**天津师范大学管理学院**

**结课报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 上课学期： |  | 课程名称： |  |
| 姓 名： |  | 学 院： |  |
| 班 级： |  | 学 号： |  |

**参考评分标准：（命题教师可根据课程考核要求自行制定评分标准）**

|  |  |  |  |
| --- | --- | --- | --- |
| **项目** | **指标** | **分值** | **评分** |
| 开发文档完整性 | 包含了规划、分析、设计、实施、运行维护等全部环节必要内容 | 10 |  |
| 开发文档规范性 | 图形符合规范、文字介绍清晰流畅，格式美观，没有错别字等低级错误 | 10 |  |
| 源代码 | 源代码完整清晰，重点功能部分包含了必要的注释，能当场解释代码的功能 | 5 |  |
| 运行可靠性 | 可以正常运行**全部**功能，中间没有出现错误或者崩溃，考虑到安全因素 | 5 |  |
| 功能完整性 | 最终系统实现的功能不少于分析阶段的计划，且必要功能完备，至少包含5个非管理性功能 | 5 |  |
| 交互性 | 界面美观，交互操作易于上手，用户学习和使用成本低 | 5 |  |
| 成功交付 | 系统满足了用户需求，用户签字认可该系统可以解决实际痛点 | 30 |  |
| 公益类别附加 | 经监理组审定，题目属于公益范畴 | 10 |  |
| 其他附加分数 | 上级优胜组附加分，期末演示优秀组与优胜组附加分 | 20 |  |
| 合计 | | 100 |  |
| 评语  阅卷人  年 月 日 | | | |

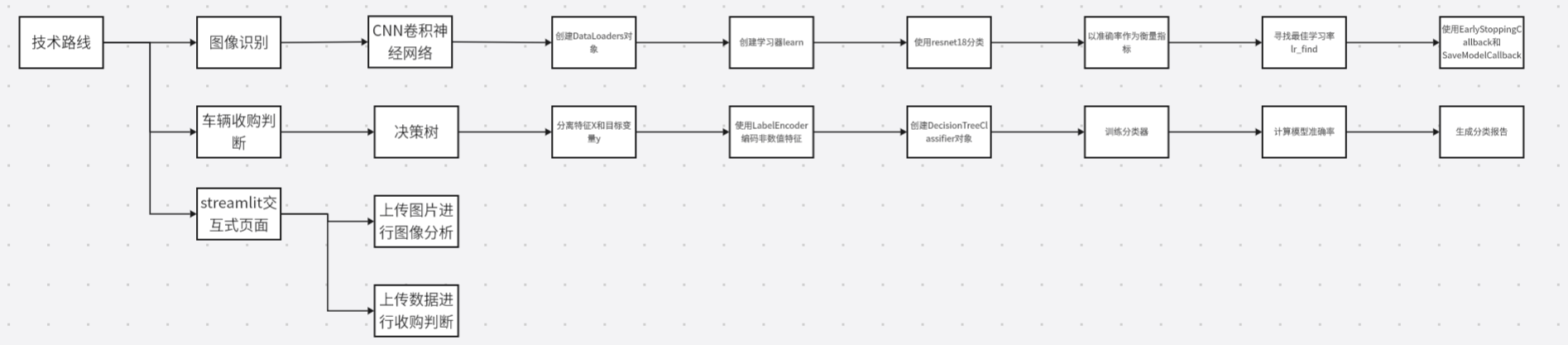
深度学习期末报告

引言

如今，汽车已经成为我们生活中必不可少的一部分。随着经济的发展和汽车行业的日益成熟，二手汽车收购已经成为我们生活中普遍的一件事。越来越多的人选择购买二手车作为代步工具，因为它既经济实惠，又能满足基本的出行需求。二手车市场因此迎来了繁荣，收购二手汽车的业务也日益兴旺。从个人到企业，从线上到线下，二手汽车收购已经成为一种常见的交易方式。这不仅有助于资源的合理利用，也推动了汽车行业的可持续发展。

