Stata 实战：
地级市宏观数据和上市公司数据的清洗与合并

主办方：连享会

主讲人：初虹（山东财经大学 · 经济学院）

2023 年 6 月 27 日

chuhong@mail.sdufe.edu.cn

课程导学

想完成一篇学术论文，Stata 应该学习到什么程度呢？苦学了 Stata 数月有余，为啥面对数据却还是茫然无措呢？想必这也是很多同学入门 Stata 时的困惑。

硬啃枯燥的语法，或许也能学到些东西，但缺少实践的土壤，很容易学了就忘，忘了也没了兴趣再学。对着下载的数据，清洗起来没有思路，敲出来的代码自然也缺少逻辑。我们常说要学以致用，就是要 **学中干、干中学**。数据清洗的过程也是软件学习的过程。

经管论文中，常见的数据主要为宏观数据、上市公司数据和微观调查数据。微观调查数据需要注意的点更多，更讲究「一库一策」。连享会 已有不少推文详细讲解了国内常见微观调查数据库的清洗步骤，并且给出了 Stata 代码。相比而言，宏观数据和上市公司数据的规范化程度更高、上手难度更低，因此这次讲座便以此二者为例，给大家过一遍 **数据处理** 这步都有哪些基本操作。

课程导学

宏观数据按照行政区划层级划分，主要可以分为省级、地市级和区县级。省级数据太简单，区县级又相对少见，于是这次讲座我们就用宏观数据里出镜率最高的地级市层面，选取常见的三大宏观数据库——**EPS、CNRDS 和统计年鉴**，以及提到上市公司数据，就绝对没跑儿的 **国泰安 CSMAR**，共四大数据库作为示例给大家演示。

我们会先讲数据下载，再讲数据清洗，最后讲数据合并。下载数据，也别小瞧，还真有些小技巧，有时候选择不同的数据下载样式，决定了不同的数据清洗难度。对于 CSMAR 数据库，我们还会给大家展示如何使用 CSMAR Stata API 来简化数据下载流程。

目标是在两个小时之内构造一份 **对接了地级市相关指标的上市公司面板数据**，以糅合数据下载、数据清洗与数据合并三大块，力图贴近现实中学术论文的应用化场景。

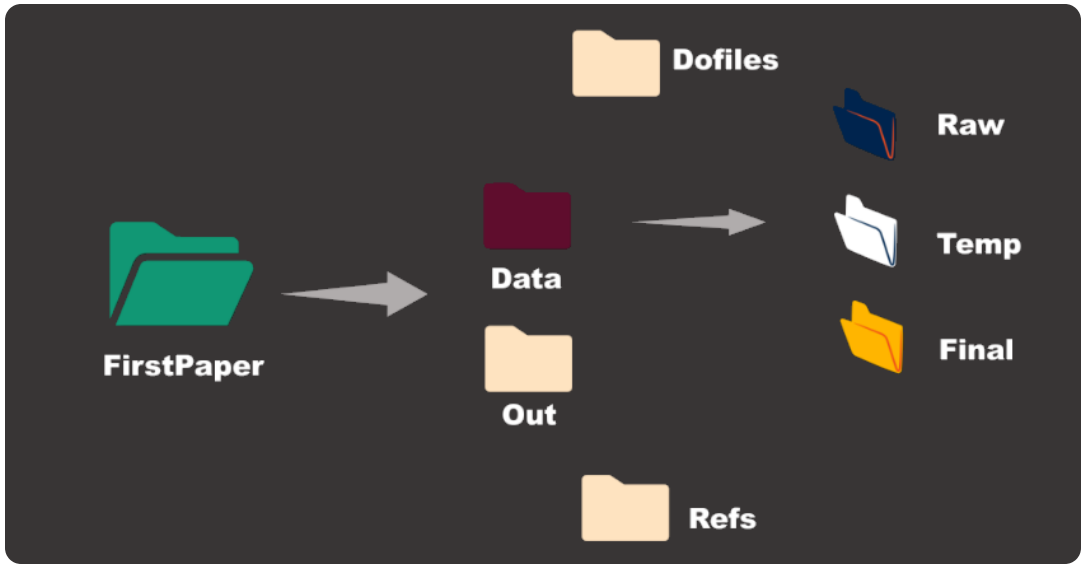
听完之后，希望会让你有「噢，原来 Stata 也没有那么难」的感觉。再辅以强化练习，想必你一定能够轻松驾驭 **Stata 进行数据清洗** 这一关。如果还能提振你继续进行学术创作的信心，那就更好了~

