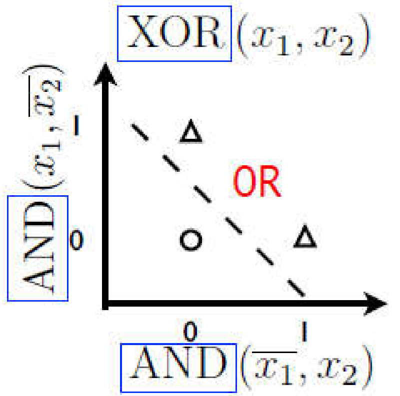
黑箱建模——深度学习介绍

■前馈神经网络

- 感知机
  - 用感知机表示XOR



感知机

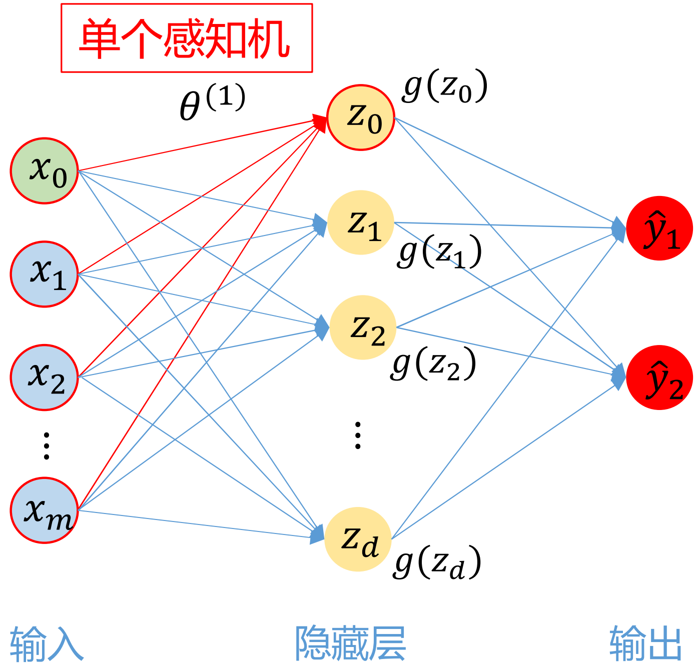


感知机的感知机

通过对感知机进行叠加来增强感知机的表达能力！ 多层感知机

■前馈神经网络

- 多层感知机

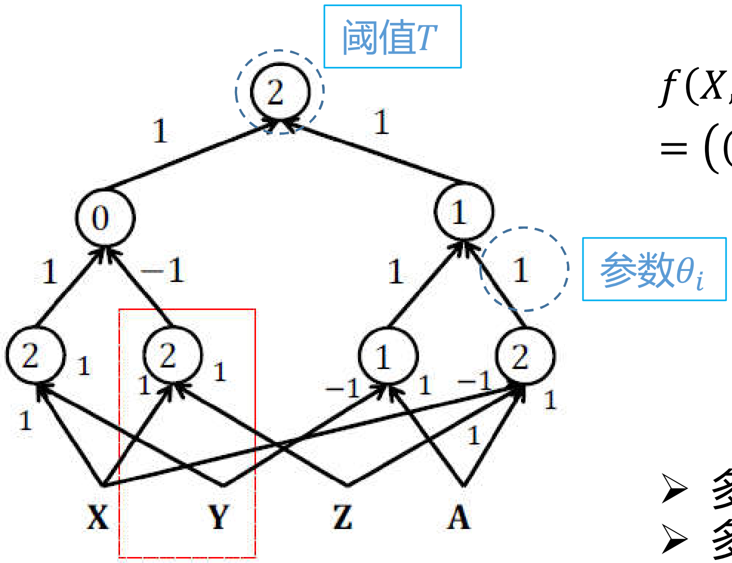


XOR： $x \oplus y = (x \vee y) \wedge (\bar{x} \vee \bar{y})$

Marvin Minsky and Seymour Papert. Perceptrons. An Introduction to Computational Geometry. 1969

■前馈神经网络

- 多层感知机：布尔函数



$$f(X,Y,Z,A) = \left((A\wedge \bar{X}\wedge Z)\vee(A\wedge\bar{Y})\right)\wedge\left((X\wedge Y)\vee\overline{(X\wedge Z)}\right)$$

➢ 多层感知机可以表示任意复杂布尔函数
➢ 多层感知机的连接可以是稀疏的

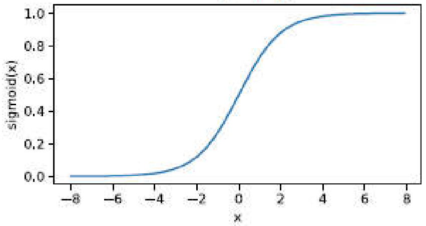
Marvin Minsky and Seymour Papert. Perceptrons. An Introduction to Computational Geometry. 1969

■前馈神经网络

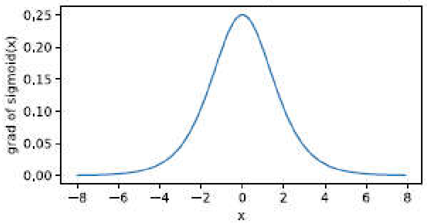
- 多层感知机：激活函数

sigmoid函数

$$g(z)=\frac{1}{1+e^{-z}}$$



$$g'(z)=g(z)\big(1-g(z)\big)$$

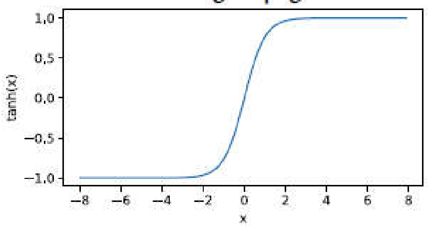


输出范围(0,1)，用于表达概率

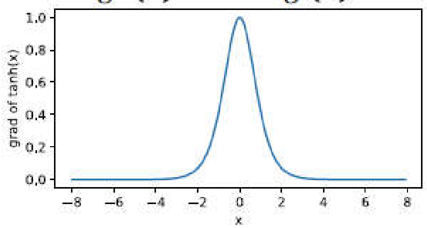
存在梯度饱和区

Tanh函数

$$g(z)=\frac{e^{z}-e^{-z}}{e^{z}+e^{-z}}$$



$$g'(z)=1-g(z)^2$$

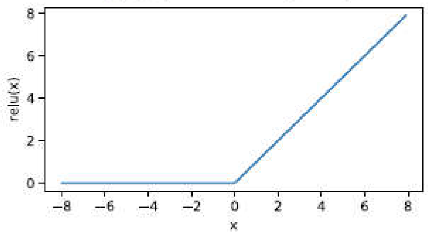


0-中心化，输出范围(-1,1)

