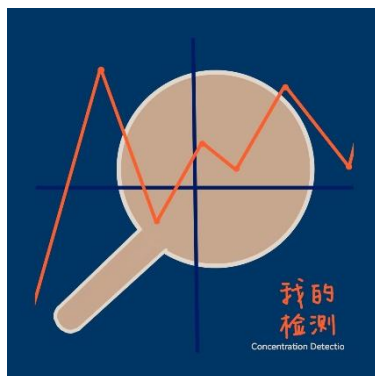


# 2022 年秋《Android 实践》课程设计

## 我的检测 APP



### 一、 小组成员

学号	姓名	工作占比	分工
			线性拟合、错误检测、函数图像绘制以及分工后的功能整合
			模型数据保存本地与删除、模型信息呈现、数据输入自定义控件的设计与实现、各 Activity 间数据传递
			Logo 设计、界面设计与美化、加载界面的实现
			图片上传、图片裁剪、浓度预测
			预测结果的呈现与分享、APP 功能测试

## 二、 使用说明

我的检测 APP 完整实现了检测农药 2, 4-二氯苯氧乙酸浓度的功能, 并具备美观简洁的界面呈现、出色的健壮性和丰富的设计细节, 适用于相关农药检测领域。

我组 APP 设计的功能与页面贴近于目前市面上的常见应用软件, 其主要功能及操作流程如下:

点击“我的检测”图标, 打开程序进入加载页面等待, 后进入程序的首页, 即模型列表界面, 页面展示出所有已经创建的模型。点击模型列表里的模型, 可以选择使用它或者删除, 点击使用它即进入模型信息界面。

在模型列表界面点击下方的创建模型按键, 进入创建模型界面。在创建模型界面, 初始会呈现 3 个输入图片和对应浓度的自定义控件, 点击自定义控件的“+”图标即可上传图片, 图片可以来自相册或拍照, 同时提供了裁剪功能。在界面的上方文本框输入模型名称。点击下方返回按键可以返回上一界面, 点击添加按键可以添加自定义控件以添加样本, 点击拟合按键即可进行拟合操作。进行拟合前会进行一些列检查: 检查是否输入模型名称、样本数是否小于 2 个, 通过此处检查后会展示采集到的样本数量, 供用户确认。拟合前先进行图片灰度值的计算, 灰度值作为自变量, 浓度作为自变量, 进行正式拟合前, 先计算变量的相关系数, 若相关系数过小, 认为不存在相关性, 会拒绝拟合并提示用户。拟合时, 先进行最小二乘的计算, 将最小二乘的结果作为斜率和截距参数的初值, 以平均百分比误差的平方作为损失函数, 通过梯度下降法进行训练 200 轮 (100 轮后调小学习率), 实现参数的微调。如果出现斜率为 NAN 的情况, 则说明无法拟合得到单值函数, 无法用于预测, 则拒绝保存模型, 并提示用户。拟合成功后则保存模型