CONTENTS

一、地级市宏观数据库的导出、初步清洗与对接

一个编程好习惯：使用项目管理的思维管理论文

- 一篇论文一个父文件夹，下设不同子文件夹，分门别类存放 Data、Code、Out 等文件
- 体系的核心为：原始数据文件和 Code
- 定义路径：`global` 或 `local`；操作路径的常用命令：`dir` / `cdout` / `cd`
- 推荐阅读：Stata 基础：从论文文件夹体系的建立说起）



常用地级市宏观数据库

- EPS DATA ：涵盖经济、金融、会计、贸易、能源等领域实证与投资研究所需的绝大部分数据
  - 缺点：一次只能下载 5000 条；需要手动点选行政区划
- 中国经济与社会发展统计数据库 ：由中国知网（CNKI）出版，是我国官方历年发布重要数据的大型统计资料数据库，其中仍在连续出版的统计年鉴资料有150多种。
  - 缺点：只支持单年份下载；部分年鉴有密码保护（如《中国城市建设统计年鉴》）
- CSMAR 数据库 ：主打为股票、公司、金融等数据，省市常用的是经济研究系列和绿色经济系列
- CNRDS 数据库 ：分为特色库和基础库，含有很多特色数据
- CEIC 数据库 ：分为《中国经济数据库》、《全球经济数据库》和《世界经济趋势数据库》，并提供 CDMNext 在线访问权限，可以通过注册个人账号使用。

EPS 数据库的查找与导出

- 选择 **样式三** 的格式下载，数据清洗起来最方便。

样式一：

		2016	2017	2018	2019	2020
预算数（亿元）	北京	4998.46	5421.86	5785.8	5815	5467
	天津	2714	2838.5	2240	1980	2421.1
	河北省	2795.07	3068.62	3381.25	3719.53	3774.43
	山西省	1604.49	1579.01	2032.83	2311.39	2229.89
	内蒙古自治区	2023.52	1863.84	1712.73	1914.1	1962.34

样式二：

		2016	2017	2018	2019	2020
预算数（亿元）	北京	4998.46	5421.86	5785.8	5815.0	5467.0
预算数（亿元）	天津	2714.0	2838.5	2240.0	1980.0	2421.1
预算数（亿元）	河北省	2795.07	3068.62	3381.25	3719.53	3774.43
预算数（亿元）	山西省	1604.49	1579.01	2032.83	2311.39	2229.89
预算数（亿元）	内蒙古自治	2023.52	1863.84	1712.73	1914.1	1962.34

样式三：

指标	地区	时间	数值
预算数（亿元）	北京	2016	4998.46
预算数（亿元）	北京	2017	5421.86
预算数（亿元）	北京	2018	5785.8
预算数（亿元）	北京	2019	5815.0
预算数（亿元）	北京	2020	5467.0
预算数（亿元）	天津	2016	2714.0
预算数（亿元）	天津	2017	2838.5
预算数（亿元）	天津	2018	2240.0
预算数（亿元）	天津	2019	1980.0
预算数（亿元）	天津	2020	2421.1
预算数（亿元）	河北省	2016	2795.07
预算数（亿元）	河北省	2017	3068.62
预算数（亿元）	河北省	2018	3381.25
预算数（亿元）	河北省	2019	3719.53
预算数（亿元）	河北省	2020	3774.43
预算数（亿元）	山西省	2016	1604.49
预算数（亿元）	山西省	2017	1579.01
预算数（亿元）	山西省	2018	2032.83
预算数（亿元）	山西省	2019	2311.39
预算数（亿元）	山西省	2020	2229.89
预算数（亿元）	内蒙古自治	2016	2023.52

地级市-地级市数据对接：使用城市名称的前两个字

- 使用 cityname 的前两个中文字符进行地级市数据库间的对接
- 需要特别注意「张家口」和「张家界」，同时最好统一去掉「市」字

```
gen cname = substr(cityname, 1, 6)
replace cname = subinstr(cityname, "市", "", .) if (cname == "张家")
```

- 有些数据库可能同时包含的地级市和区县两个层级，此时还需要根据数据额外处理

```
replace cname = cityname if (cname == "乌兰") // 乌兰浩特、乌兰察布
replace cname = cityname if (cname == "阿拉") // 阿拉善、阿拉山口、阿拉尔
```

EPS 数据库清洗：基础版

```
import excel "$DR/EPS_区域&县域-中国城市数据库 (1).xls", firstrow clear // 对应修改文件名
    destring 时间 数值, replace
    rename (地区 时间 数值)(cityname year 行政区域土地面积)  // 对应修改变量名
    label var 行政区域土地面积 "平方公里 - 全市"  // 对应修改变量名和标签
    drop 指标 分类
    format cityname %10s
    drop if (cityname == "")
    gen cname = substr(cityname, 1, 6)
    replace cname = subinstr(cityname, "市", "", .) if (cname == "张家")
save "$DT/EPS_行政区域土地面积.dta", replace
```

