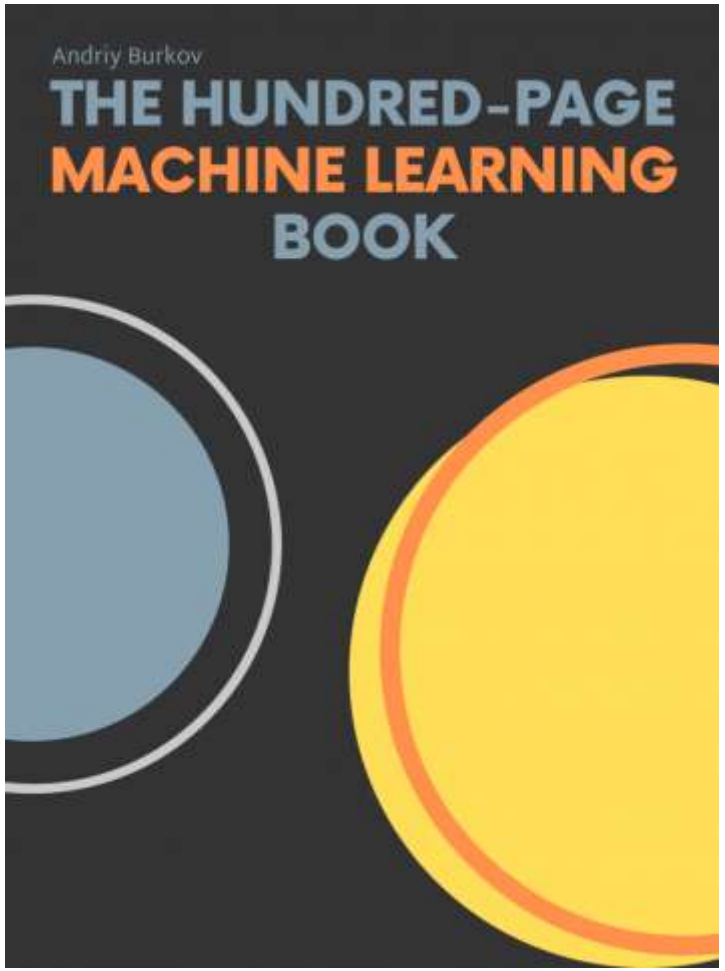
应用数据科学导论

Ht_song@163.com

2020.02

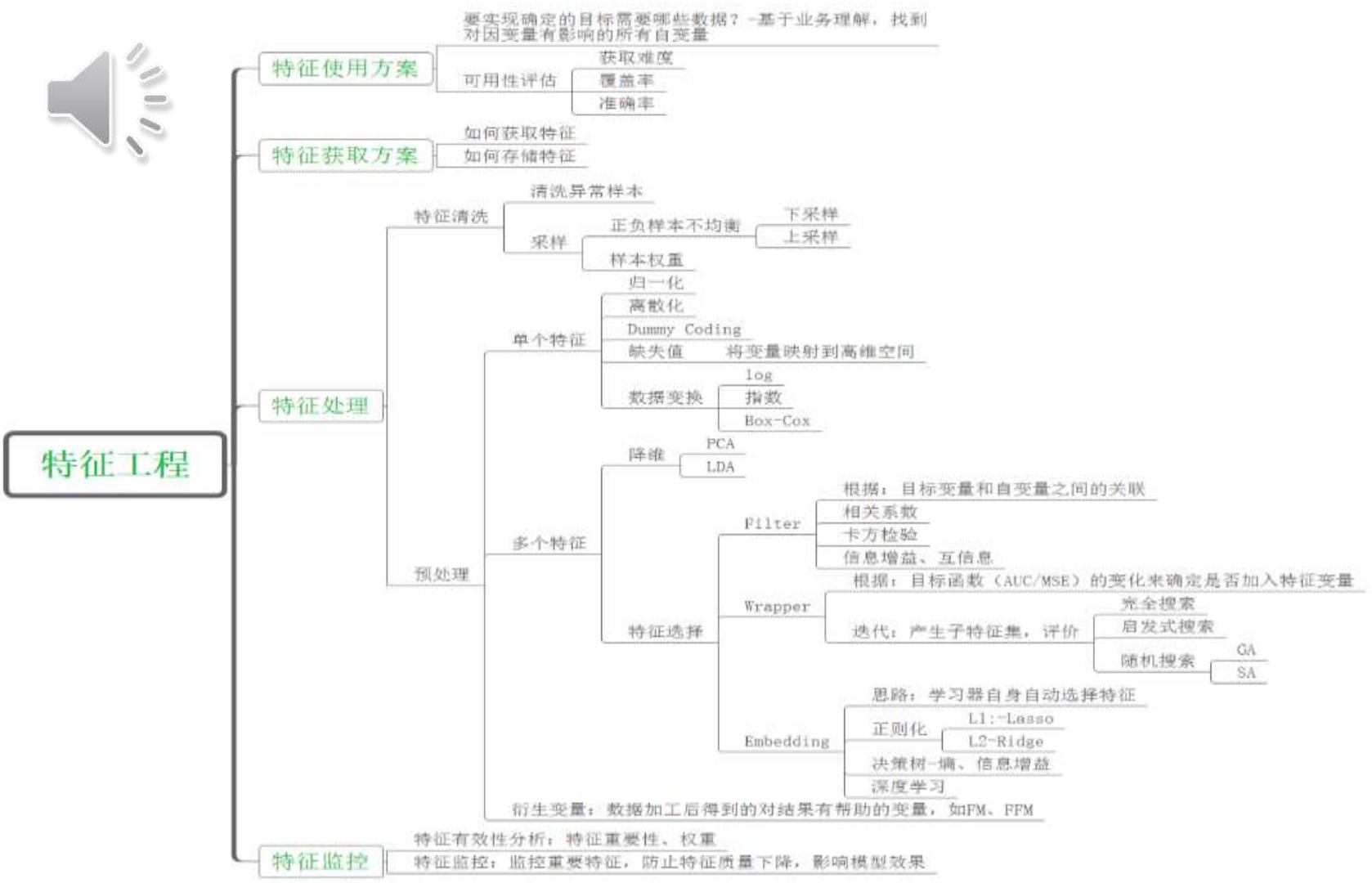
参考书


O'REILLY
TURING 图灵程序设计丛书
数据科学入门
Data Science from Scratch
[美] Joel Grus 著
高蓉 韩波 译
中国工信出版集团
人民邮电出版社
POSTS & TELECOM PRESS


Andriy Burkov
THE HUNDRED-PAGE
MACHINE LEARNING
BOOK

主要内容

- 数据科学简介
- Python语言
- 数据统计可视化
- 数据分析方法——特征工程
- 分析项目实践


特征工程
特征使用方案
要实现确定的目标需要哪些数据？-基于业务理解，找到对因变量有影响的所有自变量
可用性评估
获取难度
覆盖率
准确率
特征获取方案
如何获取特征
如何存储特征
特征处理
特征清洗
清洗异常样本
采样
正负样本不均衡
下采样
上采样
样本权重
预处理
单个特征
归一化
离散化
Dummy Coding
缺失值
将变量映射到高维空间
数据变换
log
指数
Box-Cox
多个特征
降维
PCA
LDA
特征选择
Filter
根据：目标变量和自变量之间的关联
相关系数
卡方检验
