本章目录

1.数据集划分

01 数据集划分

02 评价指标

03 正则化、偏差和方差

1.数据集划分

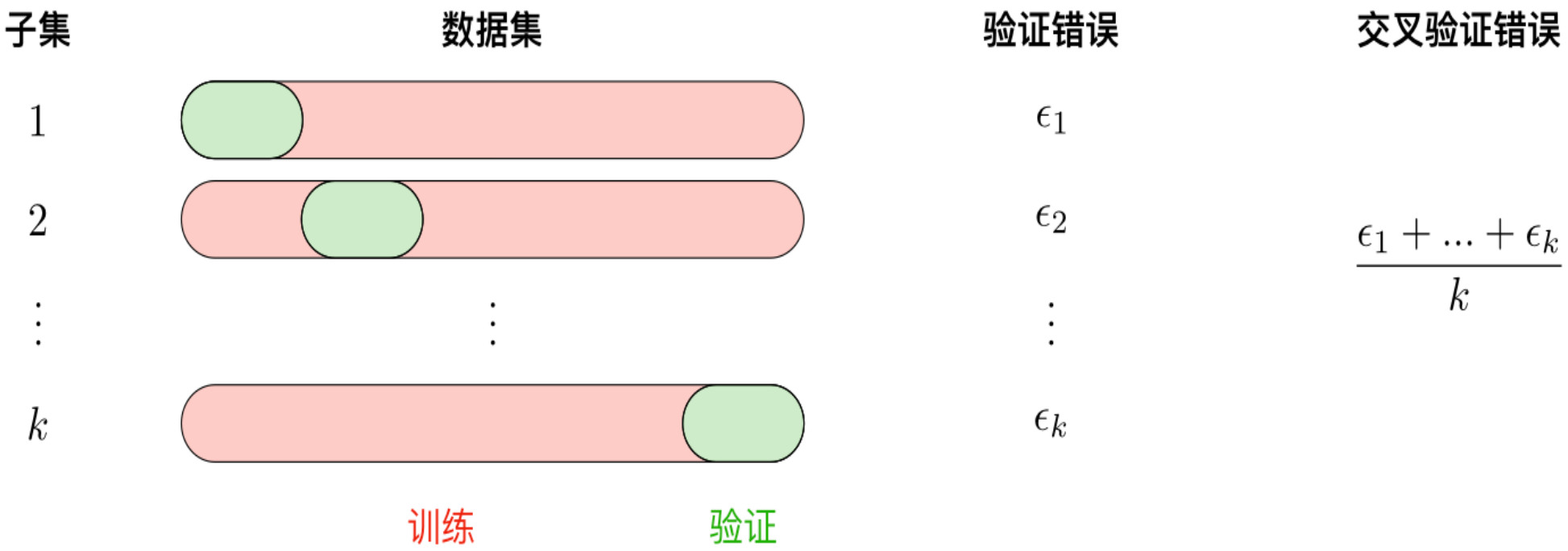
训练集（Training Set）：帮助我们训练模型，简单的说就是通过训练集的数据让我们确定拟合曲线的参数。

验证集（Validation Set）：也叫做开发集（Dev Set），用来做模型选择（model selection），即做模型的最终优化及确定的，用来辅助我们的模型的构建，即训练超参数，可选；

测试集（Test Set）： 为了测试已经训练好的模型的精确度。

三者划分：训练集、验证集、测试集
机器学习：60%，20%，20%；70%，10%，20%
深度学习：98%，1%，1% （假设百万条数据）

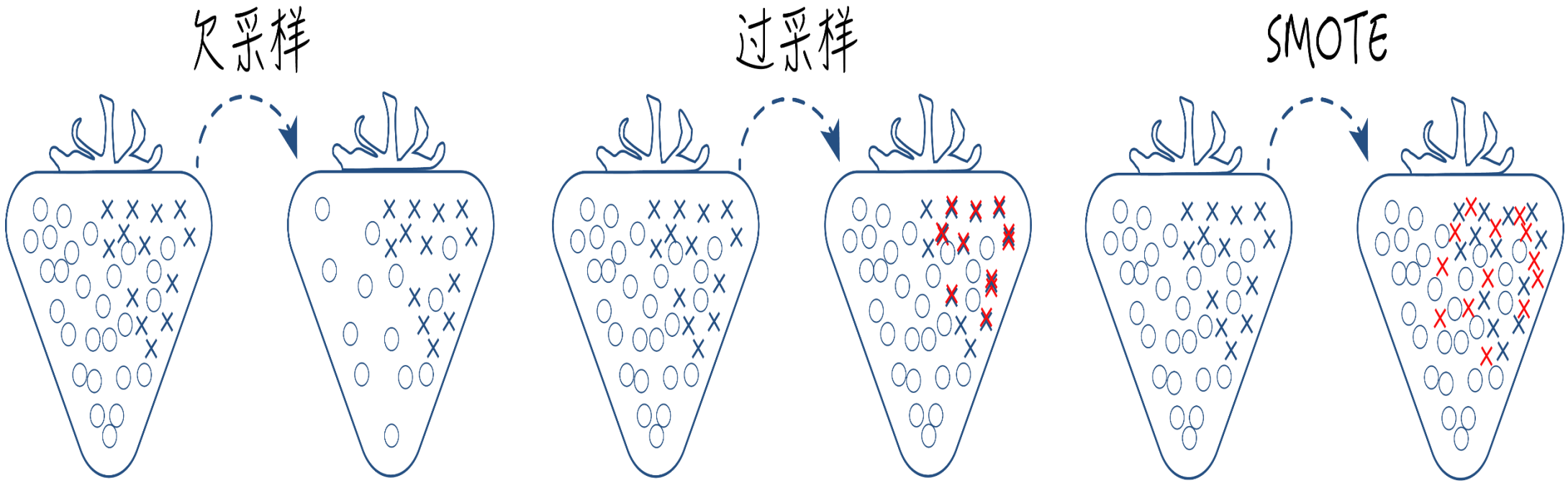
交叉验证



1. 使用训练集训练出*k*个模型
2. 用*k*个模型分别对交叉验证集计算得出交叉验证误差（代价函数的值）
3. 选取代价函数值最小的模型
4. 用步骤3中选出的模型对测试集计算得出推广误差（代价函数的值）

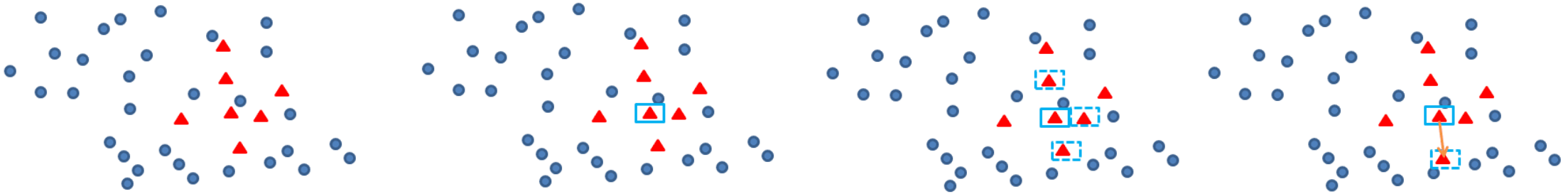
不平衡数据的处理

数据不平衡是指数据集中各类样本数量不均衡的情况.
常用不平衡处理方法有采样和代价敏感学习
采样欠采样、过采样和综合采样的方法



不平衡数据的处理

SMOTE(Synthetic Minority Over-sampling Technique)算法是过采样中比较常用的一种。算法的思想是合成新的少数类样本，而不是简单地复制样本。算法过程如图：



