

```

1
2
3      *=====
4      *      计量分析与STATA应用
5      *=====
6
7      *      主讲人：连玉君 博士
8
9      *      单   位：中山大学岭南学院金融系
10     *      电   邮：arlionn@163.com
11     *      主   页：http://blog.cnfol.com/arlion
12
13     *      ::第一部分::
14     *      Stata 操作
15
16
17
18     *      =====
19     *      +      课程目录      +
20     *      =====
21
22
23
24     *      =====
25     *      第一讲  STATA简介
26     *      =====
27
28 * 1.1 本课程简介
29 *      1.1.1 课程纲要
30 *      1.1.2 课程特点
31 *      1.1.3 课程配套资料
32 *      1.1.4 讨论和建议
33
34 * 1.2 STATA概貌
35 *      1.2.1 stata界面
36 *      1.2.2 首次使用STATA的一些基本设定
37
38 * 1.3 输入和导入数据
39 *      1.3.1 手动输入
40 *      1.3.2 从 .txt, excel 表格中粘贴
41 *      1.3.3 使用stata命令：infile, insheet, infix
42 *          1.3.3.1 以-tab-分隔的数据： -insheet- 命令
43 *          1.3.3.2 以 空格 分隔的数据： -infile- 命令
44 *          1.3.3.3 调入STATA格式的数据： -use- 命令
45 *          1.3.3.4 调入Excel格式的数据： -xmluse-命令
46 *          1.3.3.5 行列对调的数据
47 *      1.3.4 时间序列资料
48 *      1.3.5 面板资料
49 *      1.3.6 STATA官方提供的资料
50 *      1.3.7 其它软件中的数据
51
52 * 1.4 存储和导出数据
53 *      1.4.1 存储数据
54 *      1.4.2 导出和转换
55 *          1.4.2.1 -outfile-命令:导出为 .raw 文本格式
56 *          1.4.2.2 -outsheet-命令:导出为 -Tab- 分隔的文本文件
57 *          1.4.2.3 -xmlsave-命令:导出为 XML 格式
58 *          1.4.2.4 -dataout-命令:导出为 Word,Excel,Tex
59 *          1.4.2.5 -outdat- 命令:导出为 .spss, .rats, .limdep 格式
60
61 * 1.5 浏览资料
62 *      1.5.1 变量的名称
63 *      1.5.2 查看资料的结构
64 *          1.5.2.1 更改变量的存储类型
65 *          1.5.2.2 -list- 命令的使用
66 *          1.5.2.3 定义变量的显示格式
67 *          1.5.2.4 数据和变量的标签
68 *          1.5.2.5 附加说明文字
69 *          1.5.2.6 搜索变量
70 *      1.5.3 基本统计量
71 *          1.5.3.1 -summarize-命令
72 *          1.5.3.2 -codebook-命令
73 *          1.5.3.3 -inspect-命令
74 *          1.5.3.4 列表统计(table, tabulate)

```

---

|     |   |   |
|-----|---|---|
| 75  | * | 1.5.3.5 论文格式的统计表格(tabstat)                                |
| 76  |   |   |
| 77  | * | 1.6 执行指令  |
| 78  | * | 1.6.1 概览  |
| 79  | * | 1.6.2 命令的适用范围   |
| 80  | * | 1.6.2.1 列举多个变量  |
| 81  | * | 1.6.2.2 样本范围的限制   |
| 82  | * | 1.6.3 命令作用的增减: 使用选项                                       |
| 83  |   |   |
| 84  | * | 1.7 修改资料  |
| 85  | * | 1.7.1 数学表达式   |
| 86  | * | 1.7.2 变量的创建和修改  |
| 87  | * | 1.7.2.1 变量的存储类型   |
| 88  | * | 1.7.2.2 创建新变量   |
| 89  | * | 1.7.2.3 修改旧变量   |
| 90  | * | 1.7.2.4 删除变量和样本值  |
| 91  | * | 1.7.2.5 移动变量窗口中变量的位置                                      |
| 92  | * | 1.7.2.6 克隆已有变量  |
| 93  | * | 1.7.2.7 拆分变量  |
| 94  | * | 1.7.3 样本值的排序  |
| 95  |   |   |
| 96  | * | 1.8 log 文件: 记录你的分析过程                                      |
| 97  | * | 1.8.1 log 文件简介  |
| 98  | * | 1.8.2 将 log 文件转换为网页                                       |
| 99  | * | 1.8.2.1 -log2html-命令: 制作“单页”网页                            |
| 100 | * | 1.8.2.2 -hyperlog-命令: 制作“框架型”网页                           |
| 101 | * | 1.8.2.3 其他命令  |
| 102 |   |   |
| 103 | * | 1.9 do 文档: 高效快捷地执行命令                                      |
| 104 | * | 1.9.1 do 文档简介   |
| 105 | * | 1.9.1.1 打开 do 文档编辑器                                       |
| 106 | * | 1.9.1.2 保存和关闭   |
| 107 | * | 1.9.1.3 执行 do 文档  |
| 108 | * | 1.9.2 合理规划你的do文档  |
| 109 | * | 1.9.2.1 一些基本规则  |
| 110 | * | 1.9.2.2 注释语句  |
| 111 | * | 1.9.2.3 断行  |
| 112 | * | 1.9.2.4 大型 do 文档的设定                                       |
| 113 | * | 1.9.3 列印文字  |
| 114 | * | 1.9.3.1 -display-命令                                       |
| 115 | * | 1.9.3.2 列印的颜色   |
| 116 | * | 1.9.3.3 列印的位置   |
| 117 | * | 1.9.4 关于编辑器   |
| 118 | * | 1.9.5 do 文件的转换(制作网页教程)                                    |
| 119 |   |   |
| 120 | * | 1.10 stata与Excel、Word、LaTeX的亲密接触                          |
| 121 | * | 1.10.1 统计表格、矩阵的输出   |
| 122 | * | 1.10.1.1 输出基本统计量  |
| 123 | * | 1.10.1.2 输出相关系数矩阵   |
| 124 | * | 1.10.1.3 输出矩阵   |
| 125 | * | 1.10.1.4 其它说明   |
| 126 | * | 1.10.2 估计结果的输出  |
| 127 | * | 1.10.2.1 -esttab-命令: 回归结果的呈现                              |
| 128 | * | 1.10.2.2 -logout-命令: 输出【Excel、Word、TeX文档】                 |
| 129 | * | 1.10.2.3 -xml_tab-命令: 专业输出【Excel 文档】                      |
| 130 | * | 1.10.2.4 -outreg2-命令: 专业输出【Word、Excel文档】                  |
| 131 |   |   |
| 132 | * | 1.11 Stata 设定   |
| 133 | * | 1.11.1 Stata帮助  |
| 134 | * | 1.11.2 文件目录   |
| 135 | * | 1.11.3 Stata 外部命令的获取                                      |
| 136 | * | 1.11.3.1 外部命令的存储路径  |
| 137 | * | 1.11.3.2 外部命令的获取方式  |
| 138 | * | 1.11.3.3 外部命令的管理和更新                                       |
| 139 | * | 1.11.4 Stata 的系统参数  |
| 140 | * | 1.11.5 文件和文件夹的操作  |
| 141 | * | 1.11.5.1 文件的基本操作: 查找、查看、复制、编辑和删除                          |
| 142 | * | 1.11.5.2 使用stata打开-.txt-, -Word-, -Excel-, -iexplorer- 文件 |
| 143 | * | 1.11.5.2 文件夹的操作   |
| 144 | * | 1.11.6 每次启动时均需执行的命令(profile)                              |
| 145 | * | 1.11.7 常用快捷键  |
| 146 | * | 1.11.8 退出stata(exit)                                      |
| 147 |   |   |
| 148 |   |   |

