机器学习 Machine learning

2022年春季学期

机器学习简介 及 决策树

刘扬 哈工大计算机学院自然计算研究室

相关信息

- ●教材
 - ○Mitchell, **机器学习 (必需)**
 - Ohris Bishop, Pattern Recognition and Machine Learning (电子版, 必需)
 - David Mackay, Information Theory, Inference, and Learning Algorithms
- ●教师
 - ○刘扬
 - ○yliu76@hit.edu.cn

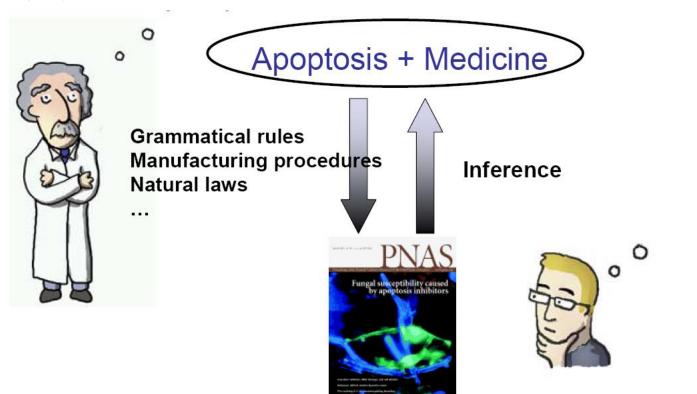
什么是学习?

- 定义1:由于经验或实践的结果而发生的持久或相对持久的适应性行为变化
- 定义2:能够使动物的行为对特定的环境条件发生适应性变化的所有过程,或者说是动物借助于个体生活经历和经验使自身的行为发生适应性变化的过程

○百度百科

什么是机器学习?

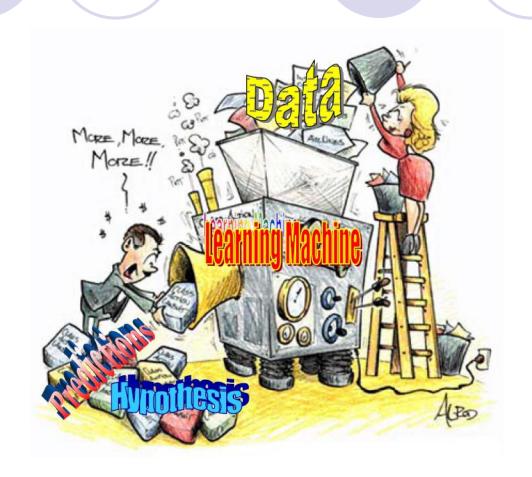
●是寻找一种对自然/人工主题、现象或活动可预测且/或可执行的机器理解方法



什么是机器学习

- 研究计算机怎样模拟或实现人类(动物)的学习行为,以获取新的知识或技能
- 重新组织已有的知识结构使之不断改善自 身的性能
- 是人工智能的核心,是使计算机具有智能的重要途径
- 其应用遍及人工智能的各个领域,它主要使用归纳、综合而不是演绎

机器学习的一个形象描述

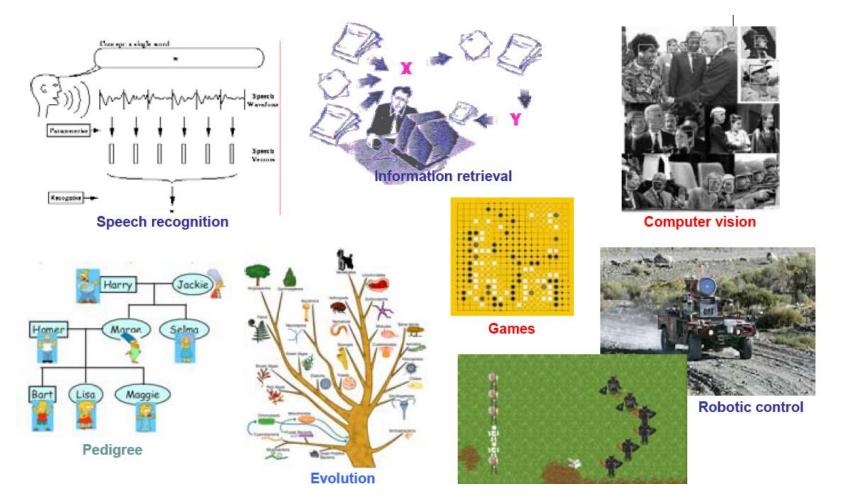


机器学习

- ●研究一种算法
 - ○提高它的性能 (P)
 - ○在某项任务中 (T)
 - ○利用一些经验 (E)

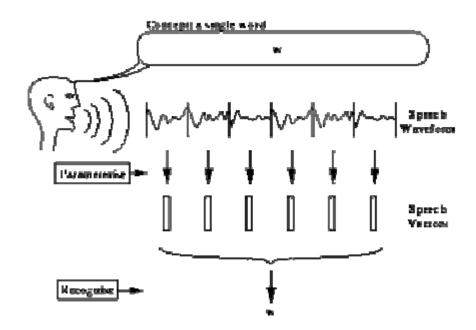
well-defined learning task: <P,T,E>

机器学习的应用



自然语言处理/语音识别

现在语音识别器或翻译器几乎都是建立在 某种具有学习能力的设备上---- 使用的越多, 则它越聪明

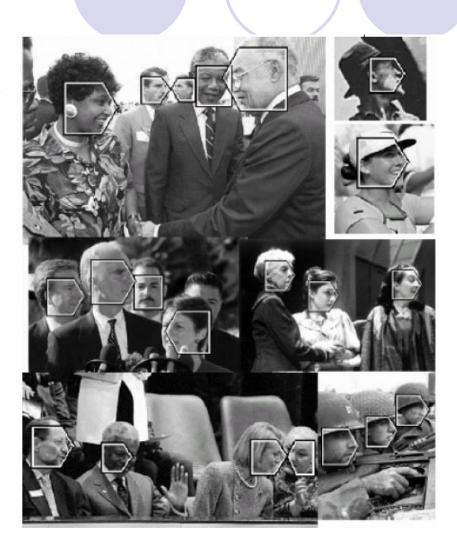


对象识别

 Behind a security camera, most likely there is a computer that is learning and/or checking!







机器人控制

• 汽车自动驾驶系统



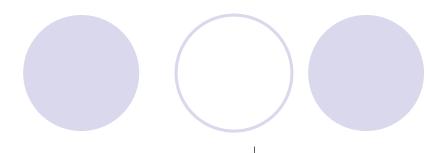
机器人控制

- The best helicopter pilot is now a computer!
 - it runs a program that learns how to fly and make acrobatic maneuvers by itself!
 - no taped instructions, joysticks, or things like that ...