EPS 数据库清洗：进阶版

```
foreach i of numlist 1/3 {
    import excel "$DR/EPS_区域&县域-中国城市数据库 (`i').xls", firstrow clear
        destring 时间 数值, replace

        local lbl = ustrregexra(指标[1], "^.*(|) ", "", .)
        // 把「指标」变量第一行括号内的内容作为「数值」变量的标签（正则表达式）
        label var 数值 "`lbl'"
        local name = ustrregexra(指标[1], " (.*", "", .)
        // 把「指标」变量第一行括号外的内容作为「数值」变量的名称

        rename (地区 时间 数值) (cityname year `name')
        drop 指标 分类
        format cityname %10s

        drop if (cityname == "")

        gen cname = substr(cityname, 1, 6)
        replace cname = subinstr(cityname, "市", "", .) if (cname == "张家")
    save "$DT/EPS_`name'.dta", replace
}
```

CNRDS 数据库清洗：基本版

```
import excel "$DR/CNRDS_学校数量_地级市.xlsx", firstrow clear
        labone, nrow(1)
        drop in 1

        destring _all, replace // Prishlnum 和 Hedistnum 含有特殊字符（: - /），无法通过 destring 转换
        gen prishlnum = real(Prishlnum)
        label var prishlnum "小学数（个）"
        gen hedistnum = real(Hedistnum)
        label var hedistnum "普通高等学校数（个）"
        drop Prishlnum Hedistnum

        rename (Cityname Year) (cityname year)
        drop if ustrregexm(Provname, "城市")
        drop if cityname == "合计"
        replace cityname = Provname if (cityname == "")  // 较常见的操作，直辖市的cityname为空，需要补上

        keep if Stacoverage == "全市"  // 默认下载「全市」和「市辖区」两个层面的数据
        drop Stacoverage

        gen cname = substr(cityname, 1, 6)
        replace cname = subinstr(cityname,  "市", "", .) if (cname == "张家")
save "$DT/CNRDS_学校数量_全市.dta", replace
```

CNRDS 数据库清洗：进阶版

```
foreach f in 学校数量 在校学生数 专任教师数 {
        import excel "$DR/CNRDS_`f'_地级市.xlsx", firstrow clear
                labone, nrow(1)  // nrow 1
                drop in 1

                destring _all, replace
                rename (Cityname Year) (cityname year)
                drop if ustrregexm(Provname, "城市")
                drop if cityname == "合计"
                replace cityname = Provname if (cityname == "")

                keep if Stacoverage == "全市"
                drop Stacoverage

                gen cname = substr(cityname, 1, 6)
                replace cname = subinstr(cityname, "市", "", .) if (cname == "张家")

                order cname year
        save "$DT/CNRDS_`f'_全市.dta", replace
}
```

年鉴取消密码保护：dxls / txls

- 统计年鉴数据的下载常会出现 **密码保护** 问题，在 Excel 里无法编辑，也无法导入 Stata。可以使用 Stata 外部命令 `dxls` 和 `txls` 进行转换。

```
net install dxls.pkg, from("https://gitee.com/kerrydu/clearpsdinexcel/raw/master")
net install txls.pkg, from("https://gitee.com/kerrydu/clearpsdinexcel/raw/master")

txls "D:\Before", todir ("D:\After")

* 最好使用全路径，使用全局暂元容易出错
* 文件路径不能超过128个字符
* 文件和文件夹不是只读
* 文件和文件夹路径不能含有<>?[]:|*和中文字符
```

推荐阅读 ：

- Stata数据处理：清洗中国城市建设统计年鉴： 推文版 、 视频版
- 推文 | Stata数据处理：批量处理被保护的年鉴数据-dxls-txls

统计年鉴数据库清洗：基础版

```
import excel "$DR/DropPassword/After/2019.xlsx", clear
        drop R
        label var B "本年完成投资-万元" // 省略其他变量的加标签
        rename A cityname  // 省略其他变量的重命名

        drop if cityname == ""
        drop if ustrregexm(cityname, "2019|续表|计量单位|continued|Measurement|城市")
        destring _all, replace
        format cityname %10s
        gen year = 2019
        order cityname year

        replace cityname = ustrregexra(cityname, " ", "", .)
        gen cname = substr(cityname, 1, 6)
        replace cname = subinstr(cityname,  "市", "", .) if (cname == "张家") // 张家口、张家界

        duplicates list cname year
        replace cname = cityname if (cname == "乌兰")  // 除张家口和张家界，还有重复值：乌兰、新疆、阿拉
        order cname year

        sort cname year complete  // 增加按照 CompletedInves 排序，目的是处理吉林（省）和吉林（市）
        duplicates drop cname year, force   // 默认保留第一次出现的
save "$DT/syb_2019", replace
```