信息增益、互信息
Wrapper
根据：目标函数（AUC/MSE）的变化来确定是否加入特征变量
迭代：产生子特征集，评价
完全搜索
启发式搜索
随机搜索
GA
SA
Embedding
思路：学习器自身自动选择特征
正则化
L1:-Lasso
L2-Ridge
决策树-熵、信息增益
深度学习
衍生变量：数据加工后得到的对结果有帮助的变量，如FM、FFM
特征监控
特征有效性分析：特征重要性、权重
特征监控：监控重要特征，防止特征质量下降，影响模型效果

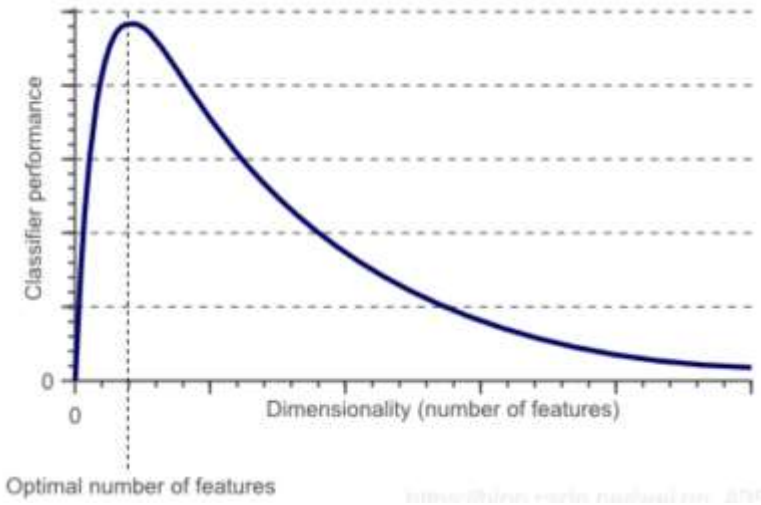
多特征处理——降维

1、降维

- **降维结果就是特征和特征之间不相关。**

2、降维的方法

- 特征选择：从高纬度的特征中寻找其中的一个子集来作为新的特征
- 特征抽取（特征降维），特征抽取则是将高纬度的特征经过某一个函数映射到低纬度来做为新的特征



图——维度与特征表现

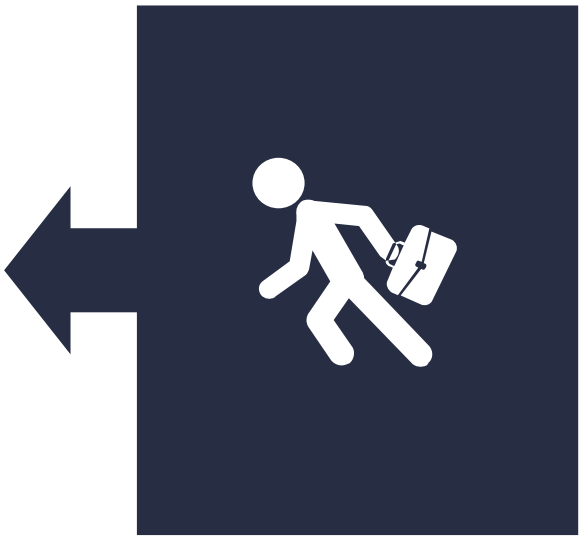
第三节

特征选择

一、特征选择

1.什么是特征选择

特征工程(Feature Selection)，也叫做特征子集选择(Feature Subset Selection, FSS)，或者叫做属性选择(Attribute Selection)。是指**从全部的数据特征中选取合适的特征**，从而确保模型变得更好。