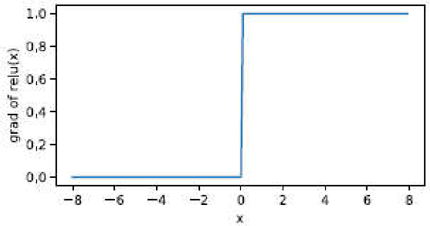
存在更大的梯度饱和区

Relu函数

$$g(z)=\max(0,z)$$



$$g'(z)=\mathrm{sgn}(z)$$



计算快速、加速收敛

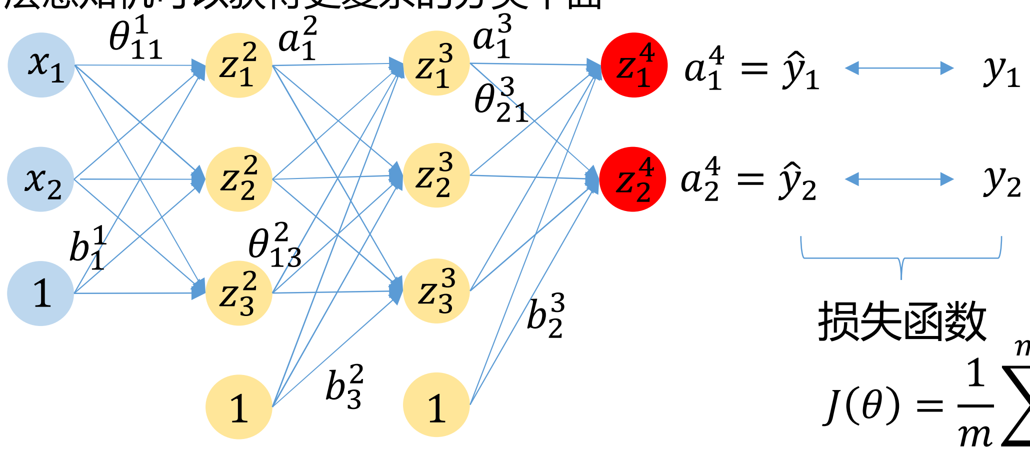
神经元可能一直不被更新

黑箱建模——深度学习介绍

■前馈神经网络

● 多层感知机：实值函数

- 单层感知机等价于线性分类器
- 多层感知机可以获得更复杂的分类平面



- 权重θ
- 偏置b
- 非线性激活函数a
- 预测值y

损失函数

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} L\big(f(x^i;\theta), y^i\big)$$

for (t = 1 to T) {
FowardPropagation(); //前向传播，计算损失函数
BackPropagation(); //反向传播，计算梯度
$\theta^{t+1} \leftarrow \theta^t - \eta \nabla_\theta J(D;\theta^t)$; //参数更新
}

Ian Goodfellow and Yoshua Bengio. Deep Learning. 2017

黑箱建模——深度学习介绍

■卷积神经网络

- 动机
  - 对于类似网格结构的数据，前馈神经网络会损失位置信息
  - 稀疏交互
  - 参数共享
  - 等变表示

对于机器学习系统的三个重要改进思想

卷积神经网络

- 局部性假设：仅使用局部信息就足够进行特征提取

每个神经元仅会与附近的神经元进行链接

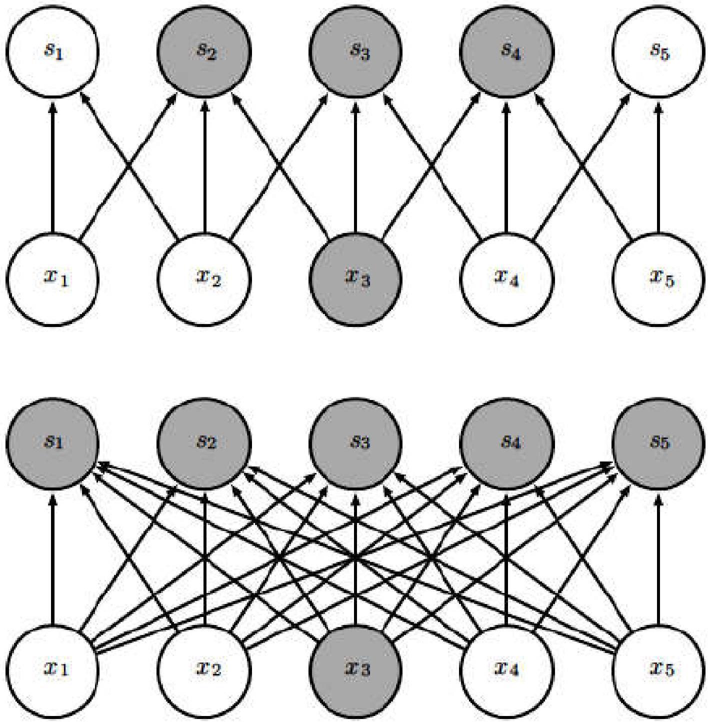
- 平移不变性假设：如果一个特征在某一位置是有用的，那么该特征换一个位置也能起到同样的作用

在所有的位置进行权重参数共享

黑箱建模——深度学习介绍

■卷积神经网络

- 动机



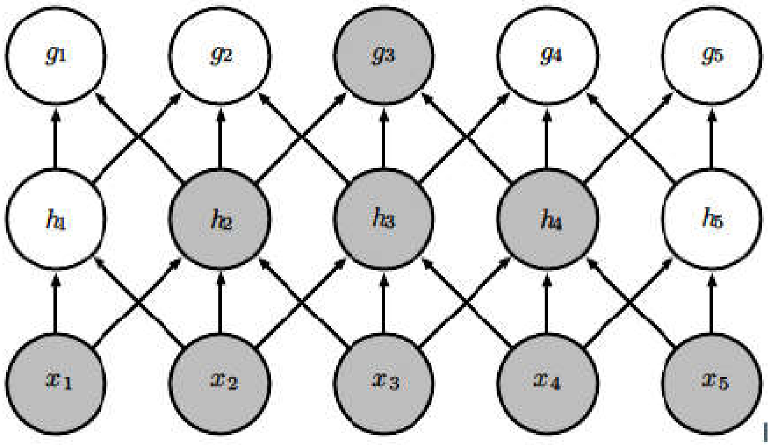
稀疏交互：
从输入层看，输入层中每一个元素在稀疏交互的情况下仅会影响输出层部分元素，而稠密链接会使得输入层的每一个元素会对所有输出层元素产生影响；
从输出层看，输出层中每一个元素在稀疏交互的情况下仅会受到部分输出层元素影响，而稠密链接会使得输出层的每一个元素都会受到所有输入层元素影响；

