

# 华大基因实习总结

## BGI Genomics Internship Summary

尹超

中国科学院大学，北京 100049

Carter Yin

University of Chinese Academy of Sciences, Beijing 100049, China

2025.7.21 – 2025.8.22

## 序言

本文为笔者在华大基因实习期间的工作总结。

望师兄师姐批评指正。

# 目录

序言	I
目录	II
<b>1 工作材料</b>	<b>1</b>
1.1 数据分析:chart 文件夹 . . . . .	1
1.2 原始数据文件修正:pre_mod 文件夹 . . . . .	1
1.2.1 代码与数据文件 . . . . .	1
1.2.2 代码实现 . . . . .	1
1.3 数据预处理:backup 文件夹 . . . . .	1
Step 0 . . . . .	2
Step 1 . . . . .	2
Step 2 . . . . .	2
Step 3 . . . . .	2
Step 4 . . . . .	2
Step 5_0base . . . . .	3
Step 5_1eye . . . . .	3
Step 5_2artery . . . . .	3
Step 5_3physiology . . . . .	3
Step 5_4all . . . . .	3
Step 5_5age . . . . .	4
Step 5_6ren . . . . .	4
Step 5_7newall . . . . .	4
1.4 模型训练和测试:train_test 文件夹 . . . . .	4
all . . . . .	4
artery . . . . .	4
eye . . . . .	5
physiology . . . . .	5
ren . . . . .	5
newall . . . . .	5
1.5 ICD10 数据的处理:ICD10 . . . . .	5
pre_mod . . . . .	5
merge . . . . .	5
Step 0 . . . . .	5
Step 1 . . . . .	5
Step 2 . . . . .	6
Step 3 . . . . .	6
Step 4 . . . . .	6
Step 5 . . . . .	6
Step 6 . . . . .	6
Step 7_0base . . . . .	7

Step 7_1eye . . . . .	7
Step 7_2artery . . . . .	7
Step 7_3physiology . . . . .	7
Step 7_4all . . . . .	7
Step 7_5age . . . . .	8
Step 7_6ren . . . . .	8
Step 7_7newall . . . . .	8
1.6 ICD10 数据的模型训练和测试:ICD10train_test 文件夹 . . . . .	8
1.7 多模态模型的训练:multi_model 文件夹 . . . . .	8
1.8 一些可能用到的文件:copy 文件夹 . . . . .	9
<b>2 工作内容 . . . . .</b>	<b>10</b>
2.1 模块 1: 数据分析工作 . . . . .	10
方法 . . . . .	10
结果 . . . . .	10
2.2 模块 2: 模型训练工作 . . . . .	12
方法 . . . . .	12
结果 . . . . .	13
<b>3 工作总结 . . . . .</b>	<b>15</b>
3.1 项目总结 . . . . .	15
3.2 个人总结 . . . . .	16
<b>4 AI 工具相关 . . . . .</b>	<b>18</b>
4.1 论文阅读 . . . . .	18
4.2 代码辅助 . . . . .	18
4.3 问题解答 . . . . .	18

# Chapter 1 工作材料

## 1.1 数据分析:chart 文件夹

该文件夹主要包含数据分析过程中生成的可视化图表和代码，具体包括：

### 数据与代码文件

- analysis\_result\_eyes\_realigned.tsv 绘图数据文件
- disease\_groups\_histogram.py: 绘制眼病患病率和其他疾病患病率的样本数量分布。
- disease\_groups\_histogram2.py: 绘制其他疾病患病率的样本数量分布。
- exclude\_max\_histogram.py: 剔除最大值后绘制样本数量分布。
- max\_vs\_others\_histogram.py: 绘制最大值与其他值的样本数量分布。

### 图片文件

- eye\_diseases\_bar.png
- other\_diseases\_bar.png
- other\_diseases\_bar\_new.png
- exclude\_max\_histogram\_ultra\_high.png
- max\_vs\_others\_histogram\_ultra\_high.png

## 1.2 原始数据文件修正:pre\_mod 文件夹

### 1.2.1 代码与数据文件

data\_preprocessing.py  
original.tsv

### 1.2.2 代码实现

- data\_preprocessing.py: 该脚本的具体实现步骤如下：
  - (1) 读取输入的 original.tsv 文件。
  - (2) 对原始数据文件进行添加列标题 samples，为保证 tsv 文件列对齐。
  - (3) 遍历数据，将除了第一列的样本编号外的“yes”替换为 1。
  - (4) 遍历数据，将除了第一列的样本编号外的“no”替换为 0。
  - (5) 将处理后的数据写入新的 preprocessing.tsv 文件中。
- preprocessing.tsv: 该文件为经过处理后的 tsv 文件。
- original.tsv.original\_backup: 该文件为原始数据文件的备份。

## 1.3 数据预处理:backup 文件夹

在数据分析之前，首先需要对原始数据进行预处理。数据预处理的主要步骤包括：(其中 Step6 为了结构完整性保留在此，仅为提及)

Step 0: 剔除没有疾病诊断信息的个体（青光眼，黄斑变性，白内障，糖尿病，中风，高血压，缺血性心脏病）

Step 1: 剔除眼病患者（青光眼，黄斑变性，白内障），用作 test(剩余 21137 samples)

Step 2: 剔除其它有记录的疾病患者（糖尿病，中风，高血压，缺血性心脏病），用作 test(剩余 11895 samples)

Step 3: 删去方差为 0 的列，并将数据集分为 training set (80%) 和 test set (20%)

Step 4: 针对 training 和 test 内的数据缺失分别进行补全（剔除缺失率过高的字段），这里我们采用了 Mean imputation

Step 5: 筛选字段-v3 眼部字段，颈动脉字段，生理代谢字段

Step 6: 基于不同的数据组合，如眼部数据、颈动脉数据、生理数据和所有数据，计算生物学年龄，训练基于决策树类算法的预测模型，包括 XGBoost, LightGBM, CatBoost, Random Forest。

