**问1：给你一个有1000列和1百万行的训练数据集。这个数据集是基于分类问题的。经理要求你来降低该数据集的维度以减少模型计算时间。你的机器内存有限。你会怎么做？（你可以自由做各种实际操作假设。）**

答：你的面试官应该非常了解很难在有限的内存上处理高维的数据。以下是你可以使用的处理方法：

1.由于我们的RAM很小，首先要关闭机器上正在运行的其他程序，包括网页浏览器，以确保大部分内存可以使用。

2.我们可以随机采样数据集。这意味着，我们可以创建一个较小的数据集，比如有1000个变量和30万行，然后做计算。

3.为了降低维度，我们可以把数值变量和分类变量分开，同时删掉相关联的变量。对于数值变量，我们将使用相关性分析。对于分类变量，我们可以用卡方检验。

4.另外，我们还可以使用PCA（主成分分析），并挑选可以解释在数据集中有最大偏差的成分。

5.利用在线学习算法，如VowpalWabbit（在Python中可用）是一个可能的选择。

6.利用Stochastic GradientDescent（随机梯度下降）法建立线性模型也很有帮助。

7.我们也可以用我们对业务的理解来估计各预测变量对响应变量的影响大小。但是，这是一个主观的方法，如果没有找出有用的预测变量可能会导致信息的显著丢失。

注意：对于第4和第5点，请务必阅读有关在线学习算法和随机梯度下降法的内容。这些是高阶方法。

**问2：在PCA中有必要做旋转变换吗？如果有必要，为什么？如果你没有旋转变换那些成分，会发生什么情况？**

答：是的，旋转（正交）是必要的，因为它把由主成分捕获的方差之间的差异最大化。这使得主成分更容易解释。

但是不要忘记我们做PCA的目的是选择更少的主成分（与特征变量个数相较而言），那些选上的主成分能够解释数据集中最大方差。通过做旋转，各主成分的相对位置不发生变化，它只能改变点的实际坐标。

如果我们没有旋转主成分，PCA的效果会减弱，那样我们会不得不选择更多个主成分来解释数据集里的方差。

注意：对PCA（主成分分析）需要了解更多。

**问3：给你一个数据集。这个数据集有缺失值，且这些缺失值分布在离中值有1个标准偏差的范围内。百分之多少的数据不会受到影响？为什么？**

答：这个问题给了你足够的提示来开始思考！由于数据分布在中位数附近，让我们先**假设这是一个正态分布**。我们知道，在一个正态分布中，约有68％的数据位于跟平均数（或众数、中位数,，在正太分布中，三者相等）1个标准差范围内的，那样剩下的约32%的数据是不受影响的。因此，约有32%**的数据将不受到缺失值的影响。**

**问4：给你一个癌症检测的数据集。你已经建好了分类模型，取得了96％的精度。为什么你还是不满意你的模型性能？你可以做些什么呢？**

答：如果你分析过足够多的数据集，你应该可以判断出来癌症检测结果是不平衡数据。在不平衡数据集中，精度不应该被用来作为衡量模型的标准，因为96％（按给定的）可能只有正确预测多数分类，但我们感兴趣是那些少数分类（4％），是那些被诊断出癌症的人。

因此，为了评价模型的性能，应该用灵敏度（真阳性率），特异性（真阴性率），F值用来确定这个分类器的“聪明”程度。如果在那4%的数据上表现不好，我们可以采取以下步骤：

1.我们可以使用欠采样、过采样或SMOTE让数据平衡。

2.我们可以通过概率验证和利用AUC-ROC曲线找到最佳阀值来调整预测阀值。

3.我们可以给分类分配权重，那样较少的分类获得较大的权重。

4.我们还可以使用异常检测。

注意：要更多地了解不平衡分类。

**问5: 为什么朴素贝叶斯如此“朴素”？**

答：朴素贝叶斯太‘朴素’了，因为它假定所有的特征在数据集中的作用是同样重要和独立的。正如我们所知，这个假设在现实世界中是很不真实的。

**问6：解释朴素贝叶斯算法里面的先验概率、似然估计和边际似然估计？**

答：先验概率就是因变量（二分法）在数据集中的比例。这是在你没有任何进一步的信息的时候，是对分类能做出的最接近的猜测。例如，在一个数据集中，因变量是二进制的（1和0）。例如，1（垃圾邮件）的比例为70％和0（非垃圾邮件）的为30％。

