二手车价格预测第十二名方案总结

wujiekd 2023-06-18 23:15:56 60 18484

代码开源链接：https://github.com/wujiekd/Predicting-used-car-prices

比赛介绍

赛题以二手车市场为背景，要求选手预测二手汽车的交易价格，这是一个典型的回归问题。
其他具体流程可以看比赛官网。

数据处理

1、box-cox变换目标值“price”，解决长尾分布。

2、删除与目标值无关的列，例如“SaleID”，“name”。这里可以挖掘一下“name”的频度作为新的特征。

3、异常点处理，删除训练集特有的数据，例如删除“seller”==1的值。

4、缺失值处理，分类特征填充众数，连续特征填充平均值。

5、其他特别处理，把取值无变化的列删掉。

6、异常值处理，按照题目要求“power”位于0～600，因此把“power”>600的值截断至600，把"notRepairedDamage"的非数值的值替换为np.nan，让模型自行处理。

特征工程

1、时间地区类

从“regDate”，“creatDate”可以获得年、月、日等一系列的新特征，然后做差可以获得使用年长和使用天数这些新特征。

“regionCode”没有保留。
因为尝试了一系列方法，并且发现了可能会泄漏“price”，因此最终没保留该特征。

2、分类特征

对可分类的连续特征进行分桶，kilometer是已经分桶了。
然后对"power"和"model"进行了分桶。

使用分类特征“brand”、“model”、“kilometer”、“bodyType”、“fuelType”与“price”、“days”、“power”进行特征交叉。
交叉主要获得的是后者的总数、方差、最大值、最小值、平均数、众数、峰度等等

这里可以获得非常多的新特征，挑选的时候，直接使用lightgbm帮我们去选择特征，一组组的放进去，最终保留了以下特征。（注意：这里使用1/4的训练集进行挑选可以帮助我们更快的锁定真正Work的特征）

```
'model_power_sum','model_power_std',
'model_power_median', 'model_power_max',
'brand_price_max', 'brand_price_median',
'brand_price_sum', 'brand_price_std',
'model_days_sum','model_days_std',
'model_days_median', 'model_days_max',
'model_amount','model_price_max',
'model_price_median','model_price_min',
'model_price_sum', 'model_price_std',
'model_price_mean'
```

3、连续特征

使用了置信度排名靠前的匿名特征“v_0”、“v_3”与“price”进行交叉，测试方法以上述一样，效果并不理想。

因为都是匿名特征，比较训练集和测试集分布，分析完基本没什么问题，并且它们在lightgbm的输出的重要性都是非常高的，所以先暂且全部保留。

4、补充特征工程

主要是对输出重要度非常高的特征进行处理

特征工程一期：

对14个匿名特征使用乘法处理得到14*14个特征

使用sklearn的自动特征选择帮我们去筛选，大概运行了半天的时间。

大致方法如下：

```
from mlxtend.feature_selection import SequentialFeatureSelector as SFS
from sklearn.linear_model import LinearRegression
sfs = SFS(LGBMRegressor(n_estimators = 1000,objective='mae' ),
           k_features=50,
           forward=True,
           floating=False,
           cv = 0)


sfs.fit(X_data, Y_data)
print(sfs.k_feature_names_)
```

最终筛选得到：

```
'new3*3', 'new12*14', 'new2*14','new14*14'
```

特征工程二期：

对14个匿名特征使用加法处理得到14*14个特征

这次不选择使用自动特征选择了，因为运行实在太慢了，笔记本耗不起。

使用的方法是删除相关性高的变量,把要删除的特征记录下来

大致方法如下：（剔除相关度>0.95的）

```
corr = X_data.corr(method='spearman')
feature_group = list(itertools.combinations(corr.columns, 2))
print(feature_group)

# 删除相关性高的变量,调试好直接去主函数进行剔除
def filter_corr(corr, cutoff=0.7):
    cols = []
    for i,j in feature_group:
        if corr.loc[i, j] > cutoff:
            print(i,j,corr.loc[i, j])
            i_avg = corr[i][corr[i] != 1].mean()
            j_avg = corr[j][corr[j] != 1].mean()
            if i_avg >= j_avg:
                cols.append(i)
            else:
                cols.append(j)
    return set(cols)

drop_cols = filter_corr(corr, cutoff=0.95)
print(drop_cols)
```

