机器学习的基本思路

1. 机器学习的一般流程
2. 数据准备
3. 数据清洗

将不好的数据处理为正常的数据

* 处理缺失值

填补 缺失值 的方法：

* 直接删除
* 人工填写
* 全局常量
* 属性统计值 某一特定类别的属性统计值
* 使用模型进行估计缺失值最可能的取值
* 处理异常值（噪声）

识别异常值的方法：首先是完全不符合逻辑的值，比如年龄为负

然后可通过3sigma原则和箱型图等来判断极端值

异常值处理方法：

* 直接删除
* 暂时保留，结合模型结果综合分析
* 均值或其他统计值替代
* 视为缺失值，用模型填补
* 处理重复值

1. 数据预处理

数据预处理与特征生成：**将正常的数据处理为优秀的数据**

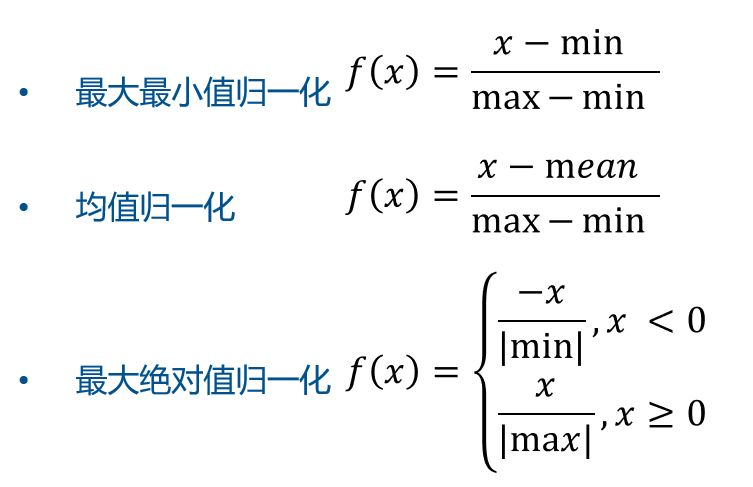
* 特征编码
* 离散化：等距分箱 等频分箱 聚类分箱[[1]](#footnote-1)
* 序号编码：将不同类别用数字进行划分，不然无法进行学习
* 独热编码：处理序号编码里单一类别会导致加入大小信息的问题，根据类别数量增加特征来描述，即增加多个0-1变量
* 特征缩放（变换）

当每一列的值范围非常不同时，我们需要将它们扩展到公共级别。

我们有不同的特征，其中一个特征的数据可能以公里表示，另一列的数据可能以米表示，最后一列的数据可能以厘米表示。在将算法应用到数据上之前，首先需要将数据放到“米”、“公里”或“厘米”的公共尺度上进行有效的分析和预测。

常见的特征变换方法：

* 归一化



但归一化仅是粗暴地将数据缩放到区间，没有考虑到数据的**正态性特征**。

这时我们可以采用标准化。

* 标准化



* Log变换

符合正态分布的数据才能让模型效果最好，但有些数据具有**长尾分布**，即拖着一个尾巴。Log函数的特征是较大的值经过变换后会比较小，于是我们可以通过log变化来获得更好的数据。

1. 特征生成

不是很重要的知识。关键是在数据处理过程中，意识到部分原始数据需要经过处理才能成为可供分析的特征（统计学中的自变量、解释变量）。

从**原始数据**中我们可以获得状态性、趋势性与统计性的特征

* 状态性特征：如平均欠款金额
* 趋势性特征：如工资增长趋势
* 统计性特征：标准差

1. 模型训练
2. 训练集、验证集与测试集

统计学（以及计量经济学）特别注重严谨性，主要通过理论与数学推导来寻找最佳的模型。[对的才是好的] 但机器学习基于工程的思想，更加关注应用中的效果，选取在测试中预测效果最好的模型作为最佳模型。[管他对不对，好用就行]

对于获得的样本数据，我们将其中的一部分做为训练集，另外一部分做为测试集，训练集用于训练模型中的参数，测试集用来测试模型的预测效果。然后，我们再从训练集中分出一部分作为验证集，用来寻找最优的超参数。

模型的参数通常是由数据来驱动调整；超参数则不需要数据来驱动，而是在训练前或者训练中人为地进行调整。[[2]](#footnote-2)

