

VIP Cheatsheet: Supervised Learning

Afshine AMIDI and Shervine AMIDI

October 13, 2018

翻译: Wang Hongnian. 由朱小虎, Chaoying Xue and Z 审阅

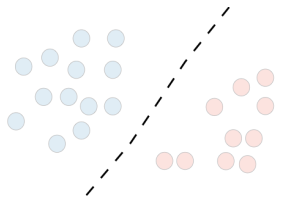
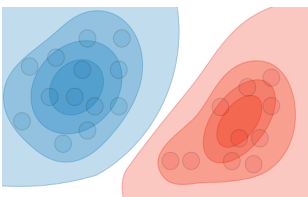
监督学习简介

给定一组数据点 $\{x^{(1)}, \dots, x^{(m)}\}$ 和与其对应的输出 $\{y^{(1)}, \dots, y^{(m)}\}$, 我们想要建立一个分类器, 学习如何从 x 预测 y 。

□ **预测类型** – 不同类型的预测模型总结如下表:

	回归	分类
输出	连续	类
例子	线性回归	Logistic回归, SVM, 朴素贝叶斯

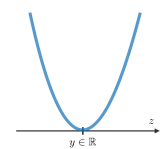
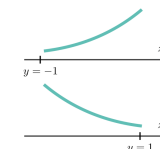
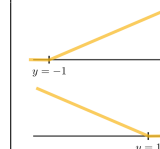
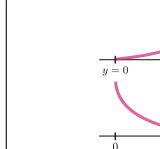
□ **型号类型** – 不同型号总结如下表:

	判别模型	生成模型
目标	直接估计 $P(y x)$	估计 $P(x y)$ 然后推导 $P(y x)$
所学内容	决策边界	数据的概率分布
例图		
示例	回归, SVMs	GDA, 朴素贝叶斯

符号和一般概念

□ **假设** – 假设我们选择的模型是 h_θ 。对于给定的输入数据 $x^{(i)}$, 模型预测输出是 $h_\theta(x^{(i)})$ 。

□ **损失函数** – 损失函数是一个 $L: (z, y) \in \mathbb{R} \times Y \mapsto L(z, y) \in \mathbb{R}$ 的函数, 其将真实数据值 y 和其预测值 z 作为输入, 输出它们的不同程度。常见的损失函数总结如下表:

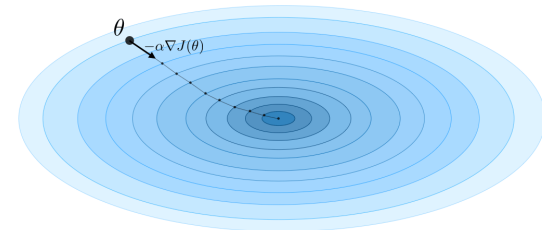
最小二乘误差	Logistic损失	铰链损失	交叉熵
$\frac{1}{2}(y - z)^2$	$\log(1 + \exp(-yz))$	$\max(0, 1 - yz)$	$-\left[y \log(z) + (1 - y) \log(1 - z)\right]$
			
线性回归	Logistic回归	SVM	神经网络

□ **成本函数** – 成本函数 J 通常用于评估模型的性能, 使用损失函数 L 定义如下:

$$J(\theta) = \sum_{i=1}^m L(h_\theta(x^{(i)}), y^{(i)})$$

□ **梯度下降** – 记学习率为 $\alpha \in \mathbb{R}$, 梯度下降的更新规则使用学习率和成本函数 J 表示如下:

$$\theta \leftarrow \theta - \alpha \nabla J(\theta)$$



备注: 随机梯度下降 (SGD) 是根据每个训练样本进行参数更新, 而批量梯度下降是在一批训练样本上进行更新。

□ **似然** – 给定参数 θ 的模型 $L(\theta)$ 的似然性用于通过最大化似然性来找到最佳参数 θ 。在实践中, 我们使用更容易优化的对数似然 $\ell(\theta) = \log(L(\theta))$ 。我们有:

$$\theta^{\text{opt}} = \arg \max_{\theta} L(\theta)$$

□ **牛顿算法** – 牛顿算法是一种数值方法, 目的是找到一个 θ 使得 $\ell'(\theta) = 0$ 。其更新规则如下:

$$\theta \leftarrow \theta - \frac{\ell'(\theta)}{\ell''(\theta)}$$

备注: 多维泛化, 也称为 *Newton-Raphson* 方法, 具有以下更新规则:

$$\theta \leftarrow \theta - \left(\nabla_{\theta}^2 \ell(\theta)\right)^{-1} \nabla_{\theta} \ell(\theta)$$

