基于网络汽车信息消费后对汽车投诉的问题分析报告

在这个信息高速发展的时代，生活中的信息网络化的产生，使得越来越多的网络信息成为如今社会判断商品质量等问题的主流趋势，它以范围广、方便迅速、不受时空限制等区别于传统询问和汽车使用过程中产品问题获取的优势成为越来越多买家和卖家渠道，在买家考虑购买汽车中存在的问题方面和卖家需要改进汽车的性能等问题中起着至关重要的作用。同时，随着互联网、云计算和大数据产业的兴起，面对海量的网络数据，数据分析、数据挖掘等相应行业也正快速发展。网络汽车信息消费投诉问题反映着汽车行业及品牌的发展现状，各汽车品牌发展水平，不同人对汽车品牌的性能、价格和存在的问题的评价，以及对汽车品牌的改进的发展动向都有着最及时有效的传达。因此，对网络汽车投诉信息进行分析研究，了解不同汽车品牌的在使用过程中存在的问题，挖掘汽车行业对汽车品牌的性能等改进的需求现状及发展趋势，为汽车品牌的发展有着重要意义。

本文利用Python网络爬虫，爬取汽车消费网（<http://tousu.315che.com/>）全国范围内与各个品牌车辆车型，对汽车品牌信息进行分析挖掘，把分析结果提供给买家卖家做参考。

# 了解汽车品牌投诉现状

由于人们对汽车行业存在的问题还不够了解，无法一时间提出自己所需要的信息，需要大量对网络汽车投诉信息进行了解，通过对汽车所存在的问题，为买家获取汽车信息所存在的问题进行了分析，为卖家提供了汽车品牌存在的问题，使汽车公司能够改进汽车产品进行分析，本文利用Python网络爬虫，爬取汽车消费网——汽车投诉（按汽车品牌）（<http://tousu.315che.com/tousulist/serial/55467/>）全国范围内与数据相关汽车投诉信息，其中爬取的信息包含有：**投诉单号、品牌、车型、投诉时间、诉求问题、投诉明细、经销商、解决状态**等信息。通过Pandas、re库的正则表达式处理爬取数据，包括数据清洗（缺失值处理、异常值处理），数据变换（规范化、一致化），然后利用matplotlib库进行数据可视化，得出数据类岗位的相关信息。

可以通过以下5个步骤来了解汽车品牌投诉问题现状。

1. 爬取汽车消费网范围内300多个汽车品牌。
2. 分析比较每个品牌投诉的数量情况，并进行可视化呈现。
3. 分析比较每个品牌车型的投诉情况，并进行可视化呈现。
4. 分析比较每个品牌投诉单号、品牌、车型、投诉时间、诉求问题、投诉明细、经销商、解决状态，从而分析品牌存在的问题。
5. 某种品牌哪个车型经常被投诉、各种汽车被投诉的次数分布、汽车故障经常发生的时间段

## 分析汽车品牌投诉信息数据

汽车消费网的投诉信息，从汽车消费网中采集数据集。字段包含了投诉单号、品牌、车型、投诉时间、诉求问题、投诉明细、经销商、解决状态，行业8个特征，如表 1‑1所示。

表 1‑1 汽车品牌投诉信息数据特征说明

|  |  |  |
| --- | --- | --- |
|  | 特征名称 | 例子说明 |
| 汽车品牌投诉信息 | 投诉单号 | 11084 |
| 品牌 | 奥迪 |
| 车型 | 奥迪A4 |
| 投诉时间 | 2019-06-09 23:12:00 |
| 诉求问题 | 天窗漏水，厂家不负责 |
| 投诉明细 | 我买车以来1年内就发生漏水七八次，我买的是车不是说船啊，厂家都不管的么，信不信我去315投诉，退款 |
| 经销商 | 龙华奥迪 |
| 解决状态 | 未解决 |