本次设计采用CNN卷积神经网络和决策树设计出评判汽车是否适合收购的系统。

1 技术路线

本APP设计主要分为两部分，第一部分图像识别，第二部分为通过用户填写数据判断该车辆是否适合收购。

* 1. CNN卷积神经网络

通过CNN卷积神经网络捞实现对图像的识别，创建一个DataLoaders对象来训练图像分类模型，创建学习器learn，使用resnet18对图像进行分类，以准确率（metrics=accuracy）作为训练时的衡量指标，并寻找最佳学习率 lr\_find。使用fastai库中的EarlyStoppingCallback和SaveModelCallback来改进模型训练过程。

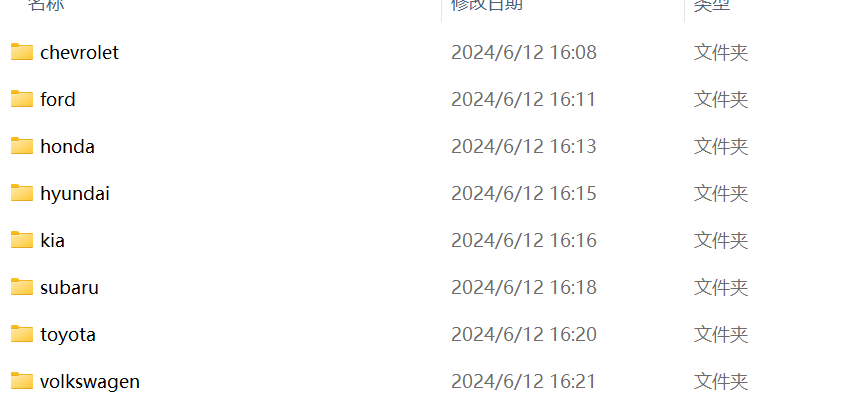
* 1. 决策树

使用决策树方法实现对汽车是否适合收购的预测，从数据中分离出特征（X）和目标变量（y），特征是用于预测的数据，而目标变量是我们要预测的结果，即车辆是否适合收购。使用LabelEncoder对非数值特征进行编码，将其转换为数值类型，以便决策树分类器可以处理这些特征。创建一个DecisionTreeClassifier对象，并将其应用于训练集数据以训练分类器。最后，计算模型的准确率，并生成一个分类报告，其中包含了精确度、召回率和F1分数等评价指标。

1.3 streamlit交互式页面

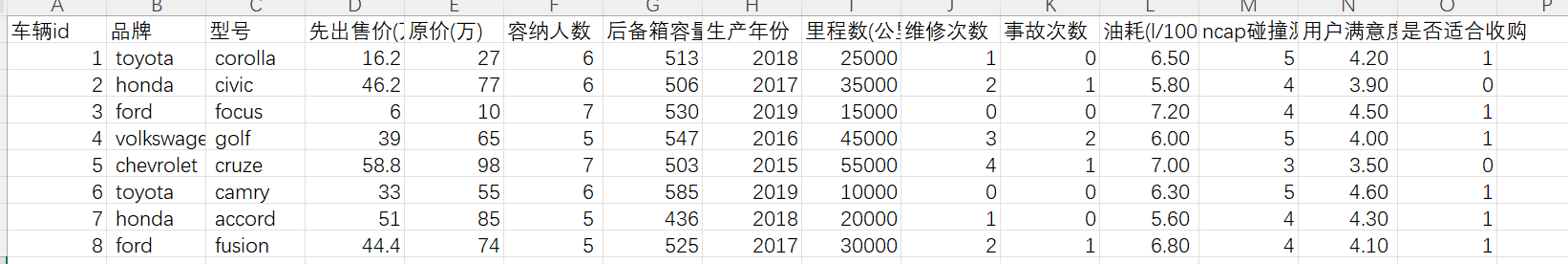
使用streamlit将两部分呈现在一个网页中，用户可以通过上传图片对汽车品类进行分析，并通过上传有关汽车售价、原价、容量等数据实现对汽车是否适合收购的预测。

1. 数据说明
   1. 图片数据

从网络中收集chevrolet、ford等8个品牌车数据各20张，确保所有图片正确匹配且没有重复，存放在以品牌命名的文件夹下。

* 1. 用户数据

从网络中收集有关二手车收购的数据，整理存放在Excel表格中，数据包含车辆id、品牌、型号、售价(万)、原价(万)、容纳人数、后备箱容量(升)、生产年份、里程数(公里)、维修次数、事故次数、油耗(l/100km)、ncap碰撞测试评分、用户满意度评分、是否适合收购，共1030条。