对比前面黑箱建模部分讲到的局部基函数！！两者思想是一致的

黑箱建模——深度学习介绍

■卷积神经网络

- 动机



稀疏交互：
尽管网络中的直接连接很稀疏，但是更深层中的元素会间接连接到更多的输入元素

■卷积神经网络

- 卷积

离散卷积

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f(m)g(n-m)$$

离散互相关

$$(f \star g)[n] = \sum_{m=-\infty}^{\infty} f(m)g(n+m)$$

两者的区别在于卷积会对卷积核进行翻转操作，互相关并不会

翻转带来的好处：卷积具有可交换性，但在神经网络中这样的性质并不重要

$$(f * g)[n] = (g * f)[n]$$

实际的深度学习库都是实现的互相关函数然后称之为卷积，卷积神经网络也采用的是互相关操作而不是卷积操作，因为学习算法可以在核的适当位置学到适当的值。

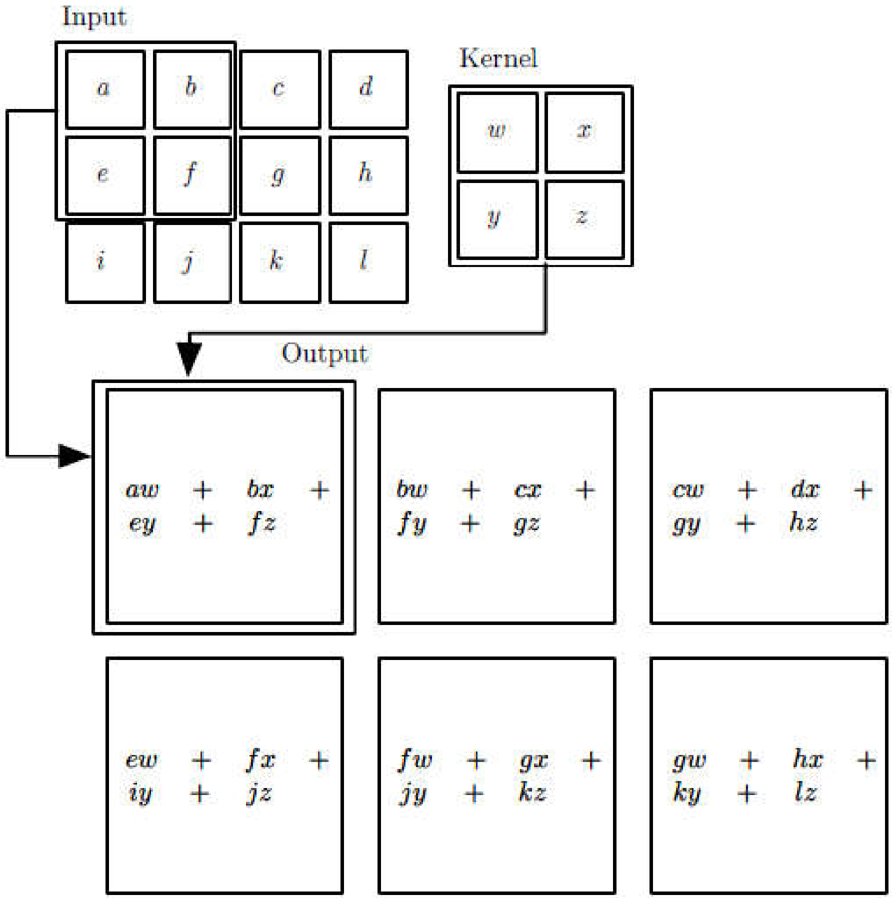
黑箱建模——深度学习介绍

■卷积神经网络

- 卷积：局部链接

两个同样大小的矩阵之间的按位乘积之后求和

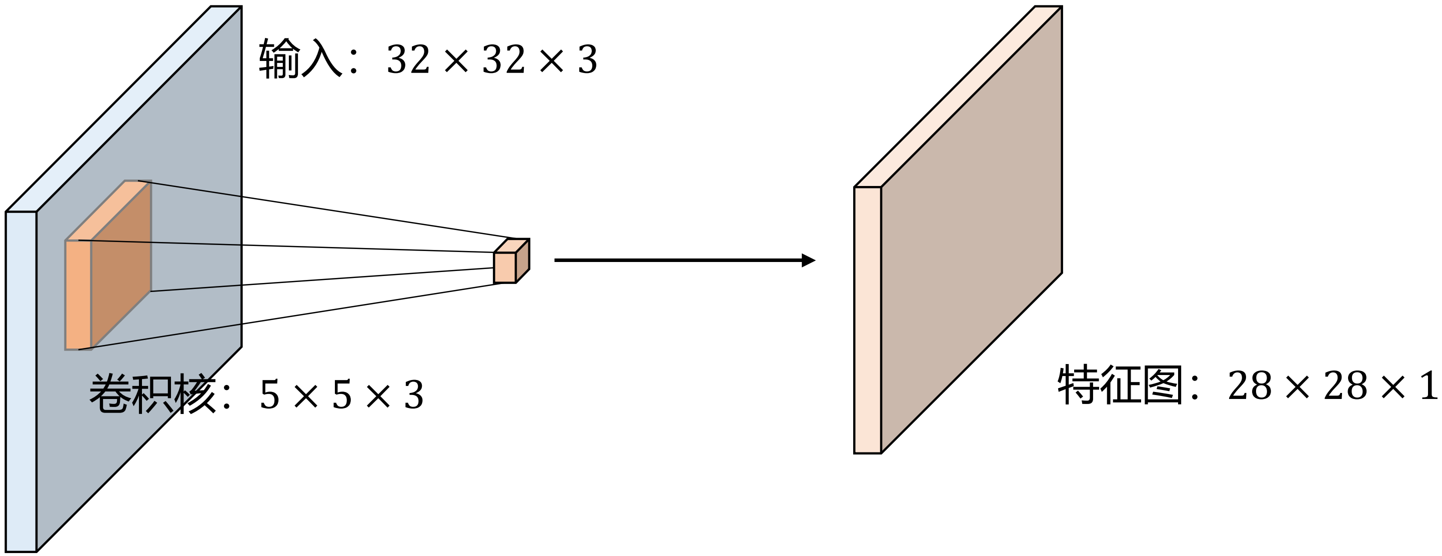
$$s(w,h) = (I \star K)(w,h)$$
$$= \sum_i \sum_j I(w+i, w+j)K(i,j)$$



