一、Map Area

福建省部分区域，中华人民共和国

数据来源 https://mapzen.com/data/metro-extracts/ 泉州市是我的家乡，所以我对搜集到的相关数据很感兴趣，通过下载、清洗、存入数据库，并进行相应的查询令我倍感熟悉。但最后数据太小了，因而扩充了到福建省的包含泉州的部分区域。

二、初期审视地图所遭遇的问题

在最初下载泉州地区的一个小样本并运行临时 quanzhou.py 文件后，我发现了数据的部分问题，我将按照以下顺序讨论：

1.因为数据当中出现了很多中文，本来想采用 python3 作为分析工具，但是在写入文件时出现了较多错误，故仍采用 python2。

2.因为数据中既有英文又有英文，因为想只针对中文进行数据清洗。故利用自己修改的代码，输出了样本数据中的相应的中文寻找错误。

```
chinese_end = re.compile(u"[\u4e00-\u9fa5]$")
def check(osmfile):
    osm_file = open(osmfile, "r")
    street_types = defaultdict(set)
    for event, elem in ET.iterparse(osm_file, events=("start",)):
        if elem.tag == "node" or elem.tag == "way":
            for tag in elem.iter("tag"):
                if chinese_end.search(tag.attrib['v']) != None:
                    print tag.attrib['v']
```

通过这段代码，我打印出了"tag“标签中，"v”属性所有以中文为结尾的字符，发现了如下问题：

（1）部分中文中混杂了小写字母，显然是人为输入错误，毕竟有些店名也包含着英文，判断是不是脏的数据比较困难，难以批量删除，只能靠人工判断。

<tag k="name" v="Bache hkepel 福建晋江市池店镇营边村南区 78 号"/>

（2）很多地方的都把“号”用“#”替换，但是为了规范我决定都把“#”换成“号”，保证数据的一致性。

<tag k="name" v="云谷花苑 1#楼"/>

（3）发现了数据中存在一些 fixme 的点，这应该是等待修复的点，应该是有问题的点，等待修复的点，本来我是想直接把这种点删除的，但后来发现它蕴含的内容还挺有意义的，故不做处理，予以导入数据库。

<tag k="fixme" v="continue / 繼續"/>

<tag k="fixme" v="position"/>

（4） 发现宁德市这个地方的地址存在不必要的地名、省份和国家，这造成了大量的数据冗余，我采取的策略是只保留 value 里面第一个空格以前的信息。

<tag k="address" v="寧德市中心部 寧德市, 福建省, 中國"/>

<tag k="address" v="蕉城南路 97 號 352100 寧德市, 福建省, 中國"/>

（5）不知道为什么宁德市中存在大量的繁体字，故我想把当中的繁体字转为简体字，经过

论坛的询问，采用 Hanzi Converter 库。
https://discussions.youdaxue.com/t/p3/49968/2?u=mr_xiao

在这部分是我最头痛的地方，想了好久都找不到有问题的数据。最后给自己一个目标先把有中文的字段打出了，然后进行观察，才发现上面的问题。因而即使是找问题，目标与方法的正确使用也是十分重要的。

三、问题数据处理
1.解决思路
因为初期发现的问题都与 value 有关，故我构造了 update_v 函数来解决这个问题。
先把所有的繁体字转为简体字；
判断是否带#并且排除是 url 的情况，是的话将#换成‘号’；
判断 k 是否是 address 以及是否有'宁德'，是的话取 v 值第一个空格前的值,判断值的末尾是否有‘,’，若有，输出去除‘,’后的值；
若条件都不满足直接输出 v。

2.相关代码
```
from hanziconv import HanziConv
def update_v(k, v):
    #繁体换成简体
    v= HanziConv.toSimplified(v)
    #判断#号
    if '#' in v and len(v) <= 20:
        return v.replace("#", u"号")
    #判断宁德市的地址
    if k == "address" and u'宁德'in v:
        v = v.split(' ')[0]
        if "," in v:
            print v.replace(",", "")
    return v
```

3.调试效果
截取部分替换后的结果：

站前路京都商贸区 宁德市，福建省，中国 => 站前路京都商贸区
蕉城北路 11 號 352100 寧德市，福建省，中國 => 蕉城北路 11 號
1# => 1 号
8#楼 => 8 号楼
食堂 4#宿舍楼 => 食堂 4 号宿舍楼