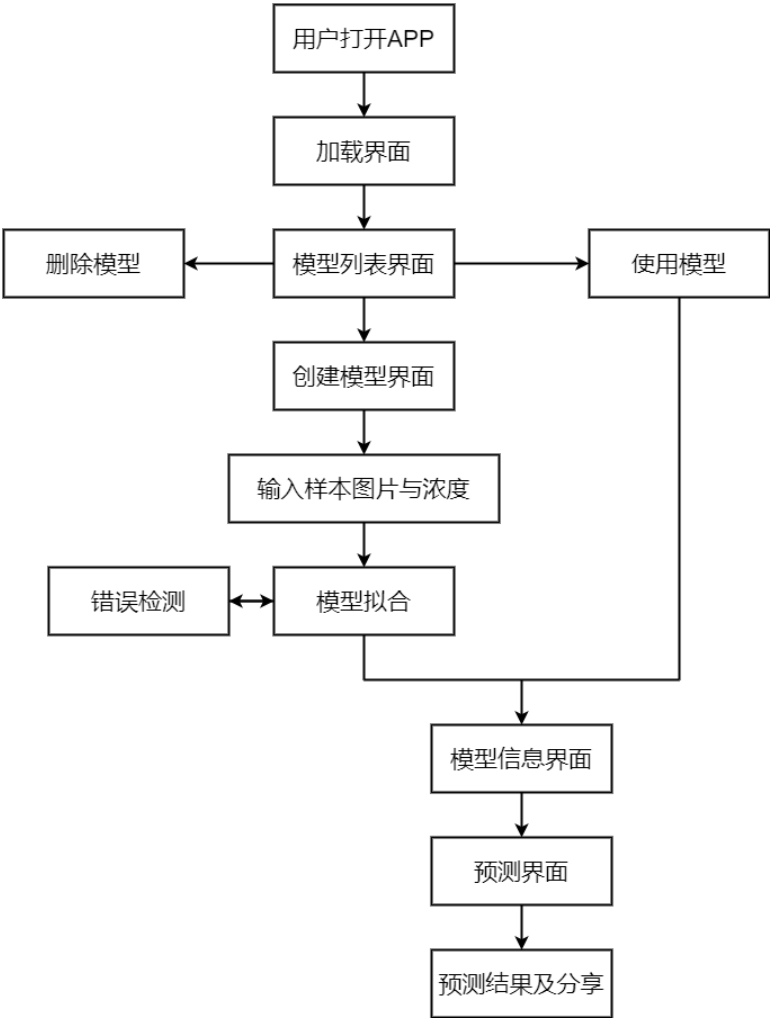
到本地，并进入模型信息界面，显示该模型的信息。

在模型信息界面，显示了模型的拟合函数的图像，和模型的名称、创建时间、方程、百分比误差等信息。点击下方返回按钮可以返回模型列表界面，点击预测按钮可以进入预测界面，使用该模型进行预测。

在预测界面，点击上方“+”图标可以上传图片（效果同创建模型界面），在下方文本框输入阈值，再点击下方的确定按钮进行预测，预测前会进行图片是否成功上传、阈值是否输入的检测。也可点击返回按钮返回模型信息界面。

预测后会根据输入的阈值和模型计算结果判断是否达标，在界面上方显示是否达标及输入的阈值，并依次显示模型函数图像和检测的图片，下方也提供了返回和分享按钮，点击分享按钮可以将预测结果界面分享到 QQ、微信等 APP。

### 三、 功能简介



### 四、 实现思路

#### 1. 加载界面

创建一个名为 Start 的 activity，将其设为初始 activity。通过 Handler 启动一个新线程，设置延迟时间，以显示加载界面，并跳转到 MainActivity，即模型列表界面。

## 2. 模型的本地存储

全部的模型信息都保存在了 MainActivity 的 MapData 中类型为 List<Map<String,String>>类型的 subject 中，为了便于保存至本地，考虑对象序列化，这里使用 Gson 库，可以用来在 Java 对象和 JSON 数据之间进行映射，存储时只需要存一个 JSON 字符串即可，而不用对不同模型的不同表项操作，易于实现，便于操作，而且不易出错。

数据直接存储在了/data/data/com.mas.name4/files/MapData.txt 中，每次打开 app 时，都会读取这个文件的字符串，进行反序列化操作。在每次对模型进行修改时，如新建模型或删除模型时，也会重新向 MapData.txt 中重新写入修改。

## 3. 创建模型的样本数据输入

在创建模型时，需要上传多组图片和对应浓度，为了便于操作和提升用户体验，我们通过 ListView 和 Adapter 实现，多组数据都能体现在一个界面中，上下滑动，传入图片，输入对应浓度数据。样本的数目理论上来说可以一直增加，点击下方的【添加】按钮便可以增加一个条目。

每次在最下方添加一个条目时，需要刷新一下 ListView，为了体验将滚动条设置到屏幕最下方，直接就可以看到新添加的条目；在每次添加好一张图片之后，也需要刷新 ListView，同样，为了用户体验，滚动条设置在添加图片之前的位置，能够带给用户更便捷的使用体验。

当所有样本添加完毕以后即可开始计算模型，同一个条目只上传图片或只输入浓度不会当做有效样本，只有既上传图片又输入浓度才算作一个有效样本。考虑到样本较多的时候，用户可能会漏传数据，最后在点击计算按钮时会提示用户