（a）原始样本　（b）选定少类样本　（c）找到靠近k的n个少类样本　（d）增加样本

不平衡数据的处理

代价敏感学习

代价敏感学习是指为不同类别的样本提供不同的权重，从而让机器学习模型进行学习的一种方法

比如风控或者入侵检测，这两类任务都具有严重的数据不平衡问题，可以在算法学习的时候，为少类样本设置更高的学习权重，从而让算法更加专注于少类样本的分类情况，提高对少类样本分类的查全率，但是也会将很多多类样本分类为少类样本，降低少类样本分类的查准率。

2.评价指标

01 数据集划分

02 评价指标

03 正则化、偏差和方差

评价指标

1. **正确肯定（True Positive,TP）：** 预测为真，实际为真
2. **正确否定（True Negative,TN）：** 预测为假，实际为假
3. **错误肯定（False Positive,FP）：** 预测为真，实际为假
4. **错误否定（False Negative,FN）：** 预测为假，实际为真

$$\text{(准确率) Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{(精确率) Precision} = \frac{TP}{TP + FP}$$

$$\text{(召回率) Recall} = \frac{TP}{TP + FN}$$

$$\text{(F1 score) F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

混淆矩阵（confusion_matrix）

		预测值	
		Positive	Negtive
实际值	Positive	TP	FN
	Negtive	FP	TN

评价指标

有100张照片，其中，猫的照片有60张，狗的照片是40张。

输入这100张照片进行二分类识别，找出这100张照片中的所有的猫。

正例（Positives）：识别对的

负例（Negatives）：识别错的

识别结果的混淆矩阵

		预测值	
		Positive	Negtive
实际值	Positive	TP=40	FN=20
	Negtive	FP=10	TN=30

评价指标

正确率（Accuracy）=(TP+ TN)/S
TP+ TN =70，S= 100，则正确率为：

Accuracy =70/100=0.7

精度（Precision）=TP/(TP+ FP)
TP=40，TP+ FP=50。

Precision =40/50=0.8

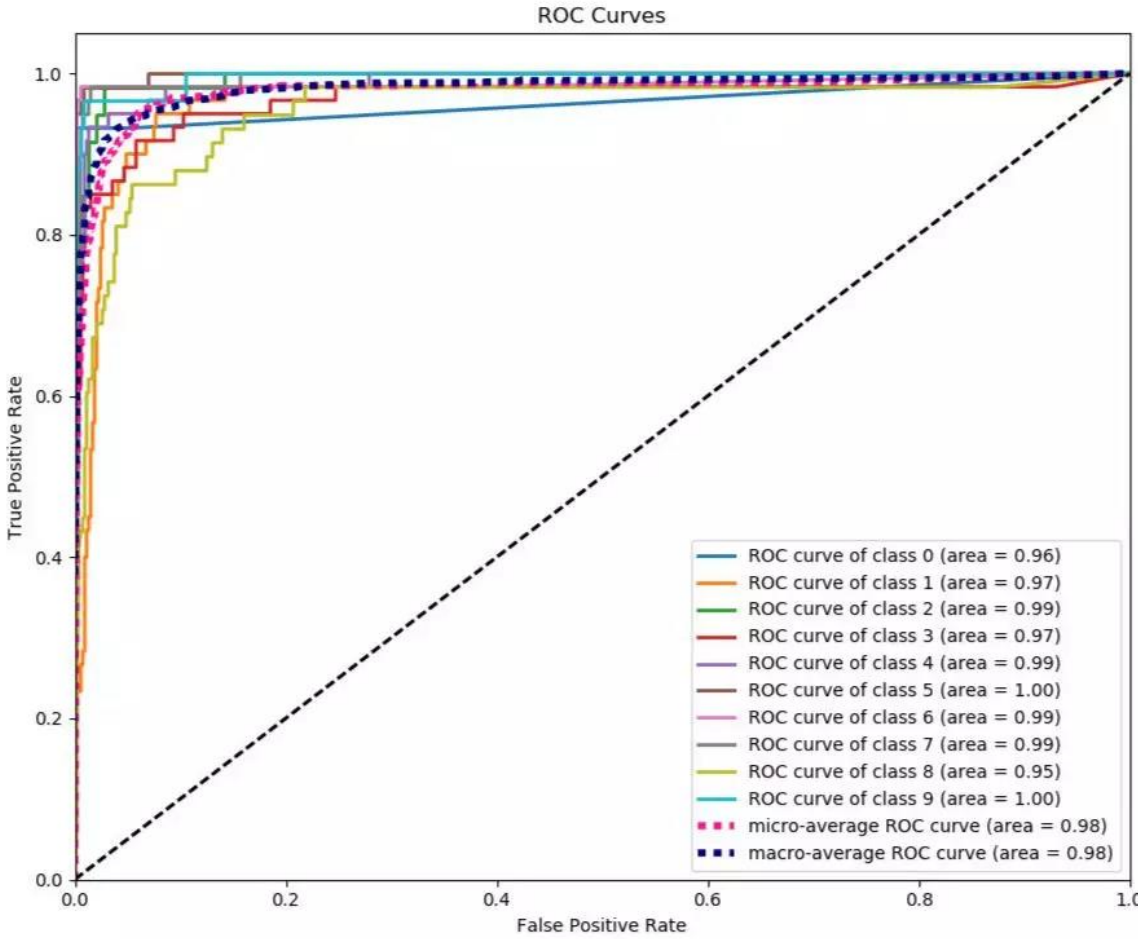
召回率（Recall）=TP/(TP+ FN)
TP=40，TP+FN =60。则召回率为：

