星战图谱 可视化

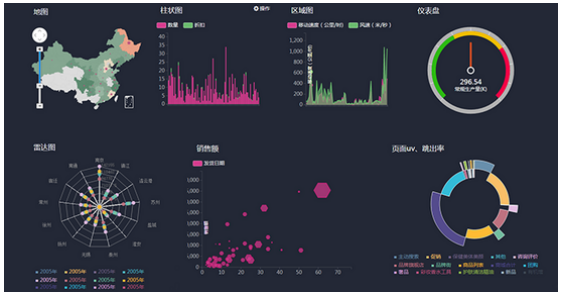
16.10.32

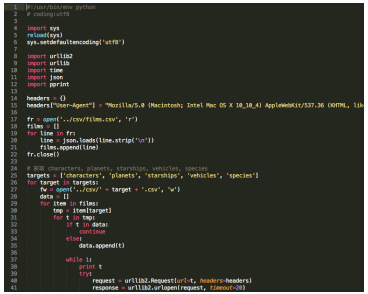
大数据

媒体
报道的

公司
宣传的

公众
理解的





我实际
干的

大数据的概念被炒得太火、太热、太烂

很多 数据
谈不上是 大数据

今天不谈大数据 只分享如何 小而美地 玩转数据

你将收获

数据采集

使用简单的爬虫获取
互联网上的公开数据

数据分析

使用代码进行简单的
统计分析和聚合运算

数据存储

将爬取的数据存储到
静态文件或数据库中

数据展示

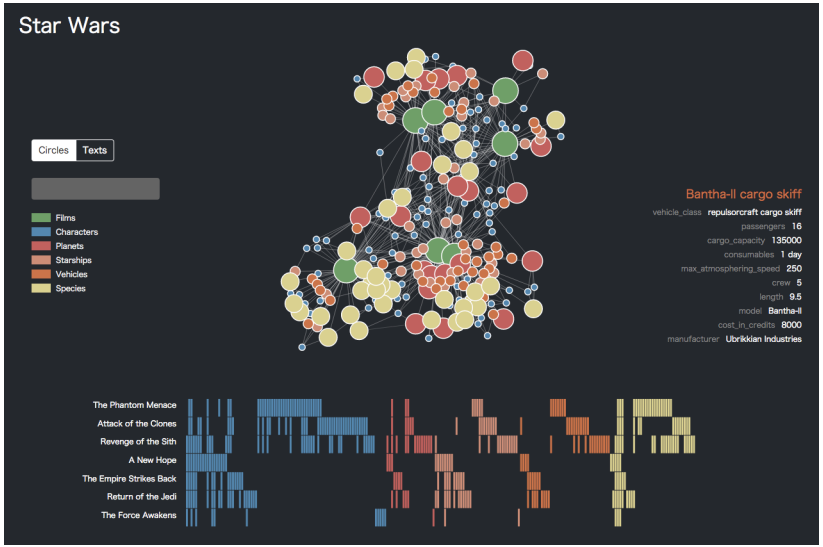
借助图形和交互网页
将分析结果进行展示

最终成果

和星战系列七部电影
有关的一些格式化数据

绘制一些用于统计和分析
的静态可视化图形

制作一份用于交互和展示
的动态可视化网页



没有看过星战系列电影的同学，一张图就可以告诉你，这7部电影中
共出现了87名角色、21颗星球、37艘飞船、39架战车、37个种族，以及这228个实体之间的1112次联系

数据采集

没有 数据 何来 玩转

数据采集

数据来自 SWAPI

全球首个量化的、可供编程使用的星战数据集
开发者经过漫长的搜集和整理
汇总了星战系列电影中涉及的多个种类实体数据

- ✓ Films: http://swapi.co/api/films/<Id>/
- ✓ People: http://swapi.co/api/people/<Id>/
- ✓ Starships: http://swapi.co/api/starships/<Id>/
- ✓ Vehicles: http://swapi.co/api/vehicles/<Id>/
- ✓ Species: http://swapi.co/api/species/<Id>/
- ✓ Planets: http://swapi.co/api/planets/<Id>/

