特征选择方法总结

特征选择综述(较全)

Li J, Cheng K, Wang S, et al. Feature selection: A data perspective[J]. ACM computing surveys (CSUR), 2017, 50(6): 1-45.

https://dl.acm.org/doi/pdf/10.1145/3136625

1. 基于特征本身属性(Filter 方法)

1.1 单个特征硬性指标

缺失率占比

常数值占比

方差(矩条件筛选)筛选

1.2 特征之间相关性

VIF, Pearson 相关系数,chi2

特征维度较小时适用，高维特征缺失较多时往往相关性不大。忽视非线性相关。

https://zhuanlan.zhihu.com/p/56793236

PCA, UMAP

降维，投影剔除，会改变原有变量的分布

https://zhuanlan.zhihu.com/p/48502371

PFA （改进版 PCA，返回值不改变原有特征）

https://pypi.org/project/principal-feature-analysis/

https://github.com/LauritzR/Parallel-Principal-Feature-Analysis

Other Filter + Fisher-score

先用其他 filter 方法对每个特征计算筛选，之后用 Fisher 值筛选，fisher 值考虑了好坏样本和本特征与其他所有特征之间的距离。

扩展方法(组合优化问题改进) https://arxiv.org/ftp/arxiv/papers/1202/1202.3725.pdf

其他： ReliefF, Laplacian score, Hilbert Schmidt Independence Criterion (HSIC), Trace Ratio criterion

1.3 特征区分度

KS （分箱计算好坏样本的区分度）

WOE,IV（分箱计算好坏样本的区分度）

1.4 特征稳定性

PSI

特征稳定性指标，距离一段时间检验

Co-variate shifting (去除分布变化较大的特征)

在数据集 a 和数据集 b（如，上半年和下半年）中随机采样（采样数量需要一致）并混合形成新的训练数据，增加一个新维度标签，取值取决于数据来源（数据集 a 标识 1,数据集 b 标识 0）。

上面得到了一个新的训练集. 将这个训练集的一部分数据(如 80%)用来训练模型(LR 等), 剩下的数据(如 20%)用来测试模型的性能。计算模型在测试集上的 AUC，如果指标较大（比如大于 0.8），便可判定发生了分布变化。根据上一步的 LR 的特征重要性来删除特征。

异常检测

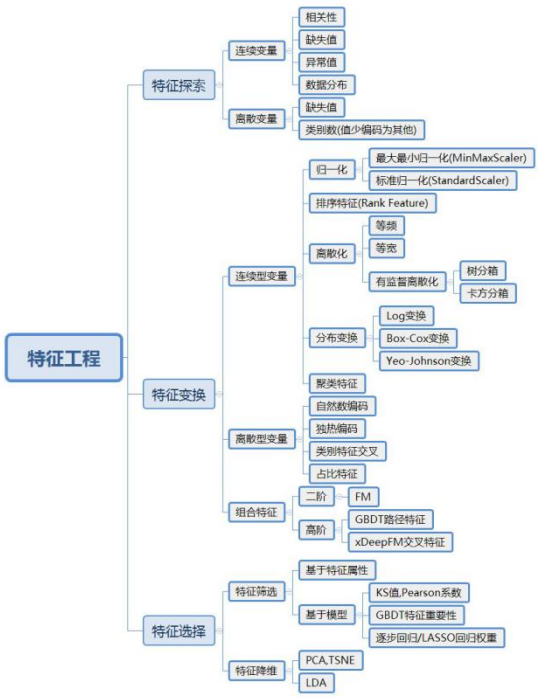
剔除异常样本对特征选择的干扰

iForest

孤立点会最先被从树的分类中分出，越易于分类的点越异常。

https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html

https://zhuanlan.zhihu.com/p/25040651



https://www.01caijing.com/article/279830.htm

2. 基于模型选择（树模型：特征对 y 的有用程度）

2.1 特征重要性

直接选择

直接根据树模型 XGB,lgb 的特征重要性计算结果保留大于 0 的特征，可以迭代使用。例子：参数在子树分裂时，用到特征的次数。

feature_imp = xgbmodel.get_booster().get_score(importance_type='weight')

特征相似性可能会影响 xgb 的分裂，如果两个特征非常相关，可能出现在 100 次抽样中，两个特征分别出现 10 次，而不是每个 20 次，即特征重要性被稀释的现象。

可能出现的问题 n_estimator, depth, colsample_bytree=1, subsample=1 参数如何选择

Boruta 方法

Boruta 也是一种基于随机森林的特征选择算法，属于 Permutation Importance 方法。

将原始特征的值随机打乱顺序生成“影子特征”集合作为基准与原始特征进行比较，从而确定每个特征的重要性。随机生成的“影子特征”一般而言重要性都很低，Boruta 算法通过不断迭代，剔除在这个过程中重要性甚至低于“影子特征”的无效特征。稳健性更好