---

|     |  |
|-----|--|
| 149 |  |
| 150 | * =====  |
| 151 | * 第二讲 数据处理   |
| 152 | * =====  |
| 153 |  |
| 154 | * 2.1 创建变量的更多技巧                                    |
| 155 | * 2.1.1 <code>_n</code> 和 <code>_N</code>          |
| 156 | * 2.1.1.1 <code>_n</code> 与 <code>_N</code>        |
| 157 | * 2.1.1.2 <code>_n</code> 与 <code>_N</code> 的应用    |
| 158 | * 2.1.2 虚拟变量的产生                                    |
| 159 | * 2.1.2.1 基本方式                                     |
| 160 | * 2.1.2.2 基于类别变量生成虚拟变量: <code>-tab-</code> 命令      |
| 161 | * 2.1.2.3 基于类别变量生成虚拟变量: <code>-xi-</code> 命令       |
| 162 | * 2.1.2.4 因子变量 ( <code>stata11</code> 的一大亮点)       |
| 163 | * 2.1.2.5 将连续变量转换为类别变量                             |
| 164 | * 2.1.2.6 利用条件函数产生虚拟变量                             |
| 165 | * 2.1.3 交乘项的产生                                     |
| 166 | * 2.1.4 <code>-egen-</code> 命令                     |
| 167 | * 2.1.4.1 <code>egen</code> 与 <code>gen</code> 的区别 |
| 168 | * 2.1.4.2 产生等差数列: <code>seq()</code> 函数            |
| 169 | * 2.1.4.3 填充数据: <code>fill()</code> 函数             |
| 170 | * 2.1.4.4 产生组内均值和中位数                               |
| 171 | * 2.1.4.5 跨变量的比较和统计                                |
| 172 | * 2.1.4.6 变量的标准化                                   |
| 173 | * 2.1.4.7 变量的平滑化 (Moving Average)                  |
| 174 | * 2.1.4.8 更多的 <code>egen()</code> 函数               |
| 175 |  |
| 176 | * 2.2 分位数  |
| 177 | * 2.2.1 分位数的基本概念                                   |
| 178 | * 2.2.2 <code>-pctile-</code> 命令                   |
| 179 | * 2.2.3 <code>-xtile-</code> 命令                    |
| 180 | * 2.2.4 <code>-_pctile-</code> 命令                  |
| 181 |  |
| 182 | * 2.3 重复样本值的处理                                     |
| 183 | * 2.3.1 检查重复的样本组                                   |
| 184 | * 2.3.2 标记和删除重复的样本组合                               |
| 185 |  |
| 186 | * 2.4 缺漏值的处理                                       |
| 187 | * 2.4.1 缺漏值简介                                      |
| 188 | * 2.4.2 缺漏值的标记                                     |
| 189 | * 2.4.3 查找/删除缺漏值                                   |
| 190 | * 2.4.3.1 缺漏值的形态                                   |
| 191 | * 2.4.3.2 删除缺漏值                                    |
| 192 | * 2.4.4 填补空缺( <code>gap</code> )                   |
| 193 | * 2.4.5 多重补漏分析( <code>multiple-imputation</code> ) |
| 194 | * 2.4.5.1 MI 简介                                    |
| 195 | * 2.4.5.2 实例分析                                     |
| 196 | * 2.4.5.3 MI <code>impute regress</code> 的假设条件     |
| 197 | * 2.4.5.4 其它补漏方法                                   |
| 198 | * 2.4.5.5 假设检验                                     |
| 199 |  |
| 200 | * 2.5 离群值的处理                                       |
| 201 | * 2.5.1 离群值的影响                                     |
| 202 | * 2.5.2 查找离群值                                      |
| 203 | * 2.5.3 离群值的处理                                     |
| 204 | * 2.5.3.1 删除                                       |
| 205 | * 2.5.3.2 对数转换                                     |
| 206 | * 2.5.3.3 缩尾处理                                     |
| 207 | * 2.5.3.4 截尾处理                                     |
| 208 |  |
| 209 | * 2.6 资料的合并和追加                                     |
| 210 | * 2.6.1 横向合并: 增加变量                                 |
| 211 | * 2.6.1.1 一对一合并                                    |
| 212 | * 2.6.1.2 多对一合并                                    |
| 213 | * 2.6.1.3 一对多合并                                    |
| 214 | * 2.6.1.4 一个例子                                     |
| 215 | * 2.6.2 横向关联: <code>-joinby-</code>                |
| 216 | * 2.6.3 纵向合并: 追加样本                                 |
| 217 | * 2.6.4 大型数据的处理                                    |
| 218 | * 2.6.5 一些有用的外部命令                                  |
| 219 |  |
| 220 | * 2.7 重新组合样本                                       |
| 221 | * 2.7.1 样本的转置                                      |
| 222 | * 2.7.2 数据的横纵变换                                    |

|     |   |                                   |
|-----|---|-----------------------------------|
| 223 | * | 2.7.3 样本的交叉组合                     |
| 224 | * | 2.7.3.1 -fillin- 命令               |
| 225 | * | 2.7.3.2 -cross-命令                 |
| 226 | * | 2.7.4 样本的堆砌                       |
| 227 |   |                                   |
| 228 | * | 2.8 文字变量的处理                       |
| 229 | * | 2.8.1 文字与数字的相互转换                  |
| 230 | * | 2.8.1.1 以文字类型存储的数字之转换             |
| 231 | * | 2.8.1.2 纯文字类别变量之转换                |
| 232 | * | 2.8.2 将数字转换成文字                    |
| 233 | * | 2.8.3 文字样本值的分解                    |
| 234 | * | 2.8.4 处理文字的函数                     |
| 235 | * | 2.8.4.1 文字函数简介                    |
| 236 | * | 2.8.4.2 例-1-: 上市公司日期、行业代码和所在地的处理  |
| 237 | * | 2.8.4.3 例-2-: 银企关系数据中银行名称的提取      |
| 238 | * | 2.8.4.4 例-3-: 处理不规则的日期            |
| 239 |   |                                   |
| 240 | * | 2.9 类别变量的分析                       |
| 241 | * | 2.9.1 类别数的统计                      |
| 242 | * | 2.9.2 交叉类别变量的生成                   |
| 243 | * | 2.9.3 分组统计量                       |
| 244 | * | 2.9.3.1 单层分组统计量                   |
| 245 | * | 2.9.3.2 二层次和三层次分组统计量              |
| 246 | * | 2.9.3.3 多层次分组统计量                  |
| 247 | * | 2.9.4 计算分组统计量的其它方法                |
| 248 | * | 2.9.4.1 -egen-命令                  |
| 249 | * | 2.9.4.2 转换原资料为分组统计量: -collapse-命令 |
| 250 | * | 2.9.5 图示分组统计量                     |
| 251 | * | 2.9.5.1 柱状图                       |
| 252 | * | 2.9.5.2 箱形图                       |
| 253 |   |                                   |
| 254 | * | 2.10 时间序列资料的处理                    |
| 255 | * | 2.10.1 简介                         |
| 256 | * | 2.10.1.1 声明时间序列: tsset 命令         |
| 257 | * | 2.10.1.2 检查是否有断点                  |
| 258 | * | 2.10.1.3 填充缺漏的日期                  |
| 259 | * | 2.10.1.4 追加样本                     |
| 260 | * | 2.10.2 时序变量的生成                    |
| 261 | * | 2.10.2.1 滞后项、超前项和差分项              |
| 262 | * | 2.10.2.2 产生增长率变量: 对数差分            |
| 263 | * | 2.10.2.3 日期变量的处理                  |
| 264 |   |                                   |
| 265 | * | 2.11 面板资料的处理                      |
| 266 | * | 2.11.1 声明面板资料: xtset 命令           |
| 267 | * | 2.11.2 公司数目和年度的统计                 |
| 268 | * | 2.11.2.1 面板资料的基本描述: xtides 命令     |
| 269 | * | 2.11.2.2 记录面板的资料形态: xtpattern 命令  |
| 270 | * | 2.11.2.3 统计公司数目: panels 命令        |
| 271 | * | 2.11.3 产生连续的公司代码                  |
| 272 | * | 2.11.4 处理为平行面板                    |
| 273 | * | 2.11.5 剔除IPO当年的数据                 |
| 274 | * | 2.11.6 行业发生变更的公司                  |
| 275 | * | 2.11.7 如何删除面板资料首尾的缺漏值?            |
| 276 | * | 2.11.8 仅保留连续 T 年以上可获得资料的公司        |
| 277 | * | 2.11.9 面板资料瘦身 I: 每隔 T 年保留一次资料     |
| 278 | * | 2.11.10 面板资料瘦身 II: 采用 P 年平均值进行估计  |
| 279 | * | 2.11.11 面板缺漏值的扩充                  |
| 280 | * | 2.11.12 变量的“去均值”和标准化处理            |
| 281 | * | 2.11.13 面板资料处理的其他主题               |
| 282 |   |                                   |
| 283 | * | 2.12 数据的查验和比较                     |
| 284 | * | 2.12.1 查验变量                       |
| 285 | * | 2.12.1.1 计数                       |
| 286 | * | 2.12.1.2 条件确认                     |
| 287 | * | 2.12.1.3 比较变量的大小                  |
| 288 | * | 2.12.2 查验两组数据                     |
| 289 | * | 2.12.2.1 查验两笔数据的观察值是否一致           |
| 290 | * | 2.12.2.2 查验两笔数据的变量是否一致            |
| 291 |   |                                   |
| 292 |   |                                   |
| 293 |   |                                   |
| 294 | * | =====                             |
| 295 | * | 第三讲 Stata绘图                       |
| 296 | * | =====                             |