## Step 0

**step0.py:** 该脚本的主要功能是剔除没有疾病诊断信息的个体。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 遍历数据，对于指定的字段，如果全为 NA，则剔除该个体。
- (3) 将处理后的数据写入新的 tsv 文件中。

## Step 1

**step1.py:** 该脚本的主要功能是剔除眼病患者。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 遍历数据，对于指定的眼科疾病诊断字段，如果值为 1，则剔除该个体。
- (3) 将处理后的数据写入新的 tsv 文件中。

## Step 2

**step2.py:** 该脚本的主要功能是剔除其它有记录的疾病患者。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 遍历数据，对于指定的其它疾病诊断字段，如果值为 1，则剔除该个体。
- (3) 将处理后的数据写入新的 tsv 文件中。

## Step 3

**step3.py:** 该脚本的主要功能是删去方差为 0 的列，并将数据集分为训练集和测试集。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 删去方差为 0 的列。
- (3) 随机打乱数据顺序。
- (4) 按照 8:2 的比例划分训练集和测试集。
- (5) 将训练集和测试集分别写入新的 tsv 文件中。

## Step 4

**step4.py:** 该脚本的主要功能如下：

- (1) 读取输入的 tsv 文件。
- (2) 遍历数据，对于缺失值超过 50% 的字段，剔除该字段。
- (3) 遍历数据，对于缺失的特定字段值，按照优先级规则进行填充。
- (4) 仅保留指定的 124 个字段。
- (5) 遍历数据，对于缺失的其它字段值，按照填充规则进行填充。

(6) 将处理后的数据写入新的 tsv 文件中。

#### 特定字段填充规则:

- (1) **bmi\_x10**: 如果为空, 使用 `bmi_calc_resurvey2 × 10` 或 `bmi_calc_baseline × 10` 填充  
优先级: `bmi_calc_resurvey2 > bmi_calc_baseline`
- (2) **standing\_height\_cm\_x10**: 如果为空, 使用 `standing_height_mm_resurvey2` 或 `standing_height_mm_baseline` 填充  
优先级: `standing_height_mm_resurvey2 > standing_height_mm_baseline`
- (3) **weight\_kg\_x10\_resurvey3**: 如果为空, 使用 `weight_kg_x10_resurvey2` 或 `weight_kg_x10_baseline` 填充  
优先级: `weight_kg_x10_resurvey2 > weight_kg_x10_baseline`

#### 其它填充规则:

- (1) 如果列只包含少数几个值 ( $\leq 10$  个唯一值), 按照填充前的比例进行随机填充
- (2) 对于 `id_ethnic_group_id` 这一列, 按照情况 1 处理
- (3) 如果是其他情况, 按照均值填充缺失值

### Step 5\_0base

该文件夹存放了整个步骤 5 中的基础数据文件。

### Step 5\_1eye

`eye.py`: 该脚本的主要功能是筛选眼科相关字段。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与眼科相关的字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

### Step 5\_2artery

`artery.py`: 该脚本的主要功能是筛选动脉相关字段。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与动脉相关的字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

### Step 5\_3physiology

`physiology.py`: 该脚本的主要功能是筛选生理相关字段。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与生理相关的字段。(不包括年龄)
- (3) 将处理后的数据写入新的 tsv 文件中。

### Step 5\_4all

`all.py`: 该脚本的主要功能是筛选所有指定字段。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。

- (2) 保留所有指定字段。(不包括年龄)
- (3) 将处理后的数据写入新的 tsv 文件中。

### Step 5\_5age

**age.py:** 该脚本的主要功能是筛选年龄相关字段。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与年龄相关的字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

### Step 5\_6ren

**ren.py:** 该脚本的主要功能是筛选人口学相关字段。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与性别、地区和职业相关的字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

### Step 5\_7newall

**newall.py:** 该脚本的主要功能是合并新增的人口学指标和原来指定的所有字段。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 合并新增的人口学指标和原来指定的所有字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

## 1.4 模型训练和测试:train\_test 文件夹

**使用不同的数据预测生物学年龄**

该文件夹下的所有 simple1\_gpu.py 位于每个子文件夹的 code 文件夹下，代码具体实施步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 对数据进行预处理，包括去除缺失值、标准化等。
- (3) 校正项计算函数。
- (4) 在四种不同的模型上进行训练。(XGBoost, LightGBM, CatBoost, RandomForest)
- (5) 训练集上进行五折交叉验证寻找最佳超参数。(以 MAE 为指标)
- (6) 在测试集上分别评估四种模型最佳参数的性能。
- (7) 输出报告。

**一个一键运行所有模型的脚本:run\_all\_models.py**

### all

**simple1\_gpu.py:** 该脚本的主要功能是对所有数据进行模型训练和测试。

### artery

**simple1\_gpu.py:** 该脚本的主要功能是对颈动脉数据进行模型训练和测试。

## eye

**simple1\_gpu.py**: 该脚本的主要功能是对眼科数据进行模型训练和测试。

## physiology

**simple1\_gpu.py**: 该脚本的主要功能是对生理数据进行模型训练和测试。

## ren

**simple1\_gpu.py**: 该脚本的主要功能是对人口学数据进行模型训练和测试。

## newall

**simple1\_gpu.py**: 该脚本的主要功能是对新的所有数据（新增了人口学数据）进行模型训练和测试。

## 1.5 ICD10 数据的处理:ICD10

### pre\_mod

- **icd10.py**: 该脚本的具体实现步骤如下:

- (1) 读取输入的 ICD10.tsv 文件。
- (2) 对原始数据文件进行添加列标题 samples，为保证 tsv 文件列对齐。
- (3) 遍历数据，将除了第一列的样本编号外的“yes”替换为 1。
- (4) 遍历数据，将除了第一列的样本编号外的“no”替换为 0。
- (5) 将处理后的数据写入新 icd10.tsv 文件中。