最终获得的应该删除的特征为：

```
['new14+6', 'new13+6', 'new0+12', 'new9+11', 'v_3', 'new11+10', 'new10+14', 'new12+4', 'new3+4', 'new11+11', 'new13+3', 'new8+1', 'n
```

特征工程三、四期：

这两期的效果不明显，为了不让特征冗余，所以选择不添加这两期的特征，具体的操作可以在feature处理的代码中看到。

5、神经网络的特征工程补充说明

以上特征工程处理都是针对于树模型来进行的，接下来，简单说明神经网络的数据预处理。

各位都知道由于NN的不可解释性，可以生成大量的我们所不清楚的特征，因此我们对于NN的数据预处理只要简单处理异常值以及缺失值。

大部分的方法都包含在以上针对树模型数据处理方法中，重点讲述几个不同点：

在对于“notRepairedDamage”的编码处理，对于二分类的缺失值，往往取其中间值。

在对于其他缺失值的填充，在测试了效果后，发现填充众数的效果比平均数更好，因此均填充众数。

选择的模型

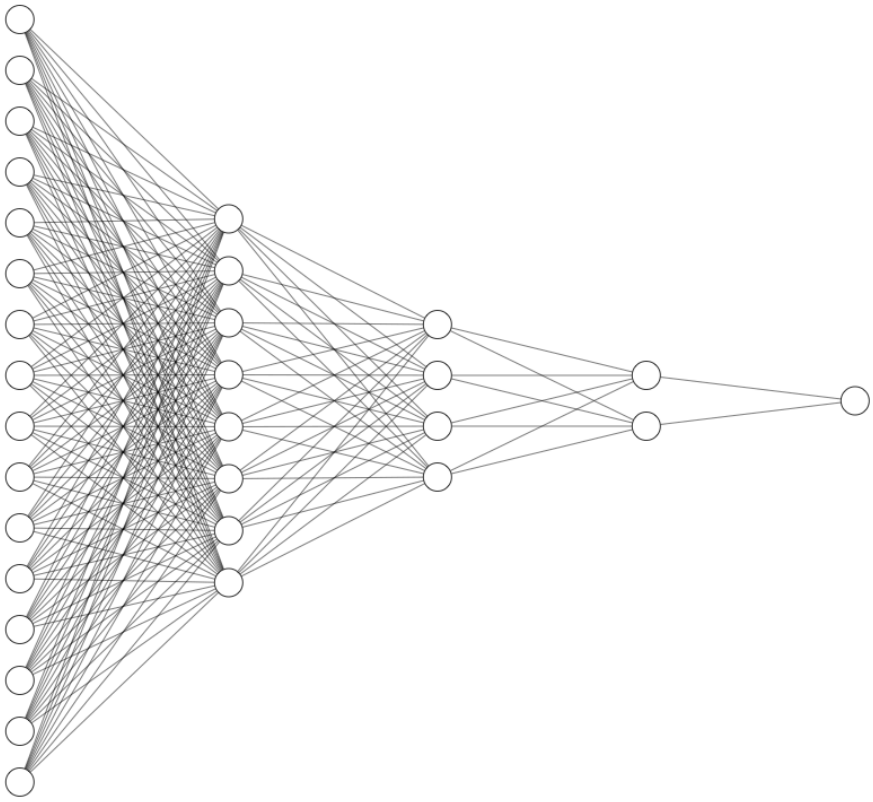
本次比赛，我选择的是lightgbm+catboost+neural network。
本来也想使用XGBoost的，不过因为它需要使用二阶导，因此目标函数没有MAE，并且用于逼近的一些自定义函数效果也不理想，因此没有选择使用它。