因此，我们可以估算出任何新的电子邮件有70％的概率被归类为垃圾邮件。似然估计是在其他一些变量的给定的情况下，一个观测值被分类为1的概率。边际似然估计就是条件似然估计。例如，“FREE”这个词在以前的垃圾邮件使用的概率就是似然估计。边际似然估计就是，“FREE”这个词在任何消息中使用的概率。

**问7：你正在一个时间序列数据集上工作。经理要求你建立一个高精度的模型。你开始用决策树算法，因为你知道它在所有类型数据上的表现都不错。后来，你尝试了时间序列回归模型，并得到了比决策树模型更高的精度。这种情况会发生吗？为什么？**

答：众所周知，时间序列数据有线性关系。另一方面，决策树算法是已知的检测非线性交互最好的算法。为什么决策树没能提供好的预测的原因是它不能像回归模型一样做到对线性关系的那么好的映射。因此，我们知道了如果我们有一个满足线性假设的数据集，一个线性回归模型能提供强大的预测。

**问8：给你分配了一个新的项目，是关于帮助食品配送公司节省更多的钱。问题是，公司的送餐队伍没办法准时送餐。结果就是他们的客户很不高兴。最后为了使客户高兴，他们只好以免餐费了事。哪个机器学习算法能拯救他们？**

答：你的大脑里可能已经开始闪现各种机器学习的算法。但是等等！这样的提问方式只是来测试你的机器学习基础。这不是一个机器学习的问题，而是一个路径优化问题。机器学习问题由三样东西组成：

1.模式已经存在。

2.不能用数学方法解决（指数方程都不行）。

3.有相关的数据。

通过判断以上三个因素来决定机器学习是不是个用来解决特定问题的工具。

**问9：你意识到你的模型受到低偏差和高方差问题的困扰。应该使用哪种算法来解决问题呢？为什么？**

答：低偏差意味着模型的预测值接近实际值。换句话说，该模型有足够的灵活性，以模仿训练数据的分布。貌似很好，但是别忘了，一个灵活的模型没有泛化能力。这意味着，当这个模型用在对一个未曾见过的数据集进行测试的时候，它会令人很失望。

在这种情况下，我们可以使用bagging算法（如随机森林），以解决高方差问题。bagging算法把数据集分成重复随机取样形成的子集。然后，这些样本利用单个学习算法生成一组模型。接着，利用投票（分类）或平均（回归）把模型预测结合在一起。另外，为了应对大方差，我们可以：

1.使用正则化技术，惩罚更高的模型系数，从而降低了模型的复杂性。

2.使用可变重要性图表中的前n个特征。可以用于当一个算法在数据集中的所有变量里很难寻找到有意义信号的时候。

**问10：给你一个数据集。该数据集包含很多变量，你知道其中一些是高度相关的。经理要求你用PCA。你会先去掉相关的变量吗？为什么？**

答：你可能会说不，但是这有可能是不对的。丢弃相关变量会对PCA有实质性的影响，因为有相关变量的存在，由特定成分解释的方差被放大。例如：在一个数据集有3个变量，其中有2个是相关的。如果在该数据集上用PCA，第一主成分的方差会是与其不相关变量的差异的两倍。此外，加入相关的变量使PCA错误地提高那些变量的重要性，这是有误导性的。

**问11：花了几个小时后，现在你急于建一个高精度的模型。结果，你建了5个GBM （Gradient Boosted Models），想着boosting算法会显示魔力。不幸的是，没有一个模型比基准模型表现得更好。最后，你决定将这些模型结合到一起。尽管众所周知，结合模型通常精度高，但你就很不幸运。你到底错在哪里？**

答：据我们所知，组合的学习模型是基于合并弱的学习模型来创造一个强大的学习模型的想法。但是，只有当各模型之间没有相关性的时候组合起来后才比较强大。由于我们已经试了5个GBM，但没有提高精度，表明这些模型是相关的。具有相关性的模型的问题是，所有的模型提供相同的信息。例如：如果模型1把User1122归类为1，模型2和模型3很有可能会做有同样分类，即使它的实际值应该是0，因此，只有弱相关的模型结合起来才会表现更好。