|     |                                    |
|-----|------------------------------------|
| 297 |                                    |
| 298 | * 3.1 简介                           |
| 299 | * 3.1.1 Stata 图形的种类                |
| 300 | * 3.1.2 二维图命令的基本结构                 |
| 301 | * 3.1.3 几种常用图形的简单示例                |
| 302 | * 3.1.4 图形的管理                      |
| 303 | * 3.1.4.1 图形的保存                    |
| 304 | * 3.1.4.2 图形的导出                    |
| 305 | * 3.1.4.3 图形的调入                    |
| 306 | * 3.1.4.4 插入 Word                  |
| 307 | * 3.1.4.5 查询                       |
| 308 | * 3.1.4.6 重新显示图形                   |
| 309 | * 3.1.4.7 图形的合并                    |
| 310 | * 3.1.4.8 删除图形                     |
| 311 | * 3.1.5 图形的显示模式(绘图模板)              |
| 312 | * 3.1.5.1 显示模式种类                   |
| 313 | * 3.1.5.2 中文投稿的黑白图                 |
| 314 | * 3.1.5.3 stata 用户提供的模板            |
| 315 | * 3.1.5.4 创建自己的图形模板                |
| 316 |                                    |
| 317 | * 3.2 二维图选项                        |
| 318 | * 3.2.1 坐标类                        |
| 319 | * 3.2.1.1 坐标轴刻度(tick)及刻度标签(label)  |
| 320 | * 3.2.1.2 坐标轴标题: ytitle() xtitle() |
| 321 | * 3.2.1.3 坐标结构: yscale() xscale()  |
| 322 | * 3.2.1.4 双坐标系                     |
| 323 | * 3.2.2 标题类                        |
| 324 | * 3.2.2.1 标题的种类                    |
| 325 | * 3.2.2.2 示例                       |
| 326 | * 3.2.2.3 标题的位置                    |
| 327 | * 3.2.3 区域类                        |
| 328 | * 3.2.3.1 Stata图形的区域划分             |
| 329 | * 3.2.3.2 控制内区和外区的边距               |
| 330 | * 3.2.3.3 控制图形的纵横比例                |
| 331 | * 3.2.3.4 绘图区的显示模式                 |
| 332 | * 3.2.3.5 绘图区和全图区背景颜色的控制           |
| 333 | * 3.2.4 图例类                        |
| 334 | * 3.2.4.1 自动产生的图例                  |
| 335 | * 3.2.4.2 从新定制图例                   |
| 336 | * 3.2.4.3 图例的位置                    |
| 337 | * 3.2.4.4 多个图例的重排                  |
| 338 | * 3.2.4.5 线型的控制                    |
| 339 | * 3.2.5 附加线类                       |
| 340 | * 3.2.5.1 选项结构                     |
| 341 | * 3.2.5.2 附加线 <位置>                 |
| 342 | * 3.2.5.3 附加线 <风格>                 |
| 343 | * 3.2.5.4 附加线 <线宽>                 |
| 344 | * 3.2.5.4 附加线 <颜色>                 |
| 345 | * 3.2.5.5 附加线 <线型>                 |
| 346 | * 3.2.5.5 附加线属性的独立性                |
| 347 | * 3.2.6 文字与文本框                     |
| 348 | * 3.2.6.1 选项类别                     |
| 349 | * 3.2.6.2 文字和文本框的整体风格              |
| 350 | * 3.2.6.3 文本框属性                    |
| 351 | * 3.2.6.4 文字属性                     |
| 352 | * 3.2.7 图标类                        |
| 353 | * 3.2.7.1 简介                       |
| 354 | * 3.2.7.2 图标的位置                    |
| 355 | * 3.2.7.3 图标的大小                    |
| 356 | * 3.2.7.4 图标的角度                    |
| 357 | * 3.2.7.5 图标的颜色                    |
| 358 | * 3.2.8 其它选项                       |
| 359 | * 3.2.8.1 分组绘图                     |
| 360 | * 3.2.8.2 重新设置变量标签                 |
| 361 | * 3.2.8.3 重新设置变量显示格式               |
| 362 | * 3.2.8.4 重设图形种类                   |
| 363 |                                    |
| 364 | * 3.3 元素代号                         |
| 365 | * 3.3.1 颜色代号                       |
| 366 | * 3.3.2 线 相关的代号                    |
| 367 | * 3.3.2.1 线型代号                     |
| 368 | * 3.3.2.2 线宽代号                     |
| 369 | * 3.3.2.3 连接方式代号                   |
| 370 | * 3.3.3 标记符号的代号                    |

|     |   |                                      |
|-----|---|--------------------------------------|
| 371 | * | 3.3.3.1 符号样式                         |
| 372 | * | 3.3.3.2 符号的边界和填充                     |
| 373 | * | 3.3.3.3 符号代号一览                       |
| 374 | * | 3.3.4 文字相关的代号                        |
| 375 | * | 3.3.4.1 文字大小代号                       |
| 376 | * | 3.3.4.2 文字角度代号                       |
| 377 | * | 3.3.4.3 文字对齐方式的代号                    |
| 378 | * | 3.3.5 边距大小的代号                        |
| 379 |   |                                      |
| 380 | * | 3.4 常用图形示例                           |
| 381 | * | 3.4.1 散点图                            |
| 382 | * | 3.4.2 折线图                            |
| 383 | * | 3.4.3 区域图                            |
| 384 | * | 3.4.4 钉形图                            |
| 385 | * | 3.4.5 直方图                            |
| 386 | * | 3.4.6 密度函数图                          |
| 387 | * | 3.4.7 累积分布函数图                        |
| 388 | * | 3.4.8 线性/非线性 拟合图                     |
| 389 | * | 3.4.9 矩阵图：显示变量间的相关性                  |
| 390 | * | 3.4.10 柱状图                           |
| 391 | * | 3.4.10.1 一维柱状图                       |
| 392 | * | 3.4.10.2 二维柱状图                       |
| 393 | * | 3.4.11 点图                            |
| 394 | * | 3.4.12 函数图                           |
| 395 | * | 3.4.13 合图示例                          |
| 396 | * | 3.4.14 三维图形                          |
| 397 | * | 3.4.15 地 图                           |
| 398 |   |                                      |
| 399 | * | 3.5 结语                               |
| 400 |   |                                      |
| 401 |   |                                      |
| 402 |   |                                      |
| 403 |   |                                      |
| 404 | * | =====                                |
| 405 | * | 第四讲 矩阵操作                             |
| 406 | * | =====                                |
| 407 |   |                                      |
| 408 | * | 4.1 矩阵的基本操作                          |
| 409 | * | 4.1.1 基本定义方式                         |
| 410 | * | 4.1.2 矩阵的管理                          |
| 411 | * | 4.1.2.1 矩阵的名称                        |
| 412 | * | 4.1.2.2 列示矩阵                         |
| 413 | * | 4.1.2.3 矩阵的行数和列数                     |
| 414 | * | 4.1.2.4 查找/删除矩阵                      |
| 415 | * | 4.1.2.5 查验矩阵中是否存在缺漏值                 |
| 416 | * | 4.1.3 矩阵的行名和列名                       |
| 417 | * | 4.1.4 选取部分矩阵                         |
| 418 | * | 4.1.4.1 选取1个元素：1*1矩阵                 |
| 419 | * | 4.1.4.2 选取子矩阵                        |
| 420 | * | 4.1.4.3 矩阵元素的修改                      |
| 421 | * | 4.1.5 更一般化的矩阵定义                      |
| 422 | * | 4.1.6 常用矩阵的定义                        |
| 423 | * | 4.1.6.1 单位矩阵                         |
| 424 | * | 4.1.6.2 常数矩阵                         |
| 425 | * | 4.1.6.3 元素为随机数的矩阵                    |
| 426 | * | 4.1.6.4 对角矩阵                         |
| 427 | * | 4.1.7 变量和矩阵的相互转换                     |
| 428 | * | 4.1.7.1 变量->矩阵                       |
| 429 | * | 4.1.7.2 矩阵->变量                       |
| 430 | * | 4.1.8 用矩阵存储统计结果                      |
| 431 | * | 4.1.8.1 以矩阵方式呈现tabstat命令的结果          |
| 432 | * | 4.1.8.2 更一般化的矩阵存储                    |
| 433 | * | 4.1.9 采用变量的方式操作矩阵                    |
| 434 | * | 4.1.9.1 对矩阵中的各列进行变换和运算               |
| 435 | * | 4.1.9.2 矩阵元素的数学变换                    |
| 436 | * | 4.1.10 矩阵的保存和调入                      |
| 437 | * | 4.1.10.1 将矩阵保存为 .dta 文档中             |
| 438 | * | 4.1.10.2 将矩阵保存到 txt, word, excel 文档中 |
| 439 |   |                                      |
| 440 | * | 4.2 矩阵运算                             |
| 441 | * | 4.2.1 矩阵的基本运算                        |
| 442 | * | 4.2.1.1 加、减、乘                        |
| 443 | * | 4.2.1.2 直乘                           |
| 444 | * | 4.2.1.3 哈式乘法                         |