3 CNN图像分类

3.1数据集导入

在环境下创立文件夹，将已经准备好的图片根据汽车品牌粘贴在对应文件夹名称的下，作为数据集。

3.2模型的构建

首先定义一个函数，用于将数据集分割为训练集、验证集和测试集，然后创建目标路径的目录结构，包括train、valid、test和每个类别下的子目录，遍历每个类别，为每个类别中的图片分配到不同的数据集，根据比例计算每个数据集中应包含的文件数量 ，分配文件到不同的数据集，遍历训练集、验证集、测试集文件并复制到目标路径。

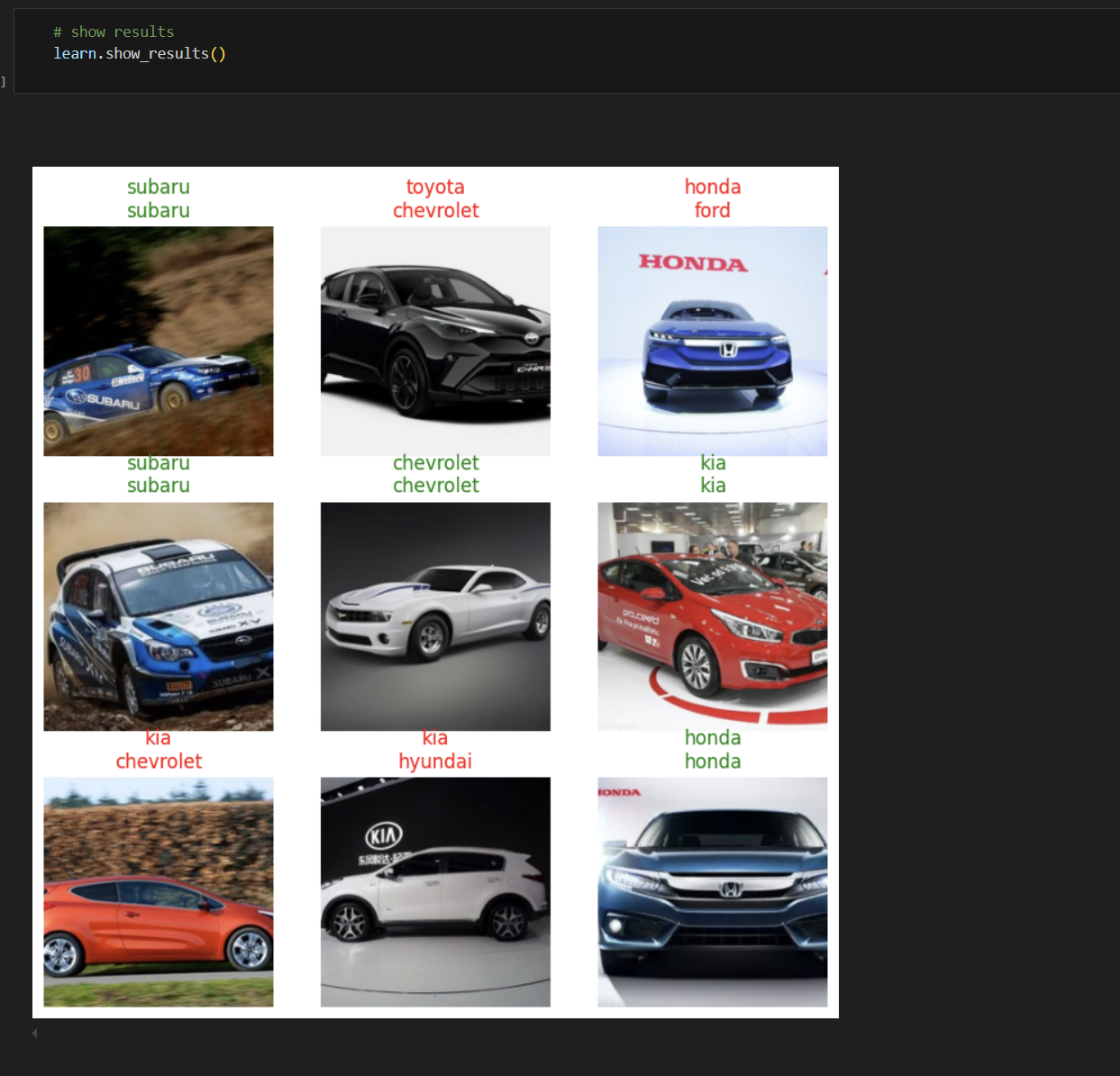
将数据集的所有图像调整为统一的大小，然后从数据集中加载图像数据，并验证数据的加载以及预处理是否正确。在进行图像分类训练采用CNN卷积神经网络，使用vision\_learner函数创建一个学习器对象。该对象封装了模型、数据加载器（dls）和优化器等信息。metrics=accuracy表示在训练和验证过程中将使用准确率作为评估指标。最后，使用fine\_tune方法开始训练模型。这里，5表示训练5个epoch（即整个数据集将被遍历5次）。3e-3是学习率，即在训练过程中用于更新模型权重的步长大小，在这里我分别用了3e-3、5e-3以及6e-3进行模型的训练，这样使训练的结果更好，准确率更高。