- **icd10.tsv**: 该文件为经过处理后的 tsv 文件。
- **ICD10.tsv.ICD10\_backup**: 该文件为原始数据文件的备份。

### merge

**merge\_files.py**: 该脚本的主要功能是将 ICD10 数据和原来的 CKB 数据进行合并。具体实现步骤如下:

- (1) 读取输入的 icd.tsv 和 preprocessing.tsv 文件。
- (2) 对数据进行合并，确保样本编号对齐。
- (3) 将合并后的数据写入新的 all.tsv 文件中。

## Step 0

**step0.py**: 该脚本的主要功能是剔除没有疾病诊断信息的个体。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。
- (2) 遍历数据，对于指定的字段，如果全为 NA，则剔除该个体。
- (3) 将处理后的数据写入新的 tsv 文件中。

## Step 1

**step1.py**: 该脚本的主要功能是剔除眼病患者。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。

(2) 遍历数据, 对于指定的眼科疾病诊断字段, 如果值为 1, 则剔除该个体。

(3) 将处理后的数据写入新的 tsv 文件中。

## Step 2

**step2.py:** 该脚本的主要功能是剔除其它有记录的疾病患者。具体实现步骤如下:

(1) 读取输入的 tsv 文件。

(2) 遍历数据, 对于指定的其它疾病诊断字段, 如果值为 1, 则剔除该个体。

(3) 将处理后的数据写入新的 tsv 文件中。

## Step 3

**step3.py:** 该脚本的主要功能是剔除 ICD10 中有记录的眼病患者。具体实现步骤如下:

(1) 读取输入的 tsv 文件。

(2) 遍历数据, 对于指定的眼科疾病诊断字段, 如果值不为 NA, 则剔除该个体。

(3) 将处理后的数据写入新的 tsv 文件中。

## Step 4

**step4.py:** 该脚本的主要功能是剔除 ICD10 中不为汉族的样本。具体实现步骤如下:

(1) 读取输入的 tsv 文件。

(2) 删除 id\_ethnic\_group\_id 不为 1 的样本 (保留 NA 值)

(3) 对于指定的 id\_ethnic\_group\_id 字段, 只删除值不为 1 且不为 NA 的样本个体

(4) 保留 NA 值和值为 1 的样本

(5) 将处理后的数据写入新的 tsv 文件中。

## Step 5

**step5.py:** 该脚本的主要功能是剔除 ICD10 中方差为 0 的样本后分成训练集和测试集。具体实现步骤如下:

(1) 读取输入的 tsv 文件。

(2) 删除方差为 0 的列 (所有值都相同的列)

(3) 将数据分为训练集和测试集, 比例为 4:1

(4) 将训练集和测试集写入新的 tsv 文件中。

## Step 6

**step6.py:** 该脚本的主要功能如下:

(1) 读取输入的 tsv 文件。

(2) 遍历数据, 对于缺失值超过 50% 的字段, 剔除该字段。

(3) 遍历数据, 对于缺失的特定字段值, 按照优先级规则进行填充。

(4) 仅保留指定的 124 个字段。

(5) 遍历数据, 对于缺失的其它字段值, 按照填充规则进行填充。

(6) 将处理后的数据写入新的 tsv 文件中。

**特定字段填充规则:**

- (1) **bmi\_x10**: 如果为空, 使用 `bmi_calc_resurvey2 × 10` 或 `bmi_calc_baseline × 10` 填充  
优先级: `bmi_calc_resurvey2 > bmi_calc_baseline`
- (2) **standing\_height\_cm\_x10**: 如果为空, 使用 `standing_height_mm_resurvey2` 或 `standing_height_mm_baseline` 填充  
优先级: `standing_height_mm_resurvey2 > standing_height_mm_baseline`
- (3) **weight\_kg\_x10\_resurvey3**: 如果为空, 使用 `weight_kg_x10_resurvey2` 或 `weight_kg_x10_baseline` 填充  
优先级: `weight_kg_x10_resurvey2 > weight_kg_x10_baseline`

**其它填充规则:**

- (1) 如果列只包含少数几个值 ( $\leq 10$  个唯一值), 按照填充前的比例进行随机填充
- (2) 对于 `id_ethnic_group_id` 这一列, 按照情况 1 处理
- (3) 如果是其他情况, 按照均值填充缺失值

**Step 7\_0base**

该文件夹存放了整个步骤 5 中的基础数据文件。

**Step 7\_1eye**

`eye.py`: 该脚本的主要功能是筛选眼科相关字段。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与眼科相关的字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

**Step 7\_2artery**

`artery.py`: 该脚本的主要功能是筛选动脉相关字段。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与动脉相关的字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

**Step 7\_3physiology**

`physiology.py`: 该脚本的主要功能是筛选生理相关字段。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与生理相关的字段。(不包括年龄)
- (3) 将处理后的数据写入新的 tsv 文件中。

**Step 7\_4all**

`all.py`: 该脚本的主要功能是筛选所有指定字段。具体实现步骤如下:

- (1) 读取输入的 tsv 文件。
- (2) 保留所有指定字段。(不包括年龄)
- (3) 将处理后的数据写入新的 tsv 文件中。

## Step 7\_5age

**age.py:** 该脚本的主要功能是筛选年龄相关字段。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与年龄相关的字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

## Step 7\_6ren

**ren.py:** 该脚本的主要功能是筛选人口学相关字段。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 仅保留与性别、地区和职业相关的字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

## Step 7\_7newall

**newall.py:** 该脚本的主要功能是合并新增的人口学指标和原来指定的所有字段。具体实现步骤如下：

- (1) 读取输入的 tsv 文件。
- (2) 合并新增的人口学指标和原来指定的所有字段。
- (3) 将处理后的数据写入新的 tsv 文件中。

## 1.6 ICD10 数据的模型训练和测试:ICD10train\_test 文件夹

该文件夹和之前的 train\_test 文件夹结构一模一样,也包含了一个一键运行所有模型的脚本 run\_all\_models.py  
值得一提的是在这里运行代码时我发现:

