实证研究中数据管理规范化
——以整理证监会上市公司行业分类为例

李刚 *

2021 年 11 月 18 日（点击查看最新版本）

摘要：以整理证监会上市公司行业分类数据为案例，分享规范化数据管理的思路和工作流（Workflow）。具体而言，包括 Python、Stata 数据处理，以及 Markdown、LaTeX 文档撰写。

关键词：数据管理；文档撰写

1 引言

如今的学术研究都将就规范，何为规范?
数据管理是指什么？是一个 IPO 的过程（Input，Process and Output）。

2 案例讲解

本次分享会将以整理证监会上市公司行业分类数据为案例，演示从数据获取、数据清理到文档撰写等规范化数据管理的流程。主要涉及的工具有 Python、Stata、Markdown 和 LaTeX，由于软件本身的学习找几本工具书看看就能掌握，“无他，但手熟尔”，所以本次分享会不拘泥于技术细节，而是想通过这个案例展现规范化数据管理的思路和工作流程。

2.1 提出问题

爬取证监会上市公司行业分类 PDF 文档，最终处理成可供 Stata 分析的 *.dta* 数据。

2.2 分析问题

经过网页数据的分析，此问题的处理思路如下：

- 数据获取：使用 Python 爬取 PDF 文件并识别、转换为 Excel ；
- 数据清理：使用 Stata 数据清理，主要涉及**字符串处理**；
- 文档撰写：简单介绍两款文档撰写工具，一是 Markdown 记录说明文档，二是 LaTeX 撰写笔记、Beamer 制作学术幻灯片。

*李刚，华中科技大学经济学院在读博士生，邮箱（gangli_econ@hust.edu.cn），个人主页（http://lgspace.top/）。

2.3 实现过程

2.3.1 Python 数据获取

准备工作

准备工作，导入需要用到的库、定义路径。

```
# 导入库
from os.path import join
import requests
import re
from bs4 import BeautifulSoup
import pdfplumber
from openpyxl import Workbook
```

习惯 Stata 通过 *global* 定义路径的方式，在 Python 也可以通过如下方式定义路径：

```
# 定义路径
os.getcwd()
# prgm = ".../Projects/Training/program"
data = join(prgm,"data")
raw = join(data, "rawData")
output = join(data, "outputData")
```

获取 PDF 链接：

```
# 获取页面标题和链接
  page_links = []
  title = []

  urls = ["http://www.csrc.gov.cn/pub/newsite/scb/ssgshyfljg/index.htm", "http://www.csrc.gov.cn/
                                                    pub/newsite/scb/ssgshyfljg/index_1.htm"]

  for url in urls:
      res = requests.get(url)
      res.encoding = "utf-8"
      soup = BeautifulSoup(res.text, 'html.parser')
      li = soup.find_all('li')
      for i in range(5, len(li)):
          page_links.append(li[i].find('a')['href'])
          title.append(li[i].find('a')['title'])

# 获取每一页面下PDF的链接
pdf_link_temp = []
for link in page_links:
    page_link = link.replace("./","http://www.csrc.gov.cn/pub/newsite/scb/ssgshyfljg/")
    res = requests.get(page_link)
    res.encoding = "utf-8"
    soup = BeautifulSoup(res.text, 'html.parser')
    if soup.find_all('a')[-5]['href'] == "/pub/newsite/fzlm/gywm":
        pdf_link_temp.append(soup.find_all('a')[-6]['href'])
    else:
        pdf_link_temp.append(soup.find_all('a')[-5]['href'])

# 构造PDF链接
pdf_links = []
```

```
for link in pdf_link_temp:
    date = link[4:10]
    if link == "./W020211026510500937710.pdf":
        date = "202107"
    pdf_links.append("http://www.csrc.gov.cn/pub/newsite/scb/ssgshyfljg/" +  date + "/" + link.
                                                        replace("./",""))
```

下载并转换 PDF:

```
# 下载 PDF 文件
for i in range(len(pdf_links)):
    pdf_link = pdf_links[i].replace("./","http://www.csrc.gov.cn/pub/newsite/scb/ssgshyfljg/")
    f_pdf = title[i] + ".pdf"
    f = requests.get(pdf_link, stream=True)
    with open(join(raw, f_pdf), 'wb') as  pdf:
        for content in f:
            pdf.write(content)
# 将PDF 转换为 Excel
wb = Workbook()
ws = wb.active
for i in range(len(title)):
    f_pdf = join(raw, title[i] + ".pdf")
    f_xlsx = title[i] + ".xlsx"
    with pdfplumber.open(f_pdf) as pdf:
        for page in pdf.pages:
            for table in page.extract_tables():
                for row in table:
                    ws.append(row)
    wb.save(join(output,f_xlsx))
```

2.3.2 Stata 数据清洗

主文档

采用项目思路，在开始写代码之前先通过 *main.do* 文档梳理思路，包括描述程序需要安装的外部命令、定义路径以及每个 *do* 文档的用途:

```
**************************************
*Function: 实证数据规范化处理
*Create date: 2021.11.16
*Last modify: 2011.11.18
**************************************

/*外部命令
ssc install fs, replace
ssc install openall, replace
ssc install nrow, replace
ssc install carryforward, replace
*/

* 定义路径
```

```
global path ".../Projects/Training"
global prgm "$path/program"
global c "$prgm/code"
global d "$prgm/data"
global raw "$d/rawData"
global o "$d/outputData"
global res "$prgm/result"

* 数据清理
do "$c/data-preparation.do"
```

数据清理

```
************************************
*Function: 清理上市公司行业分类
*Create date: 2021.11.18
*Last modify: 2011.11.18
************************************

cap mkdir $d/dtaData
cd $o
fs *.xlsx
foreach f in `r(files)'{
    import excel using "$o/`f'", clear
        *scalar fname = ustrregexrf(`"`f'"',".xlsx","")
    scalar year = substr(`"`f'"',1,4)
    scalar quarter = usubstr(`"`f'"',6,1)
    scalar fname = year + "Q" + quarter
    local fname = fname
    duplicates drop
    nrow
    carryforward _all,replace
    gen 行业门类与大类 = ustrregexs(0) + 行业大类代码 ///
    if ustrregexm(门类名称及代码,"[A-Z]") == 1
    gen year = real(year)
    save "$d/dtaData/`fname'.dta", replace
}

* 数据合并
cd $d/dtaData
openall

* 核查清洗
preserve
```

```
keep 行业门类与大类
duplicates drop
restore
```

3 总结

4 参考文献

参考文献

[1] Ben Jann. Creating LaTeX Documents from within Stata using Texdoc. 16(2):245–263.

A 附录

A.1 环境搭建搜索关键词

1. VsCode 配置 Python、Stata、LaTeX、Zotero
2. Sublime text 3 配置 Python、Stata、LaTeX
3. Zotero 安装 Better BibTeX 插件
4. Stata 与 LaTeX 联动 [1]

A.2 附录：公众号相关文章推荐

1. Zotero 入门简介
2. Python 文件操作
3. 使用 Python 提前 PDF 文本
4. Stata 与 Python 等效操作
5. Stata 书籍清单
6. Stata 杂乱文件自动分类