|     |   |                              |
|-----|---|------------------------------|
| 445 | * | 4.2.1.4 矩阵元素的数学变换            |
| 446 | * | 4.2.1.5 矩阵与单值的运算             |
| 447 | * | 4.2.2 矩阵的转置                  |
| 448 | * | 4.2.3 矩阵的逆矩阵                 |
| 449 | * | 4.2.3.1 矩阵的行列式               |
| 450 | * | 4.2.3.2 矩阵求逆                 |
| 451 | * | 4.2.4 矩阵的向量化                 |
| 452 | * | 4.2.5 矩阵的对角值                 |
| 453 | * | 4.2.6 交乘矩阵的定义                |
| 454 | * | 4.2.6.1 简单交乘矩阵               |
| 455 | * | 4.2.6.2 加权交乘矩阵               |
| 456 | * | 4.2.6.3 用户自行设定的权重            |
| 457 | * | 4.2.6.3 特殊加权交乘矩阵             |
| 458 |   |                              |
| 459 | * | 4.3 矩阵的解析                    |
| 460 | * | 4.3.1 线性相关、线性独立和正交向量         |
| 461 | * | 4.3.2 矩阵的秩                   |
| 462 | * | 4.3.3 特征根和特征向量               |
| 463 | * | 4.3.4 正定矩阵和负定矩阵              |
| 464 | * | 4.3.5 裘氏分解                   |
| 465 |   |                              |
| 466 | * | 4.4 关于矩阵的进一步说明               |
| 467 | * | 4.4.1 矩阵函数                   |
| 468 | * | 4.4.2 返回系统中的矩阵               |
| 469 | * | 4.4.3 定义约束矩阵                 |
| 470 | * | 4.4.4 矩阵与暂元的相关操作             |
| 471 | * | 4.4.5 矩阵对内存的需求               |
| 472 |   |                              |
| 473 |   |                              |
| 474 |   |                              |
| 475 |   |                              |
| 476 |   |                              |
| 477 |   | *=====                       |
| 478 | * | * 第五讲 STATA 编程初步             |
| 479 |   | *=====                       |
| 480 |   |                              |
| 481 | * | 5.1 stata程序简介                |
| 482 | * | 5.1.1 Stata 程序的基本结构          |
| 483 | * | 5.1.2 程序的执行                  |
| 484 | * | 5.1.2.1 第一种执行方式: ado 文档执行方式  |
| 485 | * | 5.1.2.2 第二种执行方式: run(Ctrl+R) |
| 486 | * | 5.1.3 程序的管理                  |
| 487 | * | 5.1.4 避免列印过多的结果              |
| 488 | * | 5.1.5 避免程序因错误而中断             |
| 489 | * | 5.1.6 避免数据在程序执行过后有所变动        |
| 490 | * |                              |
| 491 | * | 5.2 单值(scalar)               |
| 492 | * | 5.2.1 存放数值                   |
| 493 | * | 5.2.1 存放数值                   |
| 494 | * | 5.2.2 存放字符串                  |
| 495 | * | 5.2.3 执行命令后的单值结果             |
| 496 | * | 5.2.4 单值的管理                  |
| 497 | * |                              |
| 498 | * | 5.3 暂元(local)                |
| 499 | * | 5.3.1 暂元的定义和引用               |
| 500 | * | 5.3.1.1 暂元的基本功能              |
| 501 | * | 5.3.1.2 数学运算符的处理             |
| 502 | * | 5.3.1.3 `"'`暂元名`"'` (长引号)    |
| 503 | * | 5.3.1.4 暂元中的暂元               |
| 504 | * | 5.3.1.5 暂元引用机制的简化            |
| 505 | * | 5.3.2 全局暂元                   |
| 506 | * | 5.3.3 暂元的管理                  |
| 507 | * |                              |
| 508 | * | 5.4 其它暂时性物件                  |
| 509 | * | 5.4.1 暂时性变量                  |
| 510 | * | 5.4.2 暂时性矩阵和暂时性单值            |
| 511 | * | 5.4.3 暂时性文件                  |
| 512 | * |                              |
| 513 | * | 5.5 控制语句                     |
| 514 | * | 5.5.1 循环语句                   |
| 515 | * | 5.5.1.1 条件循环: while 语句       |
| 516 | * | 5.5.1.2 forvalues 语句         |
| 517 | * | 5.5.1.3 foreach 语句           |
| 518 | * | 5.5.2 条件语句                   |

---

```
519 *          5.5.2.1 if 语句
520 *          5.5.2.2 一些有用的条件函数
521 *
522 * 5.6 引用 Stata 命令的返回值
523 *      5.6.1 留存在内存中的结果
524 *      5.6.2 r-class
525 *      5.6.3 e-class
526 *      5.6.4 c-class
527
528
```



```

1
2
3
4
5      *      -----
6      *      -----  计量分析与STATA应用  -----
7      *      -----
8
9      *      主讲人：连玉君 博士
10
11     *      单   位：中山大学岭南学院金融系
12     *      电   邮：arlionn@163.com
13     *      主   页：http://blog.cnfol.com/arlionn
14
15     *      ::第一部分::
16     *      Stata 操作
17     *      =====
18     *      第一讲  STATA简介
19     *      =====
20
21
22     * cd D:\stata11\ado\personal\Net_course_A\Al_intro
23
24     cd `c(sysdir_personal)'Net_course_A\Al_intro
25
26
27 *-----
28 *   本讲目录
29 *-----
30 * 1.1  本课程简介
31 * 1.2  STATA概貌
32 * 1.3  输入和导入数据
33 * 1.4  存储和导出数据
34 * 1.5  浏览资料
35 * 1.6  执行指令
36 * 1.7  修改资料
37 * 1.8  log 文件：记录你的分析过程
38 * 1.9  do 文档：高效快捷地执行命令
39 * 1.10 stata与Excel、Word、LaTeX的亲密接触
40 * 1.11 Stata 设定
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74

```

```

75
76      *=====
77      *      计量分析与STATA应用
78      *=====
79
80      *      主讲人：连玉君 博士
81
82      *      单  位：中山大学岭南学院金融系
83      *      电  邮：arlionn@163.com
84      *      主  页：http://blog.cnfol.com/arlion
85
86      *      ::第一部分::
87      *      Stata 操作
88      *      =====
89      *      第一讲  STATA简介
90      *      =====
91      *      -1.1-  本课程简介
92
93
94
95
96      *-----
97      *->  Stata 是何方神圣?
98      *-----
99
100      *  短小精悍
101
102      *  运算速度极快
103
104      *  绘图功能卓越
105
106      *  更新和发展速度惊人
107
108
109
110      *-----
111      *->  1.1  本课程简介
112      *-----
113
114      *      ==本节目录==
115
116      *      1.1.1  课程纲要
117      *      1.1.2  课程特点
118      *      1.1.3  课程配套资料
119      *      1.1.4  课程配套资料的使用方法
120      *      1.1.5  讨论和建议
121
122
123      *
124      *  1.1.1  课程纲要
125
126
127      第一部分：Stata 操作
128      1. Stata简介
129      2. 数据处理
130      3. STATA绘图
131      4. 矩阵运算
132      5. STATA编程初步
133
134      第二部分：计量分析与Stata应用 (STATA高级班，已发布)
135      1. 普通最小二乘法 (OLS)
136      2. 广义最小二乘法 (GLS)
137      3. 非线性最小二乘法 (NLS)
138      4. 最大似然估计 (MLE)
139      5. 工具变量法与 GMM
140      6. 时间序列分析
141      7. 面板数据模型
142      8. Stata高级程序
143      9. Monte Carlo模拟与 Bootstrap(自抽样)
144
145      第三部分：Stata 应用专题 (即将发布)
146      1. Mata 语句高级程序
147      2. Logit/Probit 模型
148      3. Tobit 模型

```

---

```

149      4. Duration 模型
150      5. 事件研究法
151      6. Treatment 效应模型(Heckman, DID, PSM 等)
152      7. 分位数回归模型
153      8. 一般化线性回归模型(GLM)
154      9. 多元判别分析(discrim)
155      10. 因子分析和聚类分析
156      11. 假设检验
157      12. 广义矩估计 GMM 编程
158      13. Panel Data B(门槛面板\Panel VAR\Panel联立方程等)
159                                     */
160
161  *
162  * 1.1.2 课程特点
163
164      * 系统有序的结构安排, 帮助您快速建立Stata的学习架构
165
166      * 注重与实际应用相结合
167
168      * 翔实的配套资料
169
170      *-本讲义的 do-file 以及 PDF 格式
171
172
173
174  *
175  * 1.1.3 课程配套资料
176
177      *-本课程中使用的 do 文档和 ado 文档
178
179      *-stata do-file 格式, 可供练习操作
180      cd D:\stata11\ado\personal\Net_course_A
181      doedit Al_intro.do
182      *-or
183      doedit D:\stata11\ado\personal\Net_course_A\Al_intro.do
184
185      *-PDF 格式, 可供打印
186      cd D:\stata11\ado\personal\Net_course_A\pdf_dofiles
187      shellout Al_intro.pdf
188
189      *-课程的详细目录, 快速查询
190      shellout Course_A_contents.pdf
191
192
193      *-范例数据
194      cd D:\stata11\ado\examples  \\建议存放于此处
195      cdout
196
197      * 对于登陆国际网有困难的学员, 提供STATA官方范例数据包
198
199      * STATA外部命令包: plus(500于条)
200      ado  // 呈现已经安装的外部命令
201
202
203  *
204  * 1.1.4 课程配套资料的使用方法
205
206      *-1.1.4.1 课程配套资料的存放位置
207
208      *-我们提供的压缩包: 只需解压后放置于 D 盘根目录下即可
209      * 注意: D:\stata11 而非 D:\stata11\stata11
210
211      *-若用自己的stata软件, 需做如下设定:
212
213      * (1) profile.do 文件放置于stata安装目录下,
214      *      如 D:\stata11\profile.do
215      *      注: 若你已经自行设定了该文件,
216      *      请将我的profile文件合并到你的文件中
217
218      * (2) 重新打开 stata, 若上述文件设定无误, 则会显示
219      *      "running D:\stata11\profile.do ..."
220
221      * (3) 输入 sysdir 命令, 会显示如下信息
222      *
```