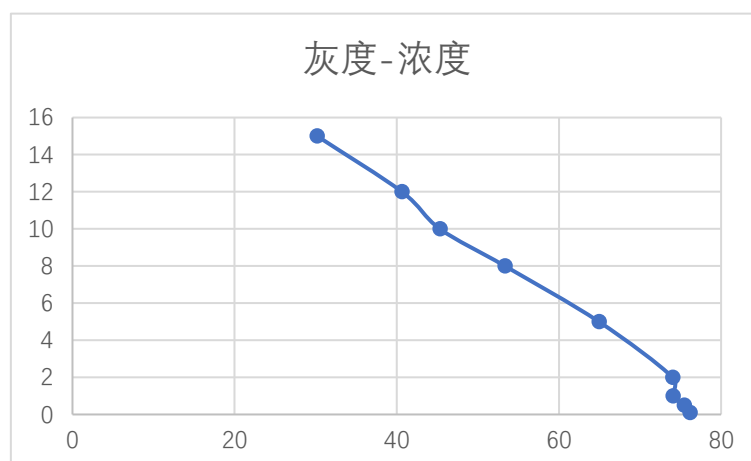
已经读取到的有效样本个数，只有点击确定时才会进行计算。

## 4. 图片上传及裁剪功能

考虑到用户可以通过直接上传本地照片或者调用相机直接拍照后上传，我们对这两种方式都提供了支持，每当用户点击上传图片的按钮时，便会弹出一个对话框，可以选择拍照或上传已有照片，这里通过系统功能即可。当选择好照片后需要对照片进行裁剪。

原计划调用系统的裁剪功能 `com.android.camera.action.CROP`，但由于安卓系统的改进及安全保护的增强，在实际体验中此功能在大多 Android11 及以上系统中，由于文件路径的问题无法正常使用。我们不得不考虑其他裁剪方式。经过对比，我们采用开源的 `uCrop` 框架<sup>1</sup>，在模块的 `build.gradle` 中导入 `'com.github.yalantis:ucrop:2.2.6'`，然后在返回照片的回调函数中调用 `uCrop` 即可，裁剪效果达到预期。

## 5. 拟合方程



根据提供的 9 组采样数据的折线图（如上图所示），我们选用一次函数对灰

---

<sup>1</sup> <https://github.com/Yalantis/uCrop>

度值和浓度的函数关系进行拟合，以求取得 25%以下的平均百分比误差。由于并没有直接求解最小百分比误差一次函数的公式或通解，我们考虑采用线性回归的做法，以最小百分比误差的平方作为损失函数（考虑到平均百分比误差中存在绝对值函数，难以求导），采用梯度下降的方式训练模型至参数收敛。考虑到最小二乘法可以求解得到最小平均平方误差的一次函数，此时函数已经具有很好的拟合效果，故我们将最小二乘的求解结果作为模型参数的初值，再针对最小百分比误差进行参数微调。注意到以 0.0001 的学习率训练 100 轮后，模型并没有收敛，我们在 100 轮后将学习率调整为 0.00002，再训练 100 轮后，模型收敛。我们利用此模型，对多组不同的具有强相关系数的变量进行拟合，都得到了很好的效果。

## 6. 拟合时的错误检测

由于采样数据存在误差，也可能存在坏点，仅使用小样本训练后得到的模型也可能有好有坏。在拟合效果糟糕的情况下，我们会提示用户，不予建立模型，主要考虑了以下两种情况：

- (1) 采样数据的灰度与浓度间相关系数 $<0.3$ ，我们认为不存在相关关系，通过在拟合前的计算实现。
- (2) 存在多个灰度相同但浓度不同的采样数据，无法拟合出合适的单值函数，多值函数显然无法用于预测，故拒绝建立模型，通过拟合后对参数是否为 NAN 的检查实现。

## 7. 拟合曲线的绘制

原计划采用 JFreeChart 绘制函数图像，但受限于 Android 不能调用 java.awt，

我们不得不考虑其他绘图方式。经过对比，我们采用开源的 MPAndroidChart 框架<sup>2</sup>，利用其中的 LineChart 自定义控件实现了图像的绘制，实现了直观的视觉效果。

