1. 有监督（supervise）

在监督式学习下，输入数据被称为“训练数据”，每组训练数据有一个明确的标识或结果，如对防垃圾邮件系统中“垃圾邮件”“非垃圾邮件”，对手写数字识别中的“1“，”2“，”3“，”4“等。在建立预测模型的时候，监督式学习建立一个学习过程，将预测结果与“训练数据”的实际结果进行比较，不断的调整预测模型，直到模型的预测结果达到一个预期的准确率。监督式学习的常见应用场景如分类问题和回归问题。常见算法有逻辑回归（Logistic Regression）和反向传递神经网络（Back Propagation Neural Network）

1. 无监督

　在非监督式学习中，数据并不被特别标识，学习模型是为了推断出数据的一些内在结构。常见的应用场景包括关联规则的学习以及聚类等。常见算法包括Apriori算法以及k-Means算法。

1. 泛化能力

在机器学习方法中，泛化能力通俗来讲就是指学习到的模型对未知数据的预测能力。在实际情况中，我们通常通过测试误差来评价学习方法的泛化能力。如果在不考虑数据量不足的情况下出现模型的泛化能力差，那么其原因基本为对损失函数的优化没有达到全局最优。

1. 过拟合

如果我们把训练样本的一些特有的特点也当做潜在样本的一般性质，这样就会导致泛化能力下降，训练误差和测试误差之间的差距太大。

low bias high variance

解决办法：

early stopping: Early stopping便是一种迭代次数截断的方法来防止过拟合的方法，即在模型对训练数据集迭代收敛之前停止迭代来防止过拟合。对模型进行训练的过程即是对模型的参数进行学习更新的过程，这个参数学习的过程往往会用到一些迭代方法，如梯度下降（Gradient descent）学习算法。这样可以有效阻止过拟合的发生，因为过拟合本质上就是对自身特点过度地学习。

、数据集扩增（Data augmentation）：**数据集扩增**

  在数据挖掘领域流行着这样的一句话，“有时候往往拥有更多的数据胜过一个好的模型”。但是往往条件有限，如人力物力财力的不足，而不能收集到更多的数据，如在进行分类的任务中，需要对数据进行打标，并且很多情况下都是人工得进行打标，因此一旦需要打标的数据量过多，就会导致效率低下以及可能出错的情况。所以，往往在这时候，需要采取一些计算的方式与策略在已有的数据集上进行手脚，以得到更多的数据。   
  通俗得讲，数据机扩增即需要得到更多的符合要求的数据，即和已有的数据是独立同分布的，或者近似独立同分布的。一般有以下方法：

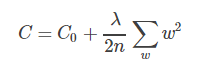
* 从数据源头采集更多数据
* 复制原有数据并加上随机噪声
* 重采样
* 根据当前数据集估计数据分布参数，使用该分布产生更多数据等

、正则化（Regularization）: **正则化：**

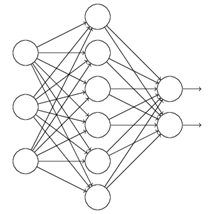
指的是在目标函数后面添加一个正则化项，一般有L1正则化与L2正则化。L1正则是基于L1范数，即在目标函数后面加上参数的L1范数和项，即参数绝对值和与参数的积项

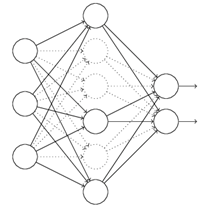


L2正则是基于L2范数，即在目标函数后面加上参数的L2范数和项，即参数的平方和与参数的积项：



Dropout：**DropOut：**

而在神经网络中，有一种方法是通过修改神经网络本身结构来实现的，其名为Dropout。该方法是在对网络进行训练时用一种技巧（trick），对于如下所示的三层人工神经网络：   
 

对于上图所示的网络，在训练开始时，随机得删除一些（可以设定为一半，也可以为1/3，1/4等）隐藏层神经元，即认为这些神经元不存在，同时保持输入层与输出层神经元的个数不变，这样便得到如下的ANN：   
 

然后按照BP学习算法对ANN中的参数进行学习更新（虚线连接的单元不更新，因为认为这些神经元被临时删除了）。这样一次迭代更新便完成了。下一次迭代中，同样随机删除一些神经元，与上次不一样，做随机选择。这样一直进行瑕疵，直至训练结束。

1. 欠拟合

即训练样本的性质都没有学习完全，模型不能在训练集上获得足够低的误差。

high bias low variance

解决方法：

1）添加其他特征项，有时候我们模型出现欠拟合的时候是因为特征项不够导致的，可以添加其他特征项来很好地解决。例如，“组合”、“泛化”、“相关性”三类特征是特征添加的重要手段，无论在什么场景，都可以照葫芦画瓢，总会得到意想不到的效果。除上面的特征之外，“上下文特征”、“平台特征”等等，都可以作为特征添加的首选项。

