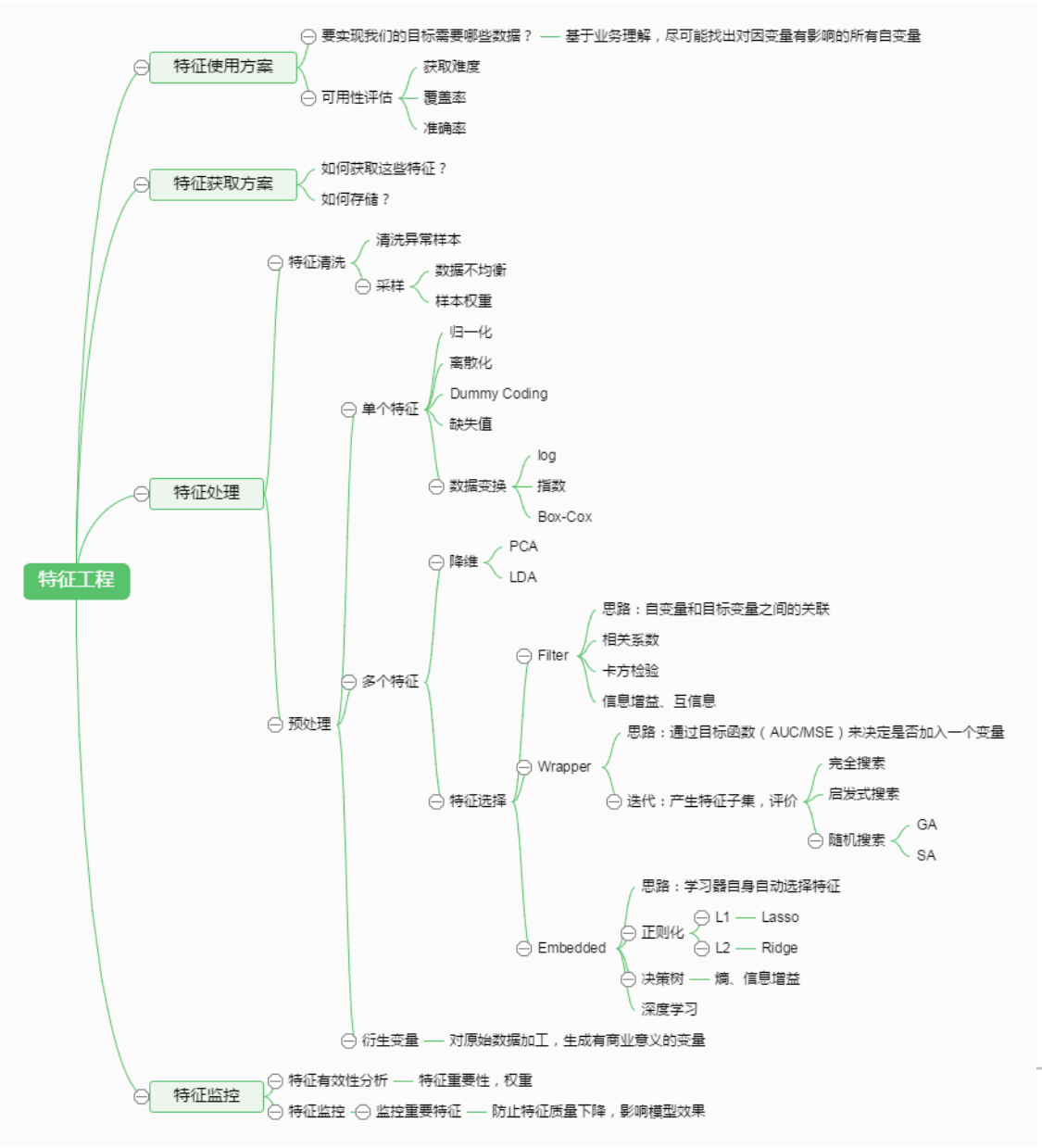
Weekend1：特征工程初步

CSDN学院

机器学习任务的一般步骤

CSDN
不 止 于 代 码

- 确定特征
  - 可能是最重要的步骤!（收集训练数据）
- 确定模型
  - 目标函数
- 模型训练：根据训练数据估计模型参数
  - 优化计算
- 模型评估：在校验集上评估模型预测性能
- 注意：Deep Learning可学习特征
  - 对视觉/语音等非结构化数据尤其重要


特征工程
特征使用方案
要实现我们的目标需要哪些数据？ — 基于业务理解，尽可能找出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
如何获取这些特征？
如何存储？
特征处理
特征清洗
清洗异常样本
采样
数据不均衡
样本权重
预处理
单个特征
归一化
离散化
Dummy Coding
缺失值
数据变换
log
指数
Box-Cox
多个特征
降维
PCA
LDA
特征选择
Filter
思路：自变量和目标变量之间的关联
相关系数
卡方检验
信息增益、互信息
Wrapper
思路：通过目标函数（AUC/MSE）来决定是否加入一个变量
迭代：产生特征子集，评价
完全搜索
启发式搜索
随机搜索
GA
SA
Embedded
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 — 对原始数据加工，生成有商业意义的变量
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