统计年鉴数据库清洗：进阶版

```
fs "$DR/DropPassword/After/*.xlsx"
foreach xlsx in `r(files)' {
        import excel "$DR/DropPassword/After/`xlsx'", clear
                labone, nrow(5 6)   // 根据5/6行的内容添加标签
                drop R
                local year = substr(A[1], 7, 4) // 根据A列第一行的内容生成year
                gen year = `year'
                drop if (B == "") & (C == "") & (D == "") // 删除非城市名称的行
                drop if ustrregexm(A, "全 国|城市.*")
                foreach v of varlist _all { // 清洗标签、将标签去除中文字符后作为变量名
                        local lbl: var label `v'
                        local u = ustrregexra("`lbl'", "\s", "", .) // 去除标签中的空白字符（空格、换行、TAB等）
                        label var `v' `"`u'"'   // 注意：复合双引号
                        local j = ustrregexra("`u'", ".[\u4E00-\u9FA5]+", "", .) // 去除标签中的中文字符
                        cap rename `v' `j'
                }
                destring _all, replace
                rename NameofCities cityname
                replace cityname = ustrregexra(cityname, " ", "", .)
                gen cname = substr(cityname, 1, 6)
                replace cname = subinstr(cityname,  "市", "", .) if (cname == "张家")
                replace cname = cityname if (cname == "乌兰") // 乌兰、新疆、阿拉
                order cname year
                sort cname year CompletedInves  // 增加按照 CompletedInves 排序，目的是处理吉林（省）和吉林（市）
                duplicates drop cname year, force   // 默认保留第一次出现的
```

二、上市公司数据库的数据下载与清洗

CSMAR 数据下载：CSMAR Stata API

CSMAR API 提供了 Python、R、MATLAB 和 Stata 四种接口，支持我们直接 **使用代码下载数据**，而不必每次都要进入 CSMAR 官网、点选框格进行数据的筛选和下载，为我们从 CSMAR 上下载数据提供了极大的便利性。

- 前提：Stata 16+、仅支持 Windows 系统
- 下载安装 CSMAR-STATA
- 解压到 personal 文件夹，然后运行 `run.bat` 文件

> **推荐阅读**：
>
> - 国泰安数据下载：CSMAR Stata API： 推文版、 视频版

CSMAR 数据下载：CSMAR Stata API

- 用户登录：`login "account" "pwd"`
  - 账号只能为个人账号，不能使用机构账号
- 查看已购买的数据库：`getDbs`
- 查看已购买的数据表：`getTables "databaseName"`
- 查看数据表中所有的字段：`getFields"tableName"`
- 预览数据：`preview "tableName"`
- 查询数据表记录条数：`getDataCount "columns" "condition" "tableName" "startTime" "endTime"`
  - `condition`：条件，类似 SQL 条件语句，`"Stkcd like'3%'"`，但不支持 order by（该命令有默认的排序方式）
  - 时间关键字参数（非必填，如需填写格式为：`YYYY-MM-DD`）

CSMAR 数据下载：CSMAR Stata API

- 查询数据表数据：`getData "columns" "condition" "tableName" "startTime" "endTime"`
  - 一次最多只能加载 20 万条记录
  - `condition`：若超过 20 万记录，需分页查询，假设是 40 万条，需分两次进行条件设置为：
    - 第一次：`"Stkcd like'3%' limit 0, 200000"`
    - 第二次：`"Stkcd like'3%' limit 200000, 200000"`
- 打包数据：`pack "columns" "condition" "tableName" "startTime" "endTime"`
  - 该命令返回一个唯一标识：signCode
- 获取 Stata 下载记录详情：`getRecord "signCode"`
- 下载数据：`copy "sourcePath""targetPath"`

```
clear
login "accout" "pwd"

* 资产负债表
getFields "FS_Combas"  // getTables "财务报表"
pack "Stkcd,Accper,Typrep,A0b1103000,A001101000, A001111000,A001123000, ///
      "Typrep = 'A' AND Accper like '20%-12-31'" "FS_Combas" "2007-12-31" "2021-12-31"

/* In the process of packaging, you can check the packaging progress ///
through the identification code, which is [9474249550914232332] */

getRecord "9474284398324326400"
list

+-------------------------------------------------------------------------------------+
|                downloadPath        |          status          |          fileSize          |
|         财务报表/资产负债表          |          success         |              1.7MB         |
|-------------------------------------------------------------------------------------|
|                                                                 filePath |     downloadTime |
| http://file.csmar.com/group1/M00/24/EF/CuIKV2IHay6AHNYLABuWcY5UmCM860.zip | 2022-02-12 16:09 |
+-------------------------------------------------------------------------------------+

local filePath = filePath[1]
copy "`filePath'" "资产负债表.zip", replace
```

