**Machine Learning**

**1. 如何理解判别式模型和生成式模型？**

1. 判别方法：由数据直接学习决策函数，或者由条件分布概率作为预测模型，即判别模型。常见的判别模型有：K近邻、SVM、决策树、感知机、线性判别分析（LDA）、线性回归、传统的神经网络、逻辑斯蒂回归、boosting、条件随机场。
2. 生成方法：由数据学习联合概率密度分布函数,然后求出条件概率分布作为预测的模型，即生成模型。 由生成模型可以得到判别模型，但由判别模型得不到生成模型。常见的生成模型有：朴素贝叶斯、隐马尔可夫模型、高斯混合模型、文档主题生成模型（LDA）、限制玻尔兹曼机.

**2. 有监督学习和无监督学习分别是什么？**

1. 监督学习的训练集要求包括输入输出，也可以说是特征和目标。训练集中的目标是由人标注的。
2. 输入数据没有被标记，也没有确定的结果，样本数据类别未知，需要根据样本间的相似性对样本集进行分类。

**3. 归一化是什么？有什么作用？常见的归一化方法有哪些？哪些机器学习模型需要进行归一化？**

归一化：对不同特征维度的伸缩变换的目的是使各个特征维度对目标函数的影响权重是一致的，即使得那些扁平分布的数据伸缩变换成类圆形。这也就改变了原始数据的一个分布。

好处：

1. 归一化：对不同特征维度的伸缩变换的目的是使各个特征维度对目标函数的影响权重是一致的，即使得那些扁平分布的数据伸缩变换成类圆形。这也就改变了原始数据的一个分布。
2. 作用：
3. 提高迭代求解的收敛速度；
4. 提高迭代求解的精度；
5. 深度学习中数据归一化可以防止模型梯度爆炸。
6. 常见的归一化方法：
7. min-max标准化：
8. z-score标准化：
9. 概率模型不需要归一化，因为它们不关心变量的值，而是关心变量的分布和变量之间的条件概率，如决策树、随机森林。而像AdaBoost、SVM、LR、KNN、K-Means之类的最优化问题就需要归一化。

**4. 线性分类器与非线性分类器的区别以及优劣是什么？**

1. 区别：如果模型是参数的线性函数，并且存在线性分类面，那么就是线性分类器，否则不是。
2. 常见的线性分类器有：LR，贝叶斯分类，单层感知机，线性回归。

常见的非线性分类器：决策树，RF，GBDT，多层感知机。

SVM两种都有（看线性核还是高斯核）。

1. 优劣势：线性分类器速度快、编程方便，但是可能拟合效果不会很好；非线性分类器编程复杂，但是效果拟合能力强。

**5. 数据预处理的方法有哪些？**

1. 缺失值：填充缺失值（离散，None；连续，均值），缺失值太多，则直接去除该列。
2. 连续值：离散化，有的模型（如决策树）需要离散值。
3. 对定量特征二值化：核心在于设定一个阈值，大于阈值的赋值为1，小于等于阈值的赋值为0。
4. 删除共线性变量：皮尔逊相关系数，去除高度相关的列。

**6. 数据不平衡问题是什么？如何解决？**

数据不平衡是指数据集中，每个类别下的样本数目相差很大。解决方法如下：

1. 进行特殊的加权，如在AdaBoost中或者SVM中。
2. 采用对不平衡数据集不敏感的算法。
3. 改变评价标准：用AUC/ROC来进行评价。
4. 采用Bagging/Boosting/ensemble等方法。
5. 采样，对小样本加噪声采样，对大样本进行下采样。
6. 考虑数据的先验分布。

**7. 多重共线性是什么？如何减小多重共线性对模型效果的影响？**

多重共线性是指线性回归模型中的解释变量之间由于存在精确相关关系或高度相关关系而使模型估计失真或难以估计准确。通俗的说，就是变量之间有较强的相关性，影响模型的预测能力。解决多重共线问题可以考虑一下几种方法：