大纲

- 数据探索
- 数据预处理
- 特征选择（下周）

数据探索

CSDN
不 止 于 代 码

- 数据探索有助于选择合适的数据预处理和建模方法。
- 数据质量分析
  - 缺失值
  - 异常值
- 特征分布特性分析
  - 统计量
  - 直方图
- 特征之间相关性分析

缺失值分析

CSDN
不 止 于 代 码

- 统计缺失样本数目及缺失率

- 缺失值处理
  - 删除含有（1个或多个）缺失值的样本（行）
  - 删除缺失值太多的特征（列）
  - 对缺失值进行插补
  - 不处理（有些学习器可处理数据缺失情况，如xgboost）

缺失值填补

CSDN
不 止 于 代 码

- 由于各种原因，实际应用中总是存在一些缺失值，通常表示为NaN/NaT（日期型变量）。
- scikit-learn的类Imputer提供一些常见填补方法
  - 均值mean（默认方法）
  - 中位数median
  - 众数most_frequent
- pandas库的fillna函数也可以处理缺失值，而且更加灵活，但是重用性较弱

```
train.loc[:, "Alley"] = train.loc[:, "Alley"].fillna("None")
```

缺失值插补

CSDN
不 止 于 代 码

插补方法	描述
均值 / 中值 / 众数插补	用该属性的均值 / 中值 / 众数进行插补 通常数值型变量用均值 / 中值、类别型变量用众数插补
固定值插补	根据背景知识用某个常量进行插补
最近邻插补	寻找最相似的样本，用该样本的属性值插补 可采用相关系数矩阵来确定哪个变量（X）与缺失值所在变量（Y）最相关。然后把所有样本按X的取值大小进行排序，变量Y的缺失值可以用排在缺失值前的那个样本的数据来代替。
回归方法	选择若干个预测缺失值的自变量X，然后建立回归方程估计缺失值
插值法	利用该变量已有数据建立合适的插值函数进行插补

例：拍拍贷“魔镜风控系统”

- 根据用户历史行为（约400维特征）预测用户在未来6个月内会逾期还款的概率

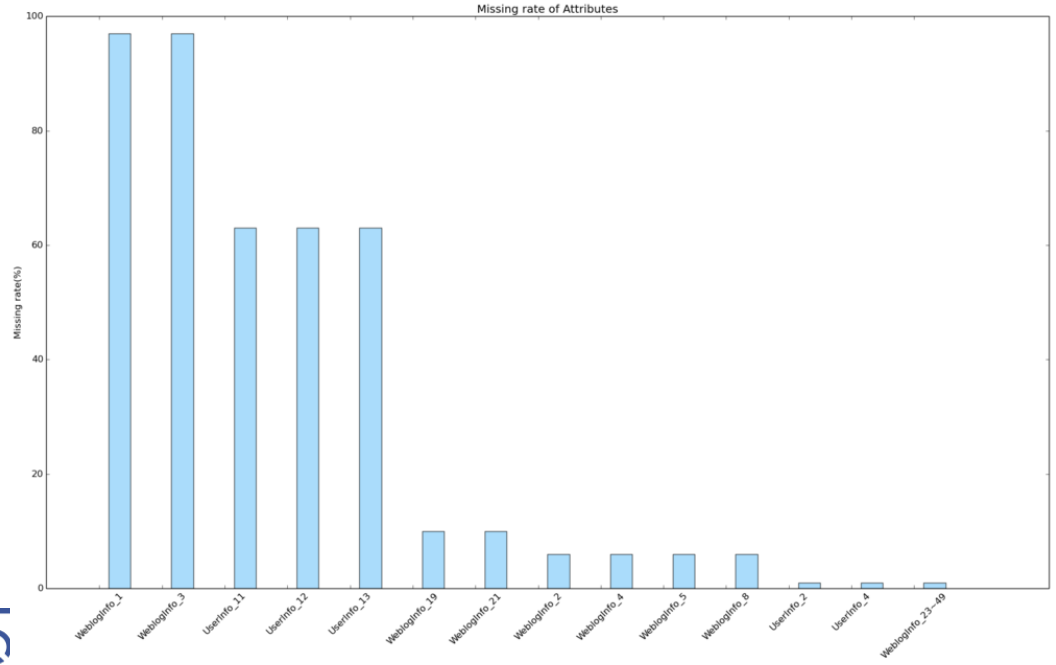
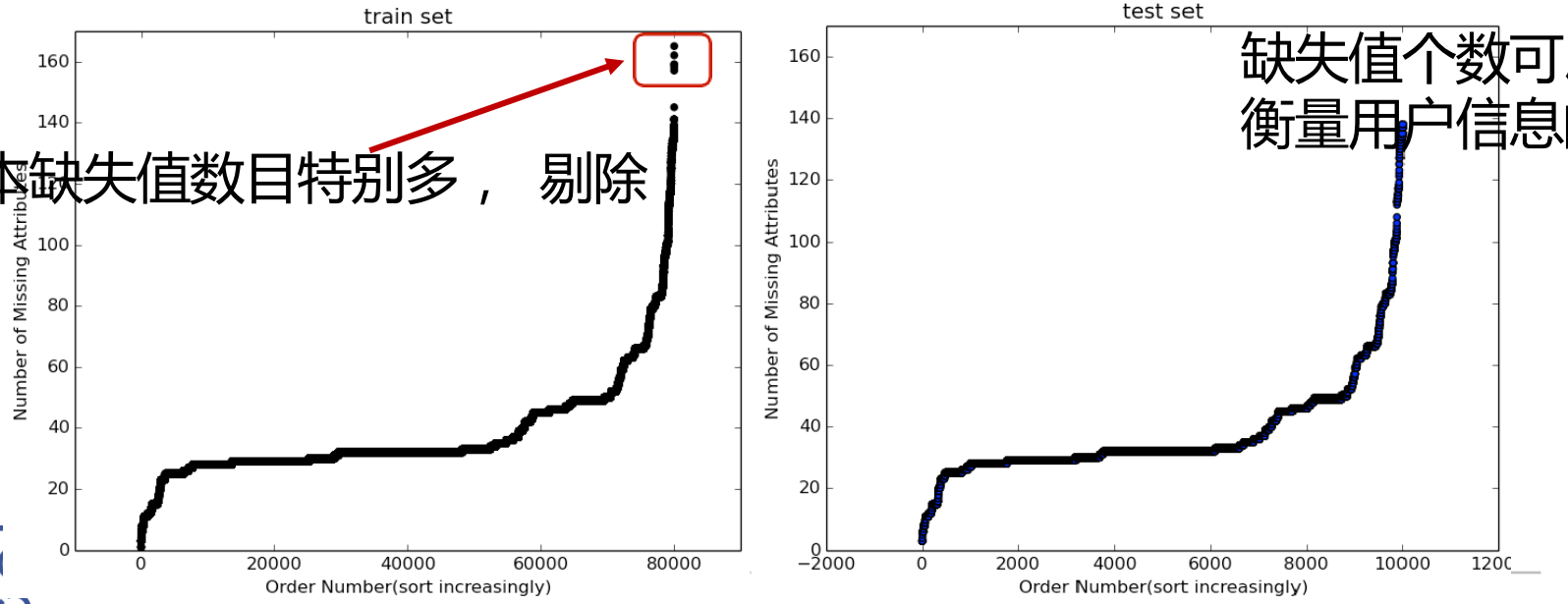