我之前的代码在训练集上进行五折交叉验证时,选出最佳参数后还在训练集上重新进行了训练,再去测试集上评估模型性能,易造成数据泄露,导致过拟合。

现在所有代码已经修改完善。

## 1.7 多模态模型的训练:multi\_model 文件夹

该文件夹下有四个模态的数据:

- (1) 颈动脉模态 train\_artery.tsv 和 test\_artery.tsv
- (2) 眼科模态 train\_eye.tsv 和 test\_eye.tsv
- (3) 生理模态 train\_physiology.tsv 和 test\_physiology.tsv
- (4) 人口学模态 train\_ren.tsv 和 test\_ren.tsv

以及 age 的数据:

- (1) train\_age.tsv
- (2) test\_age.tsv

一个晚期融合的训练脚本:multi\_model\_late\_fusion.py

- 使用了 Stacking 元学习器 (增强特征 + 正则化 + 交叉验证)
- 使用了 Blending (holdout 验证集权重学习)

## 1.8 一些可能用到的文件:copy 文件夹

模型训练时，在五折交叉验证后，选出最优参数，在完整训练集上训练的 simple1\_gpu.py

ICD10 数据合并后，筛选掉非汉族时，不保留 NA 的数据文件 preprocessing4.tsv 和 preprocessing4\_ethnic\_filter\_report.t