1. 直接删除：如果明确的知道是哪个变量引起的多重共线问题，可以将该变量直接删除。但是要注意删除的变量确定为相对不重要并从偏相关系数检验证实为产生多重共线的原因。
2. 逐步回归法：将变量逐个引入模型，每引入一个解释变量后都要进行F检验，并对已经选入的解释变量逐个进行t检验，当原来引入的解释变量由于后面解释变量的引入变得不再显著时，则将其删除。
3. 改变特征的表现形式：有些变量可以改变其表现形式，如像网页的浏览次数、点击次数等特征属于长尾分布，可以对其进行log变换，变换后的变量可以有效的降低变量之间的相关性。
4. 主成分分析PCA：通过主成分分析提取主要的特征，从而忽略次要的成分，得到相关性很低的特征。

**8. 欠拟合和过拟合的原因分别有哪些？如何避免？**

1. 欠拟合的原因：模型复杂度过低，不能很好的拟合所有的数据，训练误差大。

避免欠拟合：增加模型复杂度，如采用高阶模型（预测）或者引入更多特征（分类）等。

1. 过拟合的原因：模型复杂度过高，训练数据过少，训练误差小，测试误差大。

避免过拟合：降低模型复杂度，如加上正则惩罚项，如L1或L2，增加训练数据等。

**9. 机器学习中防止过拟合的方法有哪些？**

过拟合的原因是算法的学习能力过强，一些假设条件（如样本独立同分布）可能是不成立的；训练样本过少不能对整个空间进行分布估计。

处理方法有：

1. 早停止：如在训练中多次迭代后发现模型性能没有显著提高就停止训练。
2. 数据集扩增：原有数据增加、原有数据加随机噪声、重采样。
3. 正则化：L1或者L2正则化。
4. 特征选择或者特征降维。

**10. 正则化为什么能防止过拟合？**

1. 过拟合表现在训练数据上的误差非常小，而在测试数据上误差反而增大。其原因一般是模型过于复杂，过分得去拟合数据的噪声. 正则化则是对模型参数添加先验，使得模型复杂度较小，对于噪声的输入扰动相对较小。
2. 过拟合，就是拟合函数需要顾忌每一个点，最终形成的拟合函数波动很大。在某些很小的区间里，函数值的变化很剧烈。这就意味着函数在某些小区间里的导数值（绝对值）非常大，由于自变量值可大可小，所以只有系数足够大，才能保证导数值很大。而正则化是通过约束参数的范数使其不要太大，所以可以在一定程度上减少过拟合情况。

**11. L1和L2的异同点在哪里？**

L1范数：指向量中各个元素绝对值之和，也有个美称叫“稀疏规则算子” 。

L2范数：为x向量各个元素平方和的1/2次方。

相同点：都用于避免过拟合。

不同点：L1可以让一部分特征的系数缩小到0，从而间接实现特征选择，所以L1适用于特征之间有关联的情况。L2让所有特征的系数都缩小，但是不会减为0，它会使优化求解稳定快速，所以L2适用于特征之间没有关联的情况。L1范数可以使权值稀疏，方便特征提取。L2范数可以防止过拟合，提升模型的泛化能力。