图 1.属性缺失比率

- WeblogInfo_1 和 WeblogInfo_3 的缺失值比率为 97%，这两列属性基本不携带有用的信息，直接剔除。
- UserInfo_11、UserInfo_12 和 UserInfo_13 的缺失值比率为 63%，这三列属性是类别型的，可以将缺失值用-1 填充，相当于“是否缺失”当成一种新的类别。
- 其他缺失值比率较小的数值型属性用中值填充。

例：拍拍贷“魔镜风控系统”

- 按行统计每个样本的属性缺失值个数，将缺失值个数从小到大排序
  - 序号为横坐标， 缺失值个数为纵坐标



样本缺失值数目特别多， 剔除

缺失值个数可以作为一个特征，衡量用户信息的完善程度。

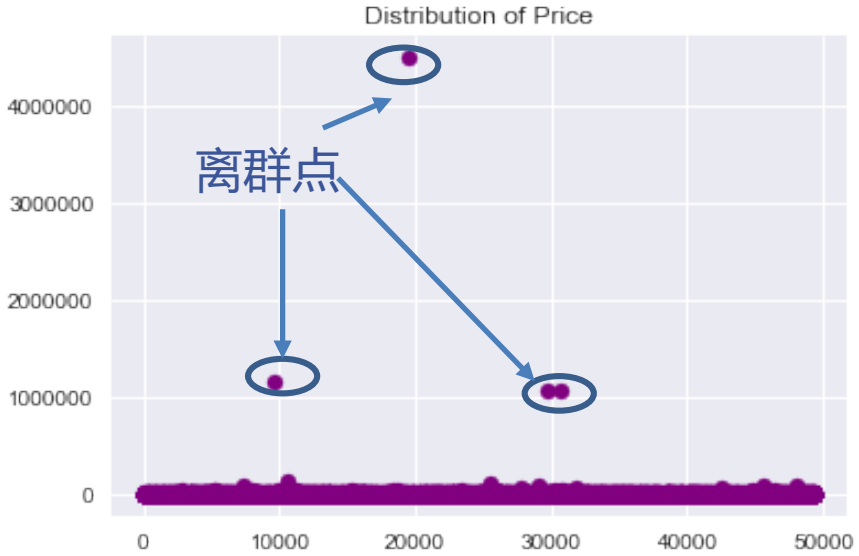
图 2.样本属性缺失个数

异常值分析

CSDN
不 止 于 代 码

- 异常值，已被称为离群点（outlier），指样本的某个属性值明显偏离其余样本的观测值。通常认为这些样本点是噪声，对模型有坏影响。

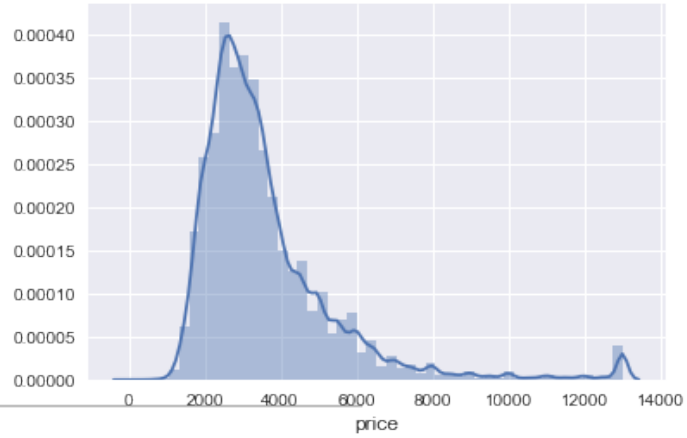
- 发现异常值
  - 最大值 / 最小值分析
  - 3σ原则
  - 箱型图
  - 直方图（尾巴）
  - 散点图（散点图上孤立的点）
  - 分位数（ 0.5%-99.5% 分位数以外）
- 异常值处理：类似缺失值处理
  - 注意分析异常值产生原因



特征分布分析