最后就是模型的使用和图片的预测结果。

3.3结果

3.3.1图像分类预测

对于测试集中的汽车图片进行训练过后可以基本准确地将机器学习出来的图片中的汽车名称与真正的汽车名称匹配上，如下图所示：

分类报告如下：

precision recall f1-score support

chevrolet 0.38 0.75 0.50 4

ford 0.29 0.50 0.36 4

honda 0.50 0.25 0.33 4

hyundai 0.50 0.25 0.33 4

kia 0.00 0.00 0.00 4

subaru 1.00 0.75 0.86 4

toyota 0.43 0.75 0.55 4

volkswagen 0.50 0.33 0.40 6

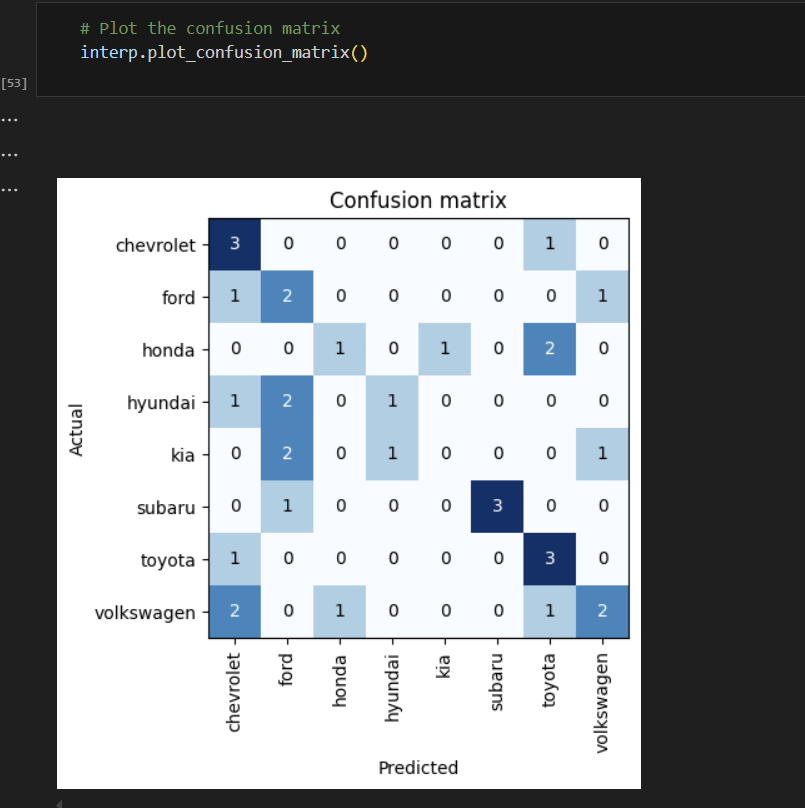
accuracy 0.44 34

macro avg 0.45 0.45 0.42 34

weighted avg 0.45 0.44 0.42 34

3.3.2混淆矩阵

混淆矩阵通过比较模型对每个类别的真实标签和预测标签来评估模型的准确性。机器学习模型对'chevrolet', 'ford', 'honda', 'hyundai', 'kia', 'subaru', 'toyota', 'volkswagen'这几个汽车品牌的分类性能进行了评估。图中的条形图显示了每个类别的真实标签与预测标签之间的对比，其中蓝色和灰色可能分别代表正确分类和错误分类的样本数量。此外，图中还列出了各个品牌的名字，并在最右侧标注了'Predicted'作为预测结果的标题，图片展示了模型对多个汽车品牌的分类性能的混淆矩阵，如图 ：



4决策树分类

4.1数据集划分

代码使用train\_test\_split函数将数据集划分为训练集和测试集，其中test\_size参数设置为0.2，这意味着20%的数据将被分配到测试集中，其余80%的数据用于训练。

4.2 模型构建与环境选择

（1）数据准备：使用pandas库的read\_excel函数读取Excel文件中的数据，并将其存储在DataFrame对象 data中。从DataFrame中分离出特征（X）和目标变量（y）.



（2）数据预处理：使用LabelEncoder将X中的非数值特征转换为数值，以便机器学习算法能够处理。

（3）数据划分：使用train\_test\_split函数将数据集划分为训练集和测试集。测试集的大小被设置为原始数据集的20%。

（4）模型构建与训练：创建一个DecisionTreeClassifier对象（clf），即决策树分类器。random\_state参数被设置为42，以确保结果的可复现性。使用训练集数据X\_train和y\_train来训练这个分类器。

（5）模型评估：是验证模型性能的重要步骤。模型对测试集X\_test进行预测，并生成预测值y\_pred。使用accuracy\_score计算预测的准确率，使用classification\_report生成更详细的分类报告，包括每个类别的精确度、召回率和F1分数等。