---

```
223      *      STATA:   D:\stata11\
224      *      UPDATES: D:\stata11\ado\updates\
225      *      BASE:    D:\stata11\ado\base\
226      *      SITE:    D:\stata11\ado\site\
227      *      PLUS:    D:\stata11\ado\plus\      // 存放和下载外部命令的位置
228      *      PERSONAL: D:\stata11\ado\personal\ // 个人文件夹
229      *      OLDPLACE: D:\stata11\ado\myado\     // 自己编写的程序
230
231
232      *-1.1.4.2 如何打来本讲义 (do-files)
233
234      *-方法1: 依次点击
235      *      "New do-file editor"-->File-->Open 指向如下路径
236      *      或输入 doedit, 然后点击 File-->Open
237      *      D:\stata11\ado\personal\Net_course_A
238      *      双击 A1_intro 即可
239
240      *-方法2: 依次输入如下命令
241      *      cd D:\stata11\ado\personal //若屏幕左下方显示的路径已在此处, 可省略
242      *      cd Net_course_A
243      *      doedit A1_intro.do
244      *-or
245      *      doedit D:\stata11\ado\personal\Net_course_A\A2_data.do
246
247
248      *-1.1.4.3 关于范例数据
249
250      *-stata官方的范例数据
251
252      *      help dta_contents // (File-->Example Datasets)
253
254      *-注: 多数已经下载, 存放于 D:\stata11\ado\Examples
255      *      打开方式 File-->Open-->D:\stata11\ado\Examples
256
257      *-本课程的范例数据
258      *      cd D:\stata11\ado\personal\Net_course_A\A1_intro
259      *      cdout
260      *      dir *.dta
261
262
263      *
264      * 1.1.5 讨论和建议
265
266      *-人大论坛【计量版】之【STATA专版】:
267      *      view browse "http://www.pinggu.org/bbs/forum-67-1.html"
268
269      *-人大论坛【统计软件培训班VIP在线答疑区】
270      *      http://www.pinggu.org/bbs/forum-114-1.html
271      *      view browse "http://www.pinggu.org/bbs/forum-114-1.html"
272
273      *      【Arlion 的博客】http://blog.cnfol.com/arlion
274      *      在百度中搜索关键词 “连玉君 博客”
275      *      view browse "http://blog.cnfol.com/arlion"
276
277      *      【E-mail】: arlionn@163.com
278
279      *      【连玉君主页】:
280      *      view browse ///
281      *      "http://www.lingnan.net/intranet/teachinfo/dispuser.asp?name=lianyj"
282
283      *-其它: 参见 1.11.1 小节
284
285
286
287
288
289
290
291
292
293
294
295
296
```

```

297
298      *=====
299      *      计量分析与STATA应用
300      *=====
301
302      *      主讲人：连玉君 博士
303
304      *      单  位：中山大学岭南学院金融系
305      *      电  邮：arlionn@163.com
306      *      主  页：http://blog.cnfol.com/arlion
307
308      *      ::第一部分::
309      *      Stata 操作
310      *      =====
311      *      第一讲  STATA简介
312      *      =====
313      *      -1.2- STATA 概貌
314      *      -1.3- 输入和导入数据
315      *      -1.4- 存储和导出数据
316
317
318      cd `c(sysdir_personal)'Net_course_A\A1_intro
319
320
321      *-----
322      *-> 1.2  STATA 概貌
323      *-----
324
325      *      ==本节目录==
326
327      *      1.2.1 stata界面
328      *      1.2.2 首次使用STATA的一些基本设定
329
330
331      *
332      * 1.2.1 STATA界面
333
334      * 四个窗口，两组菜单条
335
336      *
337      * -----
338      * 两种执行命令的方式
339
340      * 第一种：菜单
341
342      * 第二种：命令
343
344      * 实例 1->
345      * 一份简单的 do 文档
346      *      doedit L1_intro_log_cs.do
347
348      * -在 do文档中执行命令的快捷方式：Ctrl+D
349
350      * 实例 2->
351      * 连玉君,钟经樊.中国上市公司资本结构动态调整机制研究.
352      *      南方经济,2007(1):23-38.
353      *      doedit L1_intro_NFJJ.do
354
355      *
356      * - 1.2.2 首次使用STATA的一些基本设定
357
358      * -初次使用时界面偏好的设定
359
360      *      help window manage
361
362      * -设定方法
363      *      Edit-->Preference-->General Preference 按喜好设定
364      *      注：可进一步设定图形偏好、do-editor的风格等
365
366      * -保存设定
367      *      Edit-->Preference-->Save...-->New... 任意输入一个名称，如 song12
368      *      window manage prefs save song_12
369
370

```

```

371      * -调入已有的界面偏好设定:
372      *      Edit-->Preference-->Load...-->选择你喜欢的设定
373      *      window manage prefs load song_12
374
375
376      *-stata11 对中文的支持问题
377
378      *      -[Results]窗口
379      *      Edit-->Preference-->General Preference Results Color
380      *      选择 "Classic"
381      *      如此可以保证-Results-窗口中的中文字符得以正常显示
382
383      *      -[help viewer]窗口
384      *      Edit-->Preference-->General Preference Viewer Color
385      *      选择 "Custom 1"
386      *      去掉所有 "Bold" 前面的对勾, 如此可保证help文件正常显示
387
388
389      *-Stata11 手册的设定
390      *      请将stata11手册(16个pdf文档)放置于 D:\stata11\utilities
391      *      使用方法1: Help > PDF Documentation 可打开整个PDF帮助
392      *      help regress --> [section]Also see --> Manual:[R] regress
393      *      help regress
394
395
396      *-文件目录
397      *      pwd // 显示stata当前工作的路径
398      *      cd D:\stata11\ado\personal // 进入指定文件夹
399      *      sysdir // stata官方文件的路径
400      *      doedit D:\stata11\profile.do // 每次启动时需要立刻执行的命令
401      *      详见: 1.11.2 小节
402
403
404
405
406
407
408      *=====
409      *      计量分析与STATA应用
410      *=====
411
412      *      主讲人: 连玉君 博士
413
414      *      单 位: 中山大学岭南学院金融系
415      *      电 邮: arlionn@163.com
416      *      主 页: http://blog.cnfol.com/arlion
417
418      *      ::第一部分::
419      *      Stata 操作
420      *      =====
421      *      第一讲 STATA简介
422      *      =====
423      *      -1.3- 输入和导入数据
424      *      -1.4- 存储和导出数据
425
426      * 实证分析的第一步: 数据处理!
427      * 收集数据、存储、修改、分析、输出结果
428
429      *-----
430      *-> 1.3 输入和导入数据
431      *-----
432
433      *      ==本节目录==
434
435      *      1.3.1 手动输入
436      *      1.3.2 从 .txt, excel 表格中粘贴
437      *      1.3.3 使用stata命令: infile, insheet, infix
438      *      1.3.3.1 以-tab-分隔的数据: -insheet- 命令
439      *      1.3.3.2 以 空格 分隔的数据: -infile- 命令
440      *      1.3.3.3 调入STATA格式的数据: -use- 命令
441      *      1.3.3.4 调入Excel格式的数据: -xmluse-命令
442      *      1.3.3.5 行列对调的数据
443      *      1.3.4 时间序列资料
444      *      1.3.5 面板资料

```

---

```

445      *      1.3.6 STATA官方提供的资料
446      *      1.3.7 其它软件中的数据
447
448
449      *      =本节命令=
450      *      =====
451      * input, infile, insheet, type, rename, xpose, cd
452      * dataout
453      *      =====
454
455
456      *
457      * 三种方式:
458
459      *      手动输入
460      *      从 txt 或 Excel 文档中粘贴
461      *      使用 Stata 命令
462
463
464      *
465      * -1.3.1 手动输入 (极少使用)
466
467      clear
468      input x y z
469      1 2 3
470      4 5 6
471
472      end
473      save mydata, replace // 保存数据
474      use mydata, clear // 调入数据
475
476      *-1.3.1.1 -clear- 命令的使用 (stata11 更新了其功能)
477
478      *-stata运算的原理(内存的使用)
479
480      *-内存中存储的内容
481      sysuse auto, clear
482      des
483      label list
484      clear // 注意Variables窗口的变化
485      label list
486
487      sysuse auto, clear // clear 并不影响硬盘上存储的数据
488
489      sum price weigh turn
490      return list // 内存中存储的统计结果
491
492      reg price weight turn foreign
493      ereturn list // 内存中存储的回归结果
494
495      clear results
496      ret list
497      eret list
498
499      matrix A = I(5)
500      mat list A
501      mat dir
502      clear matrix
503      mat dir
504
505
506      *
507      * -1.3.2 从 .txt, excel 表格中粘贴
508
509      * 基本要求: 数据是-Tab-键分隔的
510
511      shellout d1.txt // -tab-键分隔的数据, 可以直接copy-paste
512      shellout d1.xls // Excel格式的数据, 亦可以直接copy-paste
513
514      edit // 打开数据编辑器, 贴入后可保存之
515
516
517
518      *


