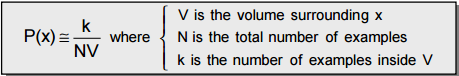
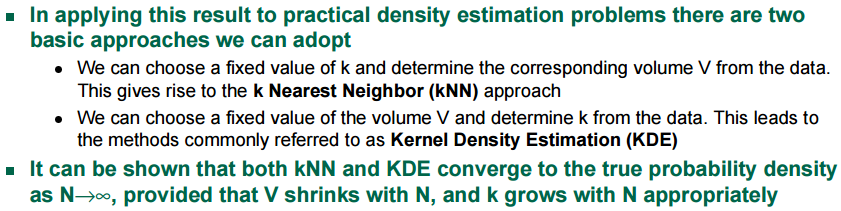
### 基于散度（分布距离）的机器学习方法

#### 非参数密度估计





#### 直方图

x轴—表示若干区间构成的样本区间

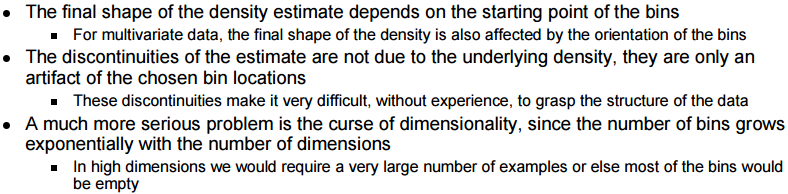
y轴—





注意：区别频率直方图和频数直方图

缺点：



#### 核密度估计

x轴—表示若干区间构成的样本区间

y轴—每一个区间内样本点的核密度估计（概率值）



表示带宽, 越大，估计的密度函数越平滑.

核函数：利用所有样本点来估计的密度，距离越近的样本点说起的作用越大。

核密度估计的加权是以数据点到的欧式距离为基准.

高维：



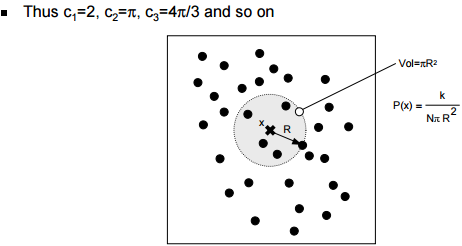
表示维空间

#### knn密度估计









表示维空间

表示到第个近邻的欧式距离，越大越光滑。

到第个近邻的欧式距离越大，说明单位体积内的样本点越稀疏，密度值越小。

knn密度估计，无论欧式距离多少，只要距离最近的k个点都可以加权.

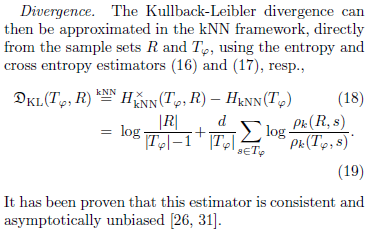
实现：

TDA::knnDE

也可以利用该函数计算散度.

#### Kullback-Leibler divergence





表示维空间

reference:

High-dimensional statistical measure for region-of-interest tracking-Boltz2009TIP\_PreprintA4

more:

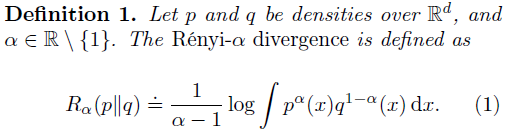
<http://math.stackexchange.com/questions/90537/what-is-the-motivation-of-the-kullback-leibler-divergence?rq=1>

<https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence#Kullback.E2.80.93Leibler_divergence_for_multivariate_normal_distributions>

实现：

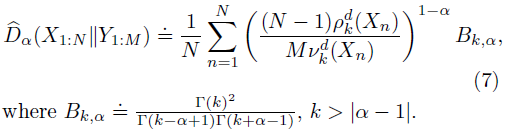
FNN::KL.dist

#### Rényi-a divergence





式（3）的估计量：

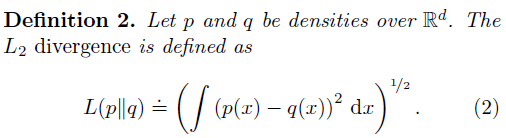


表示维空间

reference:

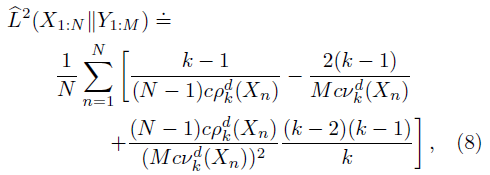
Nonparametric Divergence Estimation with Applications to Machine Learning on Distributions

#### L2 divergence





式（4）的估计量：



表示维空间

reference:

Nonparametric Divergence Estimation with Applications to Machine Learning on Distributions

#### 应用

##### 聚类

1. spectral clusting???

specClust {kknn}

##### 分类

1. knn

训练集：包含正常样本和异常样本（两类）

测试集：未知

基本思路：针对一个测试样本，寻找其在测试集中k个近邻，计算它与每一个类别的散度，将散度最小的标签作为该测试样本的类别。（k为奇数）

##### 异常值检测

1. one-class-svm ???

A Kullback-Leibler Divergence Based Kernel for SVM Classification in Multimedia Applications

Learning with Distance Substitution Kernels

Support Distribution Machines

The Kernel Trick for Distances

rank one-class svm

ksvm {kernlab}

The kernel parameter can also be set to a user defined function of class kernel by passing the function name as an argument.

1. K-LPE

散度结合K-LPE，适用于以下场景：

场景一 监督学习

训练集：只含有正常样本

测试集：未知

或

训练集：包含正常样本和异常样本

测试集：未知

场景二 非监督学习

训练集：未知

测试集：未知

基本思路一（监督学习）：针对一个测试样本，寻找其在测试集中k个近邻，并寻找其在训练集中k个近邻，计算散度作为该测试样本的得分。（非监督学习的结果是，测试样本（组）与训练样本的相似程度，相当于进行了聚类）

基本思路一（非监督学习）：针对每个特征的分布，划分正常的区间并筛选出正常样本

基本思路二（非监督学习）：聚类，大类作为正常样本（小类作为异常样本?）

可行思路（？）：针对一个测试样本，寻找其在训练集中k个近邻，并根据第k个近邻寻找自身的k个近邻，计算的散度值作为得分。

注意：

similarity(用到了高斯核函数)衡量边（edgs）可以替换为散度？

距离等相似度矩阵可以替换为散度？

nng {cccd}

##### 其他

LOF

randomforest