结合汽车品牌投诉信息的数据情况，可以实现以下目标。

1. 数据获取，利用爬虫工具爬取汽车品牌投诉的信息。
2. 数据预处理：重复值处理、投诉单号处理、品牌处理、车型处理、投诉时间处理、诉求问题处理、投诉明细处理、经销商处理、解决状态处理。
3. 对不同字段进行可视化分析，提出汽车品牌改进和购买建议。

## 分析方法与流程

汽车品牌投诉信息分析文的总体流程，主要包括以下4个步骤

1. 爬取汽车消费网的汽车品牌投诉的数据。
2. 对爬取的数据进行数据清洗操作。
3. 从不同的维度分析汽车品牌的信息，并进行可视化。
4. 针对不同的分析情况提出建议。

# 爬取汽车品牌投诉信息

根据分析要求，爬取汽车品牌、汽车品牌车型、汽车品牌投诉数量等相关岗位，爬取字段包括了投诉单号、品牌、车型、投诉时间、诉求问题、投诉明细、经销商、解决状态。

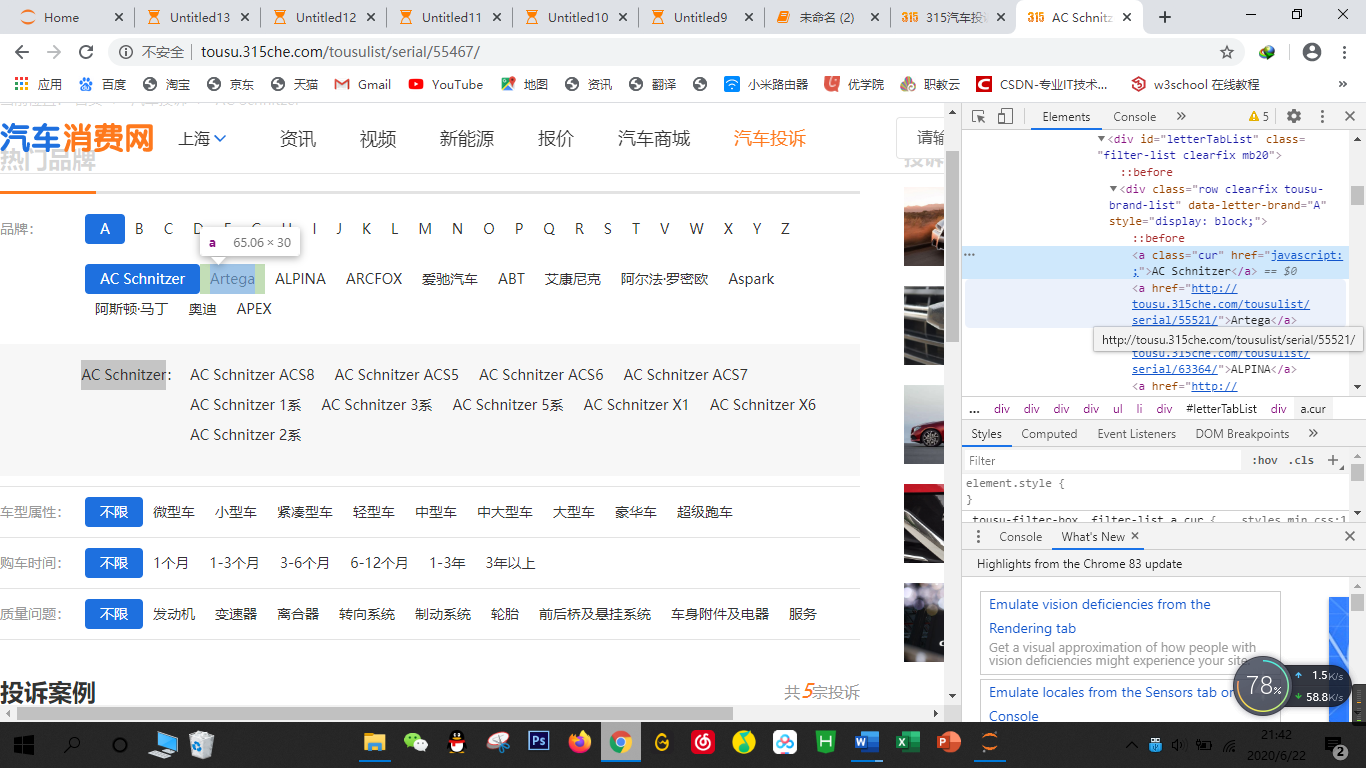
爬取步骤可以分为以下3个步骤。

1. 获取汽车品牌名称数据。
2. 获取汽车品牌的网页链接。
3. 获取汽车品牌投诉的页数。
4. 获取汽车品牌投诉案例的网页链接。
5. 对案例的网页链接进行抓取字段信息。
6. 将汽车品牌案例的链接进行保存
7. 批量爬取链接内数据保存

## 目录页信息爬取

本次爬取采用request库和lxml库，解析方式用xpath方法。在目录页中选择第一个品牌（AC Schnitzer），在品牌的位置上右击选择检查，根据XML路径语言的特点在弹出的窗口中找到汽车品牌的路径，按照类型的方法爬取汽车品牌的链接、爬取汽车投诉的页数。结合循环语句爬取多页内容。





## 二级网址信息爬取

循环打开目录页爬取下来的汽车品牌链接及页数，将有规律的翻页链接进一步判断作为爬取二级网址的链接，按照相同的方式爬取品牌每个投诉链接的投诉单号、品牌、车型、投诉时间、诉求问题、投诉明细、经销商、解决状态。然后把所有的数据保存到csv文件当中。



## 代码实现

利用request库和lxml库爬取汽车消费网的信息，如代码 2‑1所示。

代码 2‑1 爬取招聘岗位的信息

|  |  |  |
| --- | --- | --- |
|  |  |  |
| In[1]: | from lxml import etree #y用来加载网页解析  import requests #用来网页请求  import time #计时  import pandas as pd  from tkinter import \_flatten  import random  import math  import re  url = 'http://tousu.315che.com/tousulist/serial/55467/'#网址  rq = requests.get(url)  rq.encoding = 'utf-8'  dom = etree.HTML(rq.text)  #获取到每个页面下的品牌  name = dom.xpath('//\*[@id="letterTabList"]/div/a/text()')  #获取到每个页面下的网址  url\_all = dom.xpath('//\*[@id="letterTabList"]/div/a/@href')  del name[url\_all.index('javascript:;')]  url\_all = [i for i in url\_all if i != 'javascript:;']  。。。以下省略，见代码附件 |  |
| Out[1]： |  |  |

# 预处理案例信息数据

爬取下来的原始数据格式多样，我们需要处理每个字段中的数据，统一格式。

## 品牌名称数据处理

爬取下来的品牌名称多种多样，我们这里主要提取从字母A——Z来提取品牌，然后将所有品牌名称提取出来（AC Schnitzer、[Artega](http://tousu.315che.com/tousulist/serial/55521/)、[ALPINA](http://tousu.315che.com/tousulist/serial/63364/)、[ARCFOX](http://tousu.315che.com/tousulist/serial/69097/)、[爱驰汽车](http://tousu.315che.com/tousulist/serial/69166/)、[ABT](http://tousu.315che.com/tousulist/serial/75275/)、[艾康尼克](http://tousu.315che.com/tousulist/serial/76349/)[阿尔法·罗密欧](http://tousu.315che.com/tousulist/serial/80/)、[Aspark](http://tousu.315che.com/tousulist/serial/81587/)、[阿斯顿·马丁](http://tousu.315che.com/tousulist/serial/82/)[奥迪](http://tousu.315che.com/tousulist/serial/83/)、[APEX](http://tousu.315che.com/tousulist/serial/83538/)等等）

## 投诉单号数据处理

将“：”前面的投诉单号字段进行切片，删除“：”前面的字段包括“：”，只剩下单号。例如：投诉单号：91546，最后变为：91546。

## 品牌数据处理

品牌爬取出来无误，无需清洗。

## 车型数据处理

将“：”前面的品牌车型字段进行切片，删除“：”前面的字段包括“：”，只剩下车型。例如：品牌车型：朗逸，最后变为：朗逸。

## 投诉时间数据处理

将“：”前面的投诉时间字段进行切片，删除“：”前面的字段包括“：”，只剩下投诉时间。例如：投诉时间：2018-6-21 13:34:26，最后变为：2018-6-21 13:34:26。