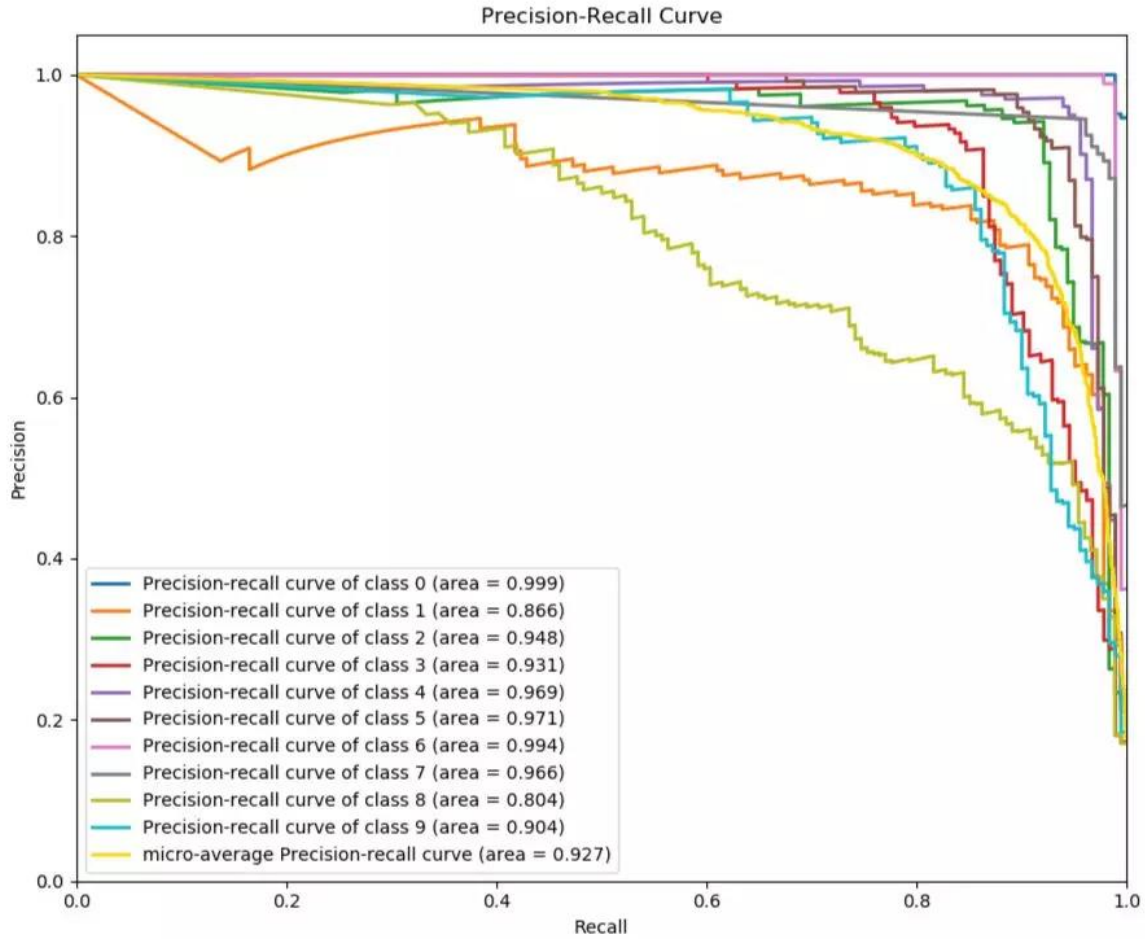
Recall =40/60=0.67

项目	符号	猫狗的例子
识别出的正例	TP+FP	40+10=50
识别出的负例	TN+FN	30+20=50
总识别样本数	TP+FP+TN+FN	50+50=100
识别对了的正例与负例	真正例+真负例=TP+TN	40+30=70
识别错了的正例与负例	伪正例+伪负例=FP+FN	10+20=30
实际总正例数量	真正例+伪负例=TP+FN	40+20=60
实际总负例数量	真负例+伪正例=TN+FP	30+10=40

评价指标

ROC和PR曲线





3.正则化、偏差和方差

01 数据集划分

02 评价指标

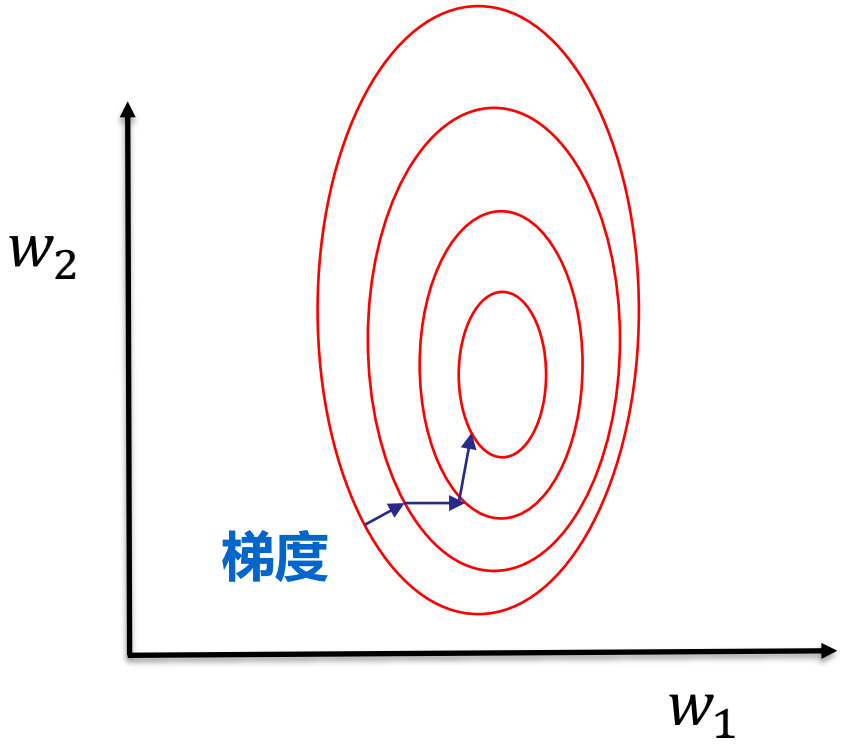
03 正则化、偏差和方差

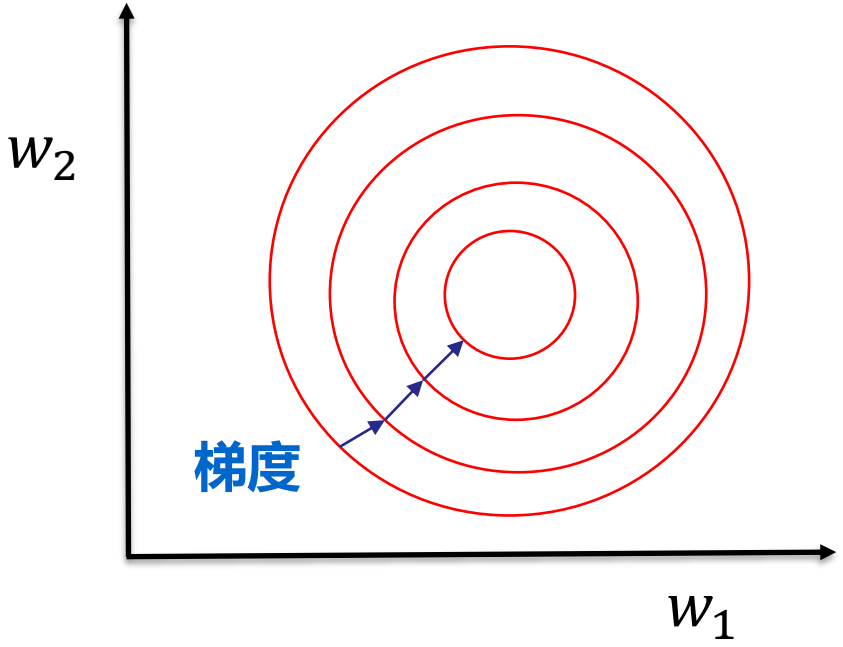
3.正则化、偏差和方差

为什么要标准化/归一化?

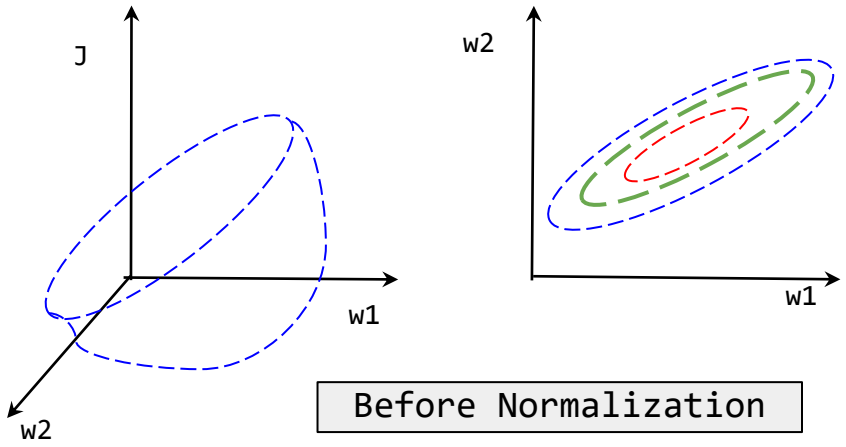
提升模型精度：不同维度之间的特征在数值上有一定比较性，可以大大提高分类器的准确性。