总体来说，发现的问题比较简单，所以处理起来也比较容易。而处理的结果也符合预期。当然肯定还存在更好的解决方法，希望在不断地探索中能找到。不过也有很多无法处理的问题，比方说部分数据人工数据错误，只能以肉眼判断等。在处理的过程中也遇到看中文编码忘加 u 等问题，但通过百度都能解决。唯一比较困难的是繁体字转简体字，还好论坛贴心的小伙

伴帮我解决了问题。明显可以感觉到找出脏数据远比处理数据更难。一颗好奇的心或许是数据分析师必不可少的。

四、数据导入数据库，探索数据部分规律

准备导入的时候发现了自己导出的 csv 文件行之间都有个空行，后来经过不断摸索才发现写入“w”改成“wb”能解决这个问题。

1.数据概述

（1）文件大小

quanzhou2.osm ——163 MB
fjquanzhou.db ——87.7 MB
nodes.csv ——71.9 MB
nodes_tags.csv ——846 KB
ways.csv ——2.16 MB
ways_nodes.csv ——22.0 MB
ways_tags.csv ——2.75 MB

（2）节点数目

sqlite>select count(*) from nodes; 888182

（3）途径数量

sqlite>select count(*) from ways; 37208

（4）唯一的用户数量

```
sqlite>select count(distinct(e.uid))
from(select uid from nodes union all select uid from ways) e;
261
```

（5）前十的贡献者用户

"来自青山下" =>"258835"
"starrysky" =>"205841"
"katpatuka" =>"126548"
"Seandebasti"=> "74906"
"jamesks" =>"37589"
"mixmaxtw" =>"24120"
"u_kubota" =>"20827"
"Supaplex" =>"15856"
"gerg1" =>"10893"
"FreedSky" =>"10412

（6）只出现过一次的用户

56

（7）前十的 amenities
"place_of_worship" "206"
"shelter" "108"
"toilets" "63"
"school" "62"
"bank" "52"
"restaurant" "51"
"fuel" "46"
"parking" "44"
"townhall" "44"
"hospital" "25"

2.进一步数据探索
(1)待处理的点
sqlite>select e.id,e.key,e.value,count(*)
from (select id,key,value from nodes_tags UNION ALL select id,key,value from ways_tags)e
where e.key='fixme'
group by e.id;（group by e.value order by count(*) desc;）

通过官方的文档知道了，fixme 键允许贡献者标记需要进一步关注的对象和地点。可以发现共有 79 个待处理的点，相对于样本总数来说是相当少的。
其中排名前两位的值是 continue 和 yes。
continue 共有 21，代表着用于构成最终已知点的节点，也就是说着 21 个点是已经解决的，等待发现者再填上就行了。 yes 有 16 个，代表标记者还不知道应该填什么内容的节点，也就是真正需要 fix 的点。通过在当地发现这样的点可以给予地图改进。

（2）最大的宗教
最大的宗教：佛教 135；道教 50；基督教 7；日本神道教 1。还是比较符合预期的，但是没想到还混进了一个神道教。

（3）废弃村庄
无意间搜索 key = 'description',发现里面 value 大部分是废弃村庄，有 26 个，经过经纬度定位工具，发现这 26 个废弃村庄大部分都在福建省福州市马 尾区，但是我通过百度也没有查清为什么这些是废弃村庄，只发现了大部分村庄的位置都比较偏僻，但是未来大家如果去福建旅游还是需要注意一下这些地方：五奇仙君殿、倒流溪、公山、半山、半岭、半溪垱坪岗、大王厝、岗仑、彭田、明浦里、楼梯企、洋车、深坑里、溪尾、琅岐楼、真珠礑、石狮、秋峰、蕃婆楼、虎仑、西洋、长垄、香炉、马额顶村、鼓鸡师、鼓鸡狮

（4）公共设施查询
select e.id,e.key,e.value,count(*)
from (select id,key,value from nodes_tags UNION ALL select id,key,value from ways_tags)e
where e.key = 'name' and e.value LIKE '%医院'（可以替换成其他）
group by value;
可以查到医院有 21 所，小学 44 所，中学 40 所，大学 7 所，公园 52 座，以街和路为结尾检