 Reading, digesting, and categorizing a vast text database is too much for human!





We want:

FILM SHOW

MOVIE

PLAY

BEST

YORK

ACTOR

NEW



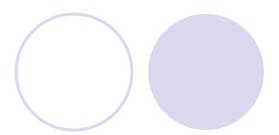
The William Randolph Heavet Foundation will give it 25 million to Lincoln Center, Metropoliton Opera Co., New York Philhermonic and Juliford School. "Our bound felt that we had a real opportunity to make a mark on the future of the performing arts with these greens as art every bit as important as one traditional areas of my in health, medical research, educacion and the social services," Beant Founda-Provident Randolph A. Heart said Vendey in measuring the greate. Lincoln Center's share will be \$100,000 for its new building, which will home young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive \$40,000 each. The Julilard School, where music and the performing arts are taught, will get \$550,000. The Heavy Francisco, a leading supporter of the Lincoln

生物信息学

cggcagcgggtgggggatcgggagccccccgaaaaaaacagaggcgcacgtcgatgccatcggggaattggaacctcaatgtgtgggaatgtttaaatattctgtgttataggtagtgtagtttcatagactatagattctcatacaga tectteqaqeeqattatacaeqaeaqeaaaatattteaqteqeqettqqqeaaaaqqettaaqeaeqaeteeceqtteceecttacatttqttetteetaaqeeeqtqqaqeeactateaaaettqttetaeqettqeaetqaaaataaaagtaaacaatcaaaaagaccaaaaaacaataacaaccagcagcagcgagtcgaacatcagtgaggcattgcaaaaatttcaaaggtcaaggttggcgtcatcgcggtctgagtcaggcggggcttgtaattgaagttgttgatg stggattgtggggaattctggtcagcatacttaacagcagccggctaattaagcaaaataaacatatcaaattccagaatgcgacggcgccatcatcctgtttgggaattcaattcgcgggcagatcgtttaattcaattaaaaggt agggagcagaagaatgcgatcgctggaatttcctaacatcacggaccccataaatttgataagcccgagctcgcttgagtcagccacatccccacatccccaaatccccgccaaaagacagctgggttgttgactcgccagat geagtggagtggacctggteaaagaagcaccgttaatgtgctgattccattcgattccatccgggaatgcgataaaggactctgatccaagcaactgcaatccggatttcgattttctatttccatttggttttgtatttacgt cattetaatgaagaettggagaagaettaegttatatteagaeeategtgegatagaggatgagteattteeatatggeegaaatttattatgtttaetategttttagaggtgttttttggaettaeeaaaagaggeatttgttt tacttgaggaattggcgatcgtagcgatttcccccgccgtaaagttcctgatcctcgttgttttttgtacatcataaagtccggattctgctcgtcgtcgcgaagatgggaacgaagctgccaaagctgagagtctgcttgaggtgctgg ccagctggataaccttgctgtacagatcggcatctgcctggagggcacgatcgaaatccttccagtggacgaacttcacctgctcgctgggaatagcgttgttgtcaagcagctcaaggagcgtattcgagttgacgggctgcacca stcottcgctggggattcccctgcgggtaagcgccttgcttggactcgtttccaaatcccatagccacgccagcagaggagtaacagagctcyhereisthegenetgattaaaaatatcctttaagaaagcccatgggtataacactqttatqattctcatqctqatqcqactqaaqcttcacctqactcctgctccacaattggtggcctttatatatagcgagatccacccgcato tgcgtggaatagaaa gcgggtgactccaggaattagcattatcgatcggaaag tcaaatagetgegateaetegggageagggtaaagtegeetegaaaeaggaagetgaageatettetataaataeaeteaaagegate/tteegaggegagtetggttagaaatttaeatggaetgeaaaaaggtatageeeeacaa ategetgegttteggeagetaattgeettttagaaattatttteeeatttegagaaetegtgtgggatgeeggatgeggettteaa/caettetggeeegggateggattgggteaeattgtetgegggetetattgtetegatee aaccaaaaaacagcagataaacaaaaacggcaaagaaaattgccacagagttgtcacgctttgttgcacaaacatttgtgcagaaaagtttttagccattattagccatta atgtteeteataaatgaaaattaatgtttgetetaegeteeaeegaactegettgtttgggggattggetagtegegetagateeeaggeggtataaeettttegetteateagttgtgaaaeeagatggetggtgttttgg eggactecetegaacgetetegaaateaagtggetttecageeggeeggeteggeegeteggeegatatteeegggeeacactgtacegeacacggatateetegeeagacteggeegataaaggeeaatg segeaggegtetatttatgecaaggacegttettetteagettteggetegagtatttgttgtgecatgttggttacgatgycaategeggtacagttatgcaaatgaaggagegatatacegeteactgacaatgaaeggegtettgt tcatgctgacattcatattcattcctttggttttttgtcttcgacggactgaaaagtgcggagaaacccaaaaacagagggcgcaaaggcgcgttaatatgcgaactcattgaagttatcacaacaccatatcca atccatatcaatatcaatatcgctattattaacgatcatgctctgctgatcaagtattcagcgctgcgctagattcgac, gattgaatcgagctcaatagactcaacag actccactcgacagatgcgcaatgccaaggacaattgc agtaaacqaggggtatqcqcaacctqcacctqqcqqacqcqqcqtatqcqcaatqtqcaattcqcttaccttctcqtt,cqqqtcaqqaactcccaqatqqqaatqqcc attactttcgccgcagtcgtcatggtgtcgttgctgcttttatgttgcgtactccgcactacacggagagttcaggggattcgtgctccgtgatctgtgatccgtgttccgtgttccggtgtgcaattgcacggttcggttgtgtaaccttcgt tttttttttagggcccaataaaagcgcttttgtggcggcttgatagattatcacttggtttcggtggctagccaagtggctttcttctgtccgacgcacttaattgaatta ggcagcgggtggggatcgggagccccccgaaaaaaaacagaggcgcacgtcgatgccatcgggaattggaacctcaatgtgtgggaatgtttaaatattctgtgtttaggtagtgtagtttcatagactatagattctcatacaga tect tegage egat tata cae gacage aaaa tatt teag tegeget tegge aaaa agget taa geae gac tee ceet tacatt tegte te tee tagge eact at caa act tegte te tagge act at caa act tegte te tagge eact at caa act tegte te tagge eact at caa act tegte te tagge each at caa act tegte each act tegte eachaaagtaaacaatcaaaaagaccaaaaaacaataacaaccagcagcagtcgaacatcagtgaggcattgcaaaaatttcaaagtcaagtttgcgtcgtcatcgcgtctgagtccgatcaagccgggcttgtaattgaagttgttgatg stggattgtggggaattetggteageataettaacageageeggetaattaageaaaataaacatateaaatteeagtgegegeeateateetgtttggggaatteaattegegggeagategtttaatteaattaaaaggt agggagcagaagaatgcgatcgctggaatttcctaacatcacggaccccataaatttgataagcccgagctcgcttgagtcagccacaccccacatccccaaatccccgccaaaaqacagctgggttgttgactcgccagat geagtggagtggacetggteaaagaageacegttaatgtgetgattecattegattecateegggaatgegataaaggatetgateeaageaactgcaateeggatttegattttetetttecatttggttttgtatttaegt cattetaatgaagaettggagaagaettaegttatatteagaeeategtgegatagaggatgagteattteeatatggeegaaatttattatgtttaetategtttttagagggtgtttttttggaeettaeeaaagaggeatttgttt ccagetggataacettgetgtacagateggeatetgeetggagggeacgatecteeagtggacgaactteacetgetegetgggaatagegttgttgteaageageteaaggagggtattegagtgaeggetgeacea