2）添加多项式特征，这个在机器学习算法里面用的很普遍，例如将线性模型通过添加二次项或者三次项使模型泛化能力更强。例如上面的图片的例子。

3）减少正则化参数，正则化的目的是用来防止过拟合的，但是现在模型出现了欠拟合，则需要减少正则化参数。

6. 交叉验证

交叉验证的基本思想是把在某种意义下将原始数据(dataset)进行分组,一部分做为训练集(train set),另一部分做为验证集(validation set or test set),首先用训练集对分类器进行训练,再利用验证集来测试训练得到的模型(model),以此来做为评价分类器的性能指标。

## 二

1.线性回归原理

将feature与result拟合成线性关系。

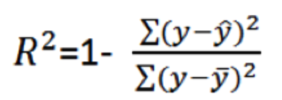
使用最小二乘法

相关指数

**R方理论评估模型**

在前一篇文章中提到了模型的准确性accuracy一词。实际上accuracy在统计学中应该称为“可决系数”，它的计算方法有两种。在这里我尝试介绍R方理论作为计算方法。

R方的计算方法如下：



## 三

1.损失函数

二分之残差平方和

2.目标函数：残差平方和

3.代价函数：二分之平均残差平方和

## 四．优化方法

1.梯度下降法

梯度下降法是按下⾯面的流程进⾏行行的：

1）⾸首先对 赋值，这个值可以是随机的，也可是让 是⼀一个全零的向量量； 2）改变 的值，使得 按梯度下降的⽅方向进⾏行行减少。

批量量梯度下降法（BGD）：每次遍历所有样本

随机梯度下降算法（SGD）: 每更更新⼀一次， 只⽤用到训练集中的⼀一个训练样本

根据样本大小判断

2.牛顿法

借鉴牛顿迭代法求导数最值，二次展开



这个式子是成立的，当且仅当 Δx 无线趋近于0。此时上式等价与：



求解：



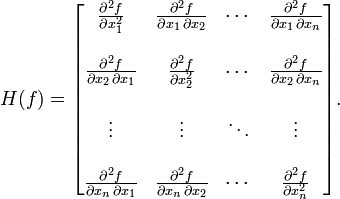
得出迭代公式：



在上面讨论的是2维情况，高维情况的牛顿迭代公式是：



其中H是hessian矩阵，定义为：



 参考<https://blog.csdn.net/sonojao/article/details/44513311>

二 拟牛顿法

拟牛顿法与原始牛顿法的区别在于增加了沿牛顿方向的一维搜索，其迭代公式为：



其中为牛顿方向，是由一维搜索的步长，也就是满足：



其实牛顿法就是阻尼牛顿法步长为1的特殊情况。

拟牛顿法算法：

输入：目标函数f(X)，梯度▽f(x),海赛矩阵H(x),精度要求ε；

输出：f(x)的极小点x\*.

步骤一：取初始点x0,置k=0

步骤二：计算梯度▽f(x)

步骤三：||▽f(x)||〈ε，那么停止计算得到的x\*=xk。

步骤四：计算H(x)

步骤五：从xk出发，沿着dk方向作一维搜索，



步骤六：

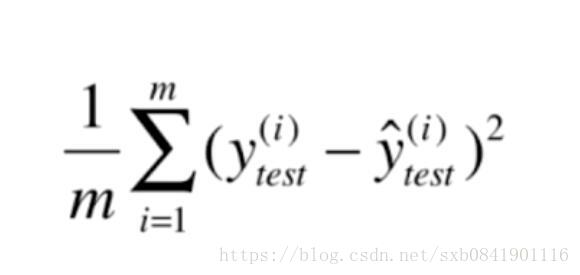
步骤七：转步骤二

参考：https://www.cnblogs.com/xiaohuahua108/p/6011105.html

## 线性回归评估标准

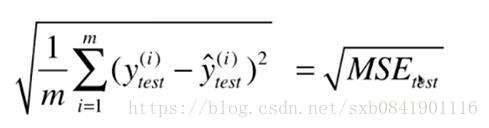
评价线性回归的指标有四种，均方误差（Mean Squared Error）、均方根误差（Root Mean Squared Error）、平均绝对值误差（Mean Absolute Error）以及R Squared方法。 sklearnz中使用的，也是大家推荐的方法是R Squared方法。

1、均方误差 MSE

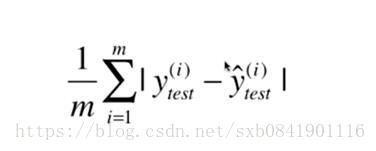


MSE的值越小，说明预测模型描述实验数据具有更好的精确度

2、均方根误差 RMSE



3、平均绝对值误差 MAE



4、 R平方 R2S

