随机森林算法实现收入层次预测

实验目标

通过本案例的学习和课后作业的练习：

1. 了解随机森林算法的基本思想；
2. 能够使用SKlearn实现随机森林算法。

你也可以将本案例相关的 ipynb 学习笔记分享到 AI Gallery Notebook (https://marketplace.huaweicloud.com/markets/aihub/notebook/list/) 版块获得成长值 (https://marketplace.huaweicloud.com/markets/aihub/article/detail/?content_id=9b8d7e7a-a150-449e-ac17-2dcf76d8b492)，分享方法请查看此文档 (https://marketplace.huaweicloud.com/markets/aihub/article/detail/?content_id=8afec58a-b797-4bf9-acca-76ed512a3acb)。

案例内容介绍

我们之前学习过分类树，随机森林就是种了很多分类树。对输入向量进行分类。每一颗树都是分类，要对这个输入向量进行"投票"。森林就是选择投票最多的那个树。

随机森林的优缺点

优点：

1.可以用来解决分类和回归问题：随机森林可以同时处理分类和数值特征

2.抗过拟合能力：通过平均决策树，降低过拟合的风险性

3.只有在半数以上的基分类器出现差错时才会做出错误的预测：随机森林非常稳定，即使数据集中出现了一个新的数据点，整个算法也不会受到过多影响，它只会影响到一颗决策树，很难对所有决策树产生影响

缺点：

1.据观测，如果一些分类/回归问题的训练数据中存在噪音，随机森林中的数据集会出现过拟合的现象

2.比决策树算法更复杂，计算成本更高

3.由于其本身的复杂性，它们比其他类似的算法需要更多的时间来训练

本案例推荐的理论学习视频：

- 《AI技术领域课程--机器学习》 随机森林 (https://education.huaweicloud.com/courses/course-v1:HuaweiX+CBUCNXE086+Self-paced/courseware/45c5a9b65ee348719ddc4f4c3801ad0f/b67e40f52a7447f79364a2de6f2b9398/)

注意事项

1. 如果您是第一次使用 JupyterLab，请查看《ModelArts JupyterLab使用指导》(https://marketplace.huaweicloud.com/markets/aihub/article/detail/?content_id=03676d0a-0630-4a3f-b62c-07fba43d2857)了解使用方法；
2. 如果您在使用 JupyterLab 过程中碰到报错，请参考《ModelArts JupyterLab常见问题解决办法》(https://marketplace.huaweicloud.com/markets/aihub/article/detail/?content_id=9ad8ce7d-06f7-4394-80ef-4dbf6cfb4be1)尝试解决问题。

实验步骤

1、导入依赖库

In [1]:
```
import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
import matplotlib.pylab as plt
import moxing as mox
import os
```

```
INFO:root:Using MoXing-v1.17.3-

INFO:root:Using OBS-Python-SDK-3.20.7
```

2、下载数据集

In [2]:
```
if not os.path.exists('car.csv'):
    mox.file.copy('obs://modelarts-labs-bj4-v2/course/hwc_edu/machine_learning/datasets/random_forests/car.csv',
                  'car.csv')
train_path = './car.csv'
```

3、读取数据文件，如果数据文件放在带有中文字符的路径，read_csv方法需要指定参数#engine='python'

In [3]:
```
data_frame_train = pd.read_csv(train_path, encoding='gbk')
print(data_frame_train.head())
```

	编号	性别_1男2女	年龄_1为60后2为70后3为80后4为90后5为00后	婚姻情况_1已婚2未婚	户口_1本地户口2外地户口	\
0	1	1	3	1	1	
1	2	2	5	1	2	
2	3	2	4	1	1	
3	4	2	2	1	1	
4	5	1	3	2	1	

	家庭收入_1小于五千2五千到一万3一万以上	居住情况_1有房户2租赁房	学历_1高中及以下2本科3硕士4博士及以上	\
0	2	1	1	
1	3	1	2	
2	3	2	2	
3	2	1	1	
4	2	1	1	

	工作情况_1上班族2自雇人员3批发和零售业4其他	工作职位_1管理2市场3技术4其他	...	地区_1中部2东南沿海3西部4东北	\
0	1	1	...	2	
1	4	1	...	1	
2	3	3	...	2	
3	1	4	...	2	
4	1	4	...	3	