进化 ancestor years

机器学习的一般泛型



●监督学习

Given
$$D = \{\mathbf{X}_i, \mathbf{Y}_i\}$$
, learn $F(\cdot; \theta)$, s.t.: $\mathbf{Y}_i = F(\mathbf{X}_i)$ $D^{\text{new}} = \{\mathbf{X}_j\}$ \Rightarrow $\{\mathbf{Y}_j\}$

无监督学习

```
Given D = \{\mathbf{X}_i\}, learn F(\cdot; \theta), s.t.: \mathbf{Y}_i = F(\mathbf{X}_i) D^{\text{new}} = \{\mathbf{X}_j\} \Rightarrow \{\mathbf{Y}_j\}
```

• 强化学习

Given $D = \{\text{env}, \text{actions}, \text{rewards}, \text{simulator/trace/real game}\}$

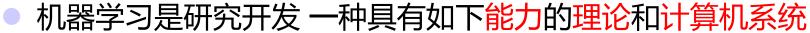
```
\begin{array}{ll} \mathsf{learn} & \mathsf{policy} : e, r \to a \\ & \mathsf{utility} : a, e \to r \end{array} \qquad \text{, s.t. } \{\mathsf{env}, \mathsf{new \ real \ game}\} \Rightarrow a_1, a_2, a_3 \dots \\ \end{array}
```

机器学习——理论

- ●对于学习到的函数F(; θ)
- ●一致性 (Consistency, 值, 模式、、)
- ●偏执与方差 (Bias versus variance)
- 采样复杂性 (Sample complexity)
- ●学习率 (learning rate)
- 收敛性 (Convergence)
- ●误差界(Error bound)
- 稳定性 (Stability)

— . . .

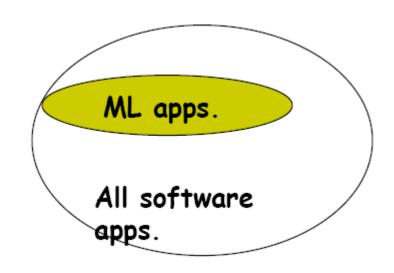
机器学习



- ○表示
- 分类,聚类和识别
- 不确定条件下推理
- 预测
- 对外界环境的反应
- O ...
- 可以在显示的模型或数学框架下,根据数据和自身经验,复杂的真实世界信息可以:
 - 被形式 (formally) 刻画和分析
 - 加入人的先验
 - 具有在数据与领域间的泛化和适应能力
 - 自动或自主操作
 - 被人类解释和感知

机器学习的快速增长

- 机器学习是最受青睐的方法
 - ○语音识别,自然语言处理
 - ○计算机视觉
 - ○医学处理
 - ○机器人控制
- 机器学习分量在增加
 - ○机器学习性能不断提高
 - ○数据快速增长,网络发展
 - ○一些软件太复杂人工撰写比较困难
 - ○新的感知器件/IO设备
 - ○环境与用户的个性化服务



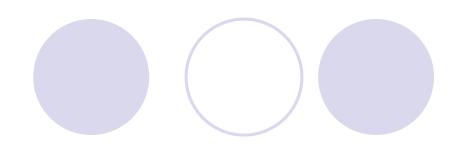
- ●推理 (Inference)
- 预测 (Prediction)
- 不确定性环境下的决策

. . . .

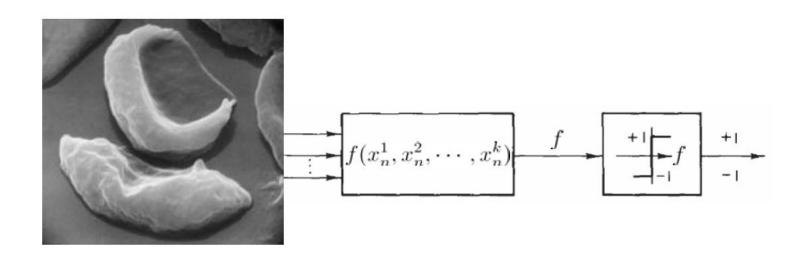
→统计机器学习

 \rightarrow 函数近似: $F(|\theta)$

分类



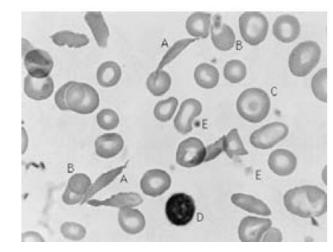
●镰刀形细胞贫血症



■函数近似

函数近似

- ●设置
 - ○示例集合X
 - ○未知的目标函数 $f: X \to Y$
 - ○函数假设集合H = {h|h: X → Y}



- •给定
 - 〇目标函数f的训练样本 $\{\langle x_i, y_i \rangle\}$

• 确定 $h \in H$,可以最好地近似f

税务欺诈检测问题

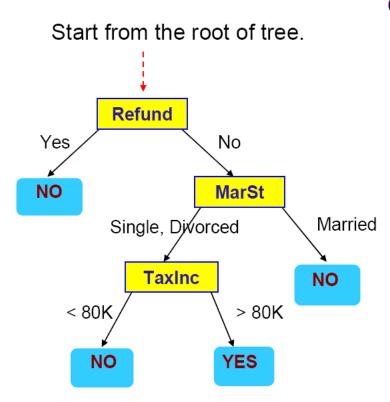


Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

- 采用什么样的函数f
 - ○假设空间是什么的样的

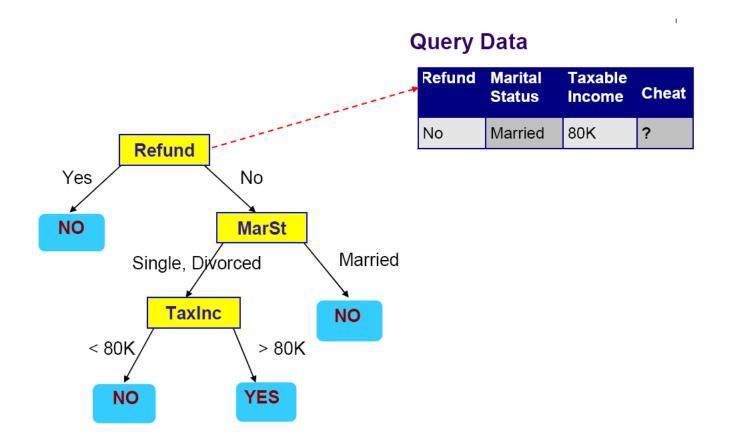
● 如何使用?