结果如下：

precision recall f1-score support

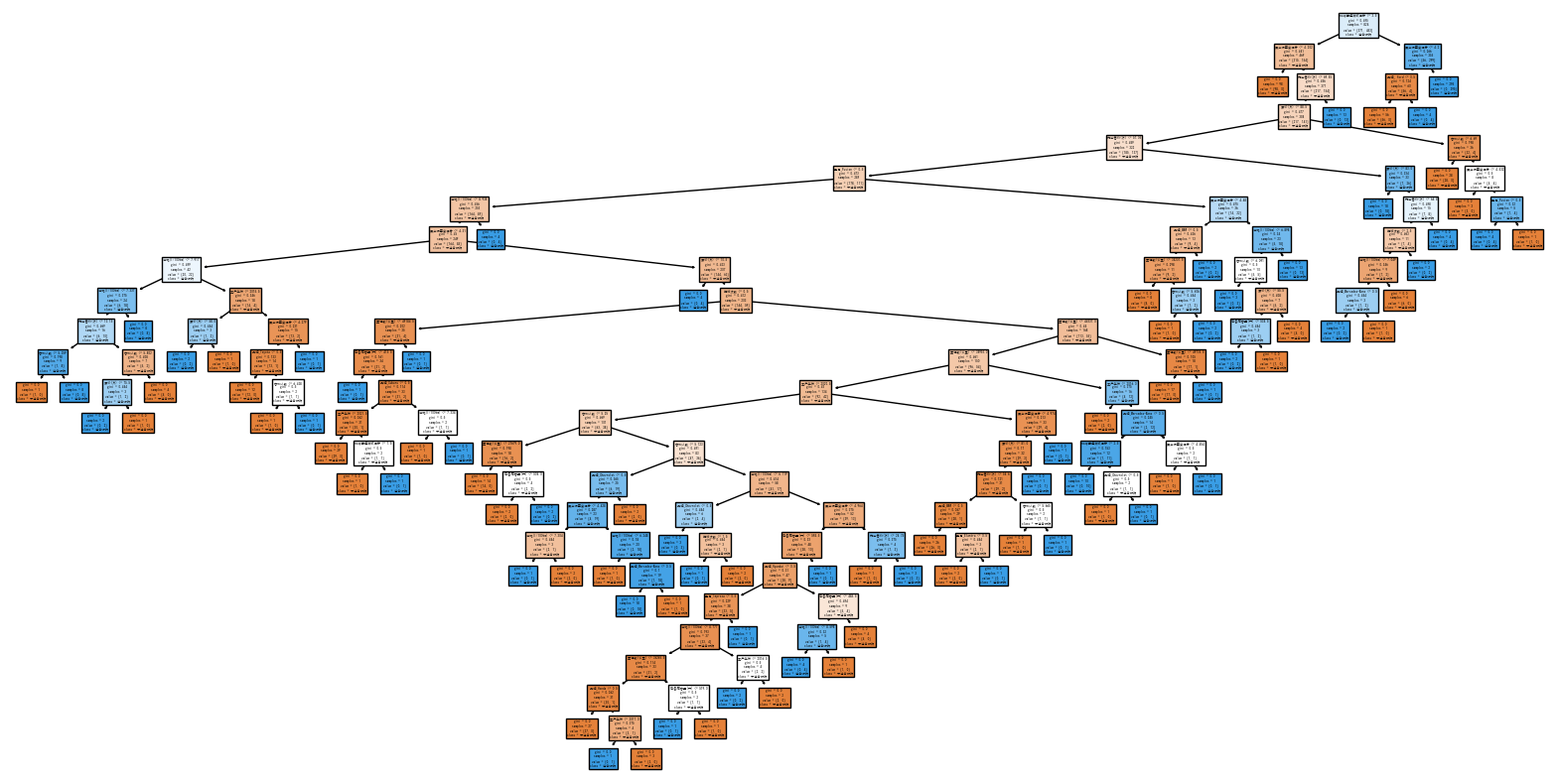
0 0.76 0.77 0.76 98

1 0.79 0.78 0.78 108

accuracy 0.77 206

macro avg 0.77 0.77 0.77 206

weighted avg 0.77 0.77 0.77 206



（6）保存模型：将训练好的模型通常保存下来，以便后续使用。

5交互式

定义了一个使用Streamlit框架的Web应用程序，它结合了图像识别和机器学习预测的功能，由汽车图像分类应用和车辆收购适宜性预测两部分组成。旨在帮助用户通过上传图片来识别汽车类型，以及根据车辆的具体特征来预测其是否适合购买。这个程序可用于汽车销售或二手车市场，帮助顾客或销售人员快速判断一辆车是否值得购买。它也可用于车辆评估服务，为车主或潜在买家提供关于车辆购买建议的参考。

以下是该脚本的主要功能和工作流程：

1. 环境设置：根据操作系统的不同，动态调整pathlib模块以确保路径处理的正确性。

2. 加载模型：确定当前脚本文件的路径，并加载一个预训练的图像分类模型（cars1.pkl）。