经过上述的数据预处理以及特征工程：
树模型的输入有83个特征；神经网络的输入有29个特征。

1、lightgbm和catboost：
因为它们都是树模型，因此我同时对这两个模型进行分析

第一：lgb和cab的训练收敛速度非常快，比同样参数的xgb快非常多。
第二：它们可以处理缺失值，计算取值的增益，择优录取。
第三：调整正则化系数，均使用正则化，防止过拟合。
第四：降低学习率，获得更小MAE的验证集预测输出。
第五：调整早停轮数，防止陷入过拟合或欠拟合。
第六：均使用交叉验证，使用十折交叉验证，减小过拟合。
其他参数设置无明显上分迹象，以代码为准，不一一阐述。

2、neural network：



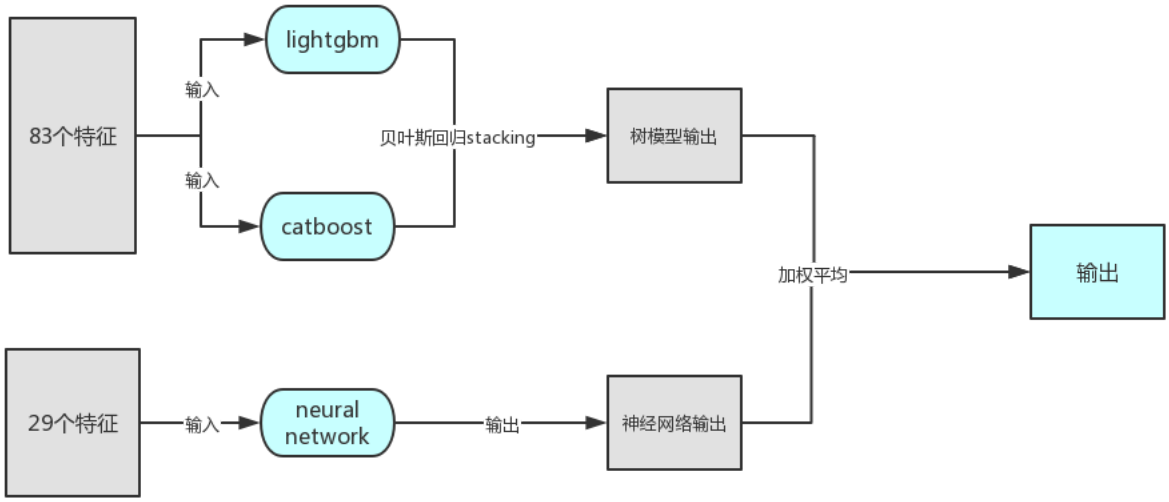
设计了一个五层的神经网络，大致框架如上图所示，但结点数由于太多只是展示部分结点画图。（无聊画了个。。）
以下为全连接层的结点个数设置，具体实施可参考代码。

接下来对神经网络进行具体分析：
第一：训练模型使用小batchsize，512，虽然在下降方向上可能会出现小偏差，但是对收敛速度的收益大，2000代以内可以收敛。
第二：神经网络对于特征工程这一类不用操心很多，就能达到与树模型相差无几的精度。
第三：调整正则化系数，使用正则化，防止过拟合。
第四：调整学习率，对训练过程的误差进行分析，选择学习率下降的时机进行调整。
第五：使用交叉验证，使用十折交叉验证，减小过拟合。
第六：选择梯度下降的优化器为Adam，它是目前综合能力较好的优化器，具备计算高效，对内存需求少等等优点。

集成的方法

由于两个树模型的训练数据一样且结构相似，首先对两个树模型进行stacking，然后再与神经网络的输出进行mix。
由于树模型和神经网络是完全不同的架构，它们得到的分数输出相近，预测值差异较大，往往在MAE上差异为200左右，因此将他们进行MIX可以取到一个更好的

结果，加权平均选择系数选择0.5，虽然神经网络的分数确实会比树模型高一点点，但是我们的最高分是多组线上最优输出的结合，因此可以互相弥补优势。



后期上分

单个神经网络和树模型的结合在本地验证集上都是420分左右；然后经过以上的模型集成，线下验证集为415分左右，线上得到的分数位于404～410，后期因为没有其他的上分方法，因此每上一分都是非常困难的，最终会使用更换种子输出多个文件，挑选了最好的三个分数进行平均最终获得线上第十三名。