**问12：KNN和KMEANS聚类（kmeans clustering）有什么不同？**

答：不要被它们的名字里的“K”误导。你应该知道，这两种算法之间的根本区别是，KMEANS本质上是无监督学习而KNN是监督学习。KMEANS是聚类算法。KNN是分类（或回归）算法。 KMEAN算法把一个数据集分割成簇，使得形成的簇是同构的，每个簇里的点相互靠近。该算法试图维持这些簇之间有足够的可分离性。由于无监督的性质，这些簇没有任何标签。NN算法尝试基于其k（可以是任何数目）个周围邻居来对未标记的观察进行分类。它也被称为懒惰学习法，因为它涉及最小的模型训练。因此，它不用训练数据对未看见的数据集进行泛化。

**问13：真阳性率和召回有什么关系？写出方程式。**

答：真阳性率=召回。是的，它们有相同的公式（TP / TP + FN）。

注意：要了解更多关于估值矩阵的知识。

**问14：你建了一个多元回归模型。你的模型R2分数为并不如你设想的好。为了改进，你去掉截距项，模型R2分数从0.3变为0.8。这是否可能？怎样才能达到这个结果？**

答：是的，这有可能。我们需要了解截距项在回归模型里的意义。截距项显示模型预测没有任何自变量，比如平均预测。公式R2= 1 – ∑(y – y^)2/∑(y – ymean)2中的y^是预测值。当有截距项时，截距点就是点（xmean，ymean），R2分数评估的是你的模型基于均值模型的表现。在没有截距项（ymean）时，当分母很大时，该模型就没有这样的估值效果了，∑(y – y^)2/∑(y – ymean)2式的值会变得比实际的小，而R2分数会比实际值大。

**问15：在分析了你的模型后，经理告诉你，你的模型有多重共线性。你会如何验证他说的是真的？在不丢失任何信息的情况下，你还能建立一个更好的模型吗？**

答：要检查多重共线性，我们可以创建一个相关矩阵，用以识别和除去那些具有75％以上相关性（决定阈值是主观的）的变量。此外，我们可以计算VIF（方差膨胀因子）来检查多重共线性的存在。 VIF值<= 4表明没有多重共线性，而值> = 10意味着严重的多重共线性。此外，我们还可以用容差作为多重共线性的指标。但是，删除相关的变量，也就是PCA可能会导致信息的丢失。为了留住这些变量，我们可以使用惩罚回归模型，如Ridge和Lasso回归。我们还可以在相关变量里添加一些随机噪声，使得变量变得彼此不同。但是，增加噪音可能会影响预测的准确度，因此应谨慎使用这种方法。

注意：多了解关于回归的知识。

**问16：什么时候Ridge回归优于Lasso回归？**

答：你可以引用ISLR的作者Hastie和Tibshirani的话，他们断言在对少量变量有中等或大尺度的影响的时候用lasso回归。在对多个变量只有小或中等尺度影响的时候，使用Ridge回归。

从概念上讲，我们可以说，Lasso回归（L1）同时做变量选择和参数收缩，而ridge回归只做参数收缩，并最终在模型中包含所有的系数。在有相关变量时，ridge回归可能是首选。此外，ridge回归在用最小二乘估计有更高的偏差的情况下效果最好。因此，选择合适的模型取决于我们的模型的目标。

注意：多了解关于ridge和lasso回归的相关知识。

**问17：全球平均温度的上升导致世界各地的海盗数量减少。这是否意味着海盗的数量减少引起气候变化？**

答：看完这个问题后，你应该知道这是一个“因果关系和相关性”的经典案例。我们不能断定海盗的数量减少是引起气候变化的原因，因为可能有其他因素（潜伏或混杂因素）影响了这一现象。全球平均温度和海盗数量之间有可能有相关性，但基于这些信息，我们不能说因为全球平均气温的上升而导致了海盗的消失。

注意：多了解关于因果关系和相关性的知识。

**问18：如何在一个数据集上选择重要的变量？给出解释。**

答：以下是你可以使用的选择变量的方法：

1.选择重要的变量之前除去相关变量

2.用线性回归然后基于P值选择变量（回归得到的截距项和斜率是通过普通最小二乘法得到的，我们对这两个参数做了估计，需要进行检验才能说明参数估计的有效性。原假设是斜率为0，备择假设是斜率不为0。p值是：在原假设成立的情况下，通过样本得到的结果发生的概率。如果这个概率非常小，那么原假设将被否定。上图的结果中，斜率的p值为8.21\*10^(-7)，非常小了，说明斜率不为0，那么参数估计中的斜率=1.1591是可信的。）