训练多模态模型时，在五折交叉验证后，选出最优参数，在完整训练集上训练的 l.py

# Chapter 2 工作内容

## 2.1 模块 1: 数据分析工作

### 方法

采用数据可视化的方法对数据进行分析，主要包括以下几个方面：

- 通过绘制直方图观察各个特征的分布情况。
- 通过绘制箱线图观察各个特征的异常值情况。
- 通过绘制热力图观察特征之间的相关性。
- 最终仅保留了直方图。

### 结果

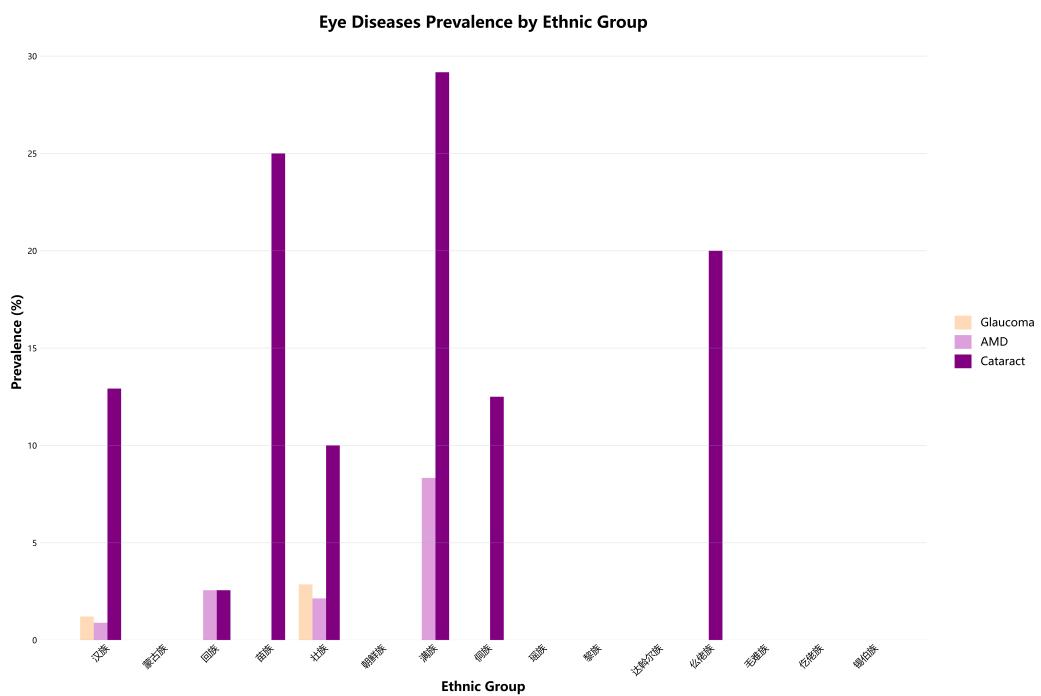


图 2.1: 眼科疾病患病率直方图

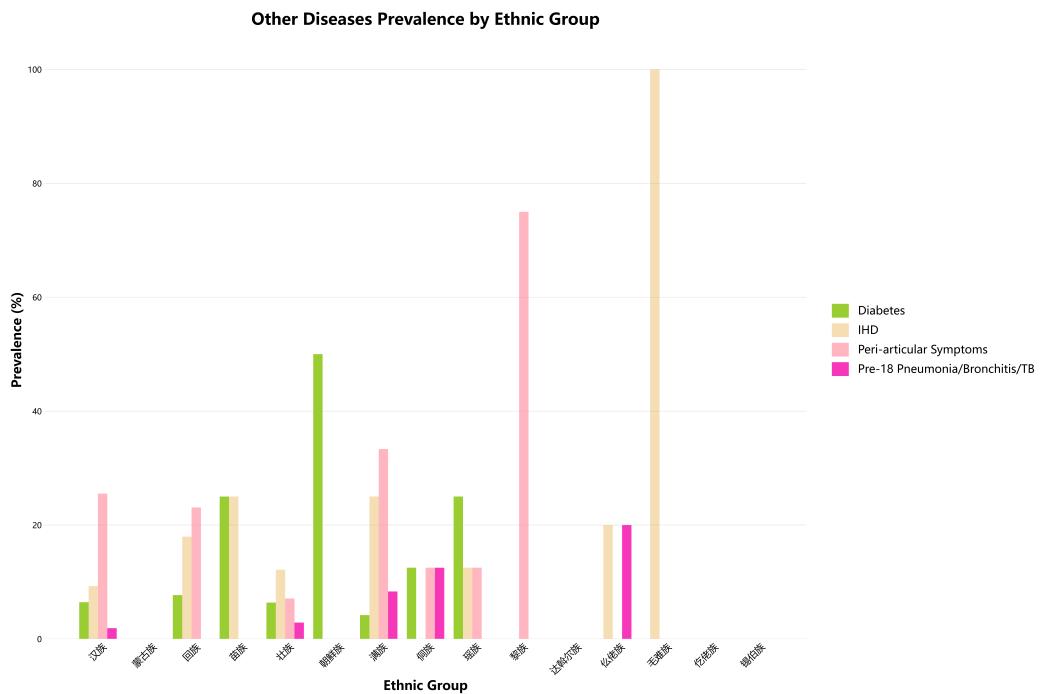


图 2.2: 其他疾病患病率直方图

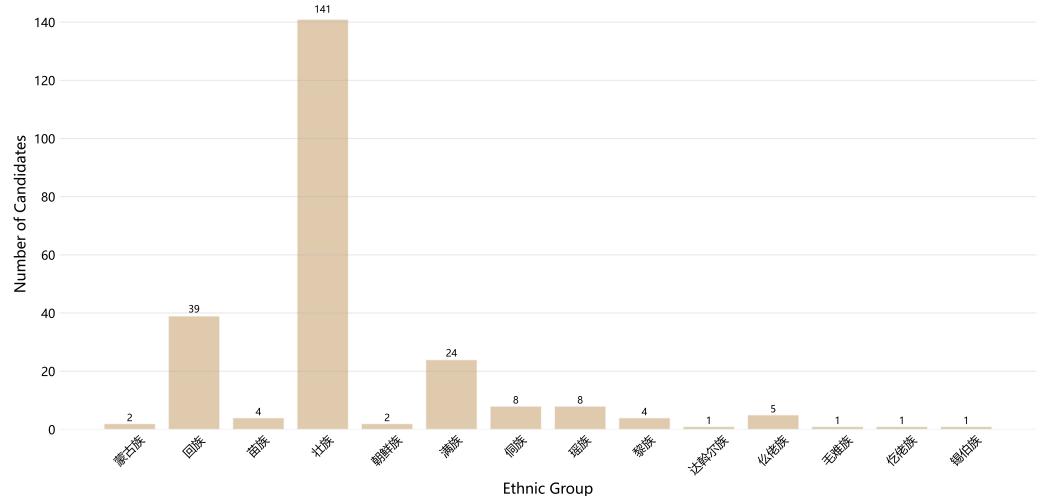


图 2.3: 排除最大值后的样本数量对比直方图

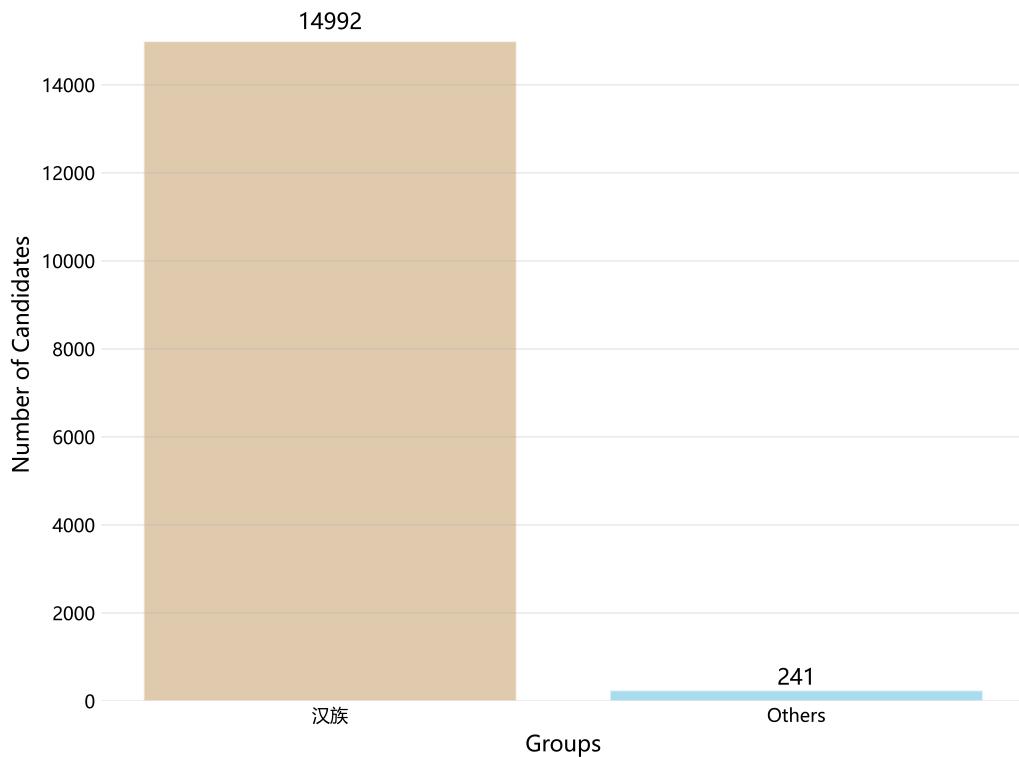


图 2.4: 最大值与其他值的样本数量对比直方图

## 2.2 模块 2: 模型训练工作

### 方法

在模型训练工作中，我们采用了以下方法：

- 使用 XGBoost、LightGBM、CatBoost 和 RandomForest 四种模型进行训练。
- 在训练集上进行五折交叉验证，以 MAE 为指标寻找最佳超参数。
- 校正项计算函数。
- 在测试集上评估四种模型最佳参数的性能。
- 更新了 ICD10 数据后，重新进行了模型训练和评估。