SWAPI
The Star Wars API

六个 API
对应六类 实体

数据采集

在浏览器中访问

http://swapi.co/api/people/1/

返回的是一个 JSON
即 键值对 形式的字符串

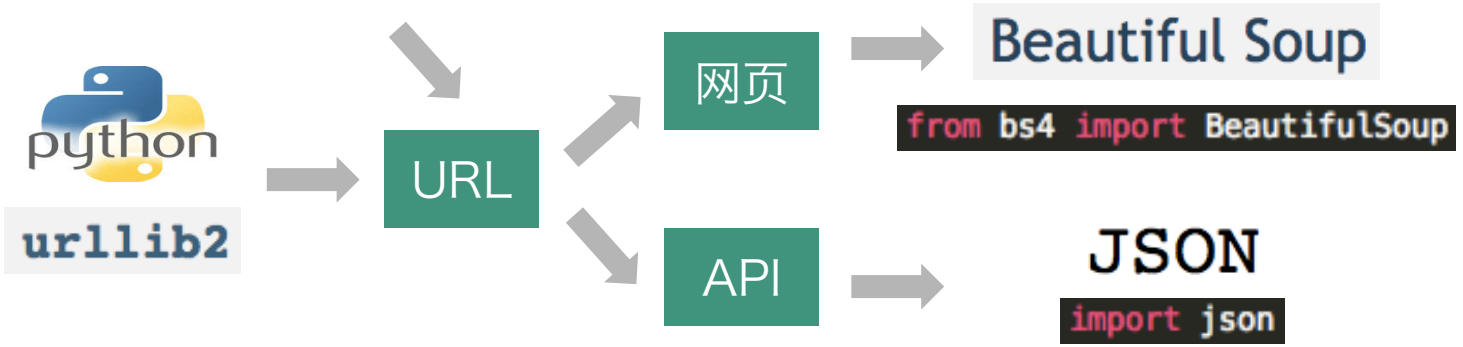
什么是JSON


```
{
    "name": "Luke Skywalker",
    "height": "172",
    "mass": "77",
    "hair_color": "blond",
    "skin_color": "fair",
    "eye_color": "blue",
    "birth_year": "19BBY",
    "gender": "male",
    "homeworld": "http://swapi.co/api/planets/1/",
    "films": [
        "http://swapi.co/api/films/6/",
        "http://swapi.co/api/films/3/",
        "http://swapi.co/api/films/2/",
        "http://swapi.co/api/films/1/",
        "http://swapi.co/api/films/7/"
    ],
    "species": [
        "http://swapi.co/api/species/1/"
    ],
    "vehicles": [
        "http://swapi.co/api/vehicles/14/",
        "http://swapi.co/api/vehicles/30/"
    ],
    "starships": [
        "http://swapi.co/api/starships/12/",
        "http://swapi.co/api/starships/22/"
    ],
    "created": "2014-12-09T13:50:51.644000Z",
    "edited": "2014-12-20T21:17:56.891000Z",
    "url": "http://swapi.co/api/people/1/"
}
```

数据采集

使用代码 而不是浏览器

http://swapi.co/api/people/1/



廖雪峰的python教程

数据采集

代码来一发

我的环境

- ✓ Mac OS
- ✓ Python2.7
- ✓ Sublime Text 2

下载传送门

用一个list存放
七部电影的API链接

```
films = []
for x in xrange(1, 8):
    films.append('http://swapi.co/api/films/' + str(x) + '/')
```

打开写文件

对每个链接
1 使用urllib2请求
2 建立连接
3 读取结果

关闭写文件

```
fw = open('films.csv', 'w')

for item in films:
    print item
    request = urllib2.Request(url=item, headers=headers)
    response = urllib2.urlopen(request, timeout=20)
    result = response.read()
    fw.write(result + '\n')

fw.close()
```