2.特征选择的作用

①减少模型的过度拟合

②提高准确度

③缩短训练时间

特征选择的方法

②Embedded：嵌入法

先使用某些机器学习的算法和模型进行训练，得到各个特征的权值系数，根据系数从大到小选择特征。

①Filter：过滤法

按照发散性或者相关性对各个特征进行评分，设定阈值或者选择阈值的个数，选择特征

……

③Wrapper：包装法

根据目标函数（通常是预测效果评分），每次选择若干特征，或者排除若干特征

常用Filter方法

1、方差选择法

方差越大差异越大，有利于特征的区分

2、卡方检验

$$\chi^2 = \sum \frac{(A-E)^2}{E}$$

检验定性自变量对定性因变量的相关性。假设自变量有N种取值，因变量有M种取值，考虑自变量等于i且因变量等于j的样本频数的观察值与期望的差距，构建统计量

3、相关系数法

根据每个特征与目标值的相关系数与P值，去掉相关系数小的特征

一、方差选择法

方差选择步骤

- 计算各个特征的方差
- 根据阈值，选择方差大于阈值的特征
- 使用feature_selection库的VarianceThreshold类来选择特征

- #参数threshold为方差的阈值

```
from sklearn.feature_selection import VarianceThreshold
#导入python的相关模块
X=[[0, 0, 1],
   [0, 1, 0],
   [1, 0, 0],
   [0, 1, 1],
   [0, 1, 0],
   [0, 1, 1]]
#其中包含6个样本，每个样本包含3个特征

sel=VarianceThreshold(threshold=((0.8*(1-0.8))))
#表示剔除特征的方差大于阈值的特征Removing features with low variance

sel.fit_transform(X)
#返回的结果为选择的特征矩阵

print(sel.fit_transform(X))#
```

```
[[0 1]
 [1 0]
 [0 0]
 [1 1]
 [1 0]
 [1 1]]
```

实例01 方差选择法

- 方差选择法对应接口

class sklearn.feature_selection.VarianceThreshold(threshold=0.0)

Parameters:	**threshold** : float, optional Features with a training-set variance lower than this threshold will be removed. The default is to keep all features with non-zero variance, i.e. remove the features that have the same value in all samples.
Attributes:	**variances_** : array, shape (n_features,) Variances of individual features.

Methods

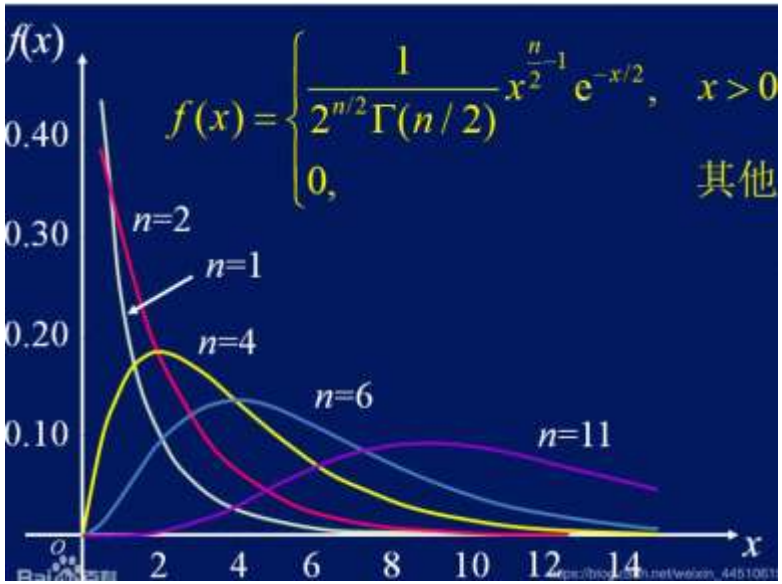
fit(X[, y])	Learn empirical variances from X.
fit_transform(X[, y])	Fit to data, then transform it.
get_params([deep])	Get parameters for this estimator.
get_support([indices])	Get a mask, or integer index, of the features selected
inverse_transform(X)	Reverse the transformation operation
set_params(**params)	Set the parameters of this estimator.
transform(X)	Reduce X to the selected features.

