膜色缺陷复盘:[2021.11.25]

* rgb颜色提取

1. 采用moyu哥给的get\_color.py脚本, 一张张过图像.

moyu\_ge建议: 绿膜数据拉高green\_lower就好, green\_lower值有一段区域变化但roi基本不变, 选这段区间的中间某处值就行了.

2. 后续考虑自动化提取rgb颜色不加入人工干预, 可能要上dl的分割来提取roi内颜色.

* rgb+ -> lab建模

其实是 rgb+ -> xyz -> lab, xyz -> lab 是一个固定参数值的矩阵乘法.

**1.** 选用的ML模型: xgboost

**2.** 精度度量方法: 10次交叉验证

**3.** 调优模型精度的过程:

1. 直接223条训练数据, 用之前的, 见过很多绿膜数据的"祖传"超参跑一遍, 精度0.88

2. 做训练数据清洗:

1. 绿膜数据出现g<b情况, 剔除这4条

2. LAB超出客户提供的正常范围, 剔除这11条 (集中出现在15 16 文件夹)

3. 15 16 文件夹数据, 相机成像模式与1-14不同, 剔除这39条

做完以上清洗后, 187条训练数据上的精度: 0.91

3. 在"祖传"超参的基础上, 开启超参搜索(另外加了点人工调lr, 之类的..), 精度到0.936

4. 加入镜片roi区域的面积信息, 精度到94.2, 简单手动调了下参, 精度到94.6

* 一些快速实验和思考

1. 根据gamma变换后的gamma\_g或gamma\_b值, 设定阈值划分训练数据. 根据gamma\_b/gamma\_g值划分数据也尝试过, 收益不大.

2. 拼接rgb值, 另外加g/b, g/r, r/b等信息, 使用k\_means做训练数据二分聚类, 分别调优两部分模型的精度. 数据被拆分为118和69, 精度分别到91+ 和93+. 整体精度没有显著提升.

1. 根据样本的lab值 **通过/不通过** 属性, 划分样本. 两部分数据都可直接到94+精度. 对于测试集, 先用两个模型各自预测lab值, 判断样本是否lab通过, 然后走专用模型预测样本的lab值. 后续验证此思路失败, 部分临近通过不通过的样本, 两个模型对lab值的预测, **不能反映样本真实的通过与否情况**. 通过/不通过属性划分样本在蓝膜数据上也一样表现较好，训练数据88张仅一张通过。直接训模型可到92.8精度.. 可见保持数据”不混杂”对精度提升很有作用. 遗憾这个通过否信息无法直接在rgb上体现..

脑洞一下, 收集LAB范围超出异常的数据单独训模型, 用于对LAB异常样本的预测，模型只需报出 **异常/不异常** 即可. [在13张LAB异常蓝膜数据上训了个模型, acc95+]

4. 可视化各炉数据的rgb值分布, 发现各炉间颜色分布有明显边界. 这也是很重要的**数据特征信息**, 反应的问题是: 对于来自，训练集完全没见过的炉，的测试数据, 模型可能无法准确预测lab值. 因为各炉间数据“自墙分明”, 由于不同炉间生产环境的影响或不同炉生产的镜片特性不同, 导致不同炉间的”规律”较难迁移...

5. **要充分模拟客户设计的实验**。

对于蔡司留下的70条测试数据, 观察到: 11 12炉被完全保留测试，即模型完全”没见过”这两炉数据. 故在给客户反馈测试集结果前，可考虑**在训练集上设计类似的实验**, 完全保留某炉数据测试，其余炉数据训练，看看模型的预测能力. 更精细的, 寻找训练集中相似的几炉数据，选择其中一炉测试，其他剩余炉训练模型，看看模型的预测能力。

在本次客户验证前做了此模拟实验，确实发现模型对没见过的炉数据，lab预测能力较差. 后面在客户的测试集上，也出现了同样的结果，好在没太严重[11错了5/20.. 12错了1/20]

* 模型调优总结

1. 清洗得到干净的训练数据很重要

2. xgboost模型, 精细调参很重要, 所以一定要上交叉验证, 防止过拟合.

3. 除以上, 就是针对项目特点的细节: 数据理解, 划分.

1. 模型输入端x, 对其分析, 划分, 扩充特征维度

1. 拆分出"相似"的数据簇, 在每一份稳定的簇内调优模型.

2. 预测lab值的上游任务: 二维图像 -> 特定长度的特征向量, 容易忽略掉除rgb之外的信息. 可以再添加一些roi形态, 面积等特征, 增加输入x的特征丰富性. 这一点需要结合客户给到的信息理解: roi区域的大小形状, 反应样本的度数闪光等信息. 度数闪光等因素可能影响镜片的光学特性[这里可能是我们关心的lab值]

[这一点在客户揭晓测试数据lab后发现, 虽然在训练集上精度调优1个点, 但在测试集上没有”成功”迁移... ] 所以**一定要注意模型的泛化能力**，但这和**充分拆解问题和挖掘数据信息**, 并不冲突...

* 最后

ML相关的**"数据"问题**, 很重要的部分在于针对特定任务 **理解,拆分 数据**. 一定牢记充分挖掘数据相关的信息, 为了精度,数据需要拆就拆, 然后再针对性的依次精准建模.

附:

11.24测试集结果, fail case 6条，acc：64/70

img: 11\_1, diff: [0.04, 0.7, 0.045]

img: 11\_4, diff: [0.04, 0.6, 0.03]

img: 11\_5, diff: [0.12, 0.54, 0.23]

img: 11\_9, diff: [0.12, 0.5, 0.23]

img: 11\_17, diff: [0.04, 0.62, 0.07]

img: 12\_14, diff: [0.18, 0.55, 0.21]