3. 创建Streamlit应用：设置应用的标题为“汽车图像分类”。 提供用户界面，允许用户上传图片（支持jpg、jpeg、png格式）。

1. 图片上传与显示：如果用户上传了图片，应用会显示这张图片，并使用加载的模型对图片进行分类预测。
2. 图像分类预测：对上传的图片进行预测，显示预测的汽车类型和相应的概率。

6. 加载另一个模型：加载另一个预训练的决策树模型（treemodel.pkl），用于预测车辆是否适合购买。

7. 定义预测函数：创建一个名为predict\_suitability的函数，它接受车辆的多个特征作为输入参数。

8. 特征处理：使用LabelEncoder将非数值型特征转换为数值型，以便模型能够处理。

9. 车辆适宜性预测：根据用户输入的车辆特征，使用决策树模型进行预测，并返回车辆是否适合购买的结果。

10. 用户界面交互：在Streamlit应用中，提供文本输入框、数字输入框、选择框、滑动条等控件，让用户输入车辆的各种特征信息。

11. 显示预测结果：用户点击“预测”按钮后，应用会调用predict\_suitability函数，根据模型的预测结果显示“该车辆适合收购”或“该车辆不适合收购”。

12. 启动应用：通过`if \_\_name\_\_ == "\_\_main\_\_":`确保当脚本被直接运行时，会执行main()函数，启动Web应用程序。