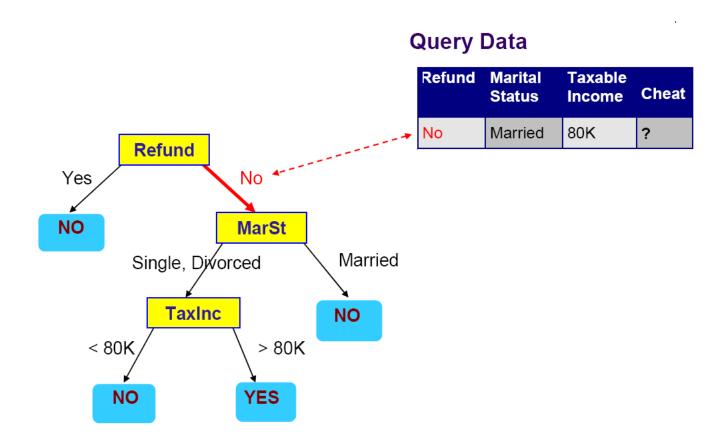


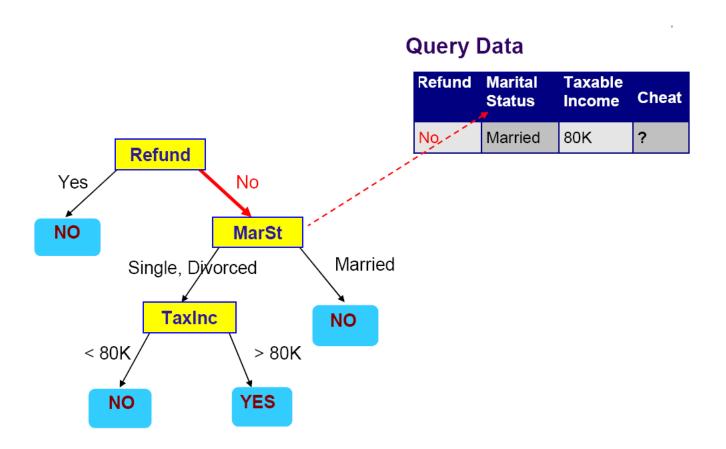


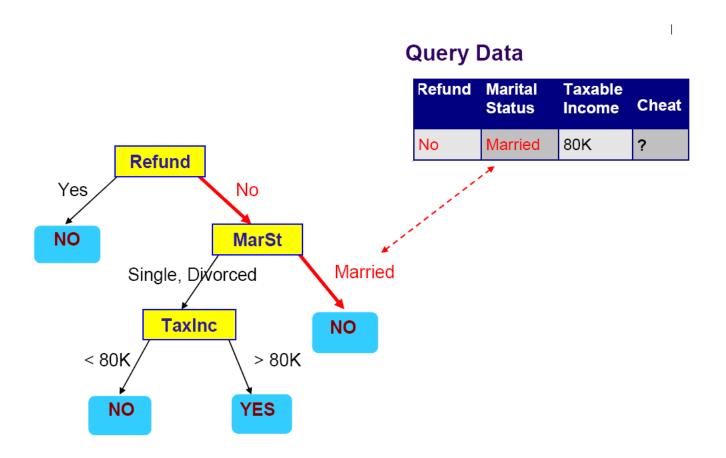
Query Data

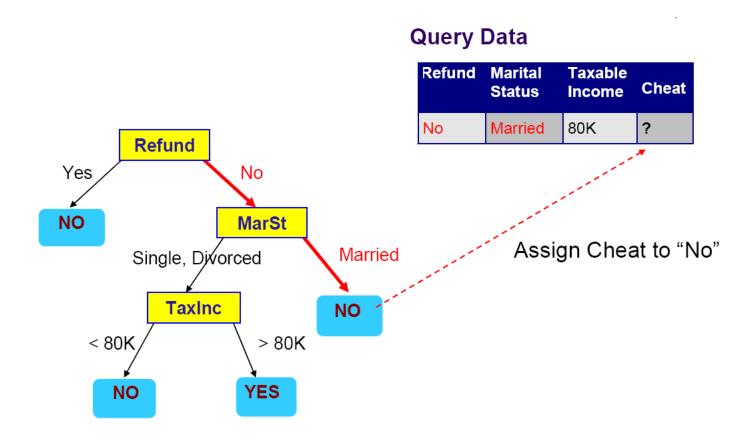
Refund	Marital Status		Cheat
No	Married	80K	?











税务欺诈的一个假设(决策树)

- 輸入:属性向量
 - X=[Refund, MarSt, TaxInc]
- 输出:
 - Y=是否欺诈
- H 就是不同的决策过程 (不同的决策树)
- Yes No

 NO

 MarSt

 Single, Divorced

 Married

 NO

 < 80K

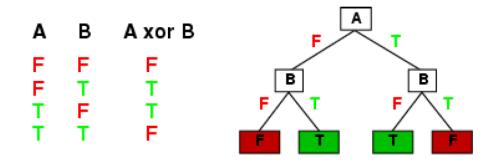
 NO

 YES

- 每一个内结点:测试一个 属性X_i
- 每个分支:选择属性X_i的 一个取值
- 每个叶结点: 预测Y

决策树的表示能力

- 决策树可以表示输入属性的任何函数
- 例如,对于布尔函数,真值表的一行→一条路径(根到叶)

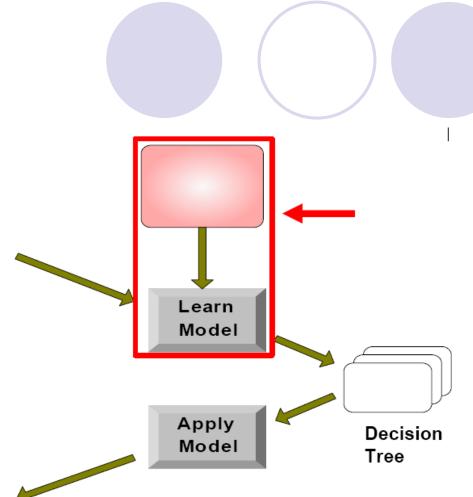


- 如果对训练数据中的每个样例都建立一条从根到叶的路径(除非f对于输入x是不确定的)就得到一个一致的决策树,但其可能没有泛化能力
- ▶ 因此希望找一个更加紧凑(小规模)的决策树