二、卡方检验

卡方检验

卡方检验就是统计样本的实际观测值与理论推断值之间的偏离程度。实际观测值与理论推断值之间的偏离程度决定卡方值的大小，卡方值越小，偏差越小，越趋于符合

$$\chi^2 = \sum \frac{(A-E)^2}{E}$$



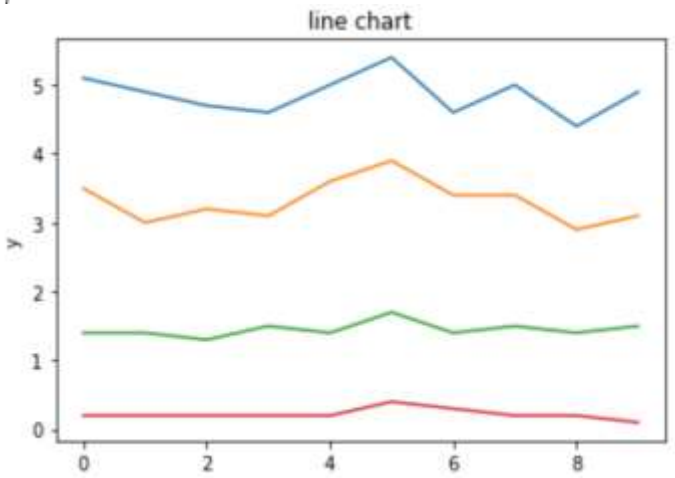
卡方检验例子

```
from sklearn.datasets import load_iris
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2
iris = load_iris()
X, y = iris.data, iris.target
X.shape
```

```
(150, 4)
```

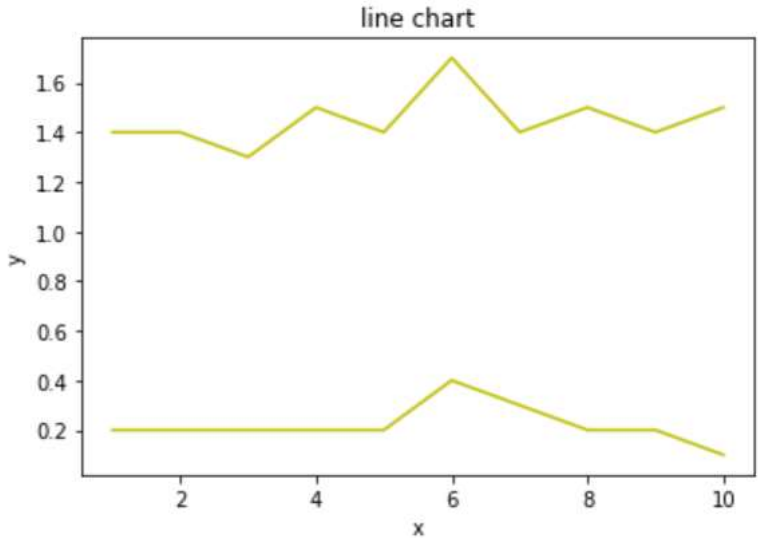
```
X_new = SelectKBest(chi2, k=2).fit_transform(X, y)
X_new.shape
```

```
(150, 2)
```




```
In [19]: # 选择K个最好的特征，返回选择特征后的数据
SelectKBest(chi2,k=2).fit_transform(iris.data,iris.target)[:1
```

```
Out[19]: array([[1.4, 0.2],
                [1.4, 0.2],
                [1.3, 0.2],
                [1.5, 0.2],
                [1.4, 0.2],
                [1.7, 0.4],
                [1.4, 0.3],
                [1.5, 0.2],
                [1.4, 0.2],
                [1.5, 0.1]])
```



三、相关系数法

1、主要方法

①皮卡森系数

相关系数可以看成协方差：一种**剔除了两个变量量纲影响、标准化后的特殊协方差**

②最大信息系数（MIC法）

MIC，即（Maximal Information Coefficient）最大信息系数，属于Maximal Information-based Nonparametric Exploration (MINE) 最大的基于信息的非参数性探索，用于衡量两个变量X和Y之间的关联程度，线性或非线性的强度

皮尔森相关系数

1、皮尔森相关系数：衡量变量之间的线性关系，

2、结果取值为[-1,1]，-1表示完全负相关，+1表示正相关，0表示不线性相关，但是并不表示没有其它关系。

3、缺陷：只衡量线性关系，如果关系是非线性的，即使连个变量具有一一对应的关系，Pearson关系也会接近0。

4、公式：样本共变异数除以X的标准差和Y的标准差的乘积。

$$r = \frac{\sum_{i=1}^{n}(xi-\bar{x})(yi-\bar{y})}{\sqrt{\sum_{i=1}^{n}(xi-\bar{x})^2 \sum_{i=1}^{n}(yi-\bar{y})^2}}$$

- 通常情况下通过以下取值范围判断变量的相关强度：
- 0.8-1.0　极强
- 0.6-0.8　强相关
- 0.4-0.6　中等程度相关

——皮尔森相关系数实例

In [27]:
```
import numpy as np
from scipy.stats import pearsonr
np.random.seed(0)
size = 300
x = np.random.normal(0, 1, size)
# pearsonr(x, y)的输入为特征矩阵和目标向量
print("Lower noise", pearsonr(x, x + np.random.normal(0, 1, size)))
print("Higher noise", pearsonr(x, x + np.random.normal(0, 10, size)))
# 输出为二元组(sorce, p-value)的数组
## 结果 Lower noise (0.71824836862138386, 7.3240173129992273e-49)
## 结果 Higher noise (0.05796429207933148, 0.31700993885324746)
```

```
Lower noise (0.7182483686213841, 7.324017312998504e-49)
Higher noise (0.05796429207933814, 0.31700993885325246)
```

Scipy的 *pearsonr* 方法能够同时计算 *相关系数* 和*p-value*.