线性回归

我们假设 $y|x; \theta \sim \mathcal{N}(\mu, \sigma^2)$

□ **正规方程** – 通过设计 X 矩阵，使得最小化成本函数时 θ 有闭式解：

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

□ **LMS算法** – 通过 α 学习率，训练集中 m 个数据的最小均方（LMS）算法的更新规则也称为Widrow-Hoff学习规则，如下：

$$\forall j, \quad \theta_j \leftarrow \theta_j + \alpha \sum_{i=1}^m [y^{(i)} - h_{\theta}(x^{(i)})] x_j^{(i)}$$

备注：更新规则是梯度上升的特定情况。

□ **LWR** – 局部加权回归，也称为LWR，是线性回归的变体，通过 $w^{(i)}(x)$ 对其成本函数中的每个训练样本进行加权，其中参数 $\tau \in \mathbb{R}$ 定义为：

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

分类和逻辑回归

□ **Sigmoid函数** – sigmoid 函数 g ，也称为逻辑函数，定义如下：

$$\forall z \in \mathbb{R}, \quad g(z) = \frac{1}{1 + e^{-z}} \in [0, 1]$$

□ **逻辑回归** – 我们假设 $y|x; \theta \sim \text{Bernoulli}(\phi)$ 。我们有以下形式：

$$\phi = p(y = 1|x; \theta) = \frac{1}{1 + \exp(-\theta^T x)} = g(\theta^T x)$$

备注：对于逻辑回归的情况，没有闭式解。

□ **Softmax回归** – 当存在超过2个结果类时，使用softmax回归（也称为多类逻辑回归）来推广逻辑回归。按照惯例，我们设置 $\theta_K = 0$ ，使得每个类 i 的伯努利参数 ϕ_i 等于：

$$\phi_i = \frac{\exp(\theta_i^T x)}{\sum_{j=1}^K \exp(\theta_j^T x)}$$

广义线性模型

□ **指数分布族** – 如果可以用自然参数 η ，也称为规范参数或链接函数，充分统计量 $T(y)$ 和对数分割函数 $a(\eta)$ 来表示，则称一类分布在指数分布族中，函数如下：

$$p(y; \eta) = b(y) \exp(\eta^T T(y) - a(\eta))$$

备注：我们经常会有 $T(y) = y$ 。此外， $\exp(-a(\eta))$ 可以看作是归一化参数，确保概率总和为1

下表是总结的最常见的指数分布：

分布	η	$T(y)$	$a(\eta)$	$b(y)$
伯努利	$\log\left(\frac{\phi}{1-\phi}\right)$	y	$\log(1 + \exp(\eta))$	1
高斯	μ	y	$\frac{\eta^2}{2}$	$\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y^2}{2}\right)$
泊松	$\log(\lambda)$	y	e^η	$\frac{1}{y!}$
几何	$\log(1 - \phi)$	y	$\log\left(\frac{e^\eta}{1 - e^\eta}\right)$	1

□ **GLM的假设** – 广义线性模型（GLM）是旨在将随机变量 y 预测为 $x \in \mathbb{R}^{n+1}$ 的函数，并依赖于以下3个假设：

$$(1) \quad y|x; \theta \sim \text{ExpFamily}(\eta) \quad (2) \quad h_{\theta}(x) = E[y|x; \theta] \quad (3) \quad \eta = \theta^T x$$

备注：普通最小二乘法和逻辑回归是广义线性模型的特例

支持向量机

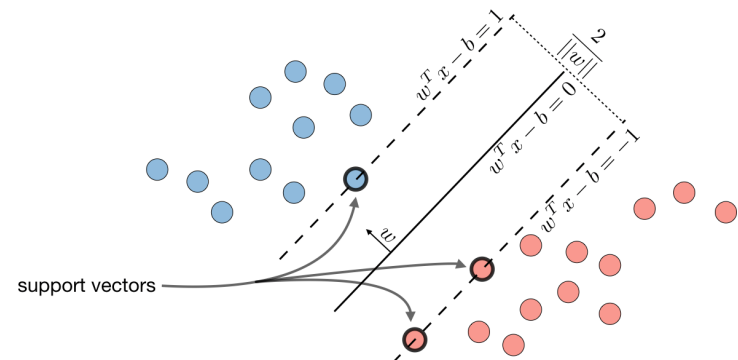
支持向量机的目标是找到使决策界和训练样本之间最大化最小距离的线。

□ **最优间隔分类器** – 最优间隔分类器 h 是这样的：

$$h(x) = \text{sign}(w^T x - b)$$

其中 $(w, b) \in \mathbb{R}^n \times \mathbb{R}$ 是以下优化问题的解决方案：

$$\min \frac{1}{2} \|w\|^2 \quad \text{使得} \quad y^{(i)}(w^T x^{(i)} - b) \geq 1$$