加速模型收敛：最优解的寻优过程明显会变得平缓，更容易正确的收敛到最优解。

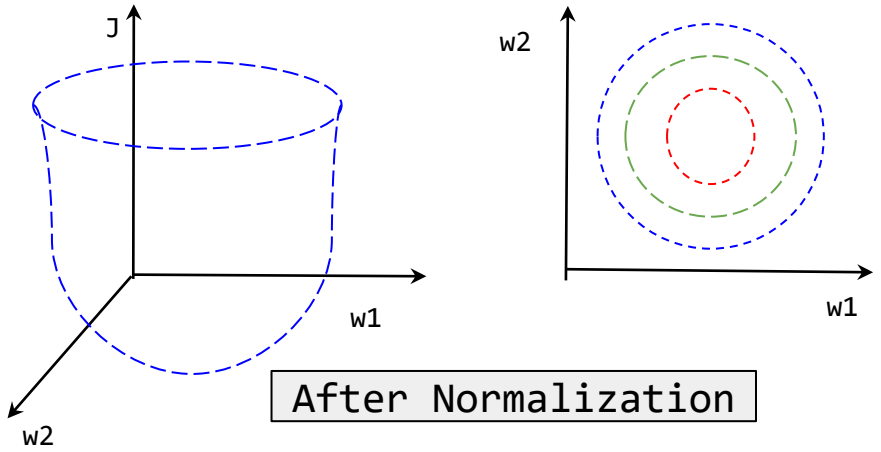




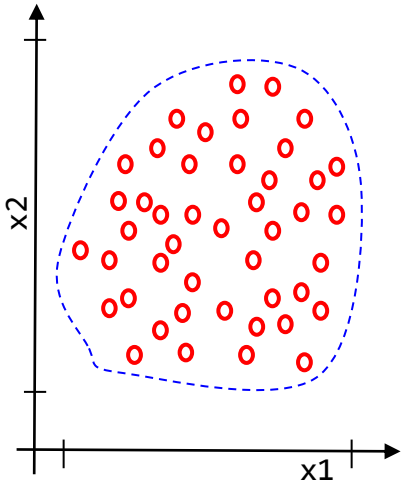
3.正则化、偏差和方差

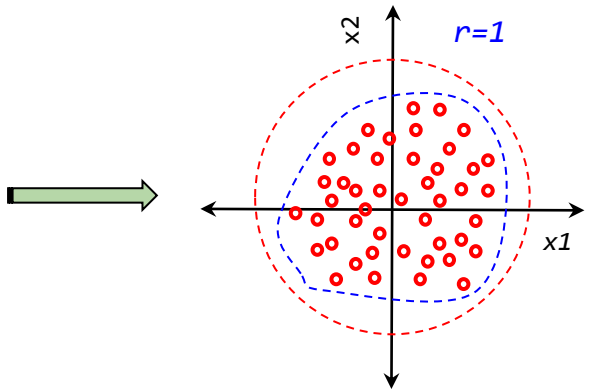


Before Normalization



After Normalization





Normalization

3.正则化、偏差和方差

归一化（最大 - 最小规范化）

$$x^* = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

将数据映射到[0,1]区间

数据归一化的目的是使得各特征对目标变量的影响一致，会将特征数据进行伸缩变化，所以数据归一化是会改变特征数据分布的。

Z-Score标准化

$$x^* = \frac{x - \mu}{\sigma}$$

$$\sigma^2 = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \mu)^2$$

$$\mu = \frac{1}{m}\sum_{i=1}^{m}x^{(i)}$$

处理后的数据均值为0，方差为1

数据标准化为了不同特征之间具备可比性，经过标准化变换之后的特征数据分布没有发生改变。

就是当数据特征取值范围或单位差异较大时，最好是做一下标准化处理。

3.正则化、偏差和方差

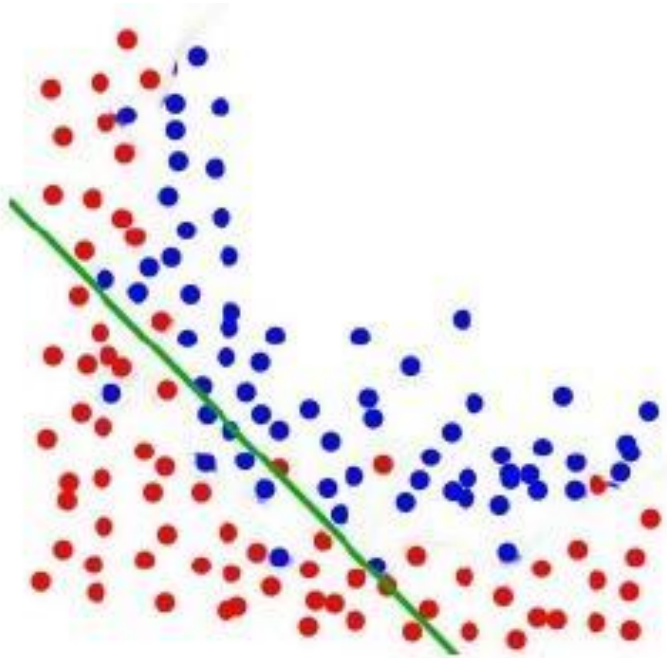
需要做数据归一化/标准化

线性模型，如基于距离度量的模型包括KNN(K近邻)、K-means聚类、感知机和SVM、神经网络。另外，线性回归类的几个模型一般情况下也是需要做数据归一化/标准化处理的。

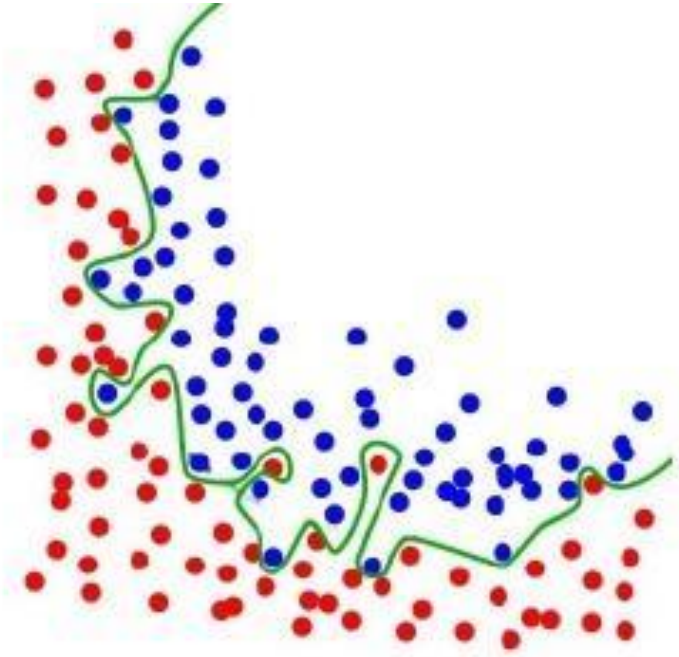
不需要做数据归一化/标准化

决策树、基于决策树的Boosting和Bagging等集成学习模型对于特征取值大小并不敏感，如随机森林、XGBoost、LightGBM等树模型，以及朴素贝叶斯，以上这些模型一般不需要做数据归一化/标准化处理。