数据采集

采集结果

- ✓ get_films.py
- ✓ get_details.py

- ✓ films.csv
- ✓ characters.csv
- ✓ planets.csv
- ✓ species.csv
- ✓ starships.csv
- ✓ vehicles.csv


```
{"title":"The Force Awakens","episode_id":7,"opening_crawl":"Luke Skywalker has vanished.\r\nIn his
absence, the sinister\r\nFIRST ORDER has risen from\r\nthe ashes of the Empire\r\nand will not rest
until\r\nSkywalker, the last Jedi,\r\nhas been destroyed.\r\n \r\nWith the support of the\r\nREPUBLIC,
General Leia Organa\r\nleads a brave RESISTANCE.\r\nShe is desperate to find her\r\nbrother Luke and
gain his\r\nhelp in restoring peace and\r\njustice to the galaxy.\r\n \r\nLeia has sent her most
daring\r\npilot on a secret mission\r\nto Jakku, where an old ally\r\nhas discovered a clue
to\r\nLuke's whereabouts....","director":"J. J. Abrams","producer":"Kathleen Kennedy, J. J. Abrams,
Bryan Burk","release_date":"2015-12-11","characters":["http://swapi.co/api/people/1/","http://swapi.
co/api/people/3/","http://swapi.co/api/people/5/","http://swapi.co/api/people/13/","http://swapi.
co/api/people/14/","http://swapi.co/api/people/27/","http://swapi.co/api/people/84/","http://swapi.
co/api/people/85/","http://swapi.co/api/people/86/","http://swapi.co/api/people/87/","http://swapi.
co/api/people/88/"],"planets":["http://swapi.co/api/planets/61/"],"starships":["http://swapi.
co/api/starships/77/","http://swapi.co/api/starships/10/"],"vehicles":[],"species":["http://swapi.
co/api/species/1/","http://swapi.co/api/species/2/","http://swapi.co/api/species/3/"],"created":"2015-04
-17T06:51:30.504780Z","edited":"2015-12-17T14:31:47.617768Z","url":"http://swapi.co/api/films/7/"}
```


每部电影对应一行 JSON

数据存储

一次 存储 随心 使用

数据存储

存储至静态文件或数据库

- 静态文件
  - ✓ TXT
  - ✓ CSV
  - ✓ JSON
- 数据库
  - ✓ MySQL等关系型数据库
  - ✓ 其他常用NoSQL

打开写文件

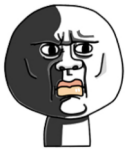
写入一行

关闭写文件

```
fw = open('films.csv', 'w')

for item in films:
    print item
    request = urllib2.Request(url=item, headers=headers)
    response = urllib2.urlopen(request, timeout=20)
    result = response.read()
    fw.write(result + '\n')

fw.close()
```