黑箱建模——深度学习介绍

■卷积神经网络

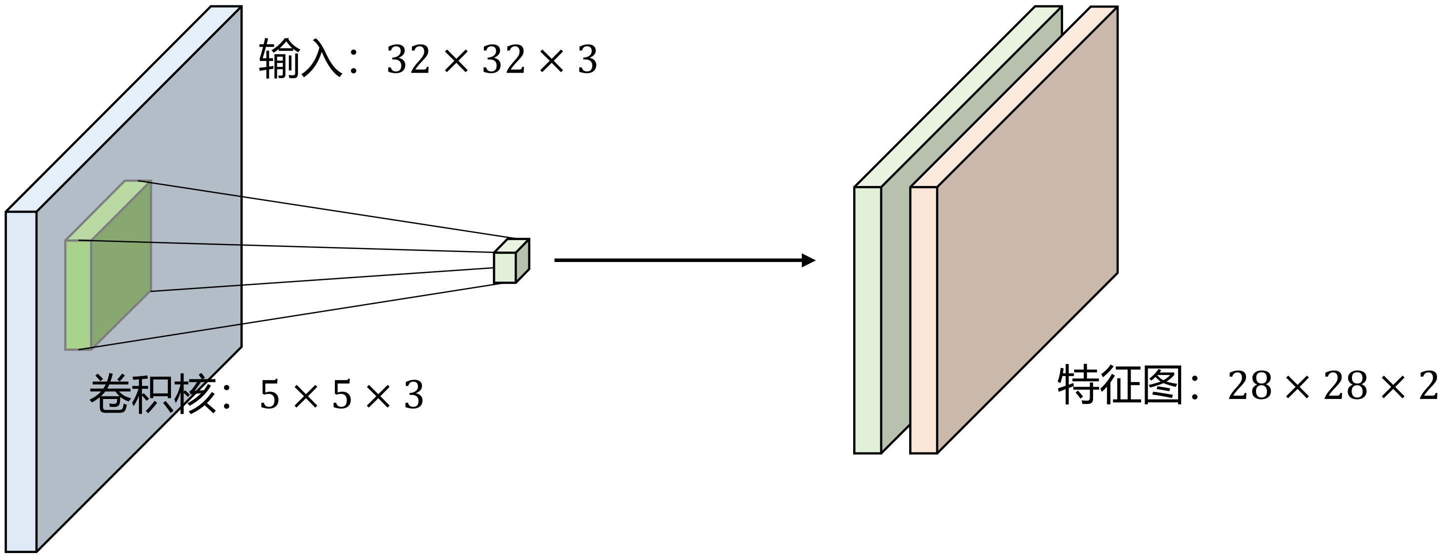
- 卷积神经网络：多特征卷积层



黑箱建模——深度学习介绍

■卷积神经网络

- 卷积神经网络：多特征卷积层



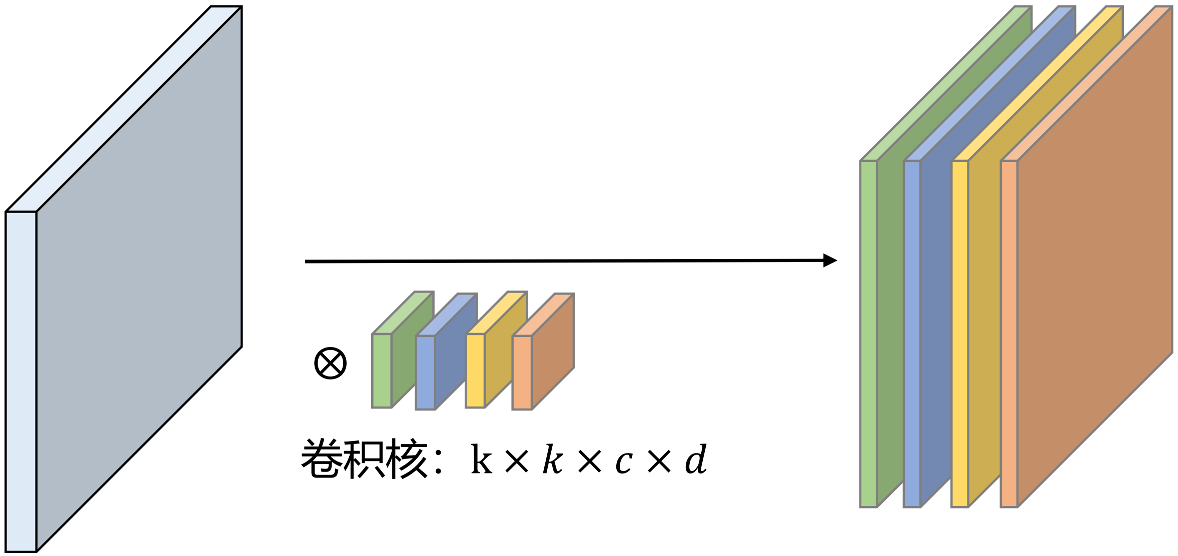
黑箱建模——深度学习介绍

■卷积神经网络

- 卷积神经网络：多特征卷积层

输入：$\mathrm{w} \times h \times c$

特征图：$(\mathrm{w} - k + 1) \times (h - k + 1) \times d$



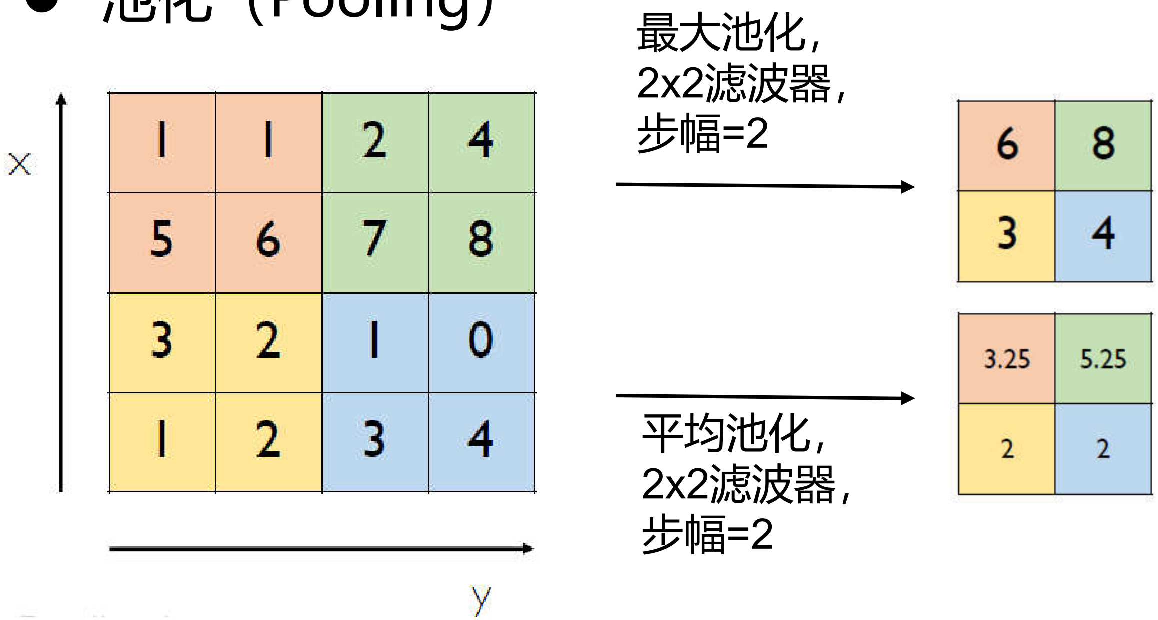
$\otimes$

卷积核：$\mathrm{k} \times k \times c \times d$

Tsinghua University

■卷积神经网络

- 池化（Pooling）



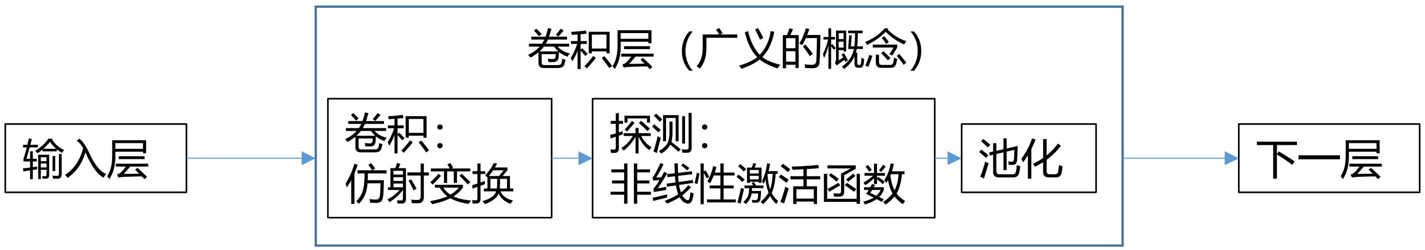
池化层：
- 降低特征对位置的敏感度
- 降低图像分辨率

帮助改善平移不变性：该层函数需要对少量平移具有输出不变性

黑箱建模——深度学习介绍

■卷积神经网络

- 典型网络结构



并不绝对，也不代表这是最优设计
根据问题特点和问题需要自行搭建网络

黑箱建模——深度学习介绍

■卷积神经网络

- More on CNN（感兴趣的同学自行学习）
  - CNN反向传播算法（CNN-BP）
  - 权重初始化（Weight Initialization）
    - Xavier Initialization
    - He Initialization
  - 正则化
    - Batch Normalization
    - Group Normalization
    - Layer Normalization
  - 数据增广（Data Augmentation）
  - 残差网络
    - ResNet
    - DenseNet