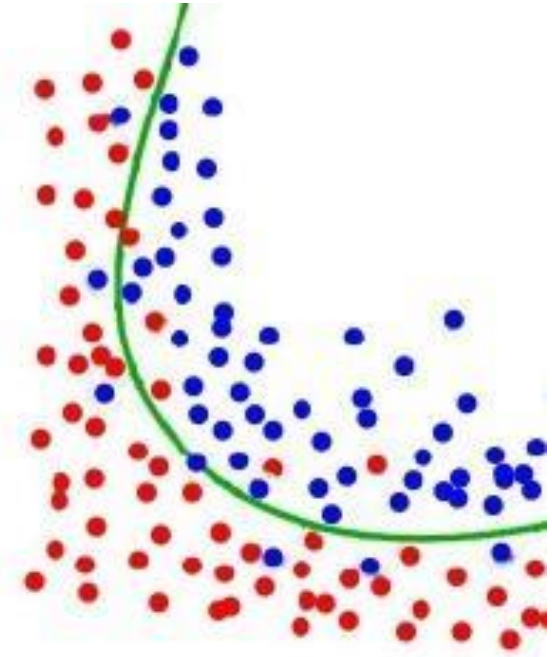
过拟合和欠拟合



欠拟合



过拟合



正合适

过拟合的处理

1.获得更多的训练数据

使用更多的训练数据是解决过拟合问题最有效的手段，因为更多的样本能够让模型学习到更多更有效的特征，减小噪声的影响。

2.**降维**

即丢弃一些不能帮助我们正确预测的特征。可以是手工选择保留哪些特征，或者使用一些模型选择的算法来帮忙（例如PCA）。

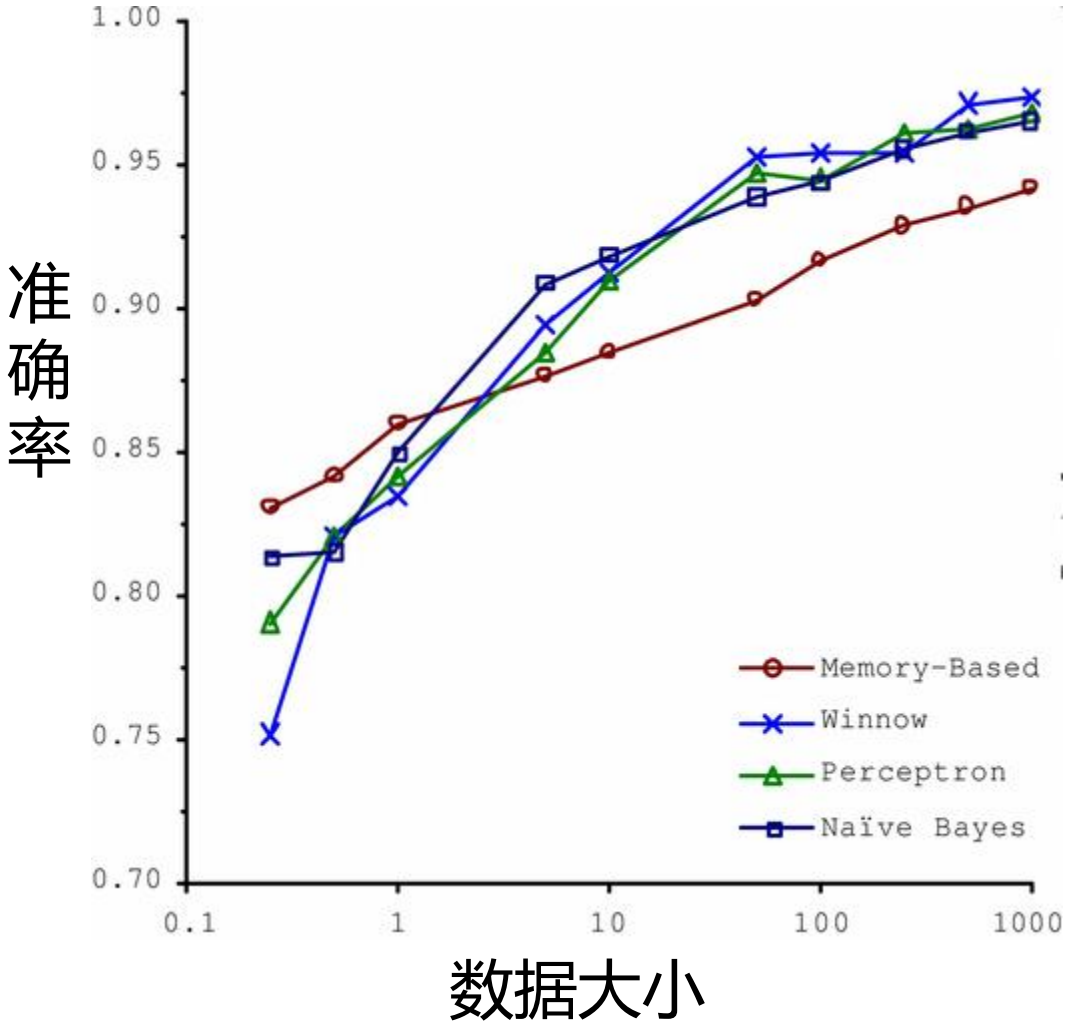
3.**正则化**

正则化(regularization)的技术，保留所有的特征，但是减少参数的大小（magnitude），它可以改善或者减少过拟合问题。

4.**集成学习方法**

集成学习是把多个模型集成在一起，来降低单一模型的过拟合风险。

数据决定一切



通过这张图可以看出，各种不同算法在输入的数据量达到一定级数后，都有相近的高准确度。于是诞生了机器学习界的名言：

成功的机器学习应用不是拥有最好的算法，而是拥有最多的数据！

欠拟合的处理

1.**添加新特征**

当特征不足或者现有特征与样本标签的相关性不强时，模型容易出现欠拟合。通过挖掘组合特征等新的特征，往往能够取得更好的效果。

2.**增加模型复杂度**

简单模型的学习能力较差，通过增加模型的复杂度可以使模型拥有更强的拟合能力。例如，在线性模型中添加高次项，在神经网络模型中增加网络层数或神经元个数等。

3.**减小正则化系数**

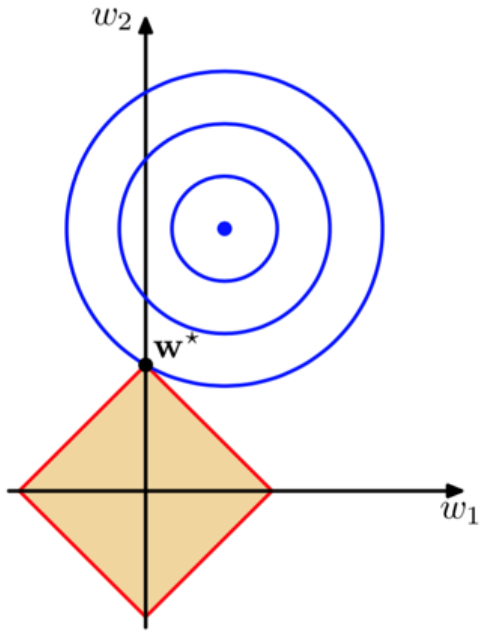
正则化是用来防止过拟合的，但当模型出现欠拟合现象时，则需要有针对性地减小正则化系数。