CSMAR 数据库清洗：基础版

```
* 清洗利润表
import excel "$DR/CSMAR_FS_Combas.xlsx", firstrow clear

    labone, nrow(1 2) concat("_")
    drop in 1/2
    destring _all, replace

    gen year = real(substr(Accper, 1, 4))  // 与下面两行命令等价
    * gen year = substr(Accper, 1, 4)
    * destring year, replace

    drop Accper Typrep ShortName
    rename Stkcd sid
    order sid year

save "$DT/CSMAR_FS_Combas.dta", replace
```

CSMAR 数据库清洗：进阶版

```
foreach f in FS_Combas FS_Comins FS_Comscfd STK_LISTEDCOINFOANL {
  import excel "$DR/CSMAR_`f'.xlsx", firstrow clear

    labone, nrow(1 2) concat("_")
    drop in 1/2
    destring _all, replace

    cap gen year = real(substr(Accper, 1, 4))
    cap gen year = real(substr(EndDate, 1, 4))

    dropvars Accper Typrep ShortName EndDate  // 思考：为什么不能使用 cap drop ?

    cap format IndustryName PROVINCE CITY %10s
    cap rename Stkcd sid
    cap rename Symbol sid
    order sid year

  save "$DT/CSMAR_`f'.dta", replace
}
```

地级市数据库与上市公司数据库合并

- CSMAR 上市公司数据库中「基本信息年度表」含有「上市公司注册地所在城市」变量 `CITY`，该变量含有部分 **县级市**，如果使用 `CITY` 与地级市数据库直接合并，则这些县级市无法匹配成功。
- 需要利用 CSMAR 数据库「基本信息年度表」中的 `CITYCODE`（行政区划代码而非行政区划名称）变量进行匹配

```
gen city_code = real(substr(strofreal(CITYCODE), 1, 4) + "00")
```

- 对应地级市数据库也需要使用地级市代码，而非地级市名称。这样就需要给地级市数据库事先匹配上地级市代码

```
use "$DT/地级市合并数据_2004_2019", clear
    merge m:1 cname using "$DR/337个地级市名称与代码对应表.dta", keep(3) nogen
save, replace
```

小结

- Stata 数据清洗常用命令：

```
import excel "文件名.xlsx", firstrow clear // 从 Excel 文件导入

destring <varlist>, replace (force) // 字符型 → 数值型
* 相比 destring, real() 函数可以处理特殊字符

rename (a b c) (A B C) // 批量重命名
* 进阶重命名：renvarlab

labone, nrow(#) // 将第 # 行的数据作为变量标签，还可以使用 concat("_") 链接不同行
nrow(#) // 将第 # 行的数据作为变量名

* 字符函数
substr(cityname, 1, 6)
subinstr(cityname, "市", "", .)
ustrregexra() ustrregexm() 使用正则表达式进行替换或查找，比上面两个更高级

format cityname %10s // 调整显示格式

merge 1:1 cname year using "B.dta", keep(3) nogen // 1:1（一对一）、1:m（一对多）、m:m（多对多）
```

小结

- 地级市数据库之间的合并：
  - 使用地级市名称前两个中文字符，需要特殊处理张家口市和张家界市
  - 下载的不同数据库的地级市数量不一致，比如统计年鉴含有大量县级市，此时必须根据 `duplicates list cname year` 进行其他重复值的处理
- 统计年鉴数据库取消密码保护：`txls / dxls`
- CSMAR 国泰安数据下载：CSMAR Stata API 简化数据下载流程
- 地级市数据库与上市公司数据库的合并：
  - 地级市数据库需要匹配地级市行政区划代码
  - 上市公司数据库「基本信息年度表」中的 **注册地所在城市名称**（`CITY`）变量，含有部分县级市，需要根据 **注册地所在城市的城市代码**（`CITYCODE`）进行匹配，而非注册地所在城市的城市名称

The End

chuhong@mail.sdufe.edu.cn

个人公众号：虹鹄山庄