■循环神经网络

- 动机
  - 对于序列数据，前馈神经网络会损失位置信息
  - 对于序列数据，卷积神经网络和前馈神经网络会限制序列长度

循环神经网络

- 局部依赖假设：过去时间段的信息可以使用隐状态进行编码

$$p(x_1,\cdots,x_T)=\prod_{t=1}^{T}p(x_t|x_1,\cdots,x_{t-1})=\prod_{t=1}^{T}g(s_{t-2},x_{t-1})$$

- 时序稳定性假设：如果一个特征在某一时间是有用的，那么该特征换一个时间戳也能起到同样的作用

$$p(x_{t_1+\tau},\cdots,x_{t_n+\tau})=p(x_{t_1},\cdots,x_{t_n})$$

黑箱建模——深度学习介绍

■循环神经网络

- 循环神经网络：原型

$$h_t = \boxed{\tanh}(Wh_{t-1} + Ux_t)$$

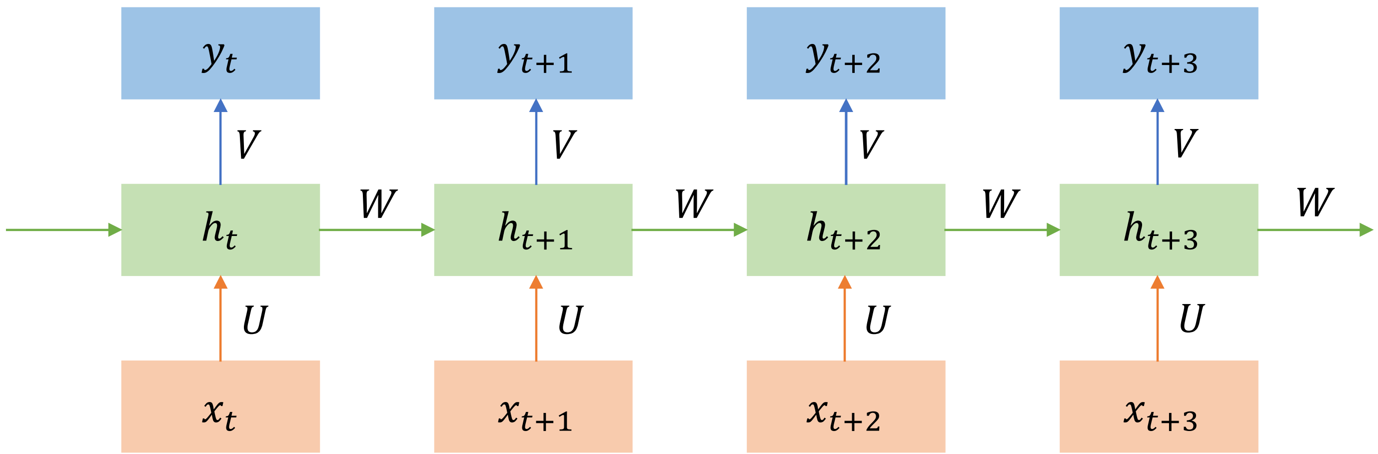
$$y_t = \boxed{softmax}(Vh_t)$$

激活函数，可以根据需要自己指定，不影响性质

满足 $\Rightarrow$

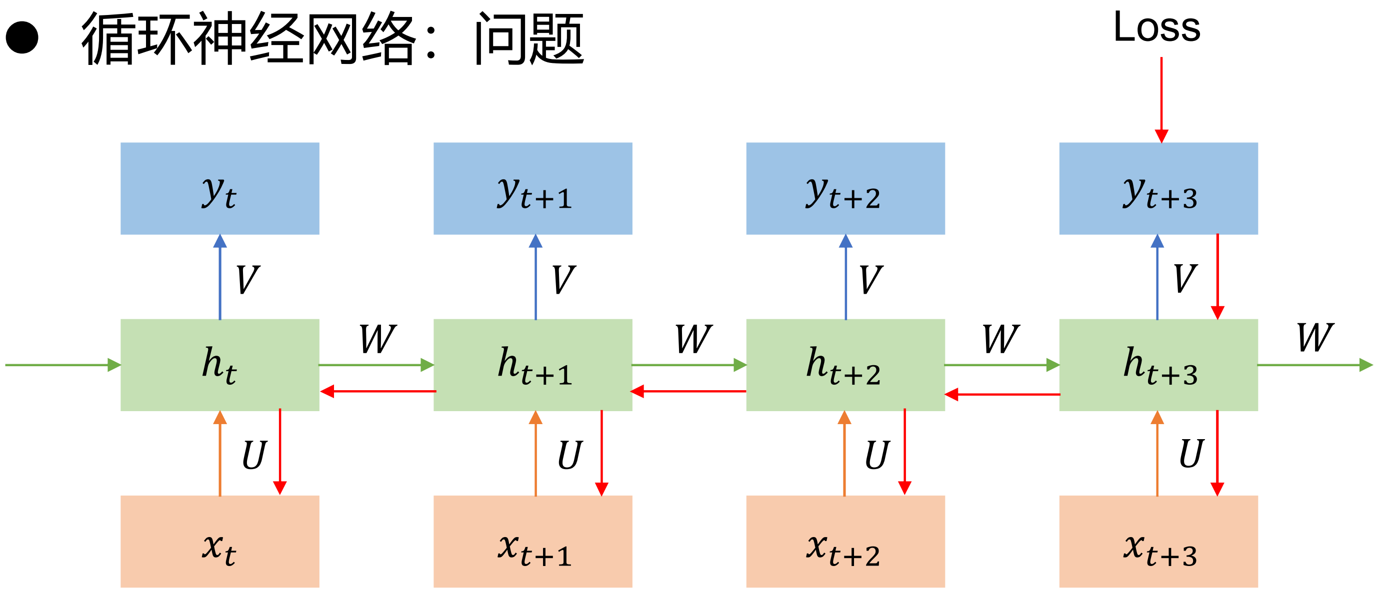
$$p(x_1, \cdots, x_T) = \prod_{t=1}^{T} p(x_t | x_1, \cdots, x_{t-1}) = \prod_{t=1}^{T} g(s_{t-2}, x_{t-1})$$