## 8. 浓度预测

通过计算用户上传的图片的灰度值，根据模型的回归方程求解该灰度值下的预测浓度，并与用户输入的阈值作比较给出是否超标的提示。

## 9. 分享检测结果

点击分享按钮时，app 会对当前界面进行截屏，并将图片保存至相册 DCIM 目录下，便于后期查找、删除等操作。当保存好截图后，通过系统功能实现分享功能。

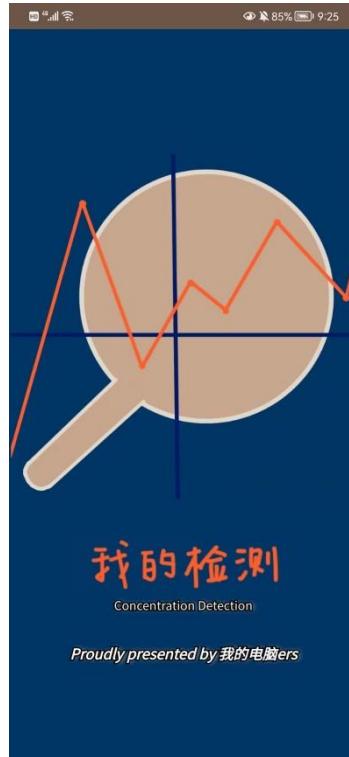
# 五、 结果展示

加载界面

---

<sup>2</sup> <https://github.com/PhilJay/MPAndroidChart>