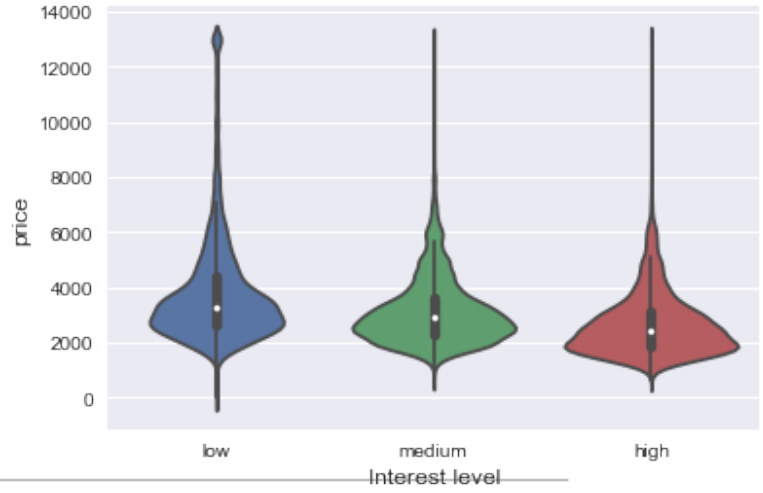
CSDN
不 止 于 代 码

- 直方图：每个取值在数据集中出现的次数，可视为概率函数（pdf）的估计（seaborn可视化工具比较简单）
  - import seaborn as sns
  - %matplotlib inline（ seaborn 是基于matplotlib 的）
  - sns.distplot(train.price.values, bins=50, kde=True)
- 核密度估计
  - Kernel Density Estimation, KDE
  - 对直方图的加窗平滑

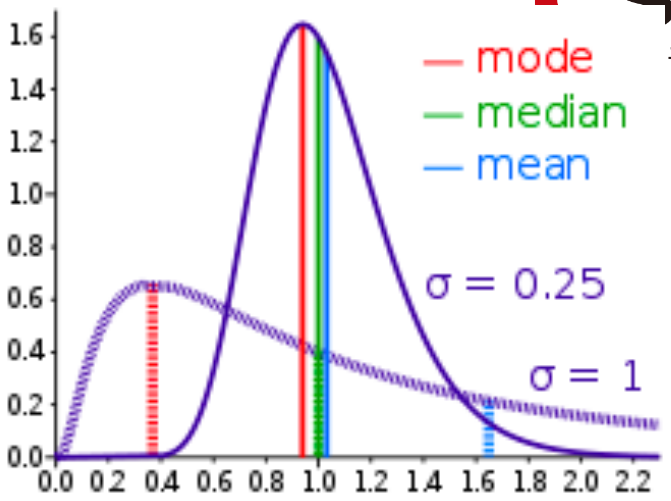


直方图（cont.）

- 在分类任务中，我们关心不同类别的特征分布
  - 核密度估计
  - order = ['low', 'medium', 'high']
  - sns.violinplot(x='interest_level', y='price', data=train, order = order)



表示集中趋势的统计量

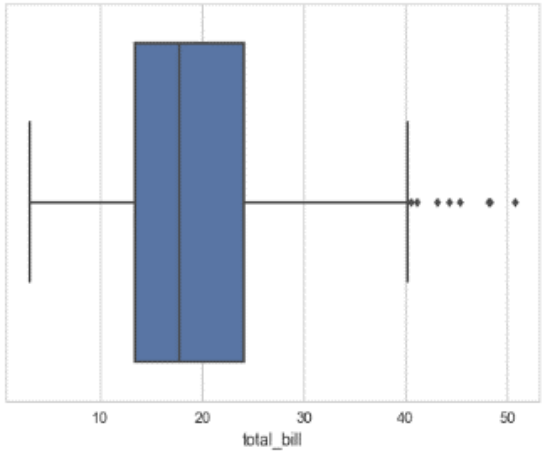


- 均值
- 中值：在一组排好序数据中
  - 数据数量为奇数，则中值为中间的那个数；
  - 如果数据数量为偶数，则中值为中间的那两个数值的平均值。
- 众数: 出现概率最大的地方