$$p(x_{t_1+\tau}, \cdots, x_{t_n+\tau}) = p(x_{t_1}, \cdots, x_{t_n})$$



■循环神经网络

- 循环神经网络：问题

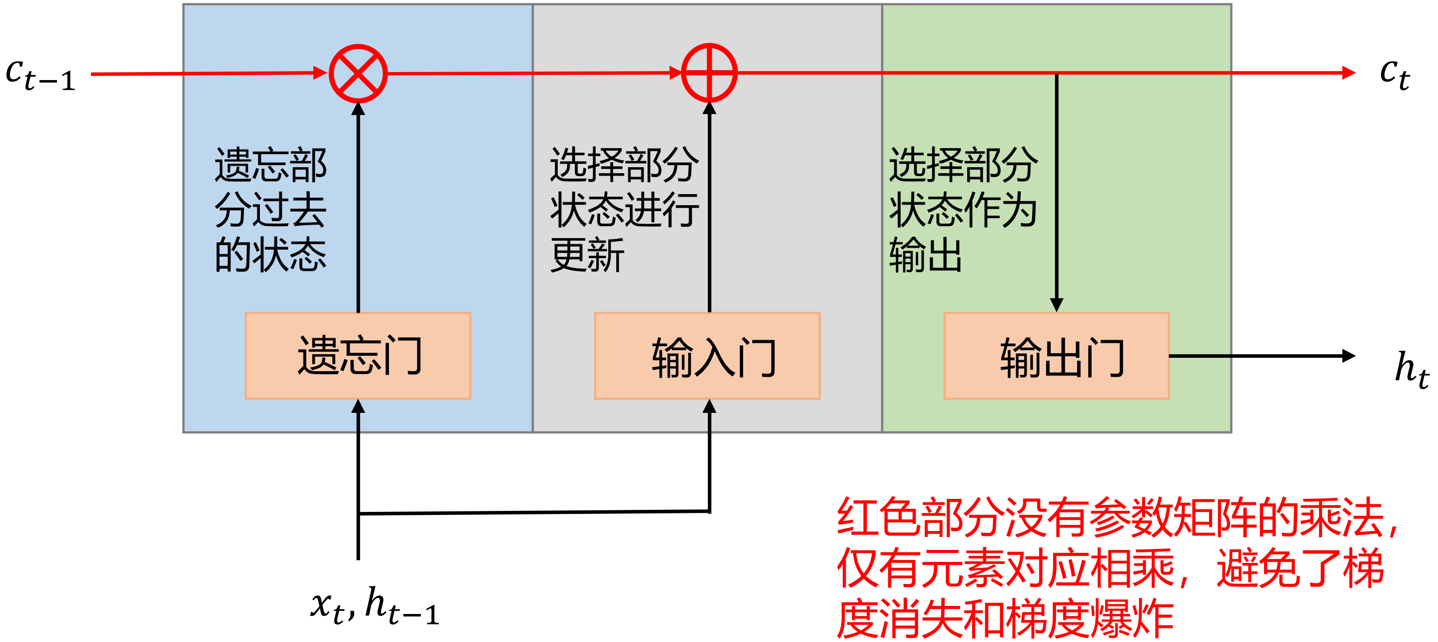


梯度反向传播时会因为权重矩阵W多次乘积（因为参数共享）导致梯度消失或者梯度爆炸

解决办法：门控循环神经网络

■循环神经网络

- 门控循环神经网络：长短期记忆网络（LSTM）



红色部分没有参数矩阵的乘法，仅有元素对应相乘，避免了梯度消失和梯度爆炸

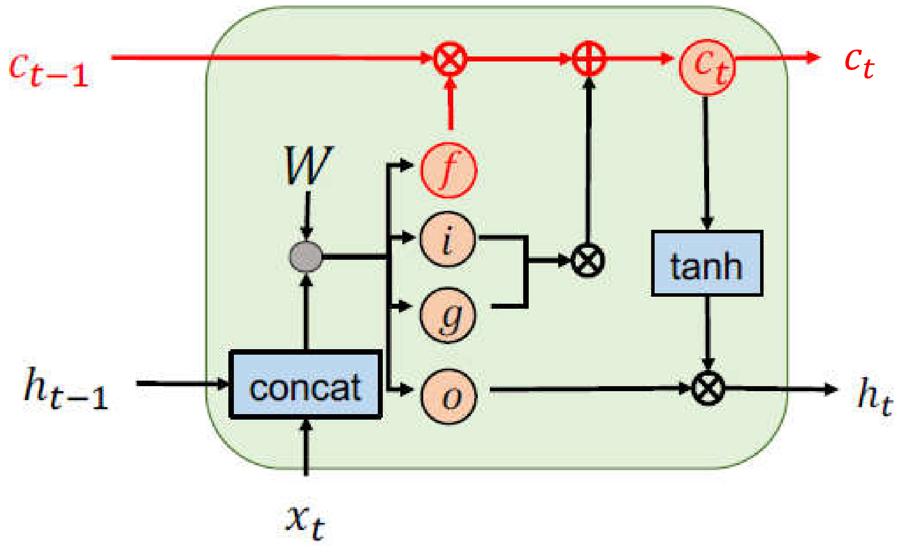
■循环神经网络

- 门控循环神经网络：长短期记忆网络（LSTM）

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$c_t = f \odot c_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(c_t)$$



不同的门可以使用不同的参数矩阵W

如果遗忘门一直处于打开状态，那么就可能获取到久远的隐状态

Tsinghua University

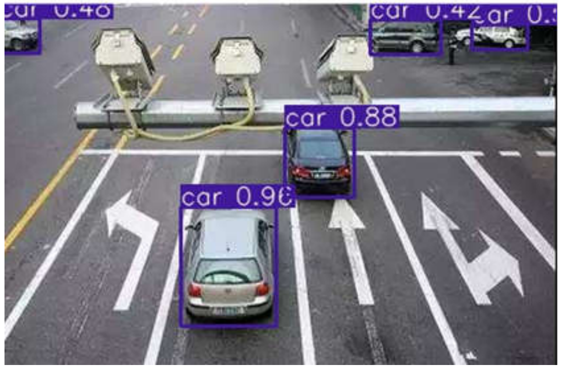
■循环神经网络

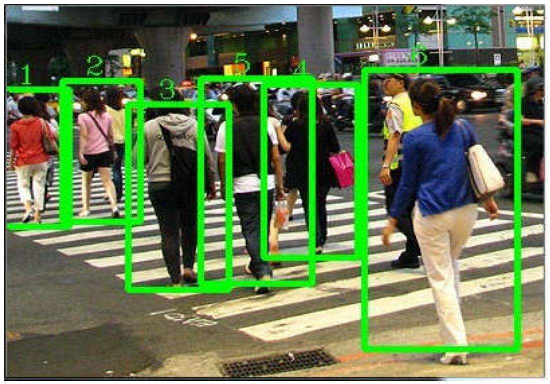
- More on RNN（感兴趣的同学自行学习）
  - RNN反向传播算法（RNN-BP）
  - RNN变体
    - 双向RNN
    - GRU(vs LSTM)
    - RNN+CNN
  - 自注意力机制(Self-Attention)
    - Seq2Seq
    - Transformer（应用广泛）
    - Bert（自然语言处理）
  - 正则化
    - Layer Normalization
    - Weight Normalization