正则化

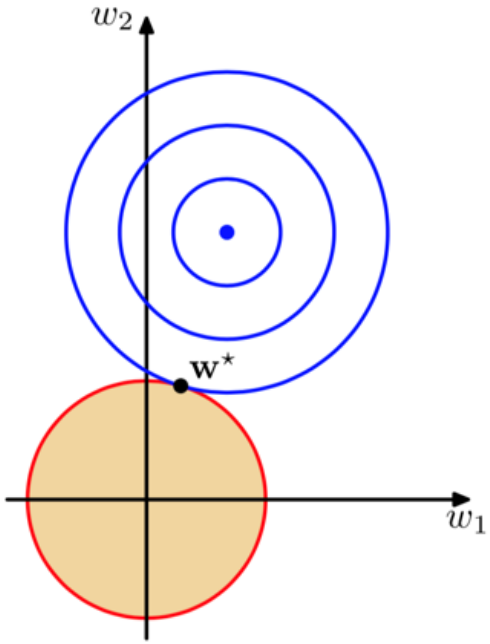
L_1正则化：$J(w) = \frac{1}{m}\sum_{i=1}^{m} L\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m}\sum_{j=1}^{n} |w_j|$，

L_2正则化：$J(w) = \frac{1}{m}\sum_{i=1}^{m} L\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$

正则化



L_1正则化是指在损失函数中加入权值向量w的绝对值之和，L_1的功能是使权重稀疏



在损失函数中加入权值向量w的平方和，L_2的功能是使权重平滑。

L_1正则化可以产生稀疏模型

L_2正则化可以防止过拟合

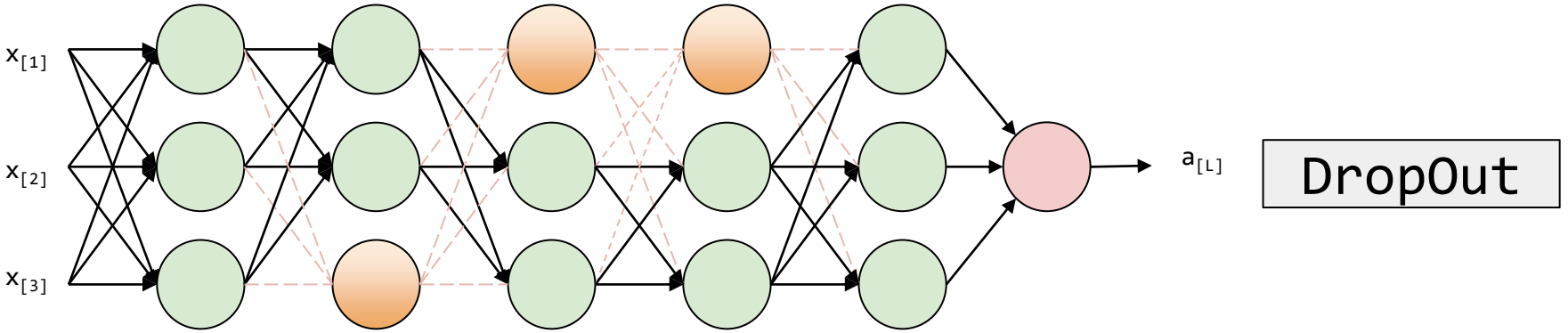
图上面中的蓝色轮廓线是没有正则化损失函数的等高线，中心的蓝色点为最优解，左图、右图分别为L_1、L_2正则化给出的限制。

可以看到在正则化的限制之下，L_2正则化给出的最优解w*是使解更加靠近原点,也就是说L_2正则化能降低参数范数的总和。

L_1正则化给出的最优解w*是使解更加靠近某些轴,而其它的轴则为0,所以L_1正则化能使得到的参数稀疏化。

正则化

Dropout正则化



Dropout的功能类似于$L2$正则化，与$L2$正则化不同的是，被应用的方式不同，**dropout**也会有所不同，甚至更适用于不同的输入范围

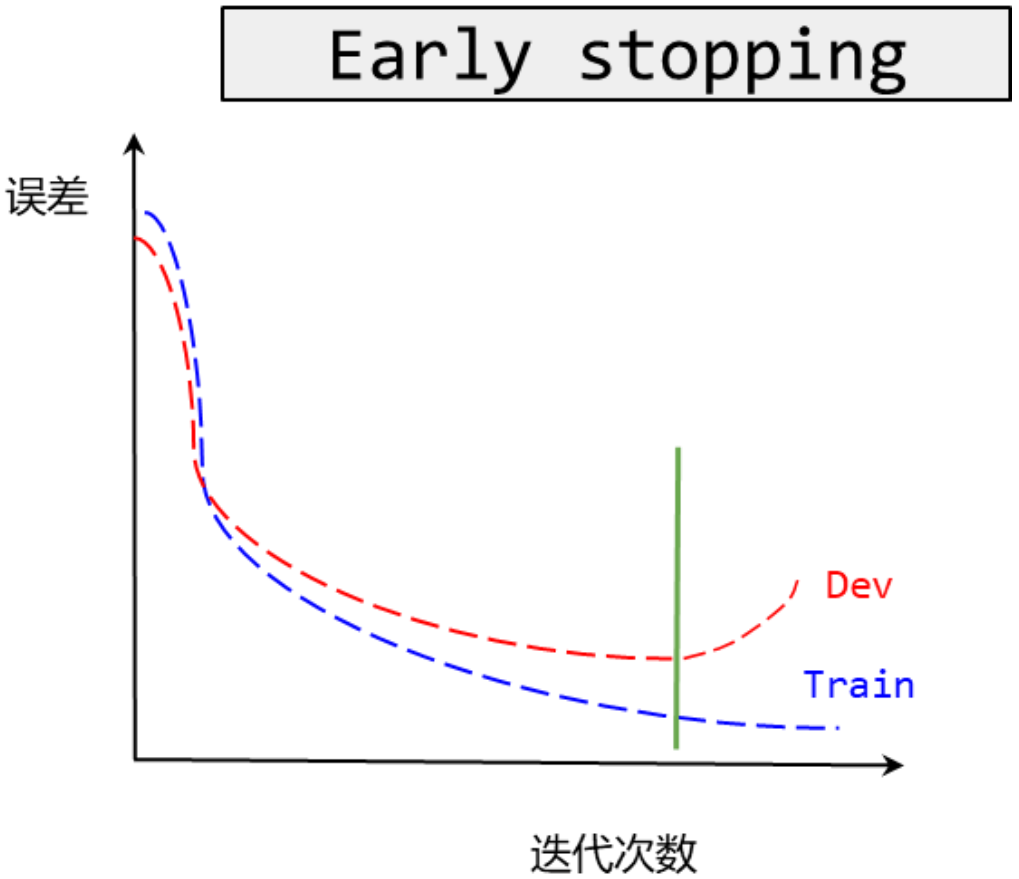
keep-prob=1(没有dropout)　keep-prob=0.5(常用取值，保留一半神经元)