3.使用前向选择，后向选择，逐步选择

4.使用随机森林和Xgboost，然后画出变量重要性图

5.使用lasso回归

6.测量可用的特征集的的信息增益，并相应地选择前n个特征量。

**问19：协方差和相关性有什么区别？**

答：相关性是协方差的标准化格式。协方差本身很难做比较。例如：如果我们计算工资（$）和年龄（岁）的协方差，因为这两个变量有不同的度量，所以我们会得到不能做比较的不同的协方差。为了解决这个问题，我们计算相关性来得到一个介于-1和1之间的值，就可以忽略它们各自不同的度量。

**问20：是否有可能捕获连续变量和分类变量之间的相关性？如果可以的话，怎样做？**

答：是的，我们可以用ANCOVA（协方差分析）技术来捕获连续型变量和分类变量之间的相关性。

**问21：Gradient boosting算法（GBM）和随机森林都是基于树的算法，它们有什么区别？**

答：最根本的区别是，随机森林算法使用bagging技术做出预测。 GBM采用boosting技术做预测。在bagging技术中，数据集用随机采样的方法被划分成使n个样本。然后，使用单一的学习算法，在所有样本上建模。接着利用投票或者求平均来组合所得到的预测。Bagging是平行进行的。而boosting是在第一轮的预测之后，算法将分类出错的预测加高权重，使得它们可以在后续一轮中得到校正。

这种给予分类出错的预测高权重的顺序过程持续进行，一直到达到停止标准为止。随机森林通过减少方差（主要方式）提高模型的精度。生成树之间是不相关的，以把方差的减少最大化。在另一方面，GBM提高了精度，同时减少了模型的偏差和方差。

注意：多了解关于基于树的建模知识。

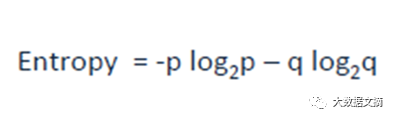
**问22：运行二元分类树算法很容易，但是你知道一个树是如何做分割的吗，即树如何决定把哪些变量分到哪个根节点和后续节点上？**

答：分类树利用基尼系数与节点熵来做决定。简而言之，树算法找到最好的可能特征，它可以将数据集分成最纯的可能子节点。树算法找到可以把数据集分成最纯净的可能的子节点的特征量。基尼系数是，如果总体是完全纯的，那么我们从总体中随机选择2个样本，而这2个样本肯定是同一类的而且它们是同类的概率也是1。我们可以用以下方法计算基尼系数：

1.利用成功和失败的概率的平方和(p^2+q^2)计算子节点的基尼系数

2.利用该分割的节点的加权基尼分数计算基尼系数以分割

熵是衡量信息不纯的一个标准（二分类）：



这里的p和q是分别在该节点成功和失败的概率。当一个节点是均匀时熵为零。当2个类同时以50%对50%的概率出现在同一个节点上的时候，它是最大值。熵越低越好。

**问23：你已经建了一个有10000棵树的随机森林模型。在得到0.00的训练误差后，你非常高兴。但是，验证错误是34.23。到底是怎么回事？你还没有训练好你的模型吗？**

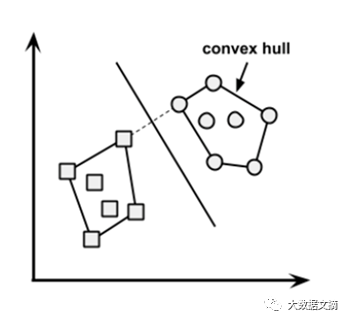
答：该模型过度拟合。训练误差为0.00意味着分类器已在一定程度上模拟了训练数据，这样的分类器是不能用在未看见的数据上的。因此，当该分类器用于未看见的样本上时，由于找不到已有的模式，就会返回的预测有很高的错误率。