备注：该线定义为 $w^T x - b = 0$ 。

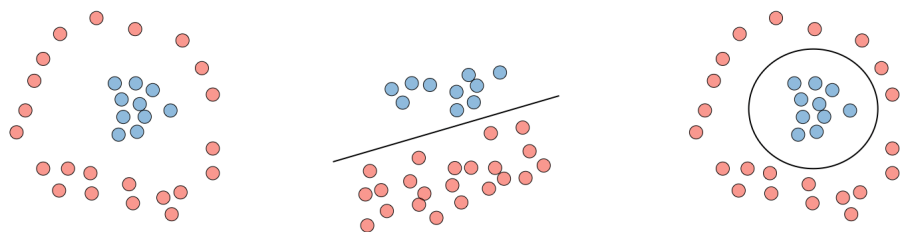
□ **合页损失** – 合页损失用于SVM，定义如下：

$$L(z, y) = [1 - yz]_+ = \max(0, 1 - yz)$$

□ **核** – 给定特征映射 ϕ ，我们定义核 K 为：

$$K(x, z) = \phi(x)^T \phi(z)$$

在实践中，由 $K(x, z) = \exp\left(-\frac{\|x - z\|^2}{2\sigma^2}\right)$ 定义的核 K 被称为高斯核，并且经常使用这种核。



非线性可分性

核映射的使用 ϕ

原始空间中的决策边界

备注：我们说我们使用“核技巧”来计算使用核的成本函数，因为我们实际上不需要知道显式映射 ϕ ，通常，这非常复杂。相反，只需要 $K(x, z)$ 的值。

□ **拉格朗日** – 我们将拉格朗日 $\mathcal{L}(w, b)$ 定义如下：

$$\mathcal{L}(w, b) = f(w) + \sum_{i=1}^l \beta_i h_i(w)$$

备注：系数 β_i 称为拉格朗日乘子。

生成学习

生成模型首先尝试通过估计 $P(x|y)$ 来模仿如何生成数据，然后我们可以使用贝叶斯法则来估计 $P(y|x)$

高斯判别分析

□ **设置** – 高斯判别分析假设 y 和 $x|y = 0$ 且 $x|y = 1$ 如下：

$$y \sim \text{Bernoulli}(\phi)$$

$$x|y = 0 \sim \mathcal{N}(\mu_0, \Sigma) \quad \text{and} \quad x|y = 1 \sim \mathcal{N}(\mu_1, \Sigma)$$

□ **估计** – 下表总结了我们在最大化似然时的估计值：

$\hat{\phi}$	$\hat{\mu}_j \quad (j = 0, 1)$	$\hat{\Sigma}$
$\frac{1}{m} \sum_{i=1}^m 1_{\{y^{(i)}=1\}}$	$\frac{\sum_{i=1}^m 1_{\{y^{(i)}=j\}} x^{(i)}}{\sum_{i=1}^m 1_{\{y^{(i)}=j\}}}$	$\frac{1}{m} \sum_{i=1}^m (x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T$

朴素贝叶斯

□ **假设** – 朴素贝叶斯模型假设每个数据点的特征都是独立的：

$$P(x|y) = P(x_1, x_2, \dots | y) = P(x_1|y)P(x_2|y)\dots = \prod_{i=1}^n P(x_i|y)$$

□ **解决方案** – 最大化对数似然给出以下解， $k \in \{0, 1\}, l \in [1, L]$

$$P(y = k) = \frac{1}{m} \times \#\{j|y^{(j)} = k\}$$

$$\text{and} \quad P(x_i = l|y = k) = \frac{\#\{j|y^{(j)} = k \text{ and } x_i^{(j)} = l\}}{\#\{j|y^{(j)} = k\}}$$

备注：朴素贝叶斯广泛用于文本分类和垃圾邮件检测。

基于树的方法和集成方法

这些方法可用于回归和分类问题。

□ **CART** – 分类和回归树（CART），通常称为决策树，可以表示为二叉树。它们具有可解释性的优点。

□ **随机森林** – 这是一种基于树模型的技术，它使用大量的由随机选择的特征集构建的决策树。与简单的决策树相反，它是高度无法解释的，但其普遍良好的表现使其成为一种流行的算法。