**12. L1和L2正则先验分别服从什么分布**

L1是拉普拉斯分布，L2是高斯分布。

给定训练数据, 贝叶斯方法通过最大化后验概率估计参数：

说明：是参数向量的先验概率。

首先看下最小二乘公式的推导：

其中，，误差。

当前已知，，假设，也就是说，那么用最大似然估计推导：

这个是通过最大似然估计的方法， 推导出线性回归最小二乘计算公式。

1. 假设1：参数向量服从高斯分布，假设，，那么用最大后验估计推导

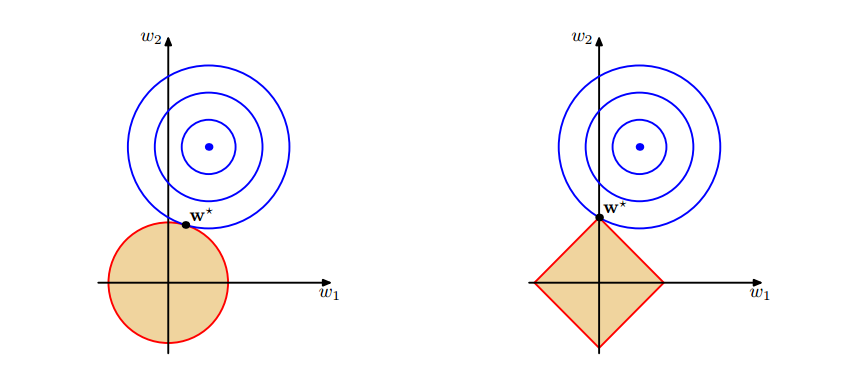
最终的公式就是岭回归计算公式。与上面最大似然估计推导出的最小二乘相比，最大后验估计就是在最大似然估计公式乘以高斯先验， 这里就理解前面L2正则就是加入高斯先验知识。

1. 假设2：参数向量服从拉普拉斯分布，假设，，那么用最大后验估计推导

最终的公式就是Lasso计算公式。与上面最大似然估计推导出的最小二乘相比，最大后验估计就是在最大似然估计公式乘以拉普拉斯先验， 这里就理解前面L1正则就是加入拉普拉斯先验知识。

**13. L1为什么可以保证稀疏？**

1. 几何解释：如下图，图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L2。可以看到在正则化的限制之下，L2正则化给出的最优解是使解更加靠近原点，也就是说L2正则化能降低参数范数的总和。L1正则化给出的最优解是使解更加靠近某些轴，而其它的轴则为0，所以L1正则化能使得到的参数稀疏化。



1. 解析解推导：假设有以下包含L1正则的目标函数

我们的目的是求得使目标函数取最小值的，上式对求导可得：

其中若，则；若，则；若，则。当，假设我们得到最优的目标解是，用秦勤公式在处展开可以得到（要注意的：

其中是关于的Hessian矩阵，为了得到更直观的解，我们简化，假设是对角矩阵，则有：

将上式代回去，简化后的目标函数可以写成这样：

从上式可以看出，各个方向的导数是不相关的，所以可以分别独立求导并使之为0，可得：

直接给出上式的解，再来看推导过程：

可以得到两点：

1. 可以看到式中的二次函数是关于对称的，那么必有，因为在二次函数值不变的程序下，这样可以使得更小。

2. 或，因为在不变的情况下， 或可以使更小。

由上述的第2点可以得到：

我们再来看一下第2点或，若，那么有，所以这时有。从这个解可以得到两个可能的结果：  
1. 若，正则化后目标中的的最优解是。因为这个方向上的影响被正则化的抵消了。