Pearson相关系数缺陷

Pearson相关系数的一个明显缺陷是，作为特征排序机制，他只对线性关系敏感。如果关系是非线性的，即便两个变量具有一一对应的关系，Pearson相关性也可能会接近0。

In [28]:
```
x = np.random.uniform(-1, 1, 100000)
print (pearsonr(x, x**2)[0])
## 结果 -0.0023080470761 2
```

```
-0.0023080470761 23308
```

https://blog.csdn.net/FontThrone/article/details/85227239?utm_medium=distribute.pc_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-2.nonecase&depth_1-utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-2.nonecase

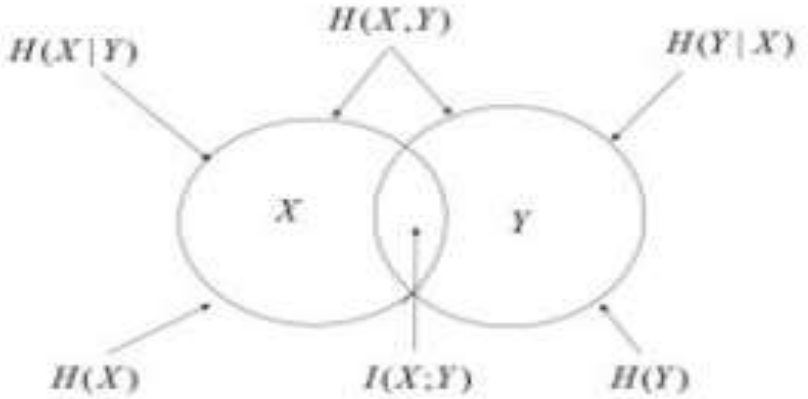
（MIC）的基础————互信息（MI）

1、互信息（MI）方法公式如下：

$$I(X,Y) = \sum_{y\in Y}\sum_{x\in X} p(x,y)\log\left(\frac{p(x,y)}{p(x)\,p(y)}\right)$$

根据熵的连锁规则，有

$$H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$$



因此，

$$H(X) - H(X|Y) = H(Y) - H(Y|X)$$

这个差叫做X和Y的互信息，记作I(X;Y)。

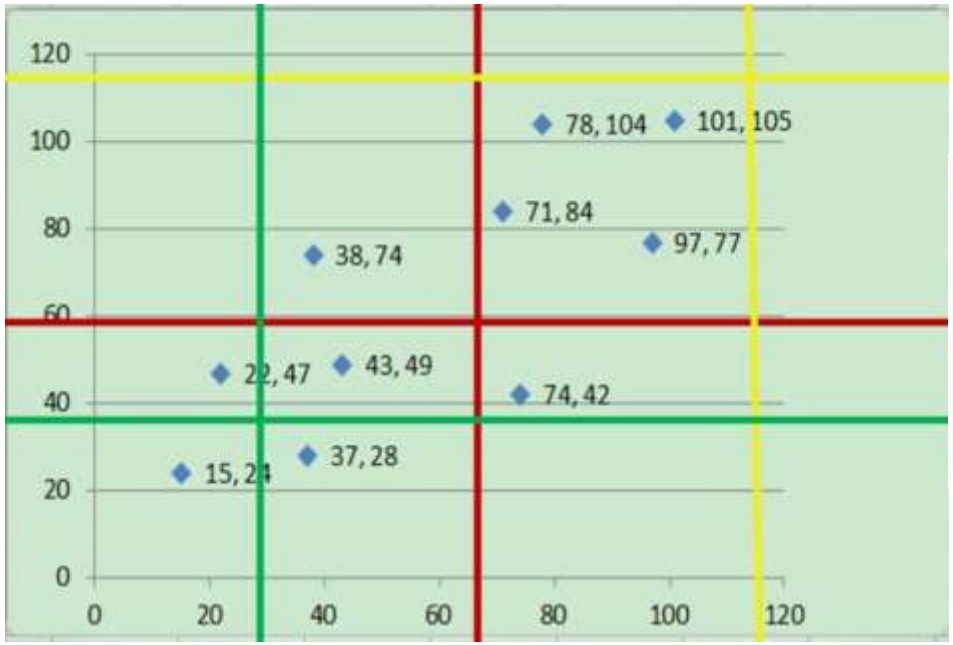
按照熵的定义展开可以得到：

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= H(X) + H(Y) - H(X,Y) \\ &= \sum_x p(x)\log\frac{1}{p(x)} + \sum_y p(y)\log\frac{1}{p(y)} - \sum_{x,y} p(x,y)\log\frac{1}{p(x,y)} \\ &= \sum_{x,y} p(x,y)log\frac{p(x,y)}{p(x)p(y)} \end{aligned}$$

MI的优化版本——MIC方法

1、最大信息系数（MIC）方法公式如下：

$$MIC[x;y] = \max_{|X||Y|<B} \frac{I[X;Y]}{\log_2(\min(|X|,|Y|))}.$$




```
from minepy import MINE
m = MINE()
x = np.random.uniform(-1, 1, 10000)
m.compute_score(x, x**2)
print(m.mic())
# 结果 1.0
```

2、特征选择

sklearn.feature_selection: Feature Selection

The sklearn.feature_selection module implements feature selection algorithms. It currently includes univariate filter selection methods and the recursive feature elimination algorithm.

