

机器学习实验 1. scikit-learn 简介

scikit-learn 概述

scikit-learn (简称为 sklearn) 是 Python 的一个开源机器学习模块, 它建立在 NumPy, SciPy 和 matplotlib 模块之上能够为用户提供各种机器学习算法接口, 可以让用户简单、高效地进行数据挖掘和数据分析。Scikit-learn 项目最早为数据科学家 David Cournapeau 于 2007 年发起的 scikits.learn 项目, 且 Scikit 的名字可视为 SciPy Toolkit, 即 SciPy 的第三方扩展。Scikit-learn 大部分都是由 Python 构建, 但还是有很多核心算法是由 Cython 完成而实现更好的效果, 例如支持向量机就是由 Cython 构建。Scikit-learn 目前主要由社区成员自发进行维护, 且专注于构建机器学习领域内经广泛验证的成熟算法。

scikit-learn 的主要功能

scikit-learn 拥有大量可以用于监督和无监督学习的方法。一般来说, 监督学习使用的更多, 因此 scikit-learn 提供了广义线性模型、支持向量机、最近邻算法、高斯过程、朴素贝叶斯、决策树和集成方法等各类监督学习(分类)算法教程, 同时还提供了特征选择、随即梯度下降算法、线性与二次判别分析等在监督学习中非常重要的算法。除了监督学习, 半监督学习中的标签传播算法、无监督学习中的多种聚类算法、以及线性降维和非线性降维等方法在 Scikit-learn 也都有相应的代码支持。此外, 在模型选择中, scikit-learn 包含了交叉验证的使用、估计器超参数的调整、模型评估方法和模型持久化概念等。

部分功能如下图所示:

分类

识别某个对象属于哪个类别

应用: 垃圾邮件检测, 图像识别

算法: SVM, nearest neighbors, random forest, ...
— 示例

回归

预测与对象相关联的连续值属性

应用: 药物反应, 股价

算法: SVR, ridge regression, Lasso, ... — 示例

聚类

将相似对象自动分组

应用: 客户细分, 分组实验结果

算法: k-Means, spectral clustering, mean-shift, ...
— 示例

降维

减少要考虑的随机变量的数量

应用: 可视化, 提高效率

算法: PCA, feature selection, non-negative matrix factorization.
— 示例

模型选择

比较, 验证, 选择参数和模型

目标: 通过参数调整提高精度

模型: grid search, cross validation, metrics.
— 示例

预处理

特征提取和归一化

应用: 把输入数据(如文本)转换为机器学习

算法可用的数据
算法: preprocessing, feature extraction.
— 示例

估计器与转化器

scikit-learn 中的大部分函数可以归为估计器(Estimator)和转化器(Transformer)两类。

估计器(Estimator)其实就是模型, 它用于对数据的预测或回归。基本上估计器都会有以下几个方法:

- `fit(x,y)`: 传入数据以及标签即可训练模型, 训练的时间和参数设置, 数据集大小以及数据本身的特点有关
- `score(x,y)`: 用于对模型的正确率进行评分(范围 0-1)。但由于在不同的问题下, 评判模型优劣的标准不限于简单的正确率, 可能还包括召回率或者是查准率等其他的指标, 特别是对于类别失衡的样本, 准确率并不能很好的评估模型的优劣, 因此在对模型进行评估时, 不要轻易的被 `score` 的得分蒙蔽。
- `predict(x)`: 用于对数据的预测, 它接受输入, 并输出预测标签, 输出的格式为 `numpy` 数组。我们通常使用这个方法返回测试的结果, 再将这个结果用于评估模型。

转化器(Transformer)用于对数据的处理, 例如标准化、降维以及特征选择等等。同与估计器的使用方法类似:

- `fit(x,y)`: 该方法接受输入和标签, 计算出数据变换的方式。
- `transform(x)`: 根据已经计算出的变换方式, 返回对输入数据 `x` 变

换后的结果（不改变 x ）

- `fit_transform(x,y)`: 该方法在计算出数据变换方式之后对输入 x 就地转换。

scikit-learn 函数的具体调用请见实验指导。