赛后总结

本次赛后，看到论坛上的单模型就可以到达400分，因此个人对本次比赛总结一下几点：

- 我的最大的败笔应该是没有充分发挥神经网络的优势，没有人为的制造多一些的特征；
- 从这单单29个特征的nn就可以媲美千挑万选特征的树模型，我猜想高分选手使用的是神经网络+更好的特征工程；
- 或者，本次比赛存在着leaky，让敏锐的前排大佬们给捕捉到了，期待前排的开源；
- 最后，感谢主办方阿里天池与Datawhale，希望能够继续举办这样有趣的赛事～

全部评论(28)

aliyun3545553975
2023-06-18 23:15:56

请问什么叫一组组的放进去测试啊，是每次结合原始特征，放一组进去，如果性能提升就保留，性能没有提升就删去吗，如果一组保留了，后面一组放进去测试的时候，是仍然用原始特征，还是增加了已经保留的特征呢

28楼 0 发表于上海市

老包要坚持学习
2022-03-04 11:20:54

@wujiekd 大佬，你好。你选择使用5个分类特征与price，power，days进行特征交叉。想问下，为啥是power，days。是因为你先用lgb跑出来 这两个特征 权重比较高吗

27楼 0

2022-03-05 12:02:53

wujiekd：

类别特征和连续特征交叉，连续特征就这几个

0

---

xu1234
2021-11-10 14:56:53

作者你好，请问为什么按照你的方案提交的是6000多分啊...小白不是很明白，麻烦解答，谢谢！

26楼 0

2022-03-14 14:14:13

Data_lukyo：

用df_testB预测提交

0

2021-11-21 14:41:32

深圳大学-董芸豪：

那个测试数据不一样的

0

2021-11-12 19:55:43

wujiekd：

看一下输出结果，是不是box-cox变换后，没有变换回来。

1

Jack666

2020-12-01 14:27:23

感谢大佬的分享，学习了

25楼 0

wjsbysdxx

2020-11-30 19:07:35

想问一下大佬的tensorflow是gpu版本的吗？同为mac本，好像只能装cpu版的😭

24楼 0

2020-12-07 11:57:34

wujiekd：

Mac只能装cpu的，A卡不支持张量运算，只有N卡可以啦，可以使用阿里提供的免费服务器或者colab

0

wjsbysdxx

2020-11-30 19:06:51

想问一下大佬的tensorflow是gpu版本的吗？同为mac本，好像只能装cpu版的

23楼 0

daxinyan

2020-09-27 20:24:13

想请问一下，用lightgbm模型进行特征选择的时候，输入的特征是只有我们和price创建出来的交互特征还是交互特征和初始特征都要加进去

22楼 0

2020-09-27 22:39:09

wujiekd：

交互特征和初始特征都要的，交互特征的目的是在初始特征的基础上进行提升，若脱离了初始特征那没有意义了

0

竹梨落
2020-09-03 17:18:46

“regionCode”没有保留。因为尝试了一系列方法，并且发现了可能会泄漏“price”，因此最终没保留该特征。请问,如果regionCode 会泄漏price,训练集和测试集都有regionCode,为什么不能用它来预测价格反而要删除掉呢?

21楼 0

2020-09-14 11:17:26

wujiekd：

有的特征不一定有效，使用regionCode 对price目标编码，出现了结果过拟合的情况

0

一叶翩舟-
2020-08-03 18:32:20

点赞分享

20楼 0

---

crystal_py罗
2020-07-31 21:30:22

楼主,您好,分箱步骤我有点不太懂,分出来后会出现很多空值,小白求助...

19楼 0

2020-08-11 14:15:59

wujiekd：

这里的分箱就是把异常大的值先给剔除掉，举个例子，假设是年龄段1-10，11-20，21-30，出现一些200、300甚至更大的，采用空值处理效果更好。

0

关于我们 法务协

tianchi_bigdata@

联系我们