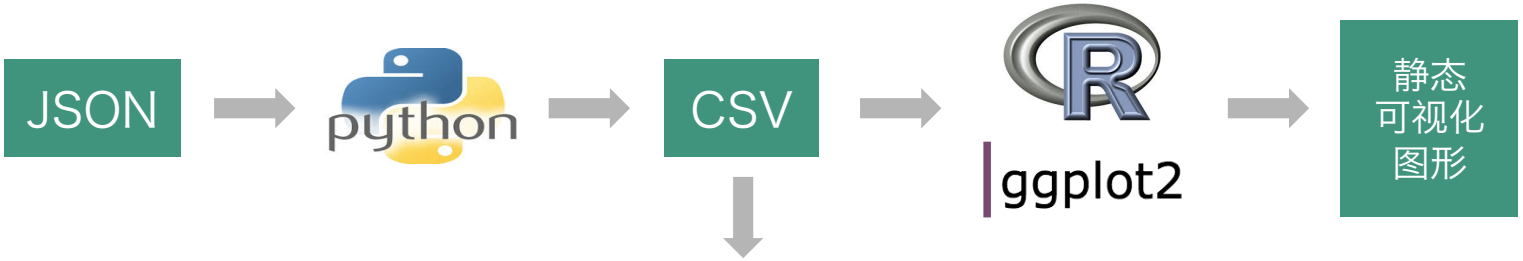
我只是习惯性地把后缀写成 CSV
它们存的明明是 JSON 啊喂

数据分析

有时 分析 多是 统计

数据分析

是时候来些分(tong)析(ji)了



每行表示 一条纪录 每列表示该纪录的 一个字段
可以理解为关系型数据库中的 表格 以及R和pandas中的 dataframe

对R和ggplot2感兴趣?
看看我的个人博客

http://zhanghonglun.cn/blog/tag/r/

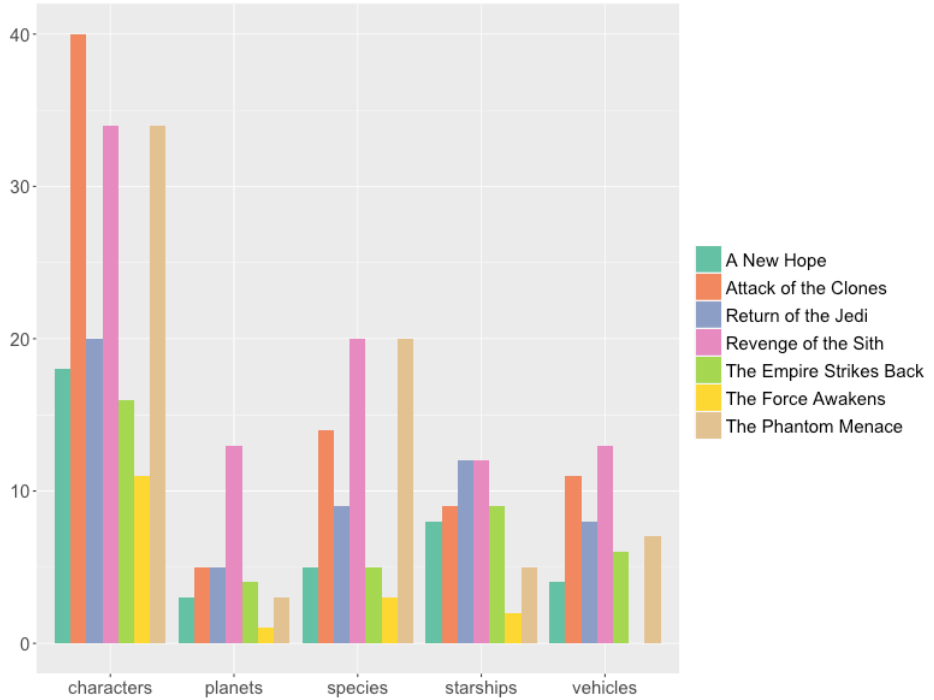
数据分析

电影实体统计

七部电影分别涉及
其他类别实体各多少

- ✓ Attack of the Clones涉及人物最多
- ✓ The Force Awakens涉及实体最少，可能是因为数据 尚未整理完全

整理代码：stat_basic.py
整理数据：stat_basic.csv
绘图代码：stat.R



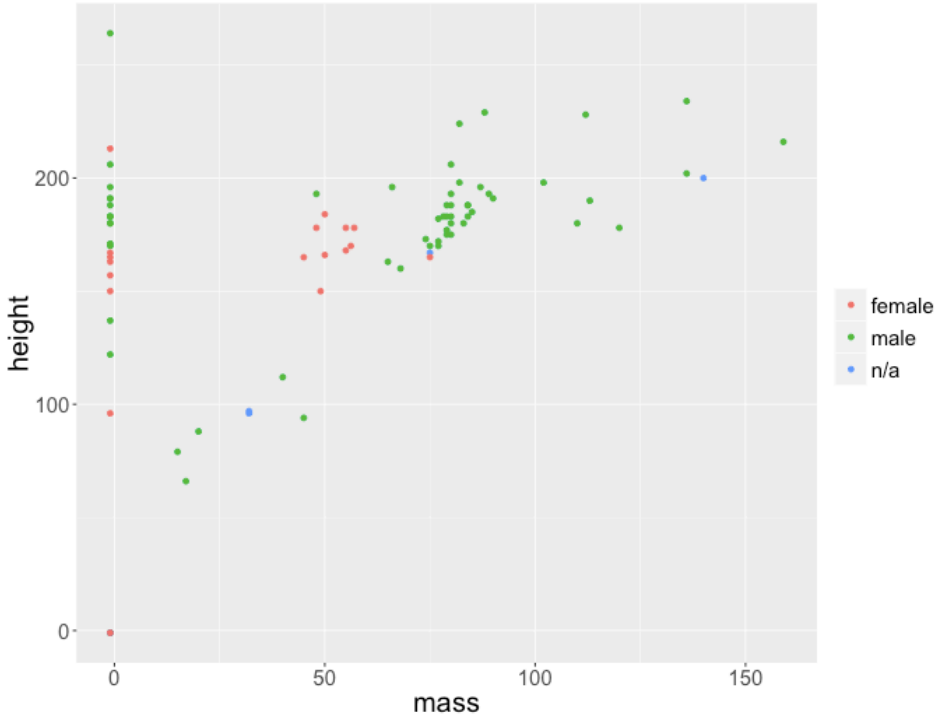
数据分析

人物统计

电影人物的性别、身高、体重分布情况

- ✓ 男性人物居多，身高和体重整体呈正相关
- ✓ 体重为 0 表示 数据缺失

整理代码：stat_character.py
整理数据：stat_character.csv
绘图代码：stat.R



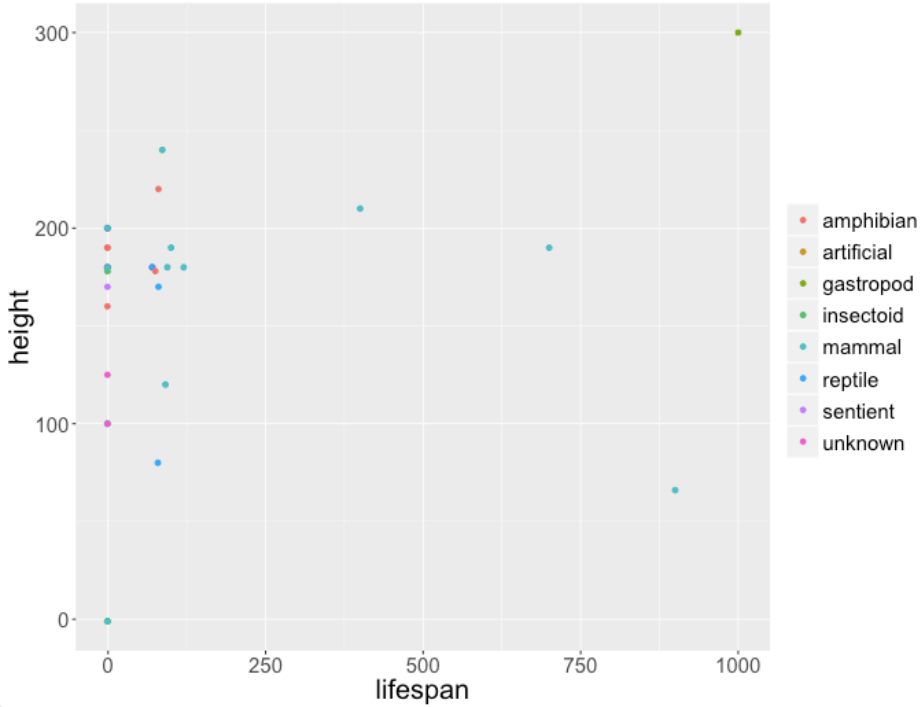
数据分析

种族统计

各个种族的种类、身高、寿命分布情况

- ✓ 大多数种族的寿命在100年以下
  少数种族寿命可达几百年
- ✓ 寿命为 0 表示 数据缺失

整理代码：stat_species.py
整理数据：stat_species.csv
绘图代码：stat.R



数据展示

更多 炫技 更高 颜值

数据展示

为什么用交互网页

可以根据用户的 交互 和 操作，选择性地动态展示 更丰富 的内容

- ✓ json_all.py
- ✓ json_force.py
- ✓ json_timeline.py

- ✓ all.json：全部实体的详细数据
- ✓ starwar.json：关系图数据