## 诉求问题数据处理

将“：”前面的诉求问题字段进行切片，删除“：”前面的字段包括“：”，只剩下诉求问题。例如：诉求问题：汽车保质期内无故自燃，最后变为：汽车保质期内无故自燃。

## 投诉明细数据处理

投诉明细爬取出来无误，无需清洗。

## 经销商数据处理

将“：”前面的经销商字段进行切片，删除“：”前面的字段包括“：”，只剩下经销商。例如：经销商：无，最后变为：无。

## 解决状态数据处理

解决状态爬取出来无误，无需清洗。

## 代码实现

如代码 3‑1所示。

代码 3‑1 岗位信息数据预处理

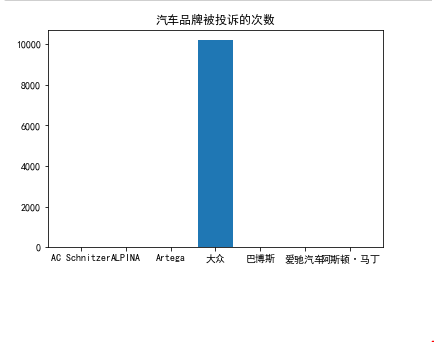
|  |  |  |
| --- | --- | --- |
|  |  |  |
| In[1]: | all\_data = pd.DataFrame()  b = 1  for yy in report\_all[:100]: #分段爬取  print('正在爬取第%d条数据...'%b)  heads = {'User-Agent':"Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3440.106 Safari/537.36"}  rqq\_info = requests.get(yy,headers = heads)  html\_info = etree.HTML(rqq\_info.content, etree.HTMLParser(encoding='utf-8'))  try:  #品牌  brand = [i for i in html\_info.xpath('//\*[@class="container breadnav"]/a[3]/text()')]  #车型  Model = [i.split('：')[1] for i in html\_info.xpath('/html/body/div[1]/div[2]/div[2]/div[1]/div[1]/div/div[3]/p[1]/text()')]  #单号  Complaint\_no = [i.split('：')[1] for i in html\_info.xpath('/html/body/div[1]/div[2]/div[2]/div[1]/div[1]/div/div[3]/p[2]/text()')]  #诉求问题  Appeal\_question = [i.split('：')[1] for i in html\_info.xpath('/html/body/div[1]/div[2]/div[2]/div[1]/div[1]/div/div[3]/p[3]/text()')]  #投诉时间  Complaint\_time = [i.split('：')[1] for i in html\_info.xpath('/html/body/div[1]/div[2]/div[2]/div[1]/div[1]/div/div[3]/p[4]/text()')]  #经销商  distributor = [i.split('：')[1] for i in html\_info.xpath('/html/body/div[1]/div[2]/div[2]/div[1]/div[1]/div/div[3]/p[5]/text()')]  #投诉明细  Complaint\_details =[i.text for i in html\_info.xpath('//\*[@class="describe"]/p')]  #解决状态  Resolution\_status = [i.text for i in html\_info.xpath('//\*[@class="article-tag unsolved"]')]    data = pd.DataFrame({  '品牌': brand,  '车型': Model,  '单号':Complaint\_no,  '诉求问题': Appeal\_question,  '投诉时间': Complaint\_time,  '经销商': distributor,  '投诉明细':Complaint\_details,  '解决状态': Resolution\_status  })  all\_data = pd.concat([all\_data, data])  b = b+1  data.to\_csv(r'C:\Users\12446\Desktop\data.csv',encoding='GBK',mode = 'a+',header = None)    except:  print('此页数据未爬取成功或者网页不存在，跳过')  pass  。。。以下省略，见代码附件 |  |