## 结果

负责人	数据	模型	MAE		MSE		RMSE		R-squared		Correlation		CA-corrected correlation test
			Training	test	Training	test	Training	test	Training	test	Training	test	
贾西超	眼部	KDM	2.02	1.97	7.27	7.066	2.7	2.65	0.9022	0.9052	0.9544	0.9558	
贾西超		KDM_naive	11.14	11.48	221.12	240.05	14.87	15.49	-1.974	-2.21	0.5016	0.4883	
贾西超		MLR	5.83	5.86	52.47	53.55	7.24	7.32	0.2942	0.2818	0.5424	0.5318	
王钊瑾		SVM	5.12	5.74	44.55	52.88	6.67	7.27	0.4043	0.2909	0.6372	0.5498	0.877
尹超		RF	5.78	5.81	51.6	52.5	7.18	7.25	0.3057	0.2959	0.5535	0.5441	0.9127
尹超		XGBoost	5.68	5.73	50.18	51.3	7.08	7.16	0.325	0.312	0.5702	0.5586	0.9084
尹超		LightGBM	5.67	5.72	50.16	51.34	7.08	7.17	0.3252	0.3115	0.5704	0.5582	0.9048
尹超		CatBoost	5.66	5.71	50.07	51.18	7.08	7.15	0.3264	0.3137	0.5714	0.5601	0.9059
贾西超		KDM	3.98	3.89	32.99	27.1	5.74	5.21	0.5562	0.6365	0.8322	0.8367	
贾西超	颈动脉	KDM_naive	12.39	9.91	318.68	175.2	17.85	13.23	-3.2864	-1.3495	0.4349	0.4223	
贾西超		MLR	6.18	6.25	58.07	59.47	7.62	7.71	0.219	0.2025	0.4679	0.4508	
王钊瑾		SVM	5.75	6.18	54.04	61.04	7.35	7.81	0.2733	0.1814	0.5303	0.4519	0.9018
尹超		RF	6.12	6.24	57.18	59.07	7.56	7.69	0.231	0.2079	0.4814	0.4572	0.9168
尹超		XGBoost	6.1	6.2	56.79	58.11	7.54	7.62	0.2362	0.2207	0.4868	0.4703	0.9176
尹超		LightGBM	6.09	6.2	56.71	58.66	7.53	7.66	0.2373	0.2134	0.488	0.4657	0.9092
尹超		CatBoost	6.08	6.19	56.82	58.32	7.52	7.64	0.2385	0.2178	0.4891	0.4679	0.915
贾西超		KDM	1.87	1.74	5.5	4.89	2.34	2.21	0.92	0.93	0.96	0.97	
贾西超		KDM_naive	11.88	11.78	222.19	222.04	14.9	14.9	-1.9885	-1.97	0.5007	0.4937	
贾西超	生理	MLR	5.48	5.61	46.81	48.64	6.84	6.97	0.3703	0.3477	0.6085	0.5897	
王钊瑾		SVM	5.06	5.39	42.02	46.1	6.48	6.79	0.4383	0.3818	0.6649	0.6233	0.8787
尹超		RF	5.39	5.49	45.5	46.89	6.75	6.85	0.388	0.3711	0.6233	0.6098	0.9049
尹超		XGBoost	5.29	5.41	44.09	45.16	6.64	6.72	0.4069	0.3944	0.6383	0.6285	0.8978
尹超		LightGBM	5.28	5.4	43.95	45.07	6.63	6.71	0.4088	0.3956	0.6398	0.6294	0.8968
尹超		CatBoost	5.26	5.33	43.67	44.36	6.61	6.66	0.4125	0.4051	0.6425	0.6366	0.8986
贾西超		KDM	0.06	0.04	0.006	0.003	0.08	0.05	0.9999	0.9999	0.9999	0.9999	
贾西超		KDM_naive	28.56	29.35	1448.94	1539.91	38.06	39.24	-18.4886	-19.6508	0.2209	0.1897	
贾西超		MLR	6.28	6.29	59.81	60.06	7.73	7.74	0.1954	0.1943	0.4428	0.4412	
王钊瑾	人口学	SVM	6.05	6.13	58.48	59.7	7.65	7.73	0.2184	0.1994	0.4788	0.458	0.9065
尹超		RF	6.2	6.17	58.64	58.32	7.66	7.64	0.2111	0.2179	0.4608	0.4675	0.9188
尹超		XGBoost	6.2	6.17	58.48	58.19	7.65	7.63	0.2133	0.2196	0.4626	0.469	0.9221
尹超		LightGBM	6.2	6.17	58.55	58.26	7.65	7.63	0.2124	0.2187	0.4619	0.4682	0.9202
尹超		CatBoost	6.2	6.17	58.52	58.31	7.65	7.64	0.2128	0.2118	0.4624	0.4675	0.9198
贾西超		KDM	5.48	5.28	59.86	48.92	7.74	6.99	0.1948	0.3439	0.7443	0.7348	
贾西超		KDM_naive	8.37	7.52	140.03	99.35	11.83	9.97	-0.8835	-0.3324	0.5889	0.5714	
贾西超		MLR	4.58	4.73	33.34	35.03	5.77	5.92	0.5516	0.5303	0.7427	0.7283	
王钊瑾		SVM	3.97	4.6	27.02	33.56	5.2	5.79	0.6388	0.5499	0.7998	0.7425	0.8887
尹超	眼部+颈动脉+生理	RF	4.67	4.81	34.32	36.24	5.86	6.02	0.5384	0.5141	0.7373	0.7194	0.9089
尹超		XGBoost	4.43	4.58	31.58	32.83	5.62	5.73	0.5753	0.5597	0.7586	0.7482	0.8963
尹超		LightGBM	4.4	4.57	31.11	32.83	5.58	5.73	0.5815	0.5597	0.7628	0.7483	0.8986
尹超		CatBoost	4.39	4.54	30.96	32.67	5.56	5.72	0.5835	0.5619	0.7641	0.7498	0.8981
贾西超		KDM	5.52	5.31	60.93	49.7	7.8	7.04	0.1804	0.3335	0.7413	0.7314	
贾西超		KDM_naive	8.31	7.48	137.89	98.49	11.74	9.92	-0.8547	-0.3208	0.5918	0.5734	
贾西超		MLR	4.34	4.48	30.12	31.44	5.49	5.61	0.5947	0.5784	0.7716	0.7605	
王钊瑾		SVM	3.47	4.32	21.53	29.69	4.64	5.45	0.7122	0.6019	0.8444	0.7764	0.8937
尹超		RF	4.62	4.75	33.54	35.37	5.79	5.95	0.5489	0.5256	0.7451	0.7281	0.9107
尹超	眼部+颈动脉+生理+人口学	XGBoost	4.23	4.34	28.74	29.62	5.36	5.46	0.6135	0.6001	0.7833	0.7749	0.9018
尹超		LightGBM	4.22	4.32	28.66	29.31	5.35	5.41	0.6146	0.607	0.7841	0.78	0.907
尹超		CatBoost	4.2	4.37	28.41	30.15	5.33	5.49	0.6179	0.5956	0.7864	0.7721	0.9029