```

---



```
519  *-1.3.3 使用stata命令: infile, insheet, infix, use, xmluse
520
521  *-1.3.3.1 以 -tab- 分隔的数据: -insheet-
522
523      type d1.txt // 查看原始资料的形态
524      type d1.txt, showtabs
525      shellout d1.txt
526      insheet using d1.txt, clear
527
528      type d11.txt // 一份没有变量名称的数据
529      insheet using d11.txt, clear
530      rename v1 price
531      rename v2 weight
532      rename v3 length
533
534  * 亦可在输入数据时, 指定变量名称
535      insheet price weight length using d11.txt, clear
536
537
538  *-1.3.3.2 以 空格 分隔的数据: -infile-
539
540      shellout d21.txt
541      insheet using d21.txt, clear
542          //空格 分隔的数据无法直接用-insheet-命令导入
543      insheet using d21.txt, clear delimiter(" ")
544          //需要通过 delimiter 选项指定"分隔符号"
545      infile v1 v2 v3 using d21.txt, clear
546          //空格 分隔的数据用-infile-命令导入比较方便*/
547
548  * 包含文字变量的情形
549      shellout d2.txt
550      infile using d2.txt, clear // 错误的方式
551      infile v1-v5 using d2.txt, clear // 文字变量全部变成了缺漏值
552      browse // 指定变量类型(下面)
553      infile str30 v1 int v2 int v3 int v4 str10 v5 ///
554          using d2.txt, clear
555      browse
556
557  * 逗号 分隔的数据
558      type d3.txt
559      shellout d3.txt
560      infile str30 v1 int v2 int v3 int v4 str10 v5 using d3.txt, clear
561
562
563  *-1.3.3.3 调入STATA格式的数据
564
565      use d3.dta, clear
566      use "D:\stata11\ado\Examples\XTFiles\invest2.dta", clear
567      sysuse auto, clear
568
569  *-说明: 使用 STATA9 无法打开 STATA10\11 版本下存储的数据,
570  *      此时可采用外部命令 -use10- 打开stata10存储的数据。
571
572
573  *-1.3.3.4 调入Excel格式的数据: -xmluse-命令
574
575  *-注意: 需要把 file.xls 另存为 file.xml (另存类型选择"XML表格")
576
577      dir *.xls
578      shellout d1.xls
579
580      xmluse d1.xls, doctype(excel) clear firstrow // 错误! .xls 不可
581
582      shellout d1.xls // "另存为" -->XML表格 更改文件的存储类型
583
584      dir d1.* // 显示当前目录下以 d1. 开头的文件
585
586      xmluse d1.xml, doctype(excel) clear firstrow // 正确! .xml 可以
587
588      browse // 第一列数据很宽, 为什么?
589      des // make 变量被自动存储为 str244
590
591      compress // 精简资料的存储结构
592      des
```

```

593     browse
594
595     *-xmluse 的其它选项
596         help xmluse
597
598
599     *-1.3.3.5 行列对调的数据
600
601         shellout d5.txt           // 常规数据
602         shellout d51.txt          // 对调数据
603
604         insheet using d51.txt, clear
605         browse
606         xpose, clear              // 对调
607         browse
608
609         rename v1 year            // 给变量重命名
610         rename v2 invest
611         rename v3 income
612         rename v4 consume
613
614
615     *
616     * 1.3.4 时间序列资料 
617
618         help tsset
619
620         sysuse gnp96.dta, clear
621
622         tsset date                // 指定时间变量
623
624         gen gg = (gnp96-L.gnp96)/L.gnp96 // 增长率
625
626         tsset, clear              // 清除时间变量
627
628         gen gg2 = (gnp96-L.gnp96)/L.gnp96 // 错误!
629
630
631     *
632     * 1.3.5 面板资料
633
634         type d6_panel.txt
635         insheet using d6_panel.txt, clear
636         tsset code year          // stata8.0 以下版本适用
637         xtset code year          // stata9.0 以上版本适用
638
639     * xpose 命令同样适用于面板数据资料
640         shellout d6_pdpose.txt
641         insheet using d6_pdpose.txt, clear
642         xpose, clear
643         list, sepby(v1)
644
645
646     *
647     * 1.3.6 STATA官方提供的资料
648
649         help dta_contents
650         help dta_examples
651         help dta_manuals
652         use http://www.stata-press.com/data/r9/educ99gdp.dta, clear
653         webuse lifeexp, clear     // 从stata官网获取数据 (等价于如下命令)
654         use http://www.stata-press.com/data/r10/lifeexp, clear
655
656
657     *
658     * 1.3.7 其它软件中数据
659
660     * -Stat/Transfer- 软件: 快捷地在不同软件数据格式之间转换
661     * 在stata内部, 可以使用 -stcmd- 命令调用 Stat/Transfer, 并完成数据的转换
662     * 对于需要转换大量数据的用户而言, 这个方法很好, 且具有可重复性
663     * 可采用 findit 命令搜索并下载如下命令, 如
664
665         findit usespss
666

```

```

667 * -usespss- 将 SPSS 格式的数据导入 STATA
668
669 * -fdasave- Save and use datasets in FDA (SAS XPORT) format
670
671 * -usesas- 将 SAS 格式的数据导入 STATA
672
673 * -bugsdta- convert a Stata datafile into the S-plus format used in Winbugs
674
675 * -Stata2mplus- Convert Stata files to Mplus files
676
677 * -outdat- module to export data to other statistical packages
678 *           such as LIMDEP, RATS, and SPSS
679
680 * -dta2ras-, -ras2dta- ArcView/ArcInfo 与 stata 数据之间的相互转换
681
682 *-How do I convert among SAS, Stata and SPSS files?
683 * http://www.ats.ucla.edu/stat/stata/faq/convert_pkg.htm
684
685
686
687
688 *-----
689 *-> 1.4 存储和导出数据
690 *-----
691
692 *           ==本节目录==
693
694 *   1.4.1 存储数据
695 *   1.4.2 导出和转换
696 *       1.4.2.1 -outfile-命令: 导出为 .raw 文本格式
697 *       1.4.2.2 -outsheet-命令: 导出为 -Tab- 分隔的文本文件
698 *       1.4.2.3 -xmlsave-命令: 导出为 XML 格式
699 *       1.4.2.4 -dataout-命令: 导出为 Word,Excel,TeX
700 *       1.4.2.5 -outdat- 命令: 导出为 spss, rats, limdep 格式
701
702
703 *
704 *- 1.4.1 存储数据
705
706     shellout d3.txt
707     infile str30 v1 int v2 int v3 int v4 str10 v5 using d3.txt, clear
708     save d3.dta, replace
709
710 * 注意: 通常只有在初次导入数据时我们需要保存之,
711 *       此后的处理都在do-file中进行, 只需保存do-file即可。
712
713
714 *
715 *- 1.4.2 导出和转换(另存为其它格式)
716
717 *-1.4.2.1 -outfile-命令: 导出为 .raw 文本格式
718
719     sysuse auto, clear
720     outfile using myauto,replace
721                                     // 存为文本格式,空格分隔,80字符/行
722
723     dir myauto*
724     winexec notepad myauto.raw // 打开输出的文本文档, .raw格式
725
726 *-选项设定 [wide] 选项
727     outfile using myauto, wide replace
728                                     // 每个观察值一行, 没有80/行的限制
729
730     dir myauto*
731     winexec notepad myauto.raw
732
733 *-导出部分变量
734     outfile price-trunk foreign using myauto, wide replace
735     dir myauto*
736     winexec notepad myauto.raw
737
738
739 *-1.4.2.2 -outsheet-命令: 导出为 -Tab- 分隔的文本文件
740
741     sysuse auto, clear
742     keep in 1/10

```

```

741      outsheet price wei len using myauto, replace
742      dir myauto*
743      winexec notepad myauto.out
744
745
746      *-1.4.2.3 -xmlsave-命令: 导出为XML格式
747
748      sysuse auto, clear
749      xmlsave auto, doctype(excel) replace
750      shellout auto.xml
751
752
753      *-1.4.2.4 -dataout-命令: 导出为 Word,Excel,TeX
754
755      *-导出当前数据
756      sysuse auto, clear
757      dataout, save(dataout01) excel replace
758      dataout, save(dataout01) word replace
759
760      keep make price weight rep78 gear foreign
761      keep in 1/30
762      dataout, save(dataout01) tex replace
763
764      *-其它数据文件的转换
765      shellout d1.txt
766      dataout using d1.txt, excel save(d1_excel) replace
767
768
769
770      *-1.4.2.5 -outdat- 命令: 导出为 .spss, .rats, .limdep 格式
771
772      help outdat
773
774
775
776
777
778
779
780
781
782
783      *=====
784      *          计量分析与STATA应用
785      *=====
786
787      *          主讲人: 连玉君 博士
788
789      *          单 位: 中山大学岭南学院金融系
790      *          电 邮: arlionn@163.com
791      *          主 页: http://blog.cnfol.com/arlion
792
793      *          ::第一部分::
794      *          Stata 操作
795      *          =====
796      *          第一讲 STATA简介
797      *          =====
798      *          -1.5- 浏览资料
799
800
801      cd `c(sysdir_personal)'Net_course_A\A1_intro
802
803
804      *-----
805      *-> 1.5 浏览资料
806      *-----
807
808      *          ==本节目录==
809
810      *          1.5.1 变量的名称
811      *          1.5.2 查看资料的结构
812      *              1.5.2.1 更改变量的存储类型
813      *              1.5.2.2 -list- 命令的使用
814      *              1.5.2.3 定义变量的显示格式

```

---

```

815      *      1.5.2.4 数据和变量的标签
816      *      1.5.2.5 附加说明文字
817      *      1.5.2.6 搜索变量
818      *      1.5.3 基本统计量
819      *      1.5.3.1 -summarize- 命令
820      *      1.5.3.2 -codebook- 命令
821      *      1.5.3.3 -inspect- 命令
822      *      1.5.3.4 列表统计(table, tabulate)
823      *      1.5.3.5 论文格式的统计表格(tabstat)
824      *      1.5.3.6 将统计结果输出到txt文档中
825
826
827      *      =本节命令=
828      *      =====
829      *      sysuse, use, describe, compress, label, summarize
830      *      codebook, inspect, histogram, kdensity
831      *      help, search, findit, recast, format
832      *      =====
833
834
835      *
836      *-1.5.1 变量的名称
837
838      *-基本规则:
839      *      (1) 由英为字母、数字或 组成, 至多不超过 32 个;
840      *      (2) 首字母必须为 字母 或 ;
841      *      (3) 英文字母的大写和小写具有不同的含义;
842
843      *-示例:
844      *      abc_1 a1 _a2 Gdp 都是合理的变量名
845      *      5gdp 2invest 则不是;
846
847      *-特别注意:
848      *      建议不要使用 ` ` 作为变量的第一个字母,
849      *      因为许多stata的内部变量都是以 _ 开头的,
850      *      如, _n, _N, _cons, _b 等等。
851
852      help _variables
853
854
855      *
856      *-1.5.2 查看资料的结构
857
858      sysuse auto, clear
859      describe
860      describe, detail
861
862      * 另一个相似的命令
863      help ds
864      sysuse nlsw88.dta, clear
865      ds
866      ds, has(type int)
867
868      ds, not(type byte)
869      ret list
870      dis "`r(varlist)'" // 编程时, 可以利用此返回值
871      browse `r(varlist)'
872
873      ds, detail
874
875
876      *-1.5.2.1 更改变量的存储类型
877
878      sysuse auto, clear
879      list gear_ratio in 1/5
880      d gear_ratio
881      recast int gear_ratio, force
882      d gear_ratio
883      list gear_ratio in 1/5
884      compress // 自动精简资料的存储格式
885
886
887      *-1.5.2.2 -list- 命令的使用      -list-
888