User guide: See the *Feature selection* section for further details.

feature_selection.GenericUnivariateSelect([...])	Univariate feature selector with configurable strategy.
feature_selection.SelectPercentile([...])	Select features according to a percentile of the highest scores.
feature_selection.SelectKBest([score_func, k])	Select features according to the k highest scores.
feature_selection.SelectFpr([score_func, alpha])	Filter: Select the pvalues below alpha based on a FPR test.
feature_selection.SelectFdr([score_func, alpha])	Filter: Select the p-values for an estimated false discovery rate
feature_selection.SelectFwe([score_func, alpha])	Filter: Select the p-values corresponding to Family-wise error rate
feature_selection.RFE(estimator[, ...])	Feature ranking with recursive feature elimination.
feature_selection.RFECV(estimator[, step, ...])	Feature ranking with recursive feature elimination and cross-validated selection of the best number of features.
feature_selection.VarianceThreshold([threshold])	Feature selector that removes all low-variance features.
feature_selection.chi2(X, y)	Compute chi-squared stats between each non-negative feature and class.
feature_selection.f_classif(X, y)	Compute the ANOVA F-value for the provided sample.
feature_selection.f_regression(X, y[, center])	Univariate linear regression tests.

第四节

特征降维

特征降维

1、特征降维的定义

- 特征抽取，即采用一个高纬度的特征经过某一个函数映射到低纬度
- 特征抽取的方法常用的就是PCA

2、特征降维方法）

- 主成分分析法（PCA）：

尽可能好的将原特征进行线性变换，映射到低维度空间

- 线性判别分析法（LDA）：

将样本点投影到低维形成类簇，即可达到分类效果，也实现了特征降维的效果

主成分分析PCA

1、PCA方法的定义

主成分分析是一种统计方法，通过正交变换将一组可能存在相关性的变量转换为线性不相关，转换后的这组变量称为主成分。

2、主要应用场景

- 数据压缩；消除冗余；消除数据噪声；数据降维，可视化

3、理解

找出数据里最主要的成分，代替原始数据并使损失尽可能的小

a) 样本点到超平面的距离足够近

b) 样本点在这个超平面的投影尽可能的分开

PCA方法介绍

PCA算法流程

1、 中心化所有样本数据（**标准化**）

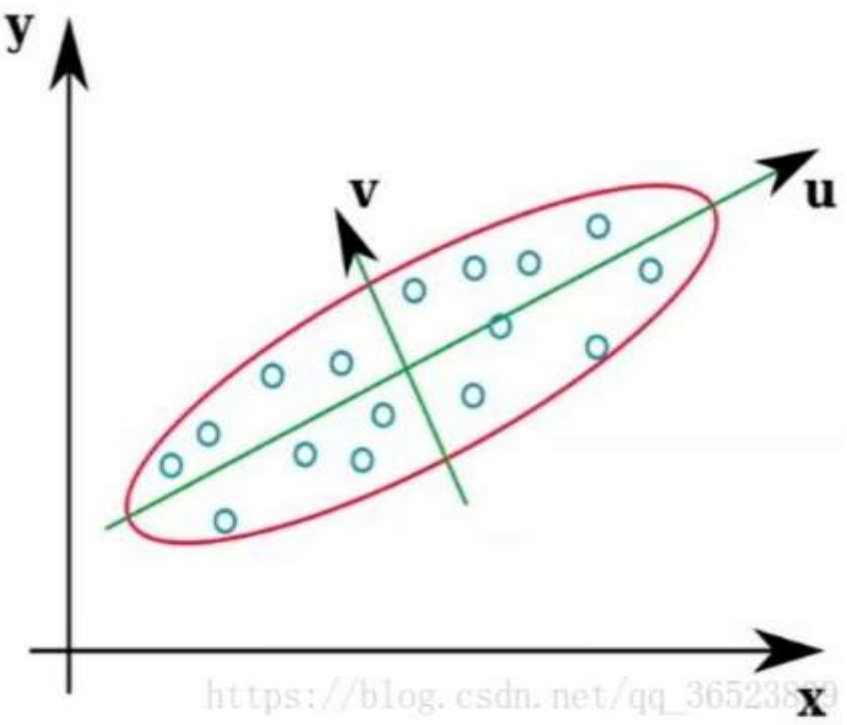
2、 计算样本集的协方差矩阵

3、 对矩阵进行**特征值分解**，获得特征值和特征向量

4、 将特征值按照从大到小的顺序排序，选择其中最大的k个特征值对应的特征向量，**标准化后组成变换矩阵w**

5、对每个样本进行投影变换以获得新（压缩后）的样本集

实际原理就是：移动坐标轴



PCA实例

#调用库的语句：
#from sklearn.decomposition import PCA

官网案例

```
X=np.array([[-1,-1],[-2,-1],[-3,-2],[1,1],[2,1],[3,2]])
pca=PCA(n_components=2)
pca.fit(X)
pca
```

```
PCA(copy=True, iterated_power='auto', n_components=2, random_state=None,
    svd_solver='auto', tol=0.0, whiten=False)
```

```
print(pca.explained_variance_ratio_)
```

```
[0.99244289 0.00755711]
```