案例：模型、超参数与参数

逻辑回归专门处理机器学习中的二分类问题，简称为Logit模型。该模型的方程为：



其中，为模型的参数，通过训练集计算出来。

我们可以调用sklearn模块的LogisticRegression类进行逻辑回归，在基于该类创建实例对象的时候，需要人为的设定一系列参数，比较重要的如下，

* penalty：指定正则化的参数，可选"l1", “l2” 默认为“l2”；

注意：l1正则化会将部分参数压缩到0，而l2正则化不会让参数取到0，只会无限接近

* C : 大于0的浮点数。C越小对损失函数的惩罚越重；
* solver：指定模型的求解方式；[[3]](#footnote-3)

这些参数则为超参数。

那么，单就逻辑回归模型而言，机器学习的思路是，循环选取不同的超参数，建立模型，通过训练集进行训练，找到最优参数；通过验证集得到预测准确率，选取预测准确率最高的超参数组合，以及该超参数组合下的最优参数，联立建立最终的模型。最后通过测试集，再一次计算模型评价的指标，与其他模型进行比较，选出研究问题的最佳模型。

我们把寻找最优超参数的过程，称之为调参。

本质上来说，机器学习的过程，就是找模型、找超参数以及找参数的过程。

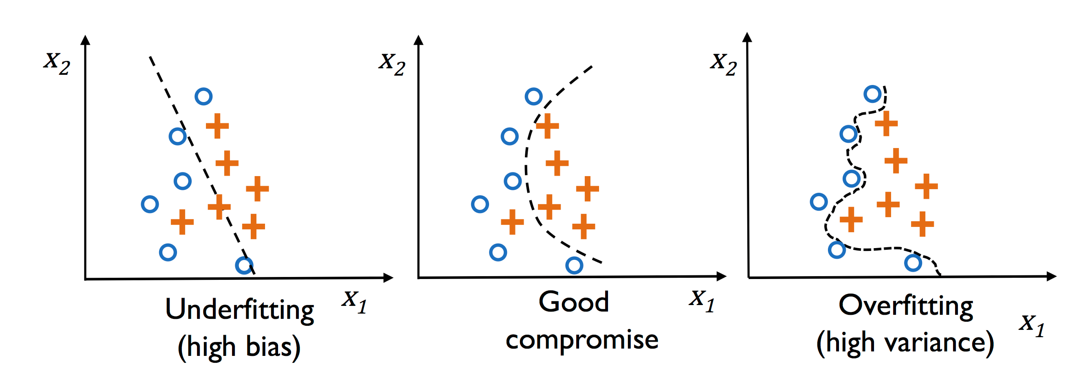
1. k-折交叉验证

在机器学习中，我们可以将20%的数据作为测试集，20%的数据作为验证集，60%的数据作为训练集。但当数据量较小的时候，训练集的训练效果可能不太好，一种处理方法是k-折交叉验证。以4-折交叉验证为例，我们将除去了测试集的数据集合随机分成4等份，每次选取一等份作为验证集，剩下的数据集作为训练集，进行四次模型的训练，并取评价指标的平均值作为最终的评价指标。？ **也可能是选取一等份为测试集，剩下部分作为测试集。**

1. 欠拟合与过拟合

通过figure 1，能很好地理解拟合、欠拟合与过拟合的关系。拟合的过程是在训练集与验证集中进行的，而模型的效果评价通过测试集计算。如果对训练集与验证集过度拟合，在测试集中测试的效果就会较差。

所以采用的模型不是越复杂越好，处理过拟合问题也是机器学习里模型调优的关键一环。



figure

1. 模型评价

机器学习中有一系列模型评价的指标，在模型训练后通过测试集 计算出来。

对于训练好的逻辑回归模型，我们称之为“分类器(Classifier)”。

为了计算有关指标，我们首先画出分类结果的混淆矩阵(Confusion Matrix)：

表 分类结果的混淆矩阵

|  |  |  |  |
| --- | --- | --- | --- |
| 项目 | | 实际值 | |
| 正例 | 负例 |
| 预测值 | 正例 | TP | FP |
| 负例 | FN | TN |

其次，我们给出这些指标的计算公式与定义。其中，精确率与召回率是相互矛盾的。

|  |  |  |
| --- | --- | --- |
| 准确率（Accuracy） |  | 预测正确的比例 |
| 精确率 (Precision) |  | 预测为正例的样例中实际为正例的比例 |
| 召回率 （Recall） |  | 实际为正例的样例中被预测为正例的比例 |
| f1-分数值 |  | 预测精度与召回率的调和平均数 |