### 《眼部生物学年龄模型表现》

Scan the QR code to open or share with friends.



负责人	数据	模型	MAE		MSE		RMSE		R-squared		Correlation		CA-corrected correlation
			Training	test	Training	test	Training	test	Training	test	Training	test	test
贾西超	眼部	KDM											
贾西超		KDM_naive											
贾西超		MLR											
王钊瑾		SVM											
尹超		RF	5.78	5.81	51.66	52.64	7.19	7.26	0.2987	0.2815	0.5473	0.5335	0.9277
尹超		XGBoost	5.7	5.68	50.48	50.86	7.11	7.13	0.3146	0.3058	0.5612	0.5549	0.918
尹超		LightGBM	5.69	5.62	50.49	50.2	7.11	7.09	0.3145	0.3148	0.5615	0.5623	0.9125
尹超		CatBoost	5.66	5.65	49.98	50.46	7.07	7.1	0.3214	0.3113	0.5674	0.559	0.9133
贾西超		KDM											
贾西超		KDM_naive											
贾西超	颈动脉	MLR											
王钊瑾		SVM											
尹超		RF	6.11	6.1	56.8	56.15	7.54	7.49	0.2288	0.2336	0.4784	0.4835	0.9201
尹超		XGBoost	6.09	6.07	56.56	56.78	7.52	7.54	0.2321	0.225	0.4821	0.4768	0.9124
尹超		LightGBM	6.09	6.24	56.59	59.21	7.52	7.69	0.2317	0.1918	0.4816	0.4567	0.9047
尹超		CatBoost	6.07	6.08	56.22	56.25	7.5	7.5	0.2367	0.2322	0.4867	0.4821	0.9182
贾西超		KDM											
贾西超		KDM_naive											
贾西超		MLR											
王钊瑾		SVM											
尹超	生理	RF	5.37	5.38	45.1	45.17	6.72	6.72	0.3878	0.3834	0.6235	0.6199	0.9068
尹超		XGBoost	5.28	5.34	44.11	44.37	6.64	6.66	0.4102	0.3943	0.6341	0.6268	0.8962
尹超		LightGBM	5.26	5.33	43.77	44.2	6.62	6.65	0.4058	0.3968	0.6376	0.6299	0.8977
尹超		CatBoost	5.23	5.29	43.07	43.81	6.56	6.62	0.4153	0.402	0.6451	0.6341	0.8987
贾西超		KDM											
贾西超		KDM_naive											
贾西超		MLR											
王钊瑾		SVM											
尹超		RF	6.17	6.17	58.05	57.77	7.62	7.6	0.212	0.2114	0.4612	0.461	0.9168
尹超		XGBoost	6.17	6.16	57.94	57.6	7.61	7.59	0.2135	0.2138	0.4624	0.463	0.9196
尹超	人口学	LightGBM	6.17	6.16	57.97	57.61	7.61	7.59	0.2131	0.2137	0.4622	0.4631	0.9182
尹超		CatBoost	6.17	6.17	57.98	57.7	7.61	7.6	0.2129	0.2124	0.462	0.4618	0.9178
贾西超		KDM											
贾西超		KDM_naive											
贾西超		MLR											
王钊瑾		SVM											
尹超		RF	4.73	4.61	34.97	34.1	5.91	5.84	0.5252	0.5346	0.7288	0.7358	0.9141
尹超		XGBoost	4.44	4.39	31.49	31.72	5.61	5.63	0.5725	0.567	0.7569	0.7538	0.8999
尹超		LightGBM	4.42	4.37	31.22	31.34	5.59	5.6	0.5761	0.5722	0.7593	0.7582	0.9034
尹超		CatBoost	4.42	4.35	31	30.7	5.58	5.54	0.5792	0.581	0.7613	0.7627	0.9014
贾西超	眼部+颈动脉+生理+人口学	KDM											
贾西超		KDM_naive											
贾西超		MLR											
王钊瑾		SVM											
尹超		RF	4.66	4.56	33.85	33.15	5.82	5.76	0.5405	0.5475	0.74	0.7449	0.9153
尹超		XGBoost	4.23	4.21	28.54	29.22	5.34	5.41	0.6125	0.6011	0.783	0.7761	0.9029
尹超		LightGBM	4.21	4.19	28.28	28.86	5.32	5.37	0.616	0.6061	0.7852	0.7797	0.9053
尹超		CatBoost	4.21	4.16	28.21	28.43	5.31	5.33	0.6171	0.612	0.786	0.783	0.9049

《眼部生物学年龄模型表现》  
Scan the QR code to open or share with friends.