黑箱建模——深度学习介绍

■深度学习在交通方向的应用

- 交通目标检测





黑箱建模——深度学习介绍

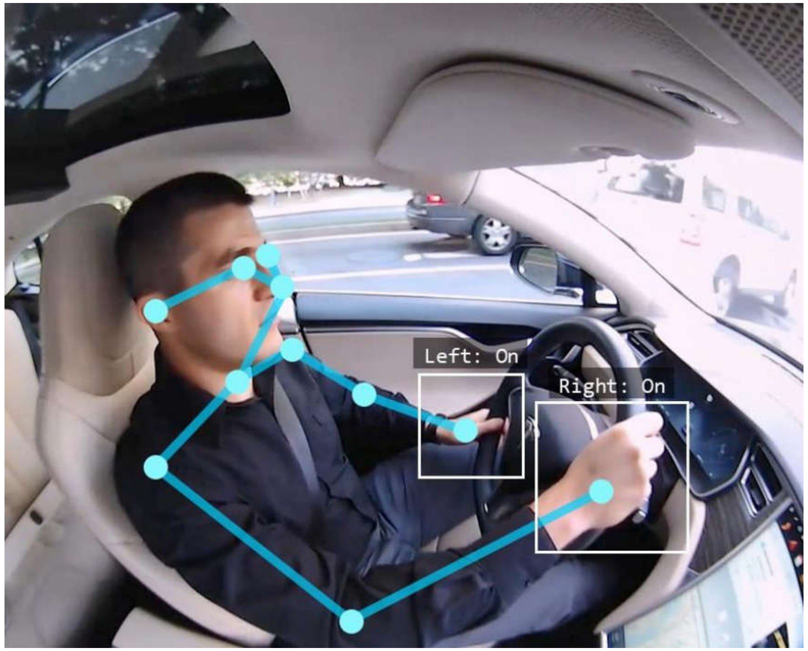
■深度学习在交通方向的应用

- 交通物体分割

黑箱建模——深度学习介绍

■深度学习在交通方向的应用

- 姿势检测

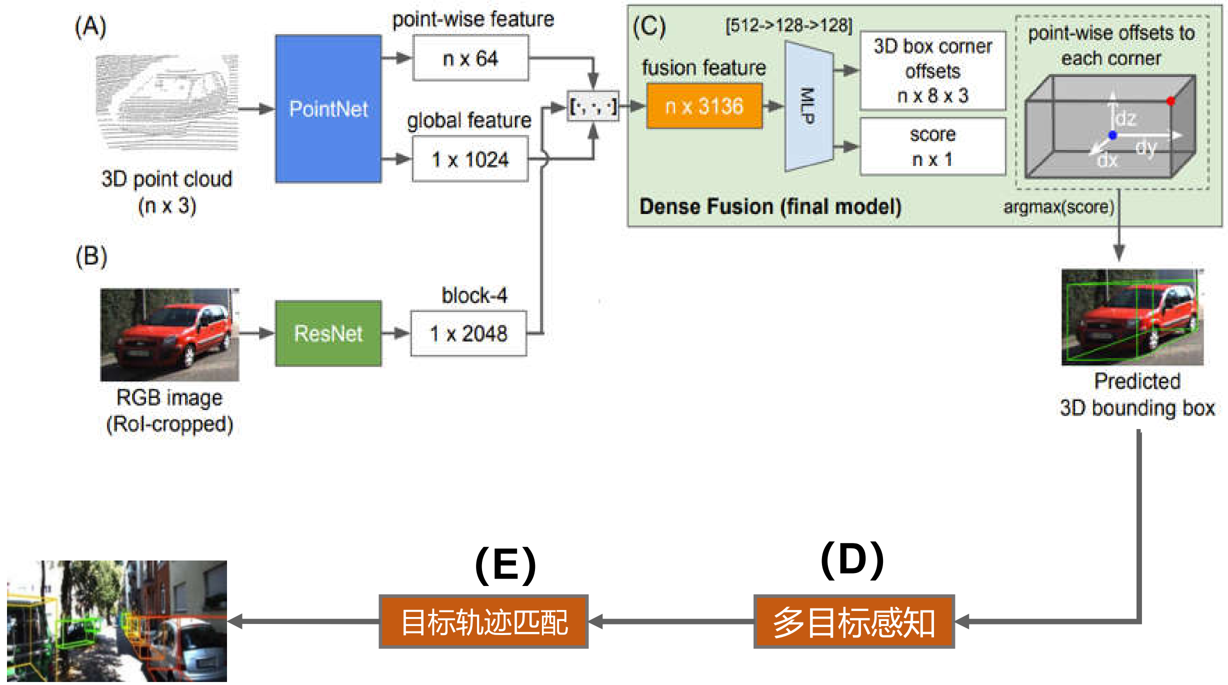


对司机的不当驾驶姿势进行预警

黑箱建模——深度学习介绍

■深度学习在交通方向的应用

- 异构传感器数据融合



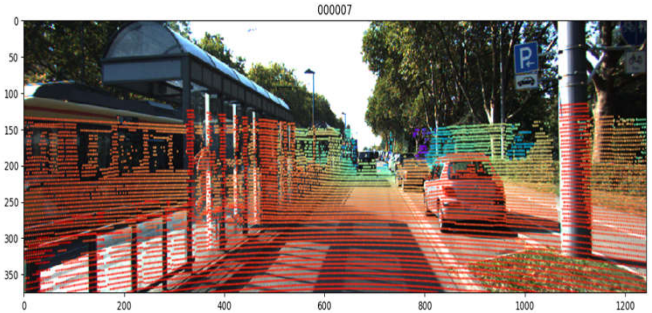
Multiple Object Tracking

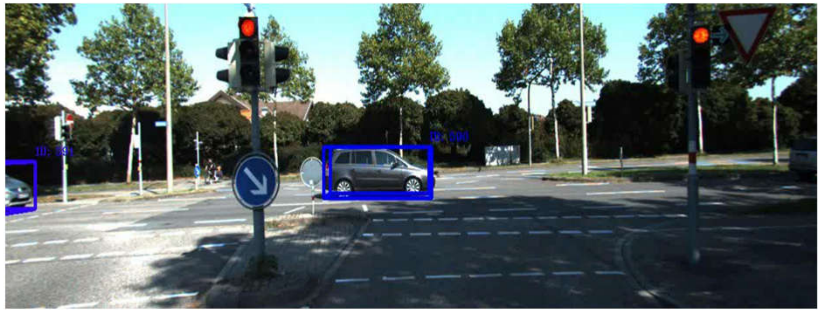
[摄像头&激光雷达数据融合框架]