```

---

```
889 list price, sep(10)
890 list price in 1/30, sep(0)
891 sort rep78
892 list make price rep78 in 1/20, sepby(rep78)
893 list price weight length, noobs
894 list price weight length, noobs clean
895
896
897 *-1.5.2.3 定义变量的显示格式 -format-
898
899 * 简介:
900 * str18 文字型变量, 每个观察值占据18个空格
901 * %-18s 靠左列印于屏幕上; 若 %18s, 则靠右列印;
902 * 若 %~18s, 则居中列印
903 * %8.0g 在 `8.0' 的原则下, 以尽量多的有效位数列出
904 * %6.2f 总共占6个空格, 小数位占两个空格
905
906 * 示例:
907 list price gear in 1/5
908 format price %6.1f
909 format gear %6.4f
910 list price gear in 1/5
911
912
913 *-1.5.2.4 数据和变量的标签 -label-
914
915 *-a 样本标签
916 sysuse auto, clear
917 des
918 label data "这是一份汽车价格资料"
919 des
920
921 *-b 变量的标签
922 label var price 汽车价格
923 label var foreign "汽车产地(1 国外; 2 国内)"
924 des
925
926 *-c 类别变量的文字标签(数字-文字对应表) -label define-
927 * label define 标签名
928 * label values 变量名 标签名 //将变量值和标签联系起来
929 browse
930 label define repair 1 "好" 2 "较好" 3 "中" 4 "较差" 5 "差"
931 label values rep78 repair
932 browse
933
934 *-d 标签的管理 -labelbook-
935 label dir
936 label list
937 label drop repair
938 label list
939 labelbook // 推荐使用
940 * 另一个例子
941 sysuse nls88, clear
942 labelbook
943
944
945 *-1.5.2.5 附加说明文字 -notes-
946
947 sysuse auto, clear
948 note: Wang:请确认-rep78-变量中缺漏值的原因
949 // 为整份数据加说明
950 notes
951 note weight: Su, 注意, 该变量与length高度共线性!
952 // 为单个变量加说明
953 notes
954
955
956 *-1.5.2.6 搜索变量 -lookfor-
957
958 use nlswork_simple.dta, clear
959 lookfor code
960 lookfor wage
961 lookfor married
962 lookfor code married
```

```
963
964     use d_lookfor.dta, clear // 对于大型数据非常方便
965     lookfor "固定资产"
966     lookfor "现金流量净额"
967     lookfor "借款"
968
969
970 *
971 *-1.5.3 基本统计量
972
973 *-1.5.3.1 -summarize- 命令
974
975     sysuse auto, clear
976     summarize
977     format price %6.2f
978     sum price, format
979     su price wei, detail
980
981
982 *-1.5.3.2 -codebook- 命令
983
984     codebook price weight
985
986     codebook rep78 // 当一个变量中的非重复值小于9个时,
987                   // Stata便会视此变量为类别变量, 并列表统计之
988
989
990 *-1.5.3.3 -inspect- 命令
991
992     inspect price weight length // 相对于 codebook 命令, 该命令还进一步绘制出直方图,
993                                // 以便对样本的分布有更直观的了解
994
995
996 *-1.5.3.4 列表统计 -table-, -tabulate-
997
998     sysuse auto, clear
999
1000     tabulate foreign
1001
1002     tab      rep78
1003
1004     table    rep78
1005
1006     tab      foreign rep78
1007
1008     table foreign rep78, c(mean price) f(%9.2f) center row col
1009
1010
1011
1012 *-1.5.3.5 论文格式的统计表格 -tabstat-
1013
1014     sysuse auto, clear
1015
1016     tabstat price weight length
1017
1018     tabstat price weight length, stats(mean p50 min max)
1019
1020     tabstat price weight length, stats(mean med min max) ///
1021                                col(s) format(%6.2f)
1022
1023     tabstat price weight length, s(mean p25 med p75 min max) ///
1024                                c(s) f(%6.2f)
1025
1026     tabstat price weight length, s(mean sd p25 med p75 min max) ///
1027                                c(s) f(%6.2f) by(foreign)
1028
1029
1030 *-1.5.3.6 将统计结果输出到txt文档中 -tabexport-
1031
1032     sysuse auto, clear
1033
1034     tabexport turn trunk length using results.txt, ///
1035             s(mean sd) replace
1036     shellout results.txt
```



```
1037
1038     tabexport turn trunk length using results.txt, ///
1039         s(mean sd) by(foreign) noreshape replace
1040
1041     tabexport turn trunk length using results.txt, ///
1042         s(count mean sd) by(foreign) replace format(%3.0f %9.2f)
1043
1044     *-说明: format() 选项与 s() 选项相对应
1045         type results.txt
1046         shellout results.txt
1047
1048
1049 *
1050 *-1.5.4 基本图形分析
1051
1052     *-1.5.4.1 直方图: 样本的总体分布情况
1053
1054     sysuse nlsw88.dta, clear
1055
1056     histogram wage
1057
1058     gen ln_wage = ln(wage)
1059     histogram ln_wage          // 对数转换后往往更符合正态分布
1060
1061     histogram hours, frequency // 纵坐标为对应的样本数, 而非比例
1062     histogram ttl_exp, normal  // 附加与该变量 N(u, s2) 参数值相同的正态分布图
1063
1064     histogram grade
1065     histogram grade, discrete // 离散变量的直方图必须附加 discrete 选项
1066
1067
1068
1069
1070     *-1.5.4.2 密度函数图
1071
1072     kdensity wage          // 它是直方图的平滑曲线
1073     kdensity ln_wage, normal
1074
1075
1076     *-1.5.4.3 散点图
1077
1078     sysuse auto, clear
1079     twoway scatter price wei
1080     scatter mpg turn
1081
1082
1083     *-1.5.4.4 相关系数矩阵
1084
1085     sysuse auto, clear
1086     graph matrix price wei len mpg
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097     *=====
1098     *          计量分析与STATA应用
1099     *=====
1100
1101     *          主讲人: 连玉君 博士
1102
1103     *          单   位: 中山大学岭南学院金融系
1104     *          电   邮: arlionn@163.com
1105     *          主   页: http://blog.cnfol.com/arlion
1106
1107     *          ::第一部分::
1108     *          Stata 操作
1109     *          =====
1110     *          第一讲 STATA简介
```

```

1111      *      =====
1112      *      -1.6-  执行命令
1113      *      -1.7-  修改资料
1114
1115
1116      cd `c(sysdir_personal)'Net_course_A\A1_intro
1117
1118
1119      *-----
1120      *-> 1.6   执行命令
1121      *-----
1122
1123      *      ==本节目录==
1124
1125      *      1.6.1 概览
1126      *      1.6.2 命令的适用范围
1127      *          1.6.2.1 列举多个变量
1128      *          1.6.2.2 样本范围的限制
1129      *      1.6.3 命令作用的增减：使用选项
1130
1131
1132      *
1133      *-----1.6.1 概览-----
1134
1135      * stata命令的通用格式：command varlist [if] [in] [ , options]
1136      * [if] [in] 用于限制样本范围
1137      * [options] “可选项”，增加了命令的弹性
1138
1139      help sum                // 解读帮助文件
1140
1141      sysuse nlsw88, clear
1142      sum wage hours ttl_exp if race==2, detail
1143      list wage grade race in 1/100, seby(race)
1144
1145      *-特别提醒：
1146      * (1) "[ ]" 为可选项，可以不填，但不在[]中的内容都必须填写
1147      * (2) 整个命令“裸露”的逗号只有一个，此前为命令主体，此后为选项
1148      *      虽然选项中可能有子选项，但子选项前的逗号并未“裸露”
1149      * 例如：
1150      sysuse sp500, clear
1151      twoway line close date, title("收盘价", place(left))
1152
1153      *
1154      *-----1.6.2 命令的适用范围-----
1155
1156      *-1.6.2.1 列举多个变量
1157
1158
1159      sum age race married never_married grade
1160      sum age-grade
1161      sum s*                // "*" 是孙悟空，可以表示`任何'长度的字母或数字
1162      sum ?a?e              // "?" 是猪八戒，只能替代`一个'长度的字母或数字
1163
1164      *-1.6.2.2 样本范围的限制
1165
1166      sum in 10/20           // 第10至第20个观察值之间的观察值
1167      sum wage in -5/-1     // 倒数...
1168      sum wage hours if race == 1 // 等于
1169      sum wage if race ~= 3  // 不等于
1170      sum wage if (race==2)&(married==1) // 且
1171      sum wage if (race==3)|(married==0) // 或
1172      sum wage if hours >= 40 // 大等于
1173
1174
1175      *
1176      *-----1.6.3 命令作用的增减：使用选项-----
1177
1178      sum wage , d
1179
1180
1181      *-说明：stata支持多数命令和选项的缩写，
1182      *      帮助文件中带下滑线的部分表示可以缩写的程度
1183
1184      sysuse sp500, clear