在随机森林算法中，用了多于需求个数的树时，这种情况会发生。因此，为了避免这些情况，我们要用交叉验证来调整树的数量。

**问24：你有一个数据集，变量个数p大于观察值个数n。为什么用OLS（最小二乘法）是一个不好的选择？用什么技术最好？为什么？**

答：在这样的高维数据集中，我们不能用传统的回归技术，因为它们的假设往往不成立。当p>nN，我们不能计算唯一的最小二乘法系数估计，方差变成无穷大，因此OLS无法在此使用的。为了应对这种情况，我们可以使用惩罚回归方法，如lasso、LARS、ridge，这些可以缩小系数以减少方差。准确地说，当最小二乘估计具有较高方差的时候，ridge回归最有效。

其他方法还包括子集回归、前向逐步回归。



**问25：什么是凸包？（提示：想一想SVM）**

答：当数据是线性可分的，凸包就表示两个组数据点的外边界。一旦凸包建立，我们得到的最大间隔超平面（MMH）作为两个凸包之间的垂直平分线。 MMH是能够最大限度地分开两个组的线。

**问26：我们知道，一位有效编码会增加数据集的维度。但是，标签编码不会。为什么？**

答：对于这个问题不要太纠结。这只是在问这两者之间的区别。

用一位有效编码编码，数据集的维度（也即特征）增加是因为它为分类变量中存在的的每一级都创建了一个变量。

例如：假设我们有一个变量“颜色”。这变量有3个层级，即红色、蓝色和绿色。其中的有效编码为one-hot编码。对“颜色”变量进行一位有效编码会生成含0和1值的Color.Red，Color.Blue和Color.Green三个新变量。在标签编码中，分类变量的层级编码为0和1，因此不生成新变量。标签编码主要是用于二进制变量。

**问27：你会在时间序列数据集上使用什么交叉验证技术？是用k倍或LOOCV？**

答：都不是。对于时间序列问题，k倍可能会很麻烦，因为第4年或第5年的一些模式有可能跟第3年的不同，而对数据集的重复采样会将分离这些趋势，我们可能最终是对过去几年的验证，这就不对了。相反，我们可以采用如下所示的5倍正向链接策略：

fold 1 : training [1], test [2]

fold 2 : training [1 2], test [3]

fold 3 : training [1 2 3], test [4]

fold 4 : training [1 2 3 4], test [5]

fold 5 : training [1 2 3 4 5], test [6]

1，2，3，4，5，6代表的是年份。

**问28：给你一个缺失值多于30%的数据集？比方说，在50个变量中，有8个变量的缺失值都多于30%。你对此如何处理？**

答：我们可以用下面的方法来处理：

1.把缺失值分成单独的一类，这些缺失值说不定会包含一些趋势信息。

2.我们可以毫无顾忌地删除它们。

3.或者，我们可以用目标变量来检查它们的分布，如果发现任何模式，我们将保留那些缺失值并给它们一个新的分类，同时删除其他缺失值。

**问29：“买了这个的客户，也买了......”亚马逊的建议是哪种算法的结果？**

答：这种推荐引擎的基本想法来自于协同过滤。

协同过滤算法考虑用于推荐项目的“用户行为”。它们利用的是其他用户的购买行为和针对商品的交易历史记录、评分、选择和购买信息。针对商品的其他用户的行为和偏好用来推荐项目（商品）给新用户。在这种情况下，项目（商品）的特征是未知的。