- ✓ timeline.json：时间线数据

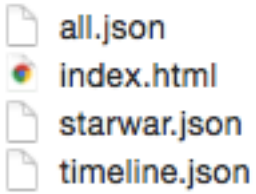
所需的数据已经整理为静态文件
所以不需要用到后端和数据库

你只需要会 

数据展示

网页一共多少代码

html文件夹 中包含了交互网页
所使用到的全部文件
一个 html，三个 json

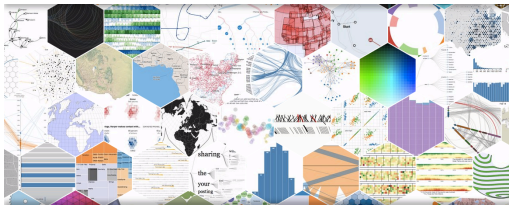


index.html
包括空格
共507行代码

这些都是基础

HTML CSS

JavaScript JQuery



这个才是重头

Data-Driven Documents

最流行的js可视化库之一

数据展示

D3的使用流程

- ✓ 在html中准备一个 svg 和 一个 g 作为绘图容器
- ✓ 在js中 select() 容器
- ✓ 在容器中 selectAll() 将要绘制的svg元素
- ✓ 使用 data() 为svg元素绑定数据
- ✓ 根据数据和元素对应状态，执行 enter().append()和 exit().remove()
- ✓ 使用 attr() 控制svg元素的外观
- ✓ 根据 用户交互，更新svg元素的外观

```
<svg width="960" height="240">
    <g></g>
</svg>
```

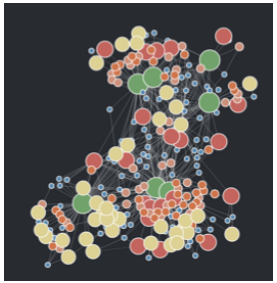
```
d3.select('#svg2 g').selectAll('text.film').data(data['films']).enter().
    append('text').text(function(d) {
    return d[0];
}).attr('transform', function(d, i) {
    return 'translate(150,' + (40 + (i + 0.5) * height2 / data['films'].
        length) + ')';
}).attr('fill', '#fff').attr('font-size', 12).attr('text-anchor', 'end')
    .attr('class', 'film');