表示散布程度的统计量

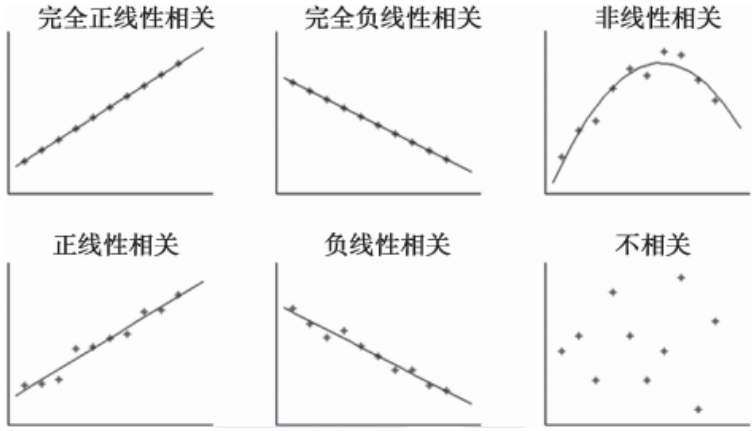
CSDN
不 止 于 代 码

- 方差
- 四分位数间距（Interquartile Range）:25%分位数到75%分位数之间的区间的宽度
- IQR在箱体图boxplot（ seaborn.boxplot）中用到：分布的图形概述
  - 长方形为IQR
  - 中间的线为中值
  - 两头的虚线：1.5 IQR



相关性

- 相关性可以通过计算相关系数或打印散点图来发现
- 相关系数：两个随机变量x,y之间的线性相关程度
  - $r = \frac{\sum_{i=0}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=0}^{n}(x_i-\bar{x})^2 \sum_{i=0}^{n}(y_i-\bar{y})^2}}$
  - $-1 \le r \le 1$
  - 通常$|r| > 0.5$，认为两者相关性比较强
  - $\begin{cases} r = 0 & \text{完全不线性相关} \\ r > 0 & \text{正相关} \\ r < 0 & \text{负相关} \end{cases}$
  - 不线性相关并不代表不相关，可能高阶相关，如$y = x^2$

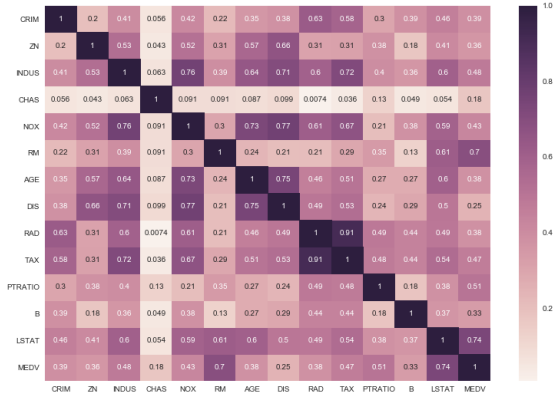


相关性(cont.)

CSDN
不 止 于 代 码

- 我们希望特征与标签强相关
  - 分类直方图可以从某种程度上看出特征与标签的相关性：不同类别的直方图差异大

- 特征与特征之间强相关的话意味着信息冗余
  - 可以两个特征可以只保留一个特征
  - 或采用主成分分析（PCA）等降维
  - 亦可让模型自动进行特征选择



波士顿房价数据相关系数热图

大纲

CSDN
不 止 于 代 码

- 数据探索
- **数据预处理**
- 特征选择

数据预处理

CSDN
不 止 于 代 码

- from sklearn.preprocessing import …
  - 数据取值范围缩放
    - 数据标准化（ Standardization ）
    - 数据归一化（ Scaling ）
    - 数据正规化（ Normalization ）
  - 特征编码
    - 二值化（ Binarizer ）
    - 多项式编码（ PolynomialFeatures ）
    - 标签编码（LabelEncoder）
    - 独热编码（OneHotEncoder）
    - 数值特征离散化

类	功能	说明
StandardScaler	无量纲化	标准化，基于特征矩阵的列，将特征值取值范围标准化（类似对正态分布标准化）
MinMaxScaler	无量纲化	区间缩放，基于最大最小值，将特征值转换到[0, 1]区间上
Normalizer	归一化	基于特征矩阵的行，将样本向量转换为“单位向量”
Binarizer	二值化	基于给定阈值，将定量特征按阈值划分
OneHotEncoder	哑编码	将定性数据编码为定量数据
Imputer	缺失值计算	计算缺失值，缺失值可填充为均值等
PolynomialFeatures	多项式数据转换	多项式数据转换
FunctionTransformer	自定义单元数据转换	使用单变元的函数来转换数据