黑箱建模——深度学习介绍

■深度学习在交通方向的应用

- 异构传感器数据融合

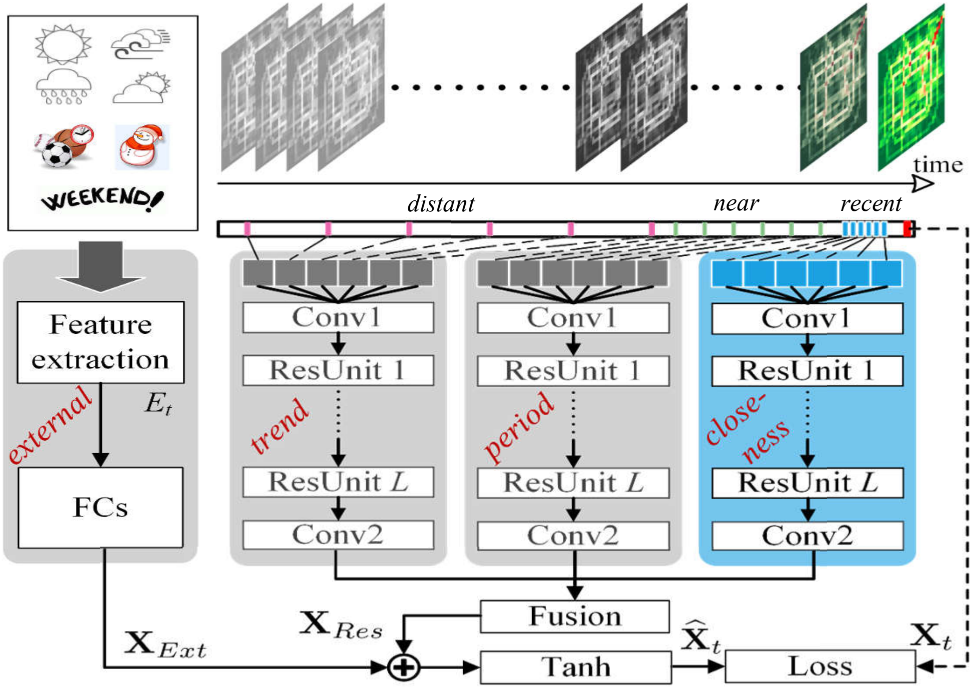




黑箱建模——深度学习介绍

■深度学习在交通方向的应用

- 交通流量预测



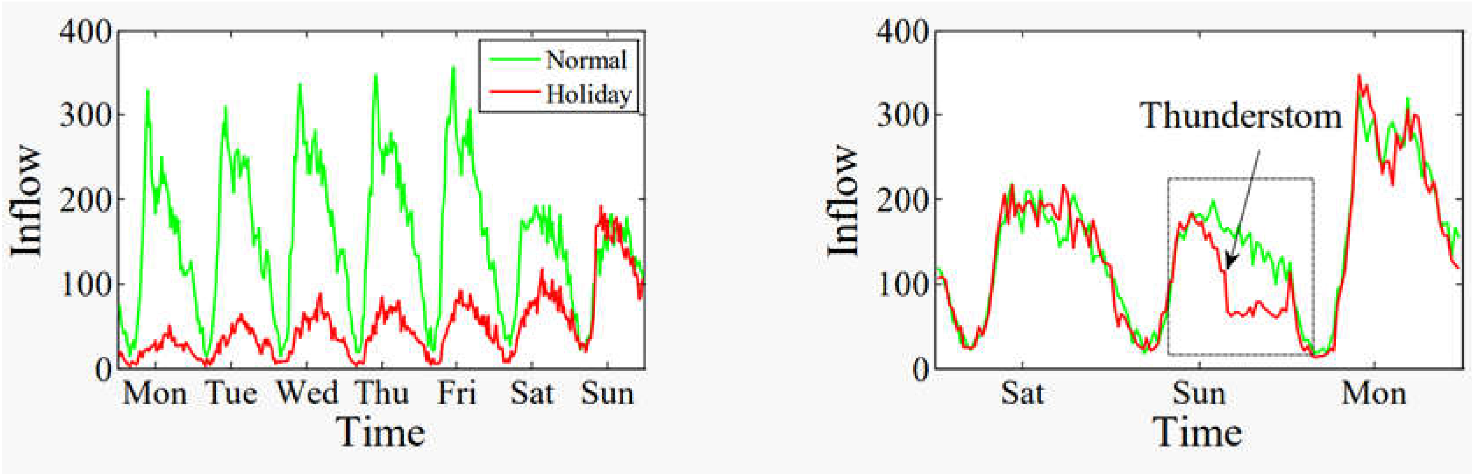
基于多元数据融合的交通流量预测

Junbo Zhang, Yu Zheng, et al. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction, AAAI 2017

黑箱建模——深度学习介绍

■深度学习在交通方向的应用

- 交通流量预测



工作日和假期

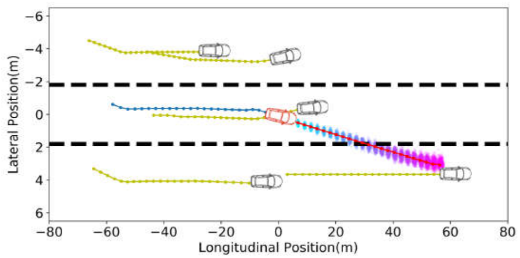
天气状况突变

Junbo Zhang, Yu Zheng, et al. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction, AAAI 2017

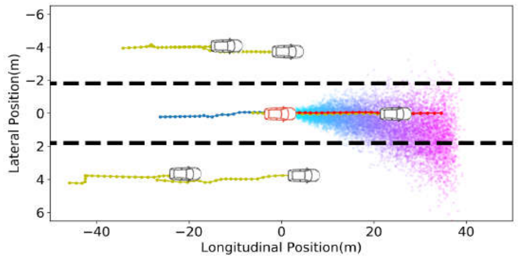
黑箱建模——深度学习介绍

■深度学习在交通方向的应用

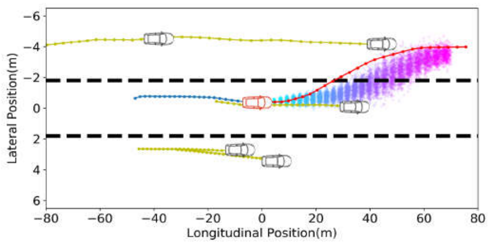
- 车辆轨迹预测



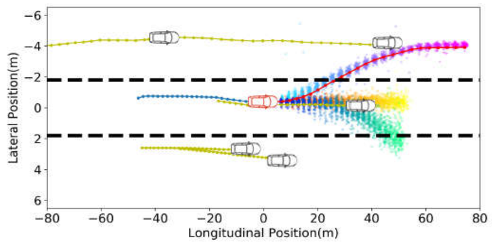
(a) Successful prediction with seq-to-seq



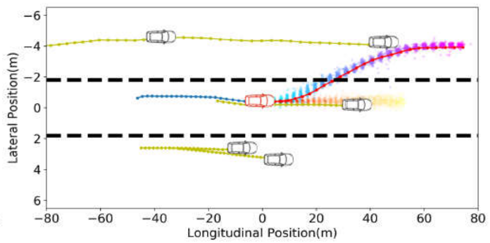
(b) Aliasing phenomenon with seq-to-seq



(c) Modified seq-to-seq



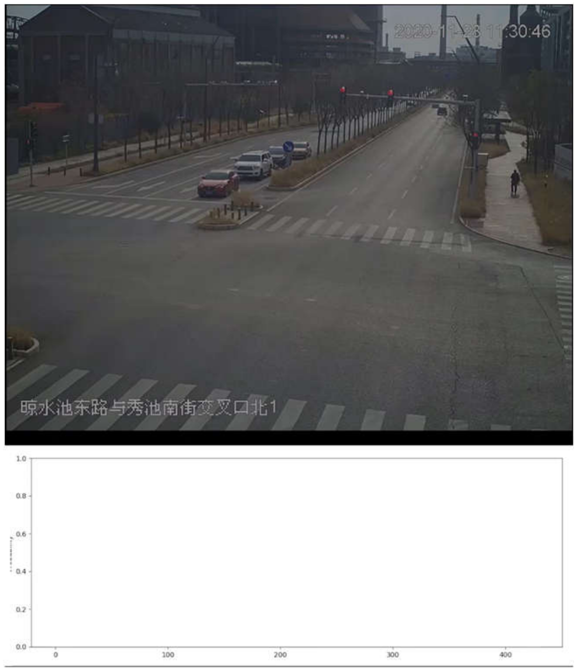
(d) All distribution with pure CVAE



(e) Final distribution with intention CVAE

■深度学习在交通方向的应用

- 交通事故风险预警与评估



- 本章小结
    - 黑箱建模的概念
    - 多项式逼近的优缺点
    - 两类典型的神经网络
    - 一元与多元线性回归
    - 病态线性回归原理与揭发
    - 基于深度学习的黑箱建模方法