决策树学习

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

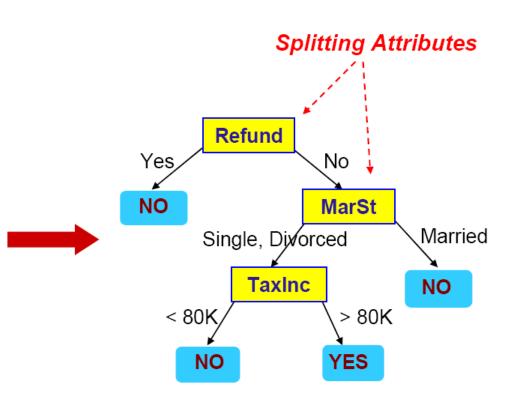
Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?



一棵决策树(欺诈问题)

categorical continuous

	•	•	•	
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



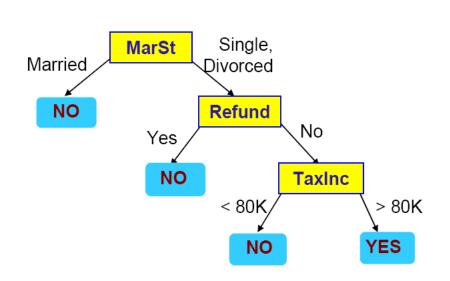
Training Data

Model: Decision Tree

另一棵决策树

categorical categorical continuous

	•	•	•	
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Training Data

同样一个训练数据集,可以有多棵树与其

Top-Down的决策树归纳算法(构造)

- Main loop:
- 1. A←下一个结点node的最好属性
- 2. 把A作为决策属性赋给结点node
- 3. 对A的每一个取值,创建一个新的儿子结点node
- 4. 把相应的训练样本分到叶结点
- 5. 如果训练样本被很好的分类,则**停止,否则在新的叶结 点上重复上述过程**

哪个属性更好?



树的归纳

- ●贪心策略
 - ○基于一个可以最优化某项准则的属性来切分示例 集和
- ●问题
 - ○如何切分示例集和
 - ●如何确定属性的测试条件?
 - ●如何确定最好的切分?
 - ○停止切分准则

树的归纳

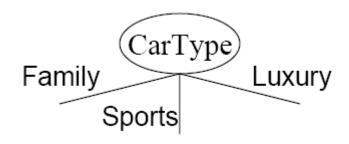
- ●贪心策略
 - ○基于一个可以最优化某项准则的属性来切分示例 集和
- ●问题
 - ○如何切分示例集和
 - ●如何确定最好的切分?
 - ●如何确定属性的测试条件?
 - ○停止切分准则

如何确定特测条件?

- 依赖于属性类型
 - ○名词性\离散 (Nominal)
 - ○有序的 (Ordinal)
 - 连续 (Continuous)
- 依赖于切分的分支个数
 - ○两路切分 (2-way)
 - ○多路切分(Multi-way)

对名词属性的切分

● 多路切分: 一个离散属性值对应一路切分

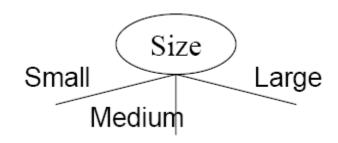


两路切分: 离散属性值被切分成两个子集, 需要寻找最优切分

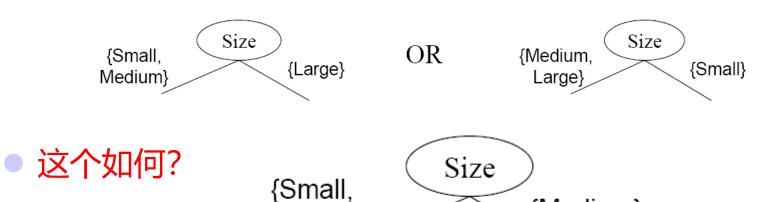


对有序属性的切分

● 多路切分: 一个属性值对应一路切分



两路切分:属性值被切分成两个子集,需要寻找最优切分



Large}

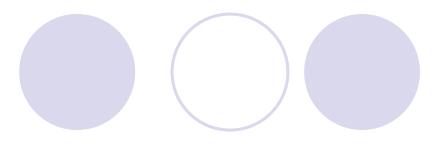
{Medium}

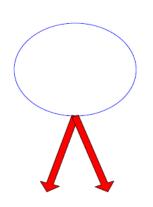
40

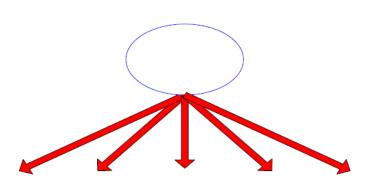
对连续属性的切分

- 不同的处理方案
 - ○离散化构造有序的类属性
 - ●静态——在起始位置一次离散化
 - ●动态——范围可以通过等区间或等频率确定,或者是 聚类
 - ○二值决策: (A<V)or(A>=V)
 - ●考虑所有可能的切分并选择最好的
 - ●计算量可能非常大

对连续属性的切分





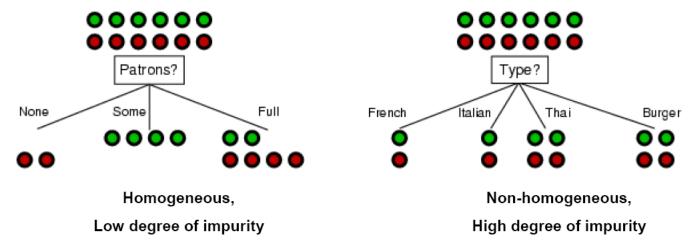


树的归纳

- ●贪心策略
 - ○基于一个可以最优化某项准则的属性来切分示例 集和
- ●问题
 - ○如何切分示例集和
 - ●如何确定最好的切分?
 - ●如何确定属性的测试条件?
 - ○停止切分准则

如何确定最好的切分

■ Idea: 一个好的属性切分是将示例集合分成若干子集, 最理想情况是每个子集为"皆为正例"或"皆为反例"



- 含心搜索
 - ○更倾向结点上的数据具有同质(homogeneous)类别分 布
- ■需要对结点混杂度 (impurity) 进行测量

如何衡量属性的"好"与"坏"

- 熵(Entropy)
 - 随机变量X的熵H(X)
 - \bigcirc H(X)是对从X随机采样值在最短编码情况下的每个值平均(期望)长度(以2为底就是0、1编码)

$$H(X) = -\sum_{i=1}^{N} P(x = i) \log_2 P(x = i)$$

OWhy?