在训练阶段使用，在测试阶段不使用！

正则化

Early stopping代表提早停止训练神经网络

Early stopping的优点是，只运行一次梯度下降，你可以找出w的较小值，中间值和较大值，而无需尝试$L2$正则化超级参数λ的很多值。



正则化

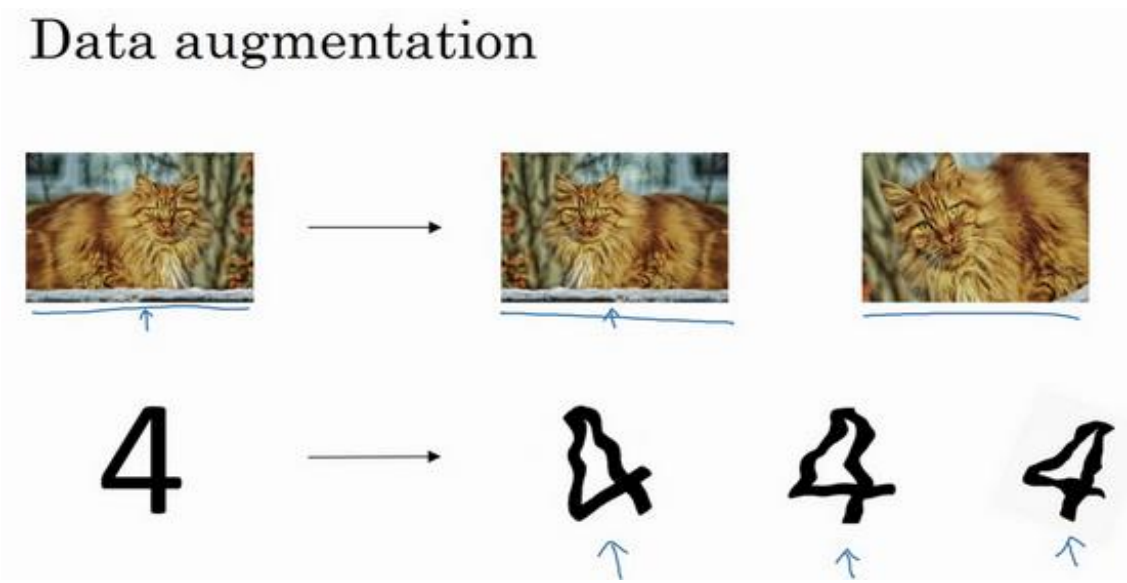
大部分的计算机视觉任务使用很多的数据，所以数据增强是经常使用的一种技巧来提高计算机视觉系统的表现。计算机视觉任务的数据增强通常以下方法实现：

(1) 随意翻转、镜像。

(2) 随意裁剪。

(3) 扭曲变形图片。

(4) 颜色转换，然后给R、G和B三个通道上加上不同的失真值。产生大量的样本，进行数据增强。



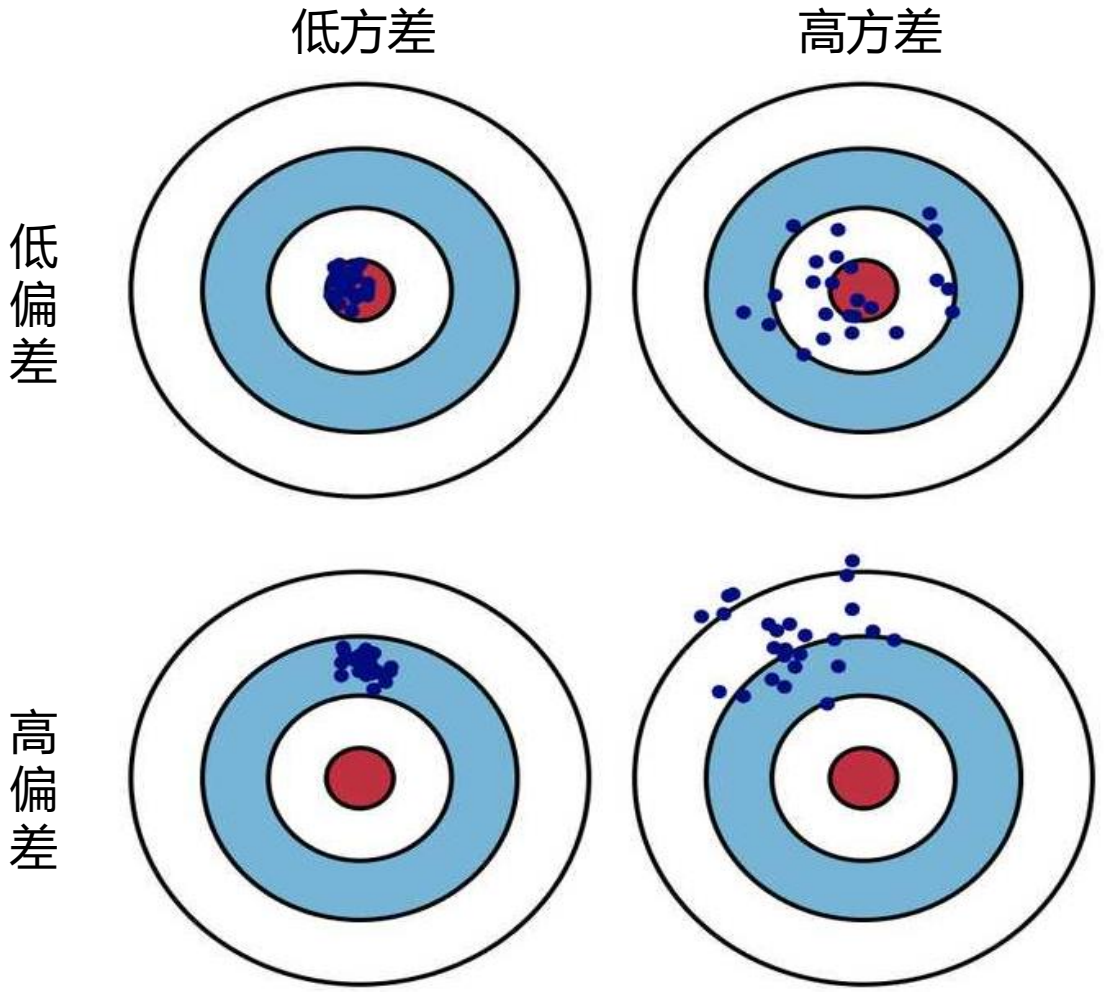
偏差和方差

方差Variance：

描述的是预测值的变化范围，离散程度，也就是离其期望值的距离。方差越大，数据的分布越分散，如右图右列所示。

偏差Bias：

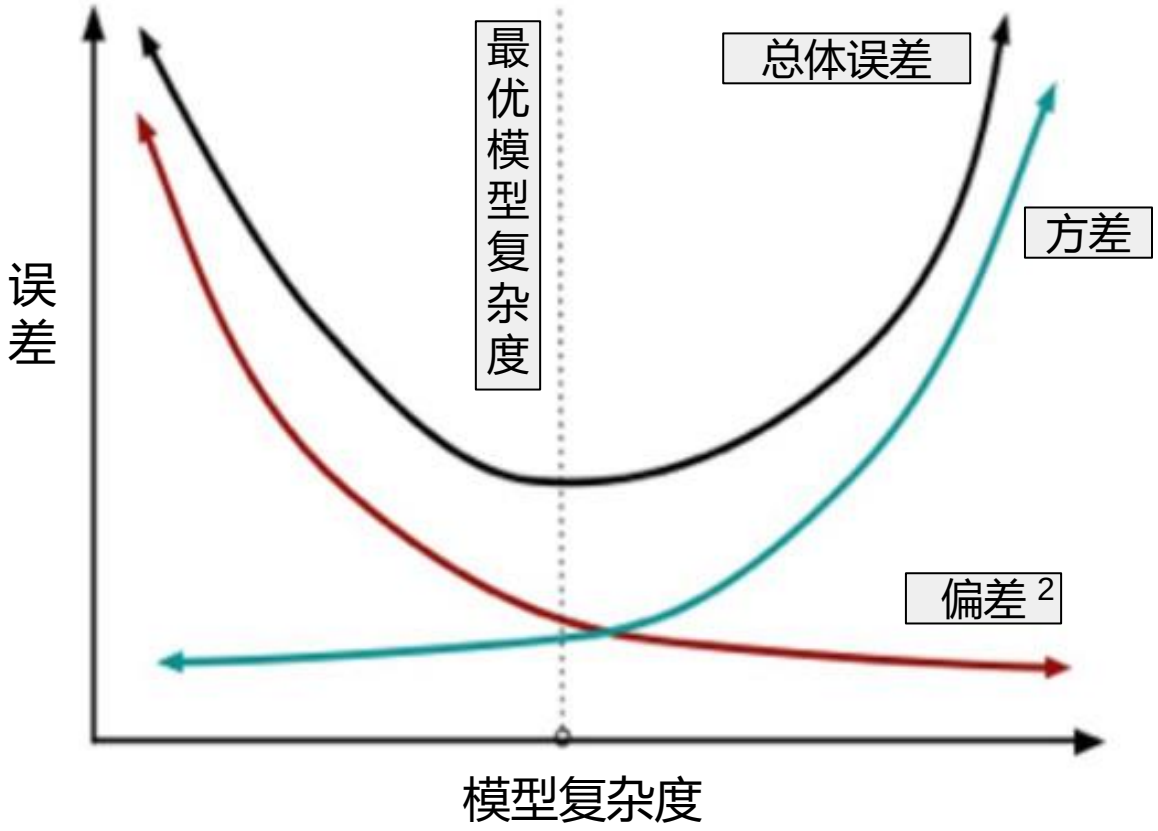
描述的是预测值（估计值）的期望与真实值之间的差距。偏差越大，越偏离真实数据，如右图第二行所示。