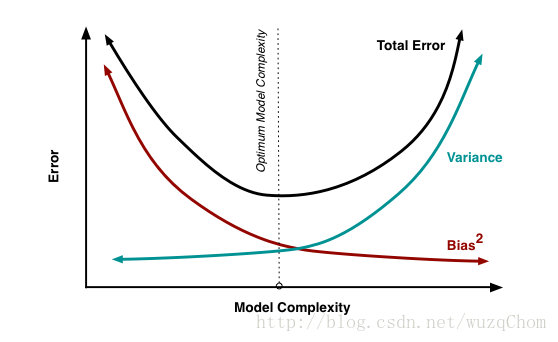
2.若，正则化不会推最优解推向0，而是在这个方面上向原点移动了的距离。

**14. 如何理解机器学习中的偏差和方差？**

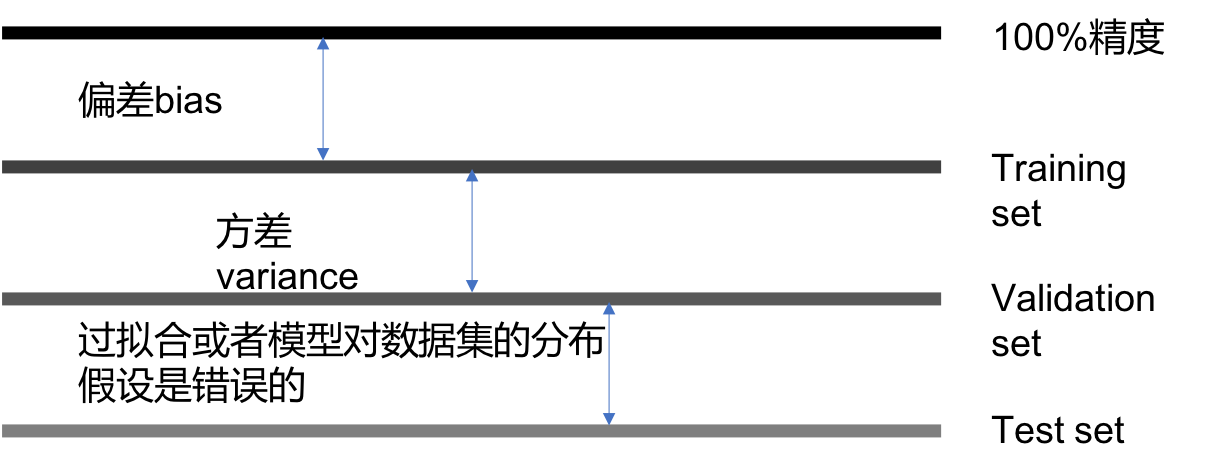
Bias-Variance Decomposition是以测试集上的平方错误为基础的，假设我们的预测值为,真实值为,,则均方误差为，在这里不考虑样本的噪声。噪声的存在是学习算法所无法解决的问题，数据的质量决定了学习的上限。并且使用来代表预测值（这里我们把它看成是随机变量，即不同数据学习得到的模型），代表真实值，代表算法的期望预测（如：可以用不同的数据集来得到，则有：

由上面的公式可知，偏差描述的是算法的预测的平均值和真实值的关系（可以想象成算法的拟合能力如何），而方差描述的是同一个算法在不同数据集上的预测值和所有数据集上的平均预测值之间的关系（可以想象成算法的稳定性如何）。

从下图可以看出，偏差和方差两者是有冲突的，称之为变差方差窘境。



假如学习算法训练不足时，此时学习器的拟合能力不够强，此时数据的扰动不会对结果产生很大的影响（可以想象成由于训练的程度不够，此时学习器指学习到了一些所有的数据都有的一些特征），这个时候偏差主导了算法的泛化能力。随着训练的进行，学习器的拟合能力逐渐增强，变差逐渐减小，但此时不同通过数据学习得到的学习器就可能会有较大的偏差，即此时的方差会主导模型的泛化能力。若学习进一步进行，学习器就可能学到数据集所独有的特征，而这些特征对于其它的数据是不适用的，这个时候就是发生了过拟合的想象。

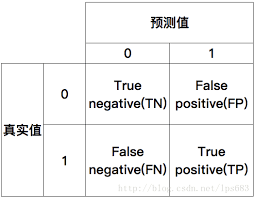


**15. 有哪些模型评价指标？请解释下AUC，准确率和召回率。**

回归问题评价指标：

1. 均方误差MSE
2. 均方跟误差RMSE
3. 平方绝对误差MAE
4. R方R Squared

分类问题评价指标：



1. 准确率：样本中预测正确的比例
2. 召回率：样本中的正例有多少被预测正确
3. 精确率：预测为正的样本中有多少是真正的正样本
4. F1指数
5. ROC曲线：ROC曲线反应了接收者操作特征，曲线上每个点反映着对同一信号刺激的感受性。横轴为负正类率FPR，表示为划分实例中所有负例占所有负例的比例，其数值越大，预测正类中实际负类越多。纵轴为真正类率灵敏度TPR，其数值越大，预测正类中实际正类越多。AUC是ROC曲线下的面积，介于0.1和1之间。AUC作为数值可以直观评价分类器的好坏，值越大越好。