信息论:

在最短编码情况下,对消息X = i分配 $-\log_2 P(X = i)$ 位, 所以其编码一个随机变量X的期望位数是

$$H(X) = -\sum_{i=1}^{N} P(x = i) \log_2 P(x = i)$$

如何衡量属性的"好"与"坏"



 $\bigcirc X$ 在给定Y = v特定条件熵H(X|Y = v)

$$H(X|y = j) = -\sum_{i=1}^{N} P(x = i|y = j) \log_2 P(x = i|y = j)$$

 $\bigcirc X$ 在给定Y条件熵H(X|Y)

$$H(X|Y) = \sum_{j \in Val(y)} P(y = j) H(X|y = j)$$

● X和Y的互信息

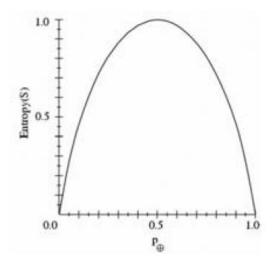
$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$$

= $H(X) + H(Y) - H(X,Y)$

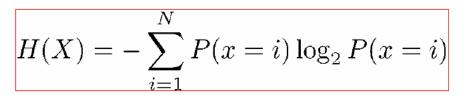
样本熵

- S是训练例子的样本集
- P₊是S中的正例比例
- P.是S中反例的比例
- ●用熵来测量S的混杂度

$$H(S) \equiv -p_{+} \log_{2} p_{+} - p_{-} \log_{2} p_{-}$$



计算熵的例子



$$P(C1) = 0/6 = 0$$
 $P(C2) = 6/6 = 1$

Entropy =
$$-0 \log 0 - 1 \log 1 = -0 - 0 = 0$$

$$P(C1) = 1/6$$
 $P(C2) = 5/6$

Entropy =
$$-(1/6) \log_2 (1/6) - (5/6) \log_2 (1/6) = 0.65$$

$$P(C1) = 2/6$$
 $P(C2) = 4/6$

Entropy =
$$-(2/6) \log_2 (2/6) - (4/6) \log_2 (4/6) = 0.92$$

信息增益

● 信息増益

$$GAIN_{split} = Entropy(p) - \left(\sum_{i=1}^{k} \frac{n_i}{n} Entropy(i)\right)$$

○父结点P被切分成k部分; n_i是每一切分的样本数即目标类变量与属性A变量在S(样本集)上的互信息

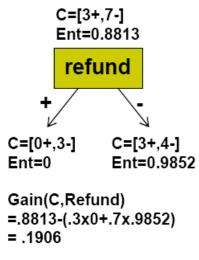


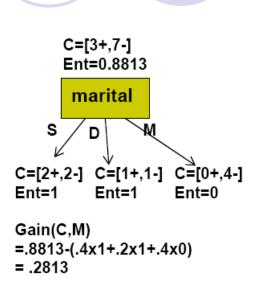
- 测量由于切分带来的熵减少量,选择具有最大减少量的切分(最大增益)
- 在ID3和C4.5中采用
- 缺点:倾向选择具有切分分支多的属性,因为每份可以很少的样本,但很纯

例子

categorical continuous

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes





选择哪个属性作为根节点?

Training Data

树归纳的停止准则

当一个结点上所有样本属于同一个类别, 停止扩展

当一个结点上所有样本具有相似的属性值, 停止扩展

•提早结束(以后会介绍)

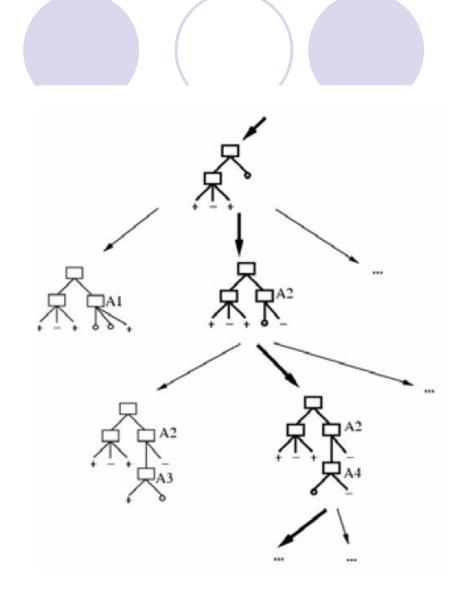
基于决策树的分类

- 优点
 - ○构建过程计算资源开销小
 - ○分类未知样本速度极快
 - ○对于小规模的树比较容易解释
 - ○在许多小的简单数据集合上性能与其它方法相近
- 例子: C4.5
 - ○深度优先构建方法
 - ○采用了信息增益
 - ○在每一个结点需要对示例依据连续属性排序
 - ○数据需要全部装入内存
 - ○不适合大规模数据

应该输出哪棵树

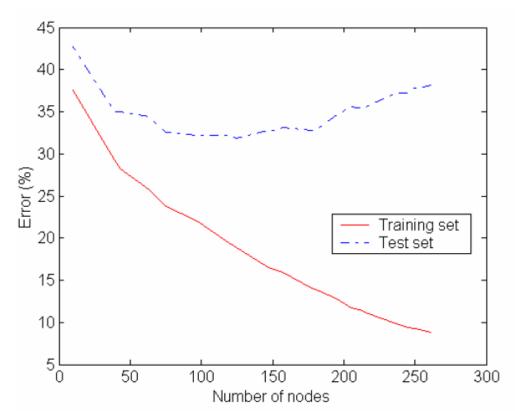
- ID3在决策树空间中执 行启发式搜索
- 当获得最小可接受的 决策树时则停止
- Why?

Occam's 剃刀:选择适合训练 集合数据的最简单假设



决策树的实践问题

- 欠拟合 (underfitting) 和过拟合 (Overfitting)
- 特征值丢失



总结: 你应当知道的

- 适定的函数近似问题
 - ○示例空间,X
 - 标注的训练样本{<x_i,y_i>}
 - ○假设空间, H={f:X→Y}
- 学习是H空间上的搜索/优化问题
 - ○各种各样目标函数
 - ●最小化训练误差(0-1损失)
 - 在所有的满足最小误差的假设中,选择最小的(?)
- 决策树学习
 - 贪心的Top-down决策树
 - ○过拟合以及剪枝
 - ○扩展

思考问题 (1)

●ID3和C4.5在树空间上执行启发式算法。为 什么不做穷尽搜索?

思考问题 (2)

●考虑目标函数f:<x1,x2>→y,这里x1和x2是实数,如果在树中每个属性只被允许使用一次,那么决策树可以表达的决策平面是什么样子的?