* 分数值，召回率的权重是精确率的β倍：



* ROC与AUC

案例分析：银行卡电信诈骗危险预测（首届“钉钉杯”大学生大数据挑战赛A题）

样本数据包含100万条记录，7个特征与1个类标签，其中特征既有连续变量，也有分类变量，变量介绍如下：

➢ distance\_from\_home：银行卡交易地点与家的距离；  
➢ distance\_from\_last\_transaction：与上次交易发生的距离；  
➢ ratio\_to\_median\_purchase\_price：近一次交易与以往交易价格中位数的比率；  
➢ repeat\_retailer：交易是否发生在同一个商户；  
➢ used\_chip：是通过芯片（银行卡）进行的交易；  
➢ used\_pin\_number：交易时是否使用了 PIN码；  
➢ online\_order：是否是在线交易订单；  
➢ fraud：诈骗行为（分类标签）。

这里**仅仅介绍模型训练**的过程。因为分类标签诈骗行为为二分类变量，故采用逻辑回归模型。我们调用LogisticRegression包，采用10-折交叉验证，调整的超参数为正则化参数penalty与惩罚参数C。

代码如下：

import numpy as np  
import pandas as pd  
from sklearn.linear\_model import LogisticRegression  
from sklearn.model\_selection import GridSearchCV  
from sklearn.model\_selection import train\_test\_split  
from sklearn.model\_selection import cross\_val\_score  
from sklearn.model\_selection import StratifiedKFold  
  
# 调用数据  
dataset = pd.read\_csv('card\_transdata.csv')  
  
# 获得特征与类标签，x为列表的集合  
x = dataset.iloc[:, :-1].values  
y = dataset.iloc[:, 7].values  
  
# 划分训练集（80%）与测试集（20%）  
x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.2, stratify=y, random\_state=1)  
  
# 设计交叉验证的参数 这里是10折，按照y分类  
Kfold = StratifiedKFold(n\_splits=10, shuffle=True, random\_state=0）  
=============================================================================

=================================================================================  
"""通过for循环与cross\_val\_score进行交叉验证"""  
# 通过双循环寻找最优超参数  
best\_score = 0  
for penalty in ['l1', 'l2']:  
 for C in [0.01, 0.1, 1, 10]:  
 md = LogisticRegression(solver='liblinear', penalty=penalty, C=C)  
 scores = cross\_val\_score(md, x\_train, y\_train, cv=Kfold) #10次交叉训练，返回10个结果  
 score = np.mean(scores)  
 if score > best\_score:  
 best\_score = score  
 best\_parameters = {'penalty': penalty, 'C': C}  
  
print('模型结果1：')  
print(best\_score)  
print(best\_parameters)  
=================================================================================

=================================================================================  
"""通过网格搜索的方法进行交叉验证"""  
# 设计超参数的调整范围  
param\_grid = {'penalty': ('l1', 'l2'), 'C': (0.01, 0.1, 1, 10)}  
  
# 训练模型，并通过网格搜索方法寻找最优超参数  
model = GridSearchCV(LogisticRegression(solver='liblinear'), param\_grid, cv=Kfold)  
model.fit(x\_train, y\_train)  
  
print('模型结果2：')  
print(model.best\_params\_)  
print(model.best\_estimator\_)

模型结果：

模型结果1：

0.9586274999999999

{'penalty': 'l1', 'C': 10}

模型结果2：

{'C': 10, 'penalty': 'l1'}

LogisticRegression(C=10, penalty='l1', solver='liblinear')

1. 聚类(Clustering)：是指把相似的数据划分到一起，具体划分的时候并不关心这一类的标签，目标就是把相似的数据聚合到一起，聚类是一种无监督学习(Unsupervised Learning)方法。

   分类(Classification)：是把不同的数据划分开，其过程是通过训练数据集获得一个分类器，再通过分类器去预测未知数据，分类是一种监督学习(Supervised Learning)方法。 [↑](#footnote-ref-1)
2. [深度学习超参数介绍及调参 - 简书 (jianshu.com)](https://www.jianshu.com/p/6602c76cc801) [↑](#footnote-ref-2)
3. [(6条消息) 机器学习sklearn----逻辑回归(LogisticRegression)使用详解\_iostreamzl的博客-CSDN博客\_logisticregression sklearn](https://blog.csdn.net/weixin_43776305/article/details/120405116) [↑](#footnote-ref-3)