图 2.6: ICD10 数据整合后所有模型结果对比

# Chapter 3 工作总结

## 3.1 项目总结

### 1. 数据分析与可视化 (chart)

本次数据分析工作主要通过可视化手段，对数据集中的疾病患病率和样本分布情况进行了探索。

- **数据可视化：**使用 Python 脚本（如 `disease_groups_histogram.py`）绘制了多种直方图，以直观展示眼科疾病和其他疾病的患病率，以及在不同筛选条件下样本数量的分布。
- **结果呈现：**最终产出了四张关键的可视化图表(`eye_diseases_bar.png`,`other_diseases_bar.png`,`exclude_max_histogram_ultra_high.png`,`max_vs_others_histogram_ultra_high.png`)，清晰地呈现了数据概况。

### 2. 原始数据修正 (pre\_mod)

- **目的：**确保原始 tsv 文件的格式统一和数据可用性。
- **执行：**
  - 利用 `realign_headers.py` 脚本为数据添加了 `samples` 列标题，保证了列的对齐。
  - 通过 `convert_yes_to_1.py` 和 `convert_no_to_0.py` 脚本，将数据中的“yes”和“no”标记统一转换为 1 和 0，为后续的数值计算做好了准备。

### 3. 数据预处理 (backup)

- **目的：**对数据进行清洗、筛选和划分，以构建高质量的训练集和测试集。
- **执行：**
  - **数据筛选：**剔除了无疾病诊断信息的个体，并根据疾病类型（眼病、其他疾病）将样本划分为训练集和测试集。
  - **数据划分：**使用 `split_train_test.py` 脚本，按 80% 训练集、20% 测试集的比例随机划分了数据集。
  - **缺失值处理：**
    - \* 可视化并剔除了缺失率超过 50% 的字段。
    - \* 对特定字段（如 `bmi`、`standing_height` 等）应用了自定义的填充规则，并对其他缺失值进行了均值填充。
  - **特征选择：**根据不同分析目的，筛选出了眼科、颈动脉、生理代谢、人口学以及所有字段组合，为后续模型训练提供不同维度的数据集。

### 4. 模型训练和测试 (train\_test)

- **目的：**在不同数据集上训练多种机器学习模型，并评估其预测生物学年龄的性能。
- **执行：**
  - **模型选择：**采用了 XGBoost、LightGBM、CatBoost 和 RandomForest 四种主流的决策树类算法。
  - **训练与验证：**在训练集上进行五折交叉验证，以平均绝对误差 (MAE) 为指标，寻找每个模型的最佳超参数。

- **性能评估:** 在独立的测试集上, 评估了每种模型在不同数据组合 (如 all, artery, eye, ren 等) 下的预测性能。

## 5. ICD10 数据预处理 (ICD10)

- **目的:** 将 ICD10 数据与原始 CKB 数据进行整合, 剔除有病的个体。
- **执行:**
  - 使用 icd10.py 脚本, 将 ICD10 数据中的“yes”和“no”标记转换为 1 和 0。
  - 通过 merge\_files.py 脚本, 将处理后的 ICD10 数据与预处理后的 CKB 数据进行合并, 生成了包含更多疾病信息的综合数据集。
  - 重复了数据预处理的步骤 (剔除无诊断信息个体、划分训练测试集、剔除眼病患者等), 确保合并后的数据集质量。
  - 增加了剔除 ICD10 数据中有病个体的步骤。
  - 增加了剔除非汉族个体的步骤 (保留 NA)。

## 6. ICD10 数据合并后-模型训练和测试 (ICD10train\_test)

- **目的:** 在不同数据集上训练多种机器学习模型, 并评估其预测生物学年龄的性能。
- **执行:**
  - **模型选择:** 采用了 XGBoost、LightGBM、CatBoost 和 RandomForest 四种主流的决策树类算法。
  - **训练与验证:** 在训练集上进行五折交叉验证, 以平均绝对误差 (MAE) 为指标, 寻找每个模型的最佳超参数。
  - **性能评估:** 在独立的测试集上, 评估了每种模型在不同数据组合 (如 all, artery, eye, ren 等) 下的预测性能。

## 7. 多模态模型的训练 (multi\_modal)

- **目的:** 通过多模态数据融合, 提高生物学年龄预测的准确性。
- **执行:**
  - 采用了晚期融合。
  - 使用了元学习器。

## 主要成果

- 成功构建了完整的数据处理和模型训练流程。
- 生成了详尽的模型性能对比报告, 为后续研究提供了坚实的数据基础。
- 验证了不同数据维度对生物学年龄预测模型性能的影响。

## 3.2 个人总结

在本项目中, 我深入参与了数据分析、超参数调优、模型训练等多个环节。在师兄们的指导下, 我逐渐掌握了项目的整体流程和各个环节的关键要点, 提升了自己的技术和项目管理能力。也懂得了不能过于依赖 AI 工具, 仍需保持对数据和模型的深入理解。

对于项目的每个环节，我都进行了详细的记录和总结，以便在未来的工作中能够更好地复用这些经验。在总结工作时，我发现了项目仍有很多可优化的地方，例如步骤过分详细导致代码和数据 tsv 文件过多，可以进行步骤的融合，减少不必要的中间文件，节省内存。  
在华大基因的一个月时间，收获良多，感谢师兄们的指导和帮助。

## Chapter 4 AI 工具相关

### 4.1 论文阅读

这里非常推荐使用豆包进行论文阅读，可以直接上传论文的 PDF 文件进行阅读，随时提问。

### 4.2 代码辅助

我使用了多种 AI 工具来辅助代码编写和调试工作，例如：

GitHub Copilot , Cursor 等第三方应用，调用了 Claude Sonnet 3.7/4.0 等模型。

一个较为明显的问题是 AI 工具在处理复杂逻辑时仍然存在局限性，有时无法完全理解开发者的意图，导致代码逻辑混乱或是结构混乱。（李祖琦师兄指出我用 AI 生成的代码有诸多问题）

这警示我们在使用 AI 工具时，必须保持对代码的高度关注和审查，不能完全依赖于 AI 的判断。

### 4.3 问题解答

对于一些术语和概念，我会使用 Gemini 2.5 Flash , Grok 4 , ChatGPT 4o/5 进行解答。