思考问题 (3)



假设空间大小

- n布尔特征有多少个决策树
- >布尔函数函数个数
- =具有 2^n 行的真值表个数= 2^{2^n}

有6个布尔属性,则有 18,446,744,073,709,551,616棵树

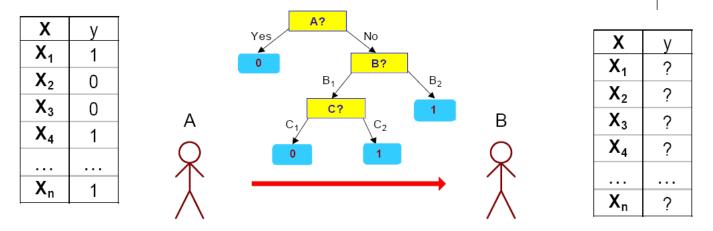
Notes on Overfitting

- Overfitting results in decision trees that are more complex than necessary
- Training error no longer provides a good estimate of how well the tree will perform on previously unseen records
- Which Tree Should We Output?
 - Occam's razor: prefer the simplest hypothesis that fits the data

Occam's Razor

- Given two models of similar generalization errors, one should prefer the simpler model over the more complex model
- For complex models, there is a greater chance that it was fitted accidentally by errors in data
- Therefore, one should include model complexity when evaluating a model

Minimum Description Length (MDL)



- Cost(Model,Data) = Cost(Data|Model) + Cost(Model)
 - Cost is the number of bits needed for encoding.
 - Search for the least costly model.
- Cost(Data|Model) encodes the misclassification errors.
- Cost(Model) uses node encoding (number of children) plus splitting condition encoding.

How to Address Overfitting

- Pre-Pruning (Early Stopping Rule)
 - Stop the algorithm before it becomes a fully-grown tree
 - Typical stopping conditions for a node:
 - Stop if all instances belong to the same class
 - Stop if all the attribute values are the same
 - More restrictive conditions:
 - Stop if number of instances is less than some user-specified threshold
 - Stop if class distribution of instances are independent of the available features (e.g., using χ 2 test)
 - Stop if expanding the current node does not improve impurity measures (e.g., Gini or information gain).

How to Address Overfitting...

Post-pruning

- Grow decision tree to its entirety
- Trim the nodes of the decision tree in a bottom-up fashion
- If generalization error improves after trimming, replace sub-tree by a leaf node.
- Class label of leaf node is determined from majority class of instances in the sub-tree
- Can use MDL for post-pruning

Handling Missing AttributeValues

- Missing values affect decision tree construction in three different ways:
 - Affects how impurity measures are computed
 - Affects how to distribute instance with missing value to child nodes
 - Affects how a test instance with missing value is classified

Computing Impurity Measure

Tid	Refund	Marital Status	Taxable Income	Class
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	?	Single	90K	Yes

Missing value

Before Splitting:

Entropy(Parent)

 $= -0.3 \log(0.3) - (0.7) \log(0.7) = 0.8813$

	Class = Yes	Class = No
Refund=Yes	0	3
Refund=No	2	4
Refund=?	1	0

Split on Refund:

Entropy(Refund=Yes) = 0

Entropy(Refund=No)

 $= -(2/6)\log(2/6) - (4/6)\log(4/6) = 0.9183$

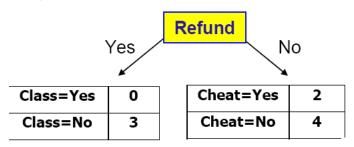
Entropy(Children)

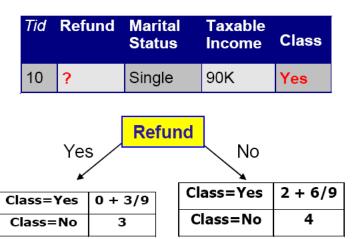
= 0.3(0) + 0.6(0.9183) = 0.551

Gain = $0.9 \times (0.8813 - 0.551) = 0.3303$

Distribute Instances

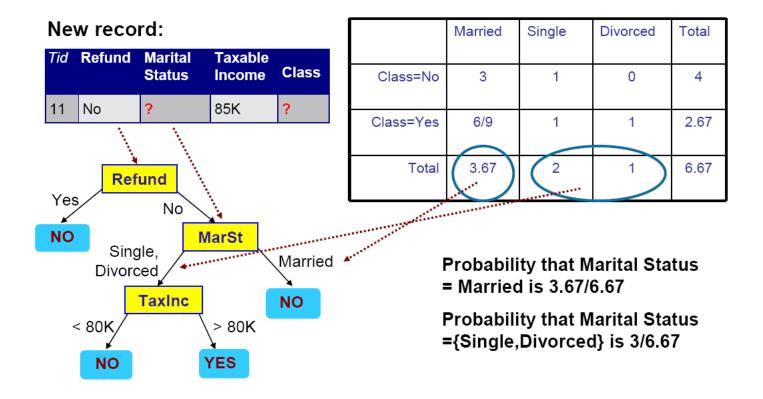
Tid	Refund	Marital Status	Taxable Income	Class
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No





Probability that Refund=Yes is 3/9
Probability that Refund=No is 6/9
Assign record to the left child with weight = 3/9 and to the right child with weight = 6/9

Classify Instances

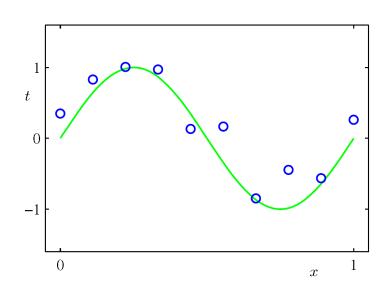


例子: 曲线拟合 (回归问题)

- ●问题: 观测一个实数值x,并用此值来预测目标值t
 - ○人工合成数据(便于研究)
 - \circ sin(2 πx)
- ●假设给定N个观测数据的*训练集合*
 - $\bigcirc X \equiv (x_1, \cdots, x_n)^T$
 - \bigcirc T $\equiv (t_1, \dots, t_n)^T$

利用 $sin(2\pi x)$ 产生样本

- N=10, x均匀分布在[0,1]
- 对每一个目标值t加一个0均值的高斯噪声



问题的目标

- ●利用训练集合
- 对每一个新 \hat{x} , 预测目标值 \hat{t}
- 实质是学习正弦函数(即真实函数f)
 - ○这是非常困难的
 - 〇采用什么样的假设空间H?
 - ○多项式函数

$$y(x, w) = w_0 + w_1 x + \dots + w_m x^m = \sum_{i=0}^m w_i x^i$$

 $\bigcirc y(x,w)$ 是x的多项式函数, w的线性函数

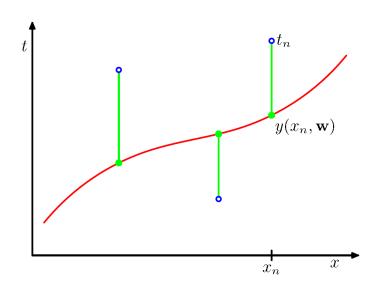
参数的确定(优化)