```

数据展示

开始动手

关系图

- ✓ 根据节点数据，添加 circle 和 text 元素
- ✓ 根据节点链接关系，添加 line 元素
- ✓ 根据节点数据，控制节点 颜色 和 大小
- ✓ 鼠标 悬浮 和 拖拽 时，触发相应事件，并更改元素的外观和显示



时间线

类似的思路和流程
不同的是使用 rect 元素

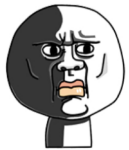
详细流程，请参考代码

数据展示

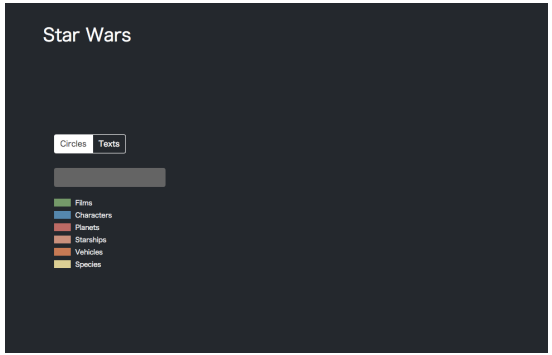
部署你的交互网页

为了达到最佳体验
良心推荐 Chrome
最好的浏览器之一

如果你 直接双击 index.html
或者用浏览器 直接打开 它



说好的可视化呢
说好的最好的浏览器之一呢



数据展示

为什么以及怎么办

跨域请求 在大多数浏览器中是 禁止的 请求不到json数据 可视化自然也就出不来

```
▶ XMLHttpRequest cannot load file:///Users/honlan/Desktop/git/starwar-visualization/star_war/html/all.json. Cross origin requests are only supported for protocol schemes: http, data, chrome, chrome-extension, https, chrome-extension-resource.   d3.v4.min.js:6
```

不管是 本地 还是 外网 你都需要一个 Web环境

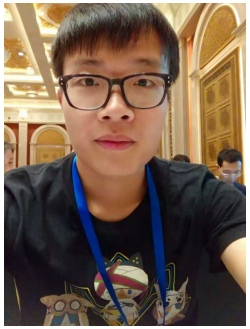
WAMP LAMP MAMP

Ubuntu LAMP环境搭建

总结一下

玩转 数据 并不 困难

个人微信公众号



张宏伦

上海交通大学直博在读

我的微信

项目链接 https://github.com/Honlan/starwar-visualization

可视化链接 http://zhanghonglun.cn/starwars/

我的个人网站 http://zhanghonglun.cn/