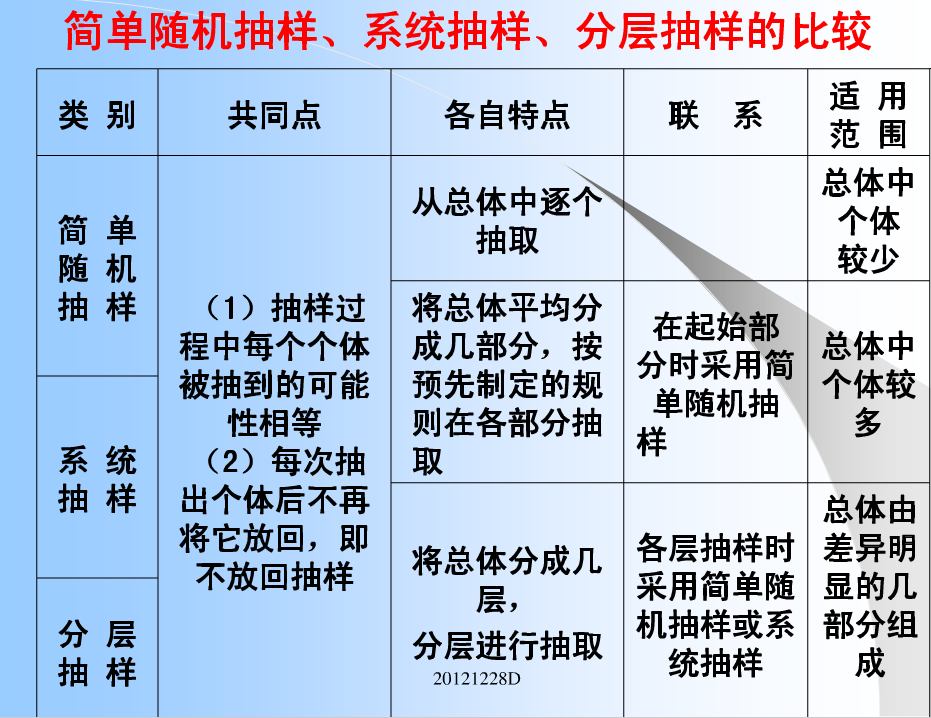
注意：了解更多关于推荐系统的知识。

**问30：你怎么理解第一类和第二类错误？**

答：第一类错误是当原假设为真时，我们却拒绝了它，也被称为“假阳性”。第二类错误是当原假设为是假时，我们接受了它，也被称为“假阴性”。在混淆矩阵里，我们可以说，当我们把一个值归为阳性（1）但其实它是阴性（0）时，发生第一类错误。而当我们把一个值归为阴性（0）但其实它是阳性（1）时，发生了第二类错误。

**问31：当你在解决一个分类问题时，出于验证的目的，你已经将训练集随机抽样地分成训练集和验证集。你对你的模型能在未看见的数据上有好的表现非常有信心，因为你的验证精度高。但是，在得到很差的精度后，你大失所望。什么地方出了错？**

答：在做分类问题时，我们应该使用分层抽样而不是随机抽样。随机抽样不考虑目标类别的比例。相反，分层抽样有助于保持目标变量在所得分布样本中的分布。



**问32：你被要求基于R2、校正后的R2和容差对一个回归模型做评估。你的标准会是什么？**

答：容差（1 / VIF）是多重共线性的指标。它是一个预测变量中的方差的百分比指标，这个预测变量不能由其他预测变量来计算。容差值越大越好。

相对于R2我们会用校正R2，因为只要增加变量数量，不管预测精度是否提高，R2都会变大。但是，如果有一个附加变量提高了模型的精度，则校正R2会变大，否则保持不变。很难给校正R2一个标准阈值，因为不同数据集会不同。

例如：一个基因突变数据集可能会得到一个较低的校正R2但仍提供了相当不错的预测，但相较于股票市场，较低的校正R2只能说明模型不好。

**问33：在k-means或kNN，我们是用欧氏距离来计算最近的邻居之间的距离。为什么不用曼哈顿距离？**

答：我们不用曼哈顿距离，因为它只计算水平或垂直距离，有维度的限制。另一方面，欧式距离可用于任何空间的距离计算问题。因为，数据点可以存在于任何空间，欧氏距离是更可行的选择。例如：想象一下国际象棋棋盘，象或车所做的移动是由曼哈顿距离计算的，因为它们是在各自的水平和垂直方向的运动。

**问34：把我当成一个5岁的小孩来解释机器学习。**

答：很简单。机器学习就像婴儿学走路。每次他们摔倒，他们就学到（无知觉地）并且明白，他们的腿要伸直，而不能弯着。他们下一次再跌倒，摔疼了，摔哭了。但是，他们学会“不要用那种姿势站着”。为了避免摔疼，他们更加努力尝试。为了站稳，他们还扶着门或墙壁或者任何靠近他们的东西。这同样也是一个机器如何在环境中学习和发展它的“直觉”的。