- ●如何确定参数w (假设给定m)
 - ○建立误差函数,测量每个样本点目标值t与预测 函数y之间的误差

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2$$

- ○显然是最小二乘问题
 - ●E是w的二次函数
 - ●求导,设倒数=0
 - ●存在唯一解w*

思考:为什么E中有1/2



多项式函数的阶

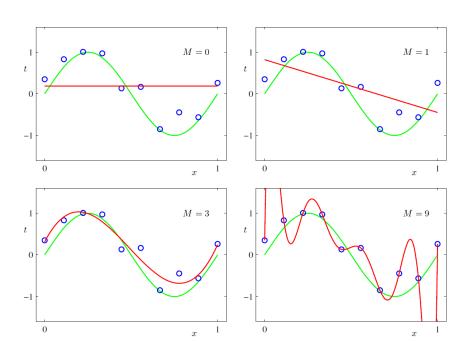
- 多项式函数阶的作用
 - ○控制模型的表达能力
 - ○模型的复杂 (灵活) 程度

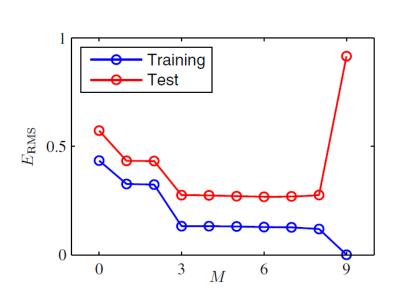
- 多项式函数阶的确定
 - ○模型比较
 - ○模型选择

模型复杂度与过学习

- 阶数M取不同值时,最佳拟合曲线情况
- ●拟合优度评价
 - ○利用优化目标函数值
 - ○方均根值(有时更方便)

$$E_{\rm RMS} = \sqrt{2E(\mathbf{w}^{\star})/N}$$





实验观察与分析

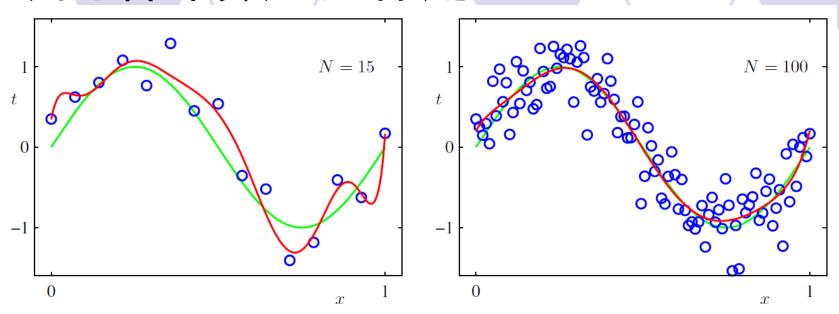
- 阶数M较小时
 - ○模型表达能力有限(不灵活)
 - ○方均根误差大
- 阶数M=3-8时
 - ○拟合较好
 - ○方均根误差较小
- 阶数M=9时
 - ○完全拟合
 - ○误差为0
 - ○但曲线已不具有正弦形状,为什么?
 - ○能力最强

实验观察与分析

- ●N=10个样本
 - OM=9,正好可以经过这N个训练样本
 - ○出现*过学习*
- ●不同阶数时最佳拟合曲线参数w*的值

	M = 0	M = 1	M = 6	M = 9
w_0^{\star}	0.19	0.82	0.31	0.35
w_1^\star		-1.27	7.99	232.37
w_2^{\star}			-25.43	-5321.83
$w_3^{\overline{\star}}$			17.37	48568.31
w_4^{\star}				-231639.30
w_5^{\star}				640042.26
w_6^{\star}				-1061800.52
w_7^{\star}				1042400.18
w_8^{\star}				-557682.99
w_{9}^{\star}				125201.43

训练样本数量的作用



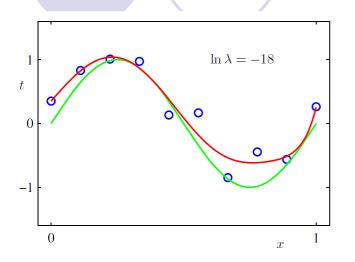
- M=9
- 增大样本数量可减少过学习程度

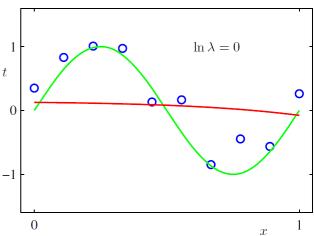
模型的复杂程度

- ●模型复杂度与什么有关?
 - ○模型参数个数?
 - ○模型参数w*的绝对值(范数)?
- ●回想模型最佳参数w*的特点
 - ○参数多时、w*往往具有大的绝对值
 - \bigcirc 可以在优化目标函数E(w)中加入对w的惩罚

$$\widetilde{E}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2} ||\mathbf{w}||^2$$

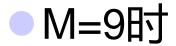
加入正则项后的实验



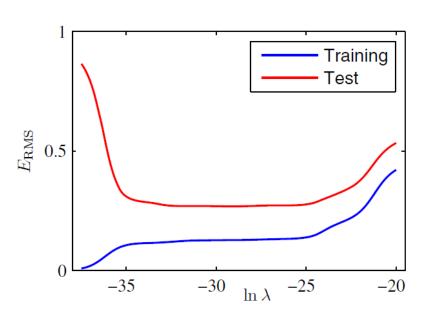


- 惩罚项比重影响
 - ○比重大时降低模型复杂度
 - ○比重适当时模型复杂度与问题匹配
 - ○比重小时、退化成原模型

加入正则项后的实验



	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
w_0^{\star}	0.35	0.35	0.13
w_1^{\star}	232.37	4.74	-0.05
w_2^{\star}	-5321.83	-0.77	-0.06
w_3^{\star}	48568.31	-31.97	-0.05
w_4^{\star}	-231639.30	-3.89	-0.03
w_5^{\star}	640042.26	55.28	-0.02
w_6^{\star}	-1061800.52	41.32	-0.01
w_7^{\star}	1042400.18	-45.95	-0.00
w_{8}^{\star}	-557682.99	-91.53	0.00
w_{9}^{\star}	125201.43	72.68	0.01



●如何设置权重参数λ?

权重参数λ确定

- 实验方法
 - ○分为训练集合 (training set)
 - ○验证集合 (validation set)
 - ○测试集合 (Test Set)