Python 包和例子
https://github.com/scikit-learn-contrib/boruta_py
英文文章
https://pdfs.semanticscholar.org/ecc2/ca3150dc4d4d8dceedab244114f191e05742.pdf
中文例子讲解
https://zhuanlan.zhihu.com/p/392325156

https://tan800630.medium.com/%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92%E4%B8%AD%E7%9A%84%E7%89%B9%E5%BE%B5%E7%AF%A9%E9%81%B8-boruta-faa536bee5bd

树模型缺点：
树模型无法在没见过的特征值出现时进行分裂。有别于线性回归，树模型无法对从未出现过的更大值或更小值的区域进行外推。
https://medium.com/nerd-for-tech/extrapolation-is-tough-for-trees-tree-based-learners-combining-learners-of-different-type-makes-659187a6f58d

缓解：删除明显超过区间的样本，Ensemble 使用多种模型

2.2 基于神经网络的选择

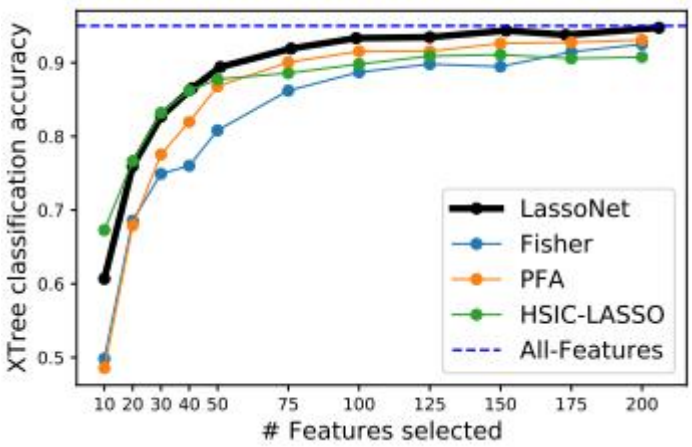
在深度学习兴起之后，根据数据类型的不同，实际上那些神经网络的模型在对数据进行分类提取时，其实已经隐形地进行了特征筛选和非线性组合，因此特征选择可能在 nlp、图像分类等方面并不显得那么重要，这点可能有别于表格型数据。在数据量和维度足够大时，神经网络可能总能学到东西。但这里的基本假设是训练验证测试集的分布不能变化过大，而当出现变化时，神经网络如何人为调整可能又是另外个问题。
这里论文例子是对于图像数据的特征筛选，对于表格型数据的使用还有待验证。

LassoNet

用 lasso 正则化的思想限制 resnet 中的参数，并用神经网络做特征提取。结合 lasso 的线性筛选和神经网络的非线性特质。

https://github.com/lasso-net/lassonet

https://arxiv.org/pdf/1907.12207.pdf



Pip install lassonet

基于深度学习 torch 这个包，初步验证发现训练慢，对设备要求比较高。

3. 特征选择方法的使用

1 对于单个特征的硬性指标可以设置稍微宽松的阈值，直接筛选。

2 其他指标可以结合模型特征重要性指标，筛选交集：这些指标表现都特别差的。

3 可以递归使用，如根据 KS 值删除后 xgb 再 KS 再 xgb，选取目标值 auc/ks 可以满意时的特征数目。(https://machinelearningmastery.com/rfe-feature-selection-in-python/)

4 注意数据本身的结构，对不同结构数据采用不同方法，如图数据，图像数据，文字数据等。

5 将特征处理写成包的形式。

sklearn.feature_selection 包
传统方法：chi2,f-regression 等

feature_selector *包 (可视化做的还可以)*
https://zhuanlan.zhihu.com/p/49479702
https://github.com/WillKoehrsen/feature-selector

skfeature *包 (包括传统方法和新方法，比较多算法，有待研究)*
https://github.com/jundongl/scikit-feature/
https://jundongl.github.io/scikit-feature/algorithms.html

6 单纯根据特征的性质进行筛选属于无监督学习的筛选，往往对噪声比较敏感，可以切分不同时间段或人为加入噪声来进行测试。

总结

待研究的方法：一是适合更高维度以及稀疏数据的更快速的方法；二是能够完善树模型的缺点，以便根据树模型本身选择特征。另外，基于深度学习的特征选择的可靠性有待探索。

特征选择方法本身具有不同的优缺点，模型如树模型本身具有优缺点，其他方面如样本的筛选和样本的数据集划分也会对特征选择干扰。因此，要结合以上优缺点使用。