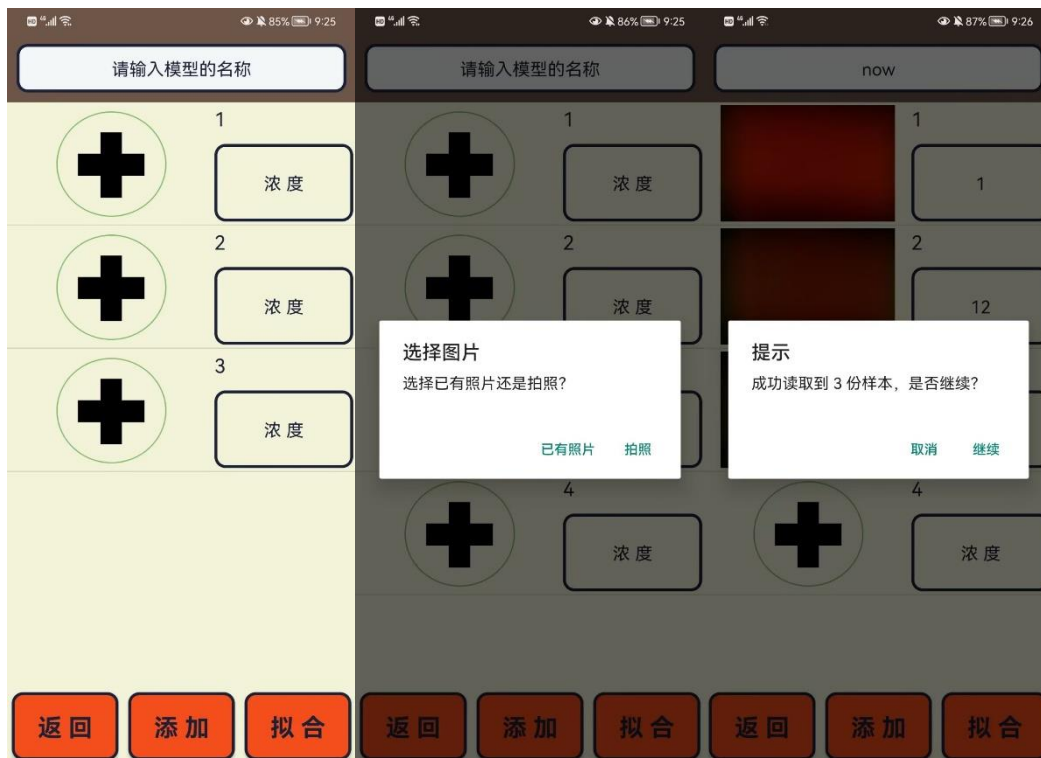




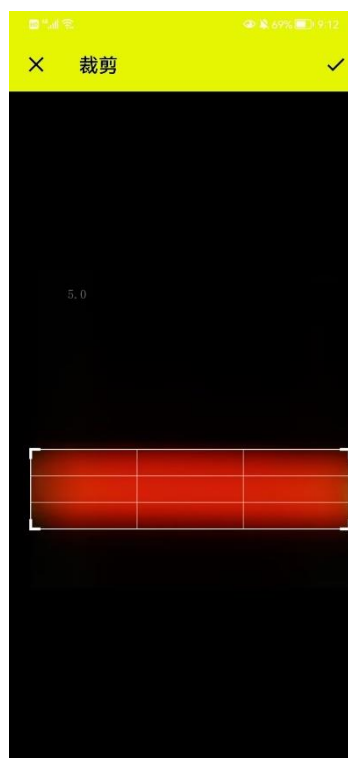
模型列表界面



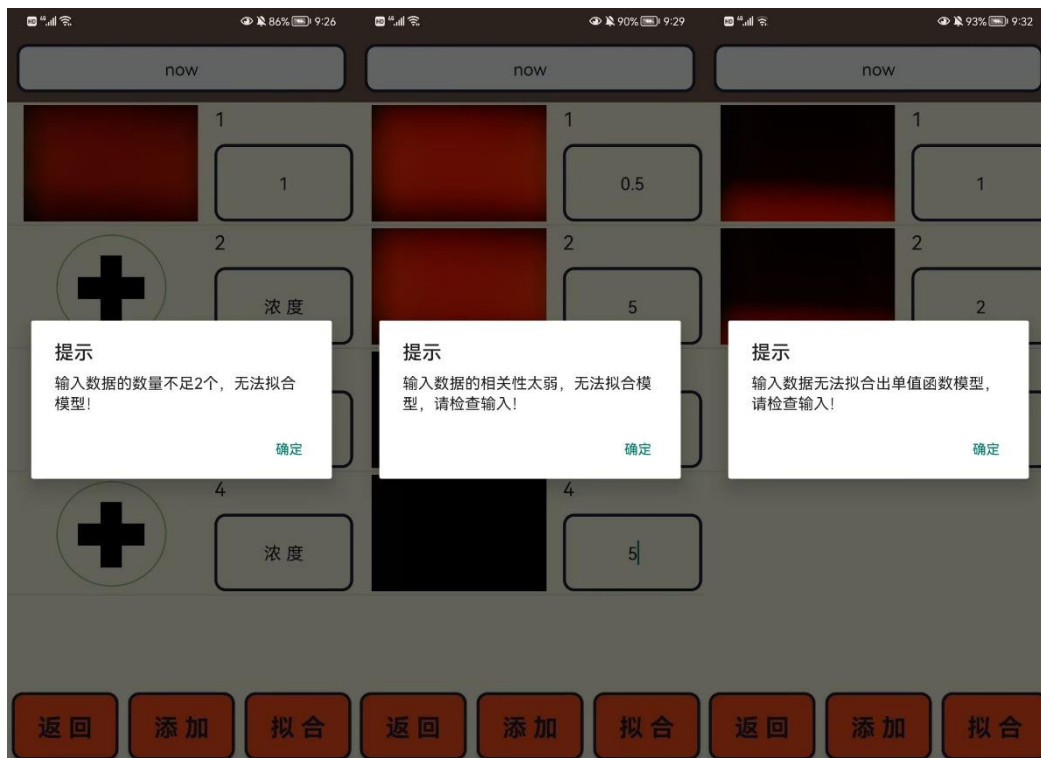
创建模型界面及数据上传



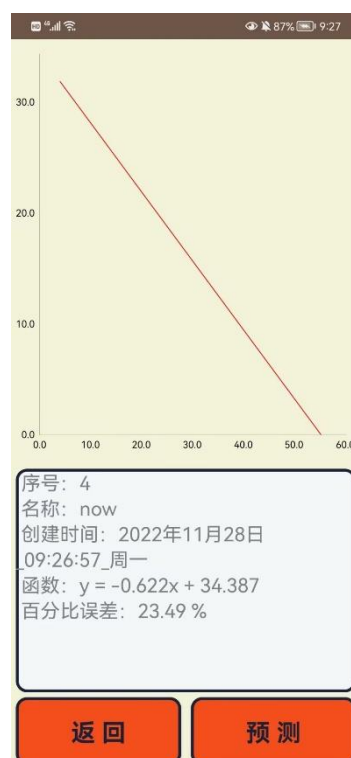
图片裁剪



模型拟合的错误检测



模型信息界面



预测界面



## 预测结果及分享