注意：这个面试问题只是想考查你是否有深入浅出地讲解复杂概念的能力。

**问35：我知道校正R2分数或者F值来是用来评估线性回归模型的。那用什么来评估逻辑回归模型？**

答：我们可以使用下面的方法：

1.由于逻辑回归是用来预测概率的，我们可以用AUC-ROC曲线以及混淆矩阵来确定其性能。

2.此外，在逻辑回归中类似于校正R2的指标是AIC。AIC是对模型系数数量惩罚模型的拟合度量。因此，我们更偏爱有最小AIC的模型。

3.空偏差指的是只有截距项的模型预测的响应。数值越低，模型越好。残余偏差表示由添加自变量的模型预测的响应。数值越低，模型越好。

了解更多关于逻辑回归的知识。

**问36：考虑到机器学习有这么多算法，给定一个数据集，你如何决定使用哪一个算法？**

答：你应该说，机器学习算法的选择完全取决于数据的类型。如果给定的一个数据集是线性的，线性回归是最好的选择。如果数据是图像或者音频，那么神经网络可以构建一个稳健的模型。如果该数据是非线性互相作用的的，可以用boosting或bagging算法。

如果业务需求是要构建一个可以部署的模型，我们可以用回归或决策树模型（容易解释和说明），而不是黑盒算法如SVM，GBM等。总之，没有一个一劳永逸的算法。我们必须有足够的细心，去了解到底要用哪个算法。

**问37：你认为把分类变量当成连续型变量会更得到一个更好的预测模型吗？**

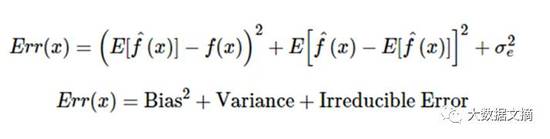
回答：为了得到更好的预测，只有在分类变量在本质上是有序的情况下才可以被当做连续型变量来处理。

**问38：什么时候正则化在机器学习中是有必要的？**

答：当模型过度拟合或者欠拟合的时候，正则化是有必要的。这个技术引入了一个成本项，用于带来目标函数的更多特征。因此，正则化是将许多变量的系数推向零，由此而降低成本项。这有助于降低模型的复杂度，使该模型可以在预测上（泛化）变得更好。

**问39：你是怎么理解偏差方差的平衡？**

答：从数学的角度来看，任何模型出现的误差可以分为三个部分。以下是这三个部分：



偏差误差在量化平均水平之上预测值跟实际值相差多远时有用。高偏差误差意味着我们的模型表现不太好，因为没有抓到重要的趋势。

而另一方面，方差量化了在同一个观察上进行的预测是如何彼此不同的。高方差模型会过度拟合你的训练集，而在训练集以外的数据上表现很差。

**问40：OLS（最小二乘法）是用于线性回归。最大似然是用于逻辑回归。解释以上描述。**

答：OLS和最大似然是使用各自的回归方法来逼近未知参数（系数）值的方法。简单地说，普通最小二乘法（OLS）是线性回归中使用的方法，它是在实际值和预测值相差最小的情况下而得到这个参数的估计。最大似然性有助于选择使参数最可能产生观测数据的可能性最大化的参数值。

**naive bayes和logistic regression的区别**

总结起来，有以下几点不同：

(1)     Naive Bayes是一个生成模型，在计算P(y|x)之前，先要从训练数据中计算P(x|y)和P(y)的概率，从而利用贝叶斯公式计算P(y|x)。

         Logistic Regression是一个判别模型，它通过在训练数据集上最大化判别函数P(y|x)学习得到，不需要知道P(x|y)和P(y)。

(2)    Naive Bayes是建立在条件独立假设基础之上的；Logistic Regression的限制则要宽松很多，如果数据满徐条件独立假设，Logistic Regression能够取得非常好的效果；当数据不满度条件独立假设时，Logistic Regression仍然能够通过调整参数让模型最大化的符合数据的分布，从而训练得到在现有数据集下的一个最优模型。

 (3)    当数据集比较小的时候，应该选用Naive Bayes，为了能够取得很好的效果，数据的需求量为O(log n)；当数据集比较大的时候，应该选用Logistic Regression，为了能够取得很好的效果，数据的需求量为O( n)

    Naive Bayes运用了比较严格的条件独立假设，为了计算P(y|x)，我们可以利用统计的方法统计数据集中P(x|y)和P(y)出现的次数，从而求得P(x|y)和P(y)。因而其所需的数据量要小一些，为O(log n).

     Logistic Regression在计算时，是在整个参数空间进行线性搜索的，需要的数据集就更大，为O( n)

**LDA中Dirichlet Distribution的定义和性质，为什么它和multinomial distribution是共轭的，什么叫叫共轭分布。**