PCA主要参数

#fit(X,y=None)算法中的“训练”这一步骤。因为PCA是无监督学习算法，y=None。fit(X)，表示用数据X来训练PCA模型。

函数返回值：调用fit方法的对象本身。比如pca.fit(X)，表示用X对pca这个对象进行训练。

#fit_transform(X)：用X来训练PCA模型，同时返回降维后的数据。

#newX=pca.fit_transform(X)，newX就是降维后的数据。

inverse_transform()

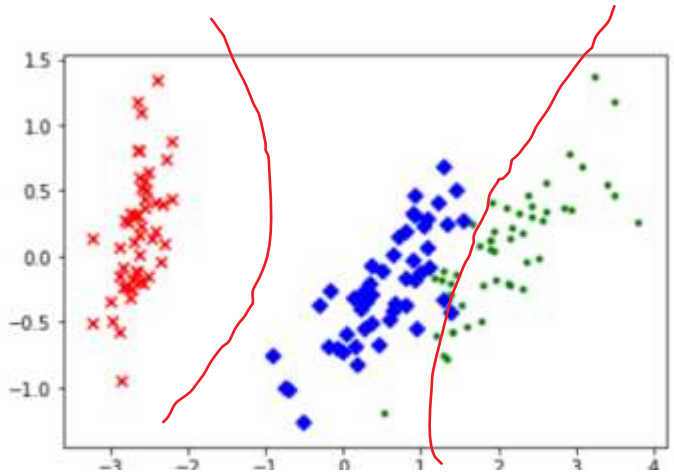
将降维后的数据**转换成原始数据**，X=pca.inverse_transform(newX)

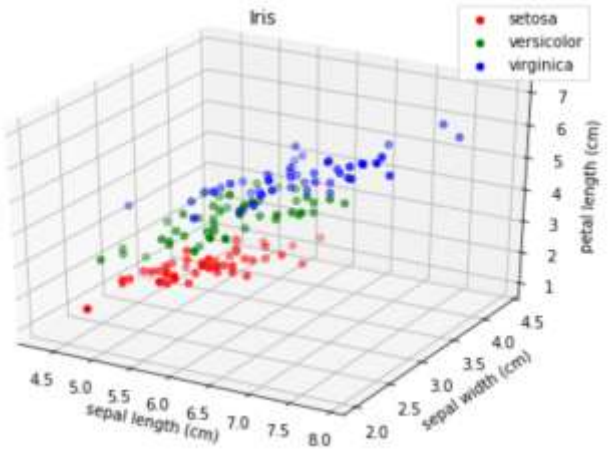
transform(X)

将数据X转换成降维后的数据。当模型训练好后，新输入的数据，都可用transform来降维。

PCA实例介绍

```
[37]: data = load_iris()#以字典形式加载数据
y=data.target#使用y表示数据集中的标签
x=data.data#使用x表示数据集的属性数据
pca=PCA(n_components=2)
#加载PCA算法，设置降维后的维度为2
reduced_x=pca.fit_transform(x)
#对原始数据进行降维，保存在reduced_x中
red_x,red_y=[],[]
blue_x,blue_y=[],[]
green_x,green_y=[],[]
#三类数据点
#按照鸢尾花的类别进行降维处理
for i in range(len(reduced_x)):
    if y[i]==0:
        red_x.append(reduced_x[i][0])
        red_y.append(reduced_x[i][1])
    elif y[i]==1:
        blue_x.append(reduced_x[i][0])
        blue_y.append(reduced_x[i][1])
    else:
        green_x.append(reduced_x[i][0])
        green_y.append(reduced_x[i][1])
#对降维后的数据可视化
plt.scatter(red_x,red_y,c='r',marker='x')
plt.scatter(blue_x,blue_y,c='b',marker='D')
plt.scatter(green_x,green_y,c='g',marker='.')
plt.show()
```





线性判别式分析LDA)