索共 509 条。

（5）长度前 10 的 alt_name

```
select e.id,e.key,e.value,length(e.value)
from (select id,key,value from nodes_tags UNION ALL select id,key,value from ways_tags)e
where    e.key = 'alt_name'
group by value
order by length(e.value) desc
limit 10;
```

```
"4253773072" "alt_name"    " 古 镛 镇 ; 将 乐 县 ; 将 乐 ;Jiangle; 古 镛 ;Chiang-le;Tsianglohsien;Tsianglo;Chiang-lo-hsien;Chiang-lo"    "80"
"2115024094" "alt_name"    " 尤 溪 城 关 镇 ; 城 关 镇 ;Youxi;Yukihsien;Yuki;Yu-hsi;Yu-ch'i;Yu-ch'i-hsien-ch'eng;Yu-ch'i-hsien"    "80"
"9216066"    "alt_name"    "Xiaolian Dao;Hsiao-lien Tao;Dori Islang;Dari Islang;Erh-lien Tao"    "64"
"3935507464" "alt_name"    "Haitan Xia;Hai-t'an Hsia;Hai-t'an Hai-hsia;Haitan Haixia"    "56"
"390385654"  "alt_name"    "Lien Shan;Tatong Tao;Ta-tung Tao;Dalian Dao;Ta-lien Tao"  "55"
"244080761"  "alt_name"    "将乐;Jiangle Xian;Chiang-le Hsien;Chiang-lo Hsien"    "47"
"2115024088" "alt_name"    "梅仙镇;Meixian Zhen;Mei-hsien;Mei-hsien-fan" "40"
"244081841"  "alt_name"    "顺昌;Shunchang Xian;Shun-ch'ang Hsien"    "35"
"4577025651" "alt_name"    "仁寿镇;Jenshow;Jen-shou;Renshou Zhen"  "33"
"4540826251" "alt_name"    "联合乡;Lianhe Xiang;Lien-ho;T'a-tou"  "32"
```

alt_name 相当于这个地点的别名，可以发现最长的别名长达 80 个字符，有 9 个别名，还是比较夸张的。

五、建议与结语

建议：

建议 1：应该先把宁德市这个地方的繁体字换为简体字，同时删除 k="address"里面的省市、国家。

（1）好处：

① 可以维持数据的一致性，使得中文都是简体字。

② 因为省市、国家信息存在大量的重复，这样做可以减少数据的冗余。

（2）预期的问题：首先要统一这一整个地区的数据记录方式，并通知当地的记录人员，只有能联系上当地的记录人员，修改还是比较容易的。

（3）预期收益是大于修改的风险，因而我建议把宁德市这个地方的数据问题修改了。

建议 2.可以定期把数据中 k="fixme"，v="yes"发布到网上，可以如采用奖励徽章或排行榜的方法，让网友知道有哪些点是需要修改的并且让其附近的网友修改。

（1）好处：可以让网友清楚哪些点是需要修改的，而巧好在那个位置附近的人可以用较低的成本提供数据信息。

（2）预期的问题：

①如何激励当地的人是一个较大的问题，因为数据获取是免费的，而给予物质上的激

励需要付出较大的成本。

②我提出的激励方法虽然是非物质的，可能难以调动网友的积极性。 同时如何正确的推送到有意向的网友也是一个较大的问题。

（3）预期的问题是比较大的，如果能有更好地激励方法这是可以实现的。但是目前来看，估计只能把这些信息挂起来。

建议 3：统一几号楼的写法，目前数据里有两种写法，一种是“#”，一种是“号”，建议都改成号，可以增加审核系统。

（1）好处：

①可以维持数据的一致性。

②增加数据的可读性。

（2）预期的问题：

①还是统一标准的难度问题，并且把还要及时把统一的格式发给信息提供者。

②审核系统可能会打击信息提供者的积极性，审核系统的建立耗时耗力。

（3）目前问题比较大的是审核系统的建立，但是统一标准还是可以并且应该做的事情，这可以避免利用数据时出现的歧义与使用错误。

结语：

1.在对这些数据进行审查之后，福建部分地区的数据显然是不完整的（也有可能是我的查询方法有问题），从基础设施的数量就可以看出来，只有一些重要的地点，未来还需要进一步完善。还有数据的一个问题就是我搞不明白为什么单单宁德市的要弄成繁体字，而且宁德市的每一个地名都跟着省市、国家，完全是不必要的，未来可以优化宁德这个地方的数据。

2.我认为这个练习的目的已经很好地锻炼了我清理数据的能力。同时也激发了我对数据的兴趣。从刚开始什么问题都找不到，什么灵感都没有，到最后有自己的想法，发现数据的问题并解决，同时写出自己的想法与建议，这一过程还是比较有成就感的。还有就是要感谢论坛在我的数据清洗过程中给予了我很大的帮助，特别是通过论坛我优化了我的代码，使得我的代码显得不那么臃肿，还有就是解决了繁体转简体的问题。

3.当然，这个数据处理过程还有很多不足，例如数据清洗大部分仅针对中文，处理过程比较简单等等，希望未来能进一步改正。