特征编码

CSDN
不 止 于 代 码

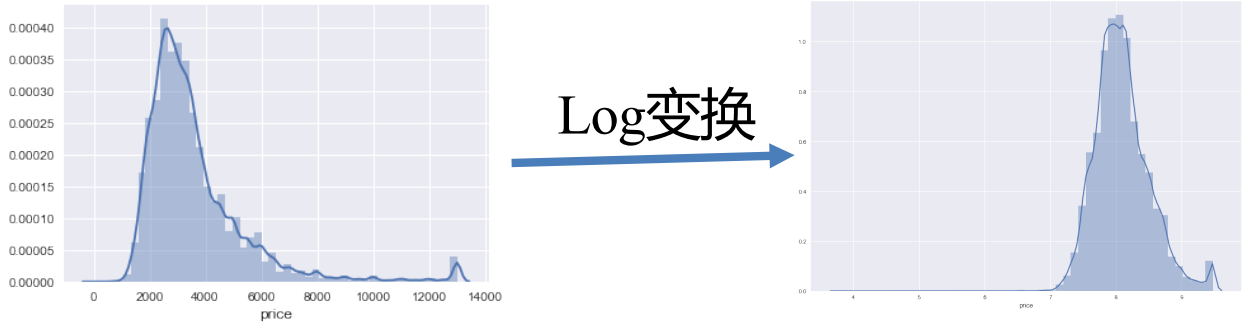
- 数值型
- 类别型
- 时间类
- 文本型

http://scikit-learn.org/stable/modules/preprocessing.html
http://scikit-learn.org/stable/modules/classes.html#module-sklearn.feature_extraction

数值型特征

CSDN
不 止 于 代 码

- 有时候数值特征并不能直接适用于线性模型，可以选择使用log(x+1)变换来处理我们的数值特征，其中x表示数值特征
  - 检查数据偏斜度（skewness），并以此来衡量数据分布的形状。
  - 当使用回归方法时，如果目标变量出现偏斜，则有必要对目标变量进行对数变换（log-transform）。通过对数变换，可以改善数据的线性度。




```
train.SalePrice = np.log1p(train.SalePrice)
```

数值型特征

CSDN
不 止 于 代 码

- 还可以根据需要进行多项式扩展数值特征
- 例如，4个特征，度为2的多项式转换公式如下：

```
from sklearn.preprocessing import PolynomialFeatures

#多项式转换
#参数degree为度，默认值为2
PolynomialFeatures().fit_transform(iris.data)
```

$$(x'_1,x'_2,x'_3,x'_4,x'_5,x'_6,x'_7,x'_8,x'_8,x'_{10},x'_{11},x'_{12},x'_{13},x'_{14},x'_{15})$$
$$= (1, x_1, x_2, x_3, x_4, x_1^2, x_1 * x_2, x_1 * x_3, x_1 * x_4, x_2^2, x_2 * x_3, x_2 * x_4, x_3^2, x_3 * x_4, x_4^2)$$

数值特征离散化

CSDN
不 止 于 代 码

- 有时候将数值型属性转换成类别表示更有意义
  - 如电商数据中对年龄分组：预测用户是否拥有某款衣服，可将年龄分布划分成1-10,11-18,19-25,26-40 …
  - 如房屋到街道的距离，分为远、中、近
- 了解领域背景知识，确定特征划分成分区有意义
  - 所有落入一个分区的数值能呈现出共同的特征
  - 在实际应用中，需要仔细考虑分区的粒度，防止过拟合
- 也可以采用聚类技术实现分组，后续我们会学习更复杂的技术如GBDT实现特征的组合及离散化
- 注意：如果数据的准确率很重要的话，离散化并不适合

► 类别型特征（ categorical attributes ）

CSDN
不 止 于 代 码

- 有些模型只能处理数值型数据。如果给定的数据是不同的类型，必须先将数据变成数值型。
- 1. 用0与1来代表样本是否有某种特性
  - 如房屋是否有中央空调（0代表没有，1代表有）
- 2. LabelEncoder：用有序数字对对不连续的数字或者文本进行编号
  - 如用数字表示房屋质量：分别用数字1、2、3表示质量差、中等、好
  - 存储量小

► 类别型特征（ categorical attributes ）

CSDN
不 止 于 代 码

- 3. 如不希望有序数关系，采用独热编码/哑编码（OneHotEncoder）：将类别型整数输入从1维→K维的稀疏编码
  - 统计该特征上有多少类，就设置几维（ K ）的向量
  - 输入必须是数值型数据（对字符串输入，先调用LabelEncoder变成数字，再用OneHotEncoder ）
  - 存储要求高，通常要求K <10（低基数，low-cardinality ）
- 4. 对高基数（high-cardinality）类别型特征：通常有成百上千个不同的取值（如邮政编码、街道名称…）
  - 可先降维 / 聚类： 1 维 → K维，K为聚类的类别数
  - 或采用均值编码：在贝叶斯的架构下，利用标签变量，有监督地确定最适合特定特征的编码方式