# 品牌数据分析与可视化

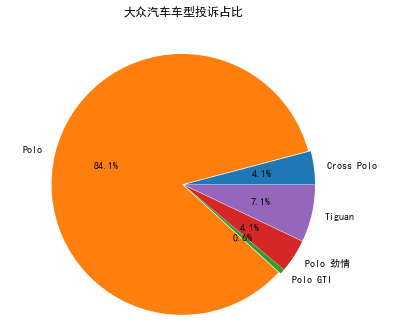
为了分析品牌对于数据分析来对品牌投诉的数量情况，本文从品牌信息中筛选出与数据分析品牌出现的次数的数据作了以下的描述分析。

## 热门品牌投诉



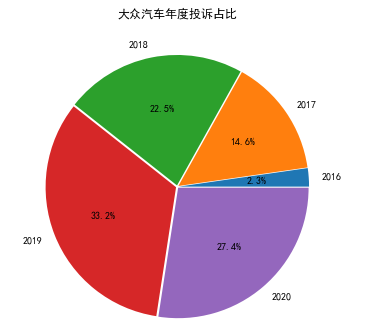
分析7个出现次数最多的品牌，其中次数出现最多的岗位是“**大众”汽车品牌**。可知大众汽车品牌价格对消费者来说是比较低的。但是“大众”汽车品牌出现的问题也是最多的。

## 大众汽车车型前五被投诉占比



取出前5个出现次数最多的“大众”品牌车型投诉进行分析，其中前三个热门投诉车型分别是“Polo”，“Tiguan”和“Cross Polo”。说明“大众”Polo系列的车型出现的问题是最多的。

## 大众汽车年度投诉占比



取出“大众”品牌被投诉最多的年份进行分析，投诉前三的年份分别是2019、2020、2018年。可以说明近几年“大众”品牌销量是比较高的。

## 代码实现

如代码 4‑1所示。

代码 4‑1 汽车消费网数据分析与可视化

|  |  |  |
| --- | --- | --- |
|  |  |  |
| In[1]: | import matplotlib.pyplot as plt  #中文显示方法  import matplotlib as mpl  mpl.rcParams['font.sans-serif'] = ['SimHei']  mpl.rcParams['font.serif'] = ['SimHei']  mpl.rcParams['axes.unicode\_minus'] = False  plt.bar(PP, number)  plt.title('汽车品牌被投诉的次数')  plt.show()  mpl.rcParams['font.sans-serif'] = ['SimHei']  mpl.rcParams['font.serif'] = ['SimHei']  mpl.rcParams['axes.unicode\_minus'] = False  p =plt.figure(figsize=(6,6))  explode = [0.01,0.01,0.01,0.01,0.01]  plt.pie(num,explode = explode,labels = PP\_A,autopct = '%1.1f%%')  plt.title('大众汽车车型投诉占比')  # plt.savefig(r'Users\Lenovo\Desktop\tousu\_CX.png')  plt.show()  mpl.rcParams['font.sans-serif'] = ['SimHei']  mpl.rcParams['font.serif'] = ['SimHei']  mpl.rcParams['axes.unicode\_minus'] = False  p =plt.figure(figsize=(6,6))  explode = [0.01,0.01,0.01,0.01,0.01]  plt.pie(number,explode = explode,labels = year,autopct = '%1.1f%%')  plt.title('大众汽车年度投诉占比')  # plt.savefig(r'Users\Lenovo\Desktop\tousu\_year.png')  plt.show()  。。。以下省略 |  |

# 5 小结

本文运用网络汽车消费信息数据，对汽车品牌进行了划分，并分析了消费者对汽车品牌的评价和投诉，分析了汽车行业质量和性能现状及其发展趋势。

结合数据分析，我们知道在汽车行业使用“大众”的消费者数量是最大的，能够得出“大众”品牌在汽车行业是使用最广泛的，使用的人越多，收到的投诉对于汽车的质量和性能会有更多的评价，“大众”公司可以去查看相关的投诉信息，从而去提高自身品牌的质量和性能。所以，就汽车行业来说，需要通过消费者购买产品来评价自身产品，从而来提高汽车的质量和性能，让消费者能有自身心理的性价比汽车。