1、LDA方法的定义

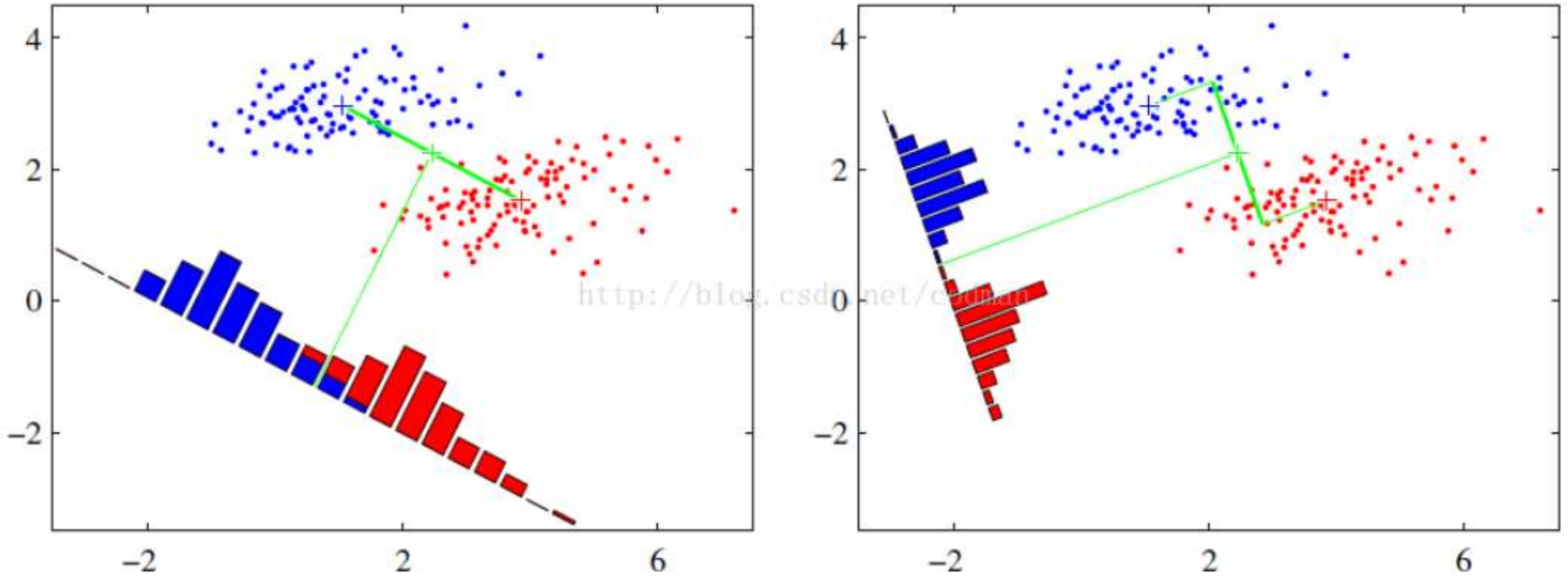
LDA方法是模式识别的经典线性学习算法，也是一种监督学习的降维技术。

2、基本思想

将高维的模式样本投影到最佳鉴别矢量空间，以达到抽取分类信息和压缩特征空间维数的效果，投影后保证模式样本在新的子空间有最大的类间距离和最小的类内距离，即模式在该空间中有最佳的可分离性。

将样本点**投影到低维形成类簇**，即可达到分类效果，也实现了特征降维的效果

线性判别式分析LDA)



PCA 与LDA对比

a) 均可对数据进行降维

b) 均使用了矩阵特征分解的思想

c) 都假设数据符合高斯分布

相同点

不同点

a) LDA有监督，PCA无监督

b) LDA最多降维到K-1的维数，PCA无限制

c) LDA除了降维，还可用来分类

d) LDA选择分类性能最好的投影方向，而PCA选择样本点投影具有最大方差的方向

- LDA在样本分类信息依赖均值而不是方差的时候，比PCA算法较优
- LDA可能过度拟合数据

Scikit_learn

scikit-learn (sklearn) 官方文档中文版

python

ApacheCN

65页 • 2022 Star • 2019年5月26日收录

开始阅读

第五节

模型监控

一、模型调优)

调参方式与工具：

1、工具选择上一般选用GridSearch，

2、调参顺序上，建议是先重要的影响大的参数，后没那么重要的，影响小的参数；

二、模型融合

1、融合的原因：一般来讲，任何一个模型在预测上都无法达到一个很好的结果，这是因为通常来说单个模型无法拟合所有数据，及不具备对所有未知数据的泛化能力，因此需要对多个模型进行融合

2、融合方式：

- **简单融合：**
  - **①分类问题：投票法融合，不考虑单个模型自身的得分；**
  - **②回归问题：假设每个模型权重一致，计算融合结果；**
- **加权融合：基本同上，区别是考虑每个模型自身得分，得分高的权重大**
- **模型进行融合：即将多个单模型的输出作为输入送入到某个模型中，让模型去做融合，通常可以达到最好的效果，但是由于用到了模型，因此要注意过拟合问题；**

模型验证

通过交叉验证对模型性能进行检验，这里通常都是一致的做法，需要注意的是在**时间序列数据预测**上，不能直接随机的划分数据，而是要考虑时间属性，因为很多特征都依赖于时间的前后关系，利用了趋势；

模型持久化

最后，最好将得到的模型持久化到磁盘，方便后续使用、优化时，不需要从头开始；

附加链接

特征降维：
https://www.cnblogs.com/bonelee/p/8632866.html
https://blog.csdn.net/datoutong_/article/details/78813233
方差选择
https://www.sohu.com/a/340983395_654419
特征抽取：
https://www.cnblogs.com/bonelee/p/8632866.html
Pca链接 ：
https://www.jianshu.com/p/69b28d2fa34d

数据，是信息时代的真相！