Pandas中的get_dummies()亦可进行one-hot编码

均值编码

CSDN
不 止 于 代 码

- 均值编码：将特征每个可能的取值k(共有K种取值)，编码为它对应的标签取c值的概率，即$p(y=c|x=k)$
  - 有监督编码方式，可用于分类和回归
- 以分类问题为例，设标签取值共有C类，则编码特征为C-1维
  - $\sum_{c=1}^{C} p(y=c|x=k)=1$,所以少一个自由度
  - 后验概率$p(y=c|x=k)$估计：
  - $p(y=c|x=k)=\frac{p(y=c,x=k)}{p(x=k)}=\frac{(y=c,且x=k)的样本数量}{(x=k)的样本数量}$

Reference: Mean Encoding: A Preprocessing Scheme for High-Cardinality Categorical Features
http://helios.mm.di.uoa.gr/~rouvas/ssi/sigkdd/sigkdd.vol3.1/barreca.pdf

均值编码（cont.）

- 真正编码采用的是先验概率$p(y=c)$和后验概率$p(y=c|x=k)$的凸组合估计：
  - $\hat{p}=\lambda p(y=c)+(1-\lambda)p(y=c|x=k)$
  - 其中先验概率$p(y=c)=\frac{(y=c)\text{的样本数量}}{\text{总样本数量}}$
- 组合权重λ：特征类别在训练集内出现的次数越多，后验概率的可信度越高，其权重越大
  - $\lambda(n)=\frac{1}{1+e^{n-k/f}}$
  - 输入参数n：特征类别在训练集中出现的次数

参考代码：https://zhuanlan.zhihu.com/p/26308272

例：均值编码

- 亦被称为直方图映射
- 优点：将特征与目标关联起来

性别	年龄							爱好
男	21	…						足球
男	48	…						散步
女	22	…						看电视剧
男	21	…						足球
女	30	…						看电视剧
女	50	…						散步

统计“性别与爱好的关系”，
性别有“男”、“女”
爱好有三种，表示成向量 [散步、足球、看电视剧]，
分别计算男性和女性中每个爱好的比例得到：男[1/3, 2/3, 0]，
女[0, 1/3, 2/3]

日期型特征

CSDN
不 止 于 代 码

- 时间型特征既可以看做连续值（持续时间、间隔时间），也可以看做离散值（星期几、几月份）

- 日期特征：年月日
- 时间特征：小时分秒
- 时间段：早中晚
- 星期，工作日 / 周末
- 特定节假日：如双11、618

时区：假如你的数据源来自不同的地理位置，别忘了利用时区将数据标准化

文本型特征

CSDN
不 止 于 代 码

- 文本型特征无法直接送到学习器，需先进行特征提取
- Scilit-learn提取常用的简单文本特征提取，更多文本特征提取可采用Natural Language Toolkit（NLTK ，http://www.nltk.org）
- 完整的NLP技术已超出本课程范围，下面我们讨论基于scikit learn的简单文本特征提取方法

文本型特征

CSDN
不 止 于 代 码

- 词向量（ 1-gram模型）：即词袋模型，统计词出现的次数（词频）
  - CountVectorizer
- n-gram模型：将词或扩展为n个词语组成的短语

文本型特征

CSDN
不 止 于 代 码

- TF-IDF特征：
  - **TF（ Term Frequency ）**：衡量一个term在文档中出现得有多频繁
  - **IDF（ Inverse Document Frequency ）**： 衡量一个term有多重要
    - 有些词出现的很多，但是并没什么作用，‘ is‘，‘ the‘，‘ and‘ …
    - 将罕见的词的重要性调高高， 常见词的重要性调低
  - TF(t) = (词t在当前文中出现次数) / (当前文档中出现的单词的总次数)
  - IDF(t) = ln(总文档数/ 含t的文档数)
  - TF-IDF(t) = TF(t) * IDF(t)

sklearn.feature_extraction.text.TfidfTransformer(*norm='l2', use_idf=True, smooth_idf=True, sublinear_tf=False*)

文本型特征

CSDN
不 止 于 代 码

- 可用词云（wordcloud）可视化
  - 文本词频统计函数，自动统计词的个数，以字典形式内部存储，在显示的时候词频大的词的字体更大
  - from wordcloud import WordCloud
  - wordcloud = WordCloud(background_color='white', width=600, height=300, max_font_size=50, max_words=40)
  - wordcloud.generate(text)