备注：随机森林是一种集成方法。

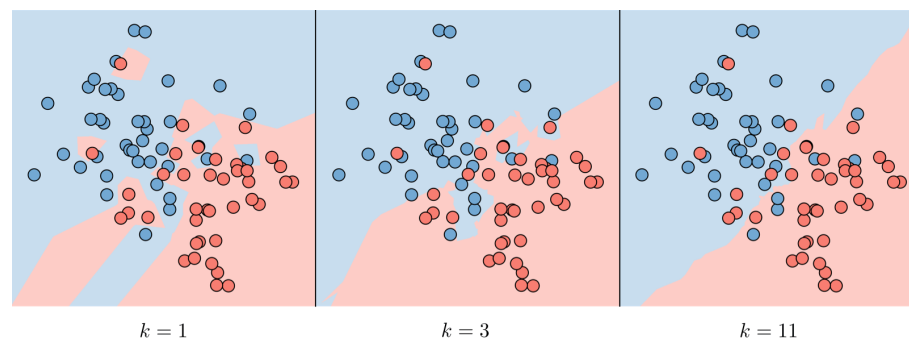
□ **提升** – 提升方法的思想是将一些弱学习器结合起来形成一个更强大的学习器。主要内容总结在下表中：

自适应增强	梯度提升
在下一轮提升步骤中，错误的会被置于高权重	弱学习器训练剩余的错误

其他非参数方法

□ **k-最近邻** – k-最近邻算法，通常称为k-NN，是一种非参数方法，其中数据点的判决由来自训练集中与其相邻的k个数据的性质确定。它可以用于分类和回归。

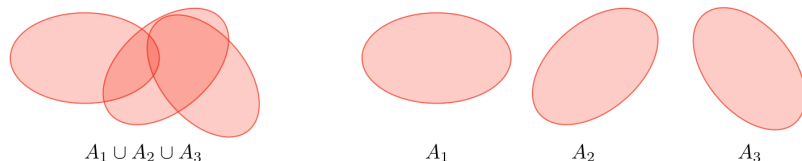
备注：参数 k 越高，偏差越大，参数 k 越低，方差越大。



学习理论

□ **联盟** – 让 A_1, \dots, A_k 成为 k 个事件。我们有：

$$P(A_1 \cup \dots \cup A_k) \leq P(A_1) + \dots + P(A_k)$$



□ **Hoeffding不等式** – 设 Z_1, \dots, Z_m 是从参数 ϕ 的伯努利分布中提取的 m iid 变量。设 ϕ 为其样本均值，固定 $\gamma > 0$ 。我们有：

$$P(|\phi - \hat{\phi}| > \gamma) \leq 2 \exp(-2\gamma^2 m)$$

备注：这种不平等也被称为 *Chernoff* 界限。

□ **训练误差** – 对于给定的分类器 h ，我们定义训练误差 $\hat{\epsilon}(h)$ ，也称为经验风险或经验误差，如下：

$$\hat{\epsilon}(h) = \frac{1}{m} \sum_{i=1}^m 1_{\{h(x^{(i)}) \neq y^{(i)}\}}$$

□ **可能近似正确(PAC)** – PAC是一个框架，在该框架下证明了许多学习理论的结果，并具有以下假设：

- 训练和测试集遵循相同的分布
- 训练样本是相互独立的

□ **打散** – 给定一个集合 $S = \{x^{(1)}, \dots, x^{(d)}\}$ 和一组分类器 \mathcal{H} ，如果对于任意一组标签 $\{y^{(1)}, \dots, y^{(d)}\}$ 都能对分，我们称 \mathcal{H} 打散 S ，我们有：

$$\exists h \in \mathcal{H}, \quad \forall i \in \llbracket 1, d \rrbracket, \quad h(x^{(i)}) = y^{(i)}$$

□ **上限定理** – 设 \mathcal{H} 是有限假设类，使得 $|\mathcal{H}| = k$ 并且使 δ 和样本大小 m 固定。然后，在概率至少为 $1 - \delta$ 的情况下，我们得到：

$$\epsilon(\hat{h}) \leq \left(\min_{h \in \mathcal{H}} \epsilon(h) \right) + 2 \sqrt{\frac{1}{2m} \log \left(\frac{2k}{\delta} \right)}$$

□ **VC维** – 给定无限假设类 \mathcal{H} 的Vapnik-Chervonenkis(VC) 维，注意 $VC(\mathcal{H})$ 是由 \mathcal{H} 打散的最大集合的大小。

备注： $\mathcal{H} = \{\text{2维线性分类器集}\}$ 的 VC 维数为 3。



□ **定理(Vapnik)** – 设 \mathcal{H} , $VC(\mathcal{H}) = d$, m 为训练样本数。概率至少为 $1 - \delta$ ，我们有：

$$\epsilon(\hat{h}) \leq \left(\min_{h \in \mathcal{H}} \epsilon(h) \right) + O \left(\sqrt{\frac{d}{m} \log \left(\frac{m}{d} \right) + \frac{1}{m} \log \left(\frac{1}{\delta} \right)} \right)$$