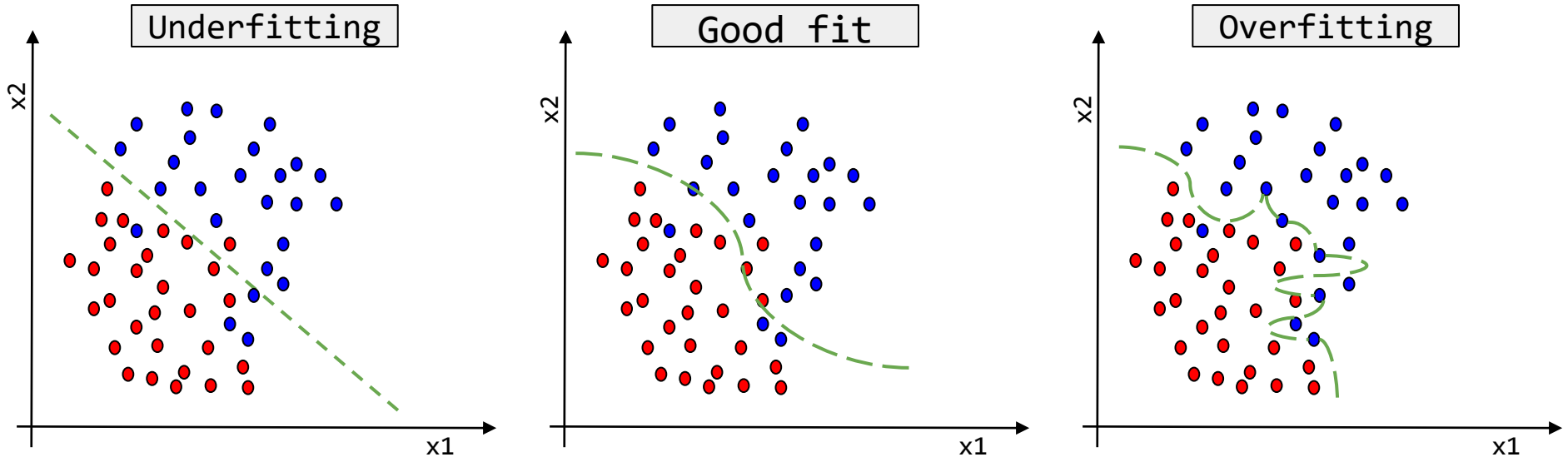
偏差和方差

方差、偏差和模型复杂度

右图是模型复杂度与误差的关系，一般来说，随着模型复杂度的增加，方差会逐渐增大，偏差会逐渐减小，在虚线处，差不多是模型复杂度的最恰当的选择，其“偏差”和“方差”也都适度，才能“适度拟合”。



偏差和方差

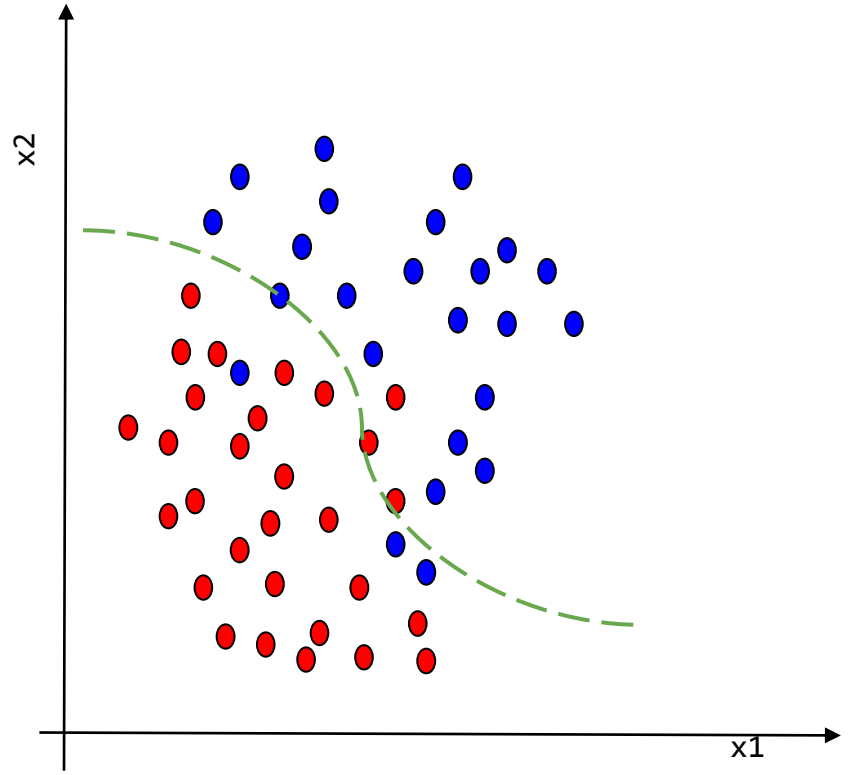


训练集误差和交叉验证集误差近似时：偏差/欠拟合

交叉验证集误差远大于训练集误差时：方差/过拟合

偏差和方差

1. 获得更多的训练实例——解决高方差
2. 尝试减少特征的数量——解决高方差
3. 尝试获得更多的特征——解决高偏差
4. 尝试增加多项式特征——解决高偏差
5. 尝试减少正则化程度λ——解决高偏差
6. 尝试增加正则化程度λ——解决高方差



参考文献

[1] Andrew Ng. Machine Learning[EB/OL]. StanfordUniversity,2014. https://www.coursera.org/course/ml

[2] Peter Harrington.机器学习实战[M]. 北京:人民邮电出版社,2013.

[3] TOM M MICHELLE. Machine Learning[M]. New York: McGraw-Hill Companies,Inc,1997.

[4] Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning[M]. New York: Springer,2001.

[5] CHRISTOPHER M. BISHOP. Pattern Recognition and Machine Learning[M]. New York: Springer,2006.

[6] Kohavi R.,Scaling up the accuracy of naïve Bayes classifiers: A decision-tree hybrid[C]// Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD), Portland, OR, 202-207, 1996.

[7] 李航. 统计学习方法[M]. 北京: 清华大学出版社,2019.

[8] CHAWLA N V, BOWYER K W, HALL L O, et al. SMOTE: Synthetic Minority Over-sampling Technique[J]. Journal of Artificial Intelligence Research, 2002, 16: 321–357.


谢 谢!