

## 模式识别

### 1. KNN

[http://courses.cs.tamu.edu/rgutier/cs790\\_w02/l8.pdf](http://courses.cs.tamu.edu/rgutier/cs790_w02/l8.pdf)

### 2. LDA :将高维数据映射到低维空间中，并使得类能比较好的区分

[http://www.csd.uwo.ca/~olga/Courses//CS434a\\_541a//Lecture8.pdf](http://www.csd.uwo.ca/~olga/Courses//CS434a_541a//Lecture8.pdf)

### 3. Isolation Forest

构建二叉树来切割数据，需要多次切割才能区分出来的数据为异常值  
二叉树的深度表示数据的异常分数

Isolation Forest 算法主要有两个参数：一个是二叉树的个数；另一个是训练单棵 iTree 时候抽取样本的数目。实验表明，当设定为 **100 棵树**，**抽样样本数为 256 条**时候，IF 在大多数情况下就已经可以取得不错的效果。这也体现了算法的简单、高效。

Isolation Forest 是**无监督**的异常检测算法，在实际应用时，并不需要黑白标签。需要注意的是：（1）如果训练样本中**异常样本的比例比较高**，违背了先前提到的异常检测的基本假设，**可能最终的效果会受影响**；（2）异常检测跟具体的应用场景紧密相关，**算法检测出的“异常”不一定是我们实际想要的**。比如，在识别虚假交易时，异常的交易未必就是虚假的交易。所以，在特征选择时，可能需要过滤不太相关的特征，以免识别出一些不太相关的“异常”。

#### 参考文献

1. F. T. Liu, K. M. Ting and Z. H. Zhou, *Isolation-based Anomaly Detection*, TKDD, 2011

## 异常值检验

### 1.1 监督异常检测：

类似于模式识别，但**他的类通常极其不平衡**。C4.5决策树分类算法不适用，**但SVM和ANN**（Artificail Neural Networks）还行。由于通常异常不是提前知道的，或者是测试阶段实时产生的，这种设置通常意义不大。

普通的DNN无法对时间序列上的变化进行建模，RNN更加适合这种需求，神经元的输出在下一个时间戳会作用到自身

RNN / LTSM <http://blog.csdn.net/prom1201/article/details/52221822>

三种神经网络的区别：<https://www.zhihu.com/question/34681168>

基于CNN （不太适合用于对时间信息不完全）<http://blog.csdn.net/wzjiayou/article/details/76573031>，抽样方法可以参考解决数据不平衡问题

基于时间序列：<https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/46283.pdf>

RBM：十分接近比赛题目

<https://github.com/aaxwaz/Fraud-detection-using-deep-learning>

### 1.2 半监督异常检测：

用无异常值的训练集训练出模型，从而偏离此模型的实例的为异常值。代表算法：**One-class SVMs**和autoencoders

异常值的几种类型：

1. 点异常检测
2. 集合异常检测（可通过相关性、分组与合计来产生新特征，再使用点异常检测。需要对数据集非常熟悉，这个过程也叫做数据视图的产生）
3. 情境异常检测（可通过加入改变情境的因素作为新特征，再使用点异常检测）

无监督算法总结：

<https://www.cnblogs.com/DianeSoHungry/p/7050254.html>

LOF 和 knn比较合适，局部和全局

LOF <http://blog.csdn.net/wangyibo0201/article/details/51705966>

KDA 基于聚类

<https://arxiv.org/pdf/1503.03208.pdf>

## 参数调整

NN hyperparameters:[http://colinraffel.com/wiki/neural\\_network\\_hyperparameters](http://colinraffel.com/wiki/neural_network_hyperparameters)

TPE, BO, random, grid:

[http://neupy.com/2016/12/17/hyperparameter\\_optimization\\_for\\_neural\\_networks.html#summary](http://neupy.com/2016/12/17/hyperparameter_optimization_for_neural_networks.html#summary)

学习曲线<https://www.cnblogs.com/lgh344902118/p/8085309.html>