	车型_共17款车型	金融产品_1一证贷2两证贷3三证贷	成交价	首付比例	贷

```
款金额   贷款期限_1为12个月2为24个月   \

0          10                   2  164000   0.4   98400
1

1           8                   1  183000   0.2  146400
1

2          10                   2  195000   0.1  175500
2

3           7                   1  240000   0.1  216000
1

4           9                   2  154000   0.2  123200
2
```

```
   客户等级_1高星级2中星级3低星级   车型级别_1高2中3低   是否贷款买车_1是0否

0                   2            2             1

1                   2            3             0

2                   2            3             1

3                   1            3             0

4                   1            1             1


[5 rows x 24 columns]
```

4、划分训练集和测试集的X，y

In [4]:
```
X_train, y_train = data_frame_train.values[:, :-1], data_frame_train.values[:, -1]
```

5、实例化模型

In [5]:
```
# 不调整参数的效果(oob_score=True:采用袋外样本来评估模型的好坏,反映了模型的泛化能力)
rfclf = RandomForestClassifier(oob_score=True, random_state=10)
```

6、模型训练

In [6]:
```
rfclf.fit(X_train, y_train)
```

```
/home/ma-user/anaconda3/envs/XGBoost-Sklearn/lib/python3.6/site-packa
ges/sklearn/ensemble/forest.py:248: FutureWarning: The default value
of n_estimators will change from 10 in version 0.20 to 100 in 0.22.

  "10 in version 0.20 to 100 in 0.22.", FutureWarning)

/home/ma-user/anaconda3/envs/XGBoost-Sklearn/lib/python3.6/site-packa
ges/sklearn/ensemble/forest.py:460: UserWarning: Some inputs do not h
ave OOB scores. This probably means too few trees were used to comput
e any reliable oob estimates.

  warn("Some inputs do not have OOB scores. "

/home/ma-user/anaconda3/envs/XGBoost-Sklearn/lib/python3.6/site-packa
ges/sklearn/ensemble/forest.py:465: RuntimeWarning: invalid value enc
ountered in true_divide

  predictions[k].sum(axis=1)[:, np.newaxis])
```

Out[6]:
```
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='
gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=Non
e,
            oob_score=True, random_state=10, verbose=0, warm_start=Fa
lse)
```

7、模型对测试集进行预测

In [7]:
```
y_pre = rfclf.predict(X_train)  # 预测值
y_prb_1 = rfclf.predict_proba(X_train)[:, 1]  # 预测为1的概率
```

8、输出oob_score以及auc

In [8]:
```
print(rfclf.oob_score_)  # 0.8407142857142857
```

```
0.8407142857142857
```

以上是 随机森林 的实现方法，受限于篇幅原因，本案例未完全覆盖 随机森林 的全部操作，欢迎你将更全面的随机森林 学习笔记分享到 AI Gallery Notebook (https://marketplace.huaweicloud.com/markets/aihub/notebook/list/) 版块获得成长值 (https://marketplace.huaweicloud.com/markets/aihub/article/detail/?content_id=9b8d7e7a-a150-449e-ac17-2dcf76d8b492)，分享方法请查看此文档 (https://marketplace.huaweicloud.com/markets/aihub/article/detail/?content_id=8afec58a-b797-4bf9-acca-76ed512a3acb)。

作业

请你利用本实验中学到的知识点，完成以下编程题：

1. 请你尝试修改 RandomForestClassifier() 函数的 n_estimators（树的数量）参数的不同取值，看看该参数的修改对模型会有怎样的影响。(https://marketplace.huaweicloud.com/markets/aihub/notebook/detail/?id=a3f89295-2d42-4995-8a7f-15c1d12570ee)
2. 请你尝试修改 RandomForestClassifier() 函数的 max_depth（最大深度）参数的不同取值，看看该参数的修改对模型会有怎样的影响。(https://marketplace.huaweicloud.com/markets/aihub/notebook/detail/?id=8e32c793-913c-4190-a938-734220a45bb0)
3. 请你尝试修改 RandomForestClassifier() 函数的所有可调参数的不同取值，看看不同参数的不同取值组合，对模型会有怎样的影响。(https://marketplace.huaweicloud.com/markets/aihub/notebook/detail/?id=9b440733-09a3-4fce-9b9d-67bdc59459ad)