Rent List Inquries任务中的features特征

案例分析

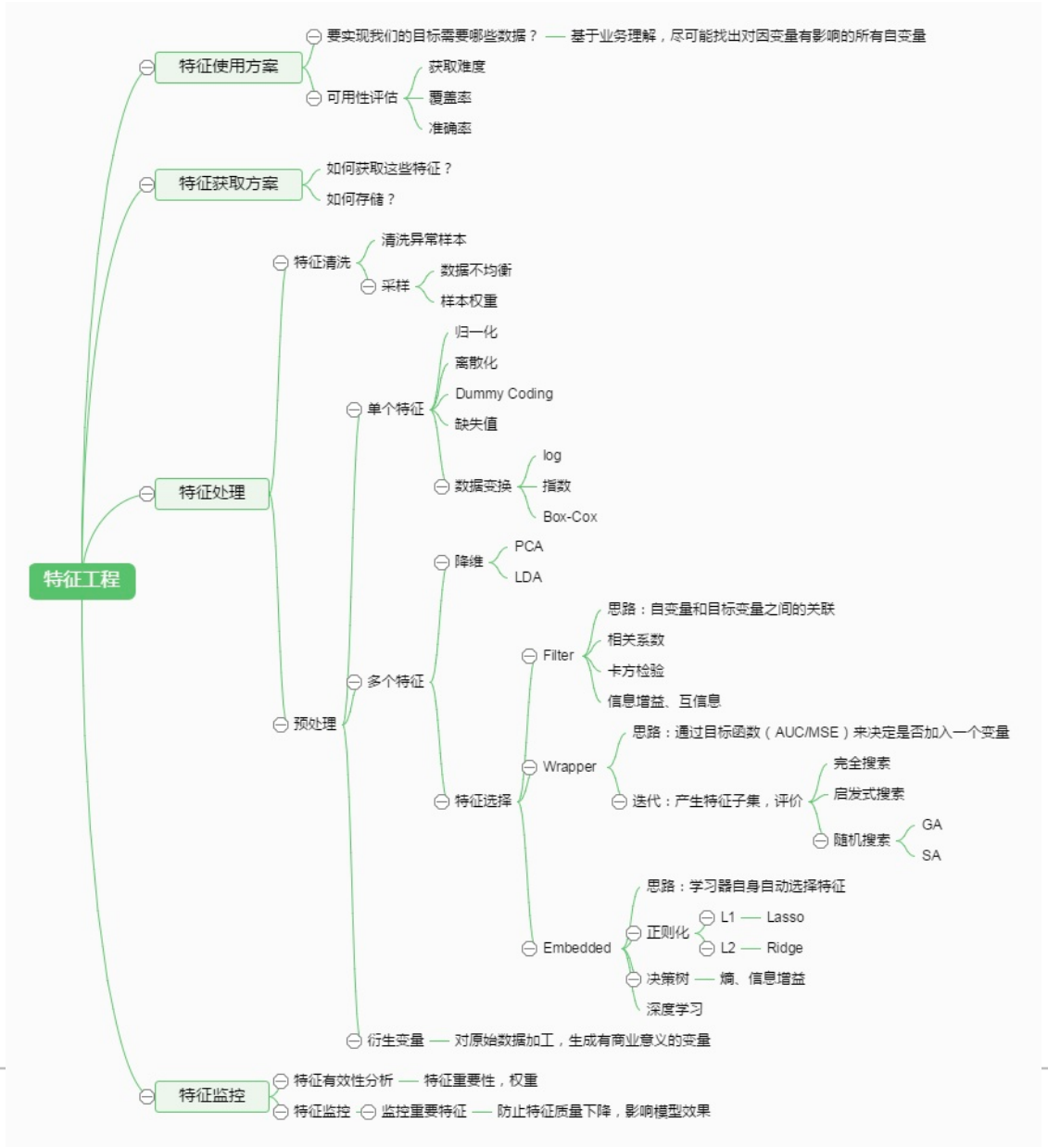
CSDN
不 止 于 代 码

- Two Sigma Connect: Rental Listing Inquiries（纽约市公寓租赁）
  - 第三周分类作业的数据集

特征工程小结

CSDN
不 止 于 代 码

- 如果知道数据的物理意义（领域专家），可能可以设计更多特征
  - 如Higgs Boson任务中有几维特征是物理学家设计的，还有些有高能物理研究经验的竞赛者设计了其他一些特征
  - 如房屋租赁任务中，利用常识可设计出一些特征
- 如果不是领域专家，一些通用的规则：
  - 字符串型特征：Label编码
  - 时间特征：年月日、时间段（早中晚）…
  - 数值型特征：加减乘除，多项式，log, exp
  - 低基数类别特征：one-hot编码
  - 高基数类别特征：先降维，再one-hot编码；均值编码


特征工程
特征使用方案
要实现我们的目标需要哪些数据？ — 基于业务理解，尽可能找出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
如何获取这些特征？
如何存储？
特征处理
特征清洗
清洗异常样本
采样
数据不均衡
样本权重
预处理
单个特征
归一化
离散化
Dummy Coding
缺失值
数据变换
log
指数
Box-Cox
多个特征
降维
PCA
LDA
特征选择
Filter
思路：自变量和目标变量之间的关联
相关系数
卡方检验
信息增益、互信息
Wrapper
思路：通过目标函数（AUC/MSE）来决定是否加入一个变量
迭代：产生特征子集，评价
完全搜索
启发式搜索
随机搜索
GA
SA
Embedded
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 — 对原始数据加工，生成有商业意义的变量
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

CSDN
不 止 于 代 码

THANK YOU

AI100