```

```

1185     replace volume = volume/1000
1186     #delimit ;
1187         twoway (rspike hi low date)
1188             (line close date)
1189             (bar volume date, barw(.25) yaxis(2))
1190             in 1/57
1191     , yscale(axis(1) r(900 1400))
1192     yscale(axis(2) r( 9 45))
1193     ylabel(, axis(2) grid)
1194     ytitle("股价 -- 最高, 最低, 收盘", place(top))
1195     ytitle("交易量 (百万股)", axis(2) bexpand just(left))
1196     xtitle(" ")
1197     legend(off)
1198     subtitle("S&P 500", margin(b+2.5))
1199     note("数据来源: 雅虎财经!");
1200     #delimit cr
1201
1202
1203
1204
1205
1206 *-----
1207 *-> 1.7 修改资料
1208 *-----
1209
1210 * 目的:
1211 * (1) 对现有变量进行修正和转换
1212 * (2) 产生新的变量
1213
1214 * ==本节目录==
1215
1216 * 1.7.1 数学表达式
1217 * 1.7.2 变量的创建和修改
1218 * 1.7.2.1 变量的存储类型
1219 * 1.7.2.2 创建新变量
1220 * 1.7.2.3 修改旧变量
1221 * 1.7.2.4 删除变量和样本值
1222 * 1.7.2.5 移动变量窗口中变量的位置
1223 * 1.7.2.6 克隆已有变量
1224 * 1.7.2.7 拆分变量
1225 * 1.7.3 样本值的排序
1226
1227
1228 * ==本节命令=
1229 * =====
1230 * gen, replace, drop, order, aorder, move, sort, gsort,
1231 * assert, count, compare, encode, decode, recode,
1232 * note, notes, notes drop, char, char list
1233 * =====
1234
1235 *
1236 *-----1.7.1 数学表达式-----
1237
1238 * 三类: 关系运算; 逻辑运算; 算术运算
1239
1240 * 关系运算符 ==; >; <; >; <; !=; ~=
1241 sysuse auto, clear
1242 list price if foreign == 0
1243 sum price if foreign != 1
1244
1245 * 逻辑运算符: & -->(与) ; | -->(或)
1246 sysuse auto, clear
1247 sum price wei if (foreign==1 & rep78<=3)
1248 sum price wei if (rep78==1) | (rep78==5) | (foreign !=0)
1249 sum price wei if (rep78>2 & rep78<5) | (price>10000)
1250
1251 * 算术运算符: + - * / ^(幂)
1252 display 5^2
1253 dis 1 - 3*2 + 4/5 - 9^3
1254 dis 2*_pi
1255
1256
1257 *
1258 *-----1.7.2 变量的创建和修改-----

```



```

1259
1260 *-1.7.2.1 变量的存储类型
1261
1262 *- 整数的存储类型
1263 * byte      字节型      (-100, +100)
1264 * int       一般整数型  (-32000, +32000)
1265 * long      长整数型    (-2.14*10^10, +2.14*10^10), 即, 正负21亿
1266
1267 *- 小数的存储类型
1268 * float     浮点型      8   位有效数字
1269 * double    双精度      16  位有效数字
1270
1271 *- 字符型变量
1272 * str#      如 str20 表示该变量最多包含 20 个字符
1273 *           每个汉字占两个字符
1274 sysuse auto, clear
1275 des
1276 gen x = "中国"      // 一个汉字占两个字符
1277 des x
1278
1279
1280 *-1.7.2.2 创建新变量      -generate-
1281
1282 *-基本方式
1283 sysuse auto, clear
1284
1285 generate price2 = price^2      // 可简写为 gen
1286 gen price2f = price^2 if foreign==1
1287 gen wlratio = weight/length
1288
1289
1290 *-数学函数转换
1291
1292 help math functions
1293
1294 sysuse nlsw88.dta, clear
1295
1296 gen ln_wage = ln(wage)      // 取对数
1297 gen sqrt_hours = sqrt(hours) // 开根号
1298
1299 gen int_wage = int(wage)      // 取整
1300 gen floor_wage = floor(wage) // 等价于取整
1301 gen ceil_wage = ceil(wage)   // 取整数上限
1302
1303 list *wage in 1/5
1304
1305
1306 *-1.7.2.3 修改旧变量      -rename-, -renvars-, -replace-
1307
1308 *-单个变量重命名      -rename-
1309 rename displacement disp
1310
1311 *-批量修改变量名称 -renvars-
1312 help renvars
1313 sysuse auto, clear
1314 renvars price weight length / p wei len
1315 renvars p-wei, postfix(_new) // 批量增加后缀
1316 renvars mpg , prefix(old_)   // 批量增加前缀
1317
1318
1319 *-修改观察值      -replace-
1320
1321 sysuse auto, clear
1322 replace price = 10000 if (price>10000)
1323 gen byte bad = 0      // 事先指明变量类型是个不错的习惯
1324 replace bad = 1 if (rep78>3)
1325 list rep78 bad
1326
1327 *-更为合理的定义方式
1328 replace bad=. if (rep78==.)
1329 list rep78 bad
1330
1331 *-文字变量观察值的修改
1332 des make

```

```

1333         list make in 50/59
1334         replace make="宝马 320i" if (make=="BMW 320i") //要加双引号!
1335         list make in 50/59
1336
1337
1338 *-1.7.2.4 删除变量和样本值 -drop-
1339
1340 *- Stata官方命令 -drop-
1341     drop price2                // 删除一个变量
1342     drop wlratio-bad2          // 删除一组变量
1343     list price in 1/5
1344     drop in 1/3                // 删除指定区间的观察值
1345     drop if (rep78==.)         // 删除满足特定条件的观察值
1346     list price in 1/5
1347     drop _all                  // 删除内存中的所有变量
1348
1349
1350 *- 一些有用的外部命令 -cap drop-; -dropvars-; -safedrop-
1351
1352 *-cap drop-
1353     help capture
1354
1355     capture drop price2
1356     gen price2 = price^2
1357     cap drop prcie wlratio bad2 // 能否删掉这三个变量?
1358     gen wlratio = weight/length
1359
1360 *-dropvars-
1361     dropvars price2 wlratio bad2 // 等价于如下三条命令
1362     * cap drop price2
1363     * cap drop wlratio
1364     * cap drop bad2
1365     gen wlratio = wei/len
1366
1367 *-safedrop-
1368     sysuse auto, clear
1369     drop forei
1370     sysuse auto, clear
1371     safedrop forei
1372     safedrop foreign gear_ratio
1373
1374
1375 *-1.7.2.5 移动变量窗口中变量的位置 -order- -aorder- -move-
1376
1377     sysuse auto, clear
1378
1379     order price weight length foreign
1380
1381     order trunk, before(weight) // 把trunk移到weight之前
1382     sysuse auto, clear
1383     move trunk weight           // 功能同上,stata11以前版本适用
1384
1385     order _all, alpha           // 按字母对变量排序
1386     aorder                      // 功能同上,stata11以前版本适用
1387
1388
1389 *-1.7.2.6 克隆已有变量 -clonevar-
1390
1391 * 把已有变量的标签, 数字-文字对应表等所有内容都复制过去
1392     help clonevar
1393
1394     sysuse auto, clear
1395
1396     clonevar foreign_c = foreign
1397
1398     gen foreign_g = foreign
1399     sort mpg
1400     list foreign* in 1/10
1401     browse
1402
1403
1404 *-1.7.2.7 拆分变量 -separate-
1405
1406     sysuse auto, clear

```

```

1407         separate mpg, by(foreign)
1408
1409         * 等价方式: 但没有变量标签
1410         gen mpg_f = mpg if (foreign==1)
1411         gen mpg_d = mpg if (foreign==0)
1412
1413         browse
1414
1415
1416     *
1417     *-1.7.3 样本值的排序      -sort-      -gsort-
1418
1419         sysuse nlsw88.dta, clear
1420
1421         sort wage                      // 默认为升序排列
1422         list wage in 1/10
1423         dis "max = " wage[_N]
1424         sum wage
1425         gen nag_wage = -wage
1426         sort nag_wage                  // 降序排列
1427
1428         gsort -wage                    // 降序排列
1429         list wage in 1/10
1430
1431         gsort wage, gen(numb)          // 产生排序编号
1432         list numb wage in 1/10
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443         *=====
1444         *      计量分析与STATA应用
1445         *=====
1446
1447         *      主讲人: 连玉君 博士
1448
1449         *      单   位: 中山大学岭南学院金融系
1450         *      电   邮: arlionn@163.com
1451         *      主   页: http://blog.cnfol.com/arlion
1452
1453         *      ::第一部分::
1454         *      Stata 操作
1455         *      =====
1456         *      第一讲  STATA简介
1457         *      =====
1458         *      -1.8- log 文件
1459
1460         cd `c(sysdir_personal)'Net_course_A\A1_intro
1461
1462
1463     *-----
1464     *-> 1.8 log 文件: 记录你的分析过程
1465     *-----
1466
1467         *      ==本节目录==
1468
1469         *      1.8.1 log 文件简介
1470         *      1.8.2 将 log 文件转换为网页
1471         *      1.8.2.1 -log2html- 命令: 制作"单页"网页
1472         *      1.8.2.2 -hyperlog- 命令: 制作"框架型"网页
1473         *      1.8.2.3 其他命令
1474
1475     *
1476     *-1.8.1 log 文件简介
1477
1478         * 记录你的分析过程: log 文件
1479
1480         help log

```

```
1481
1482     *- 示例 1:
1483         doedit L1_intro_log_cs.do
1484         dir *.log
1485         shellout paper01.log
1486
1487     *- 示例 2:
1488     *
1489     *-----记录开始-----
1490     *
1491         cd D:\stata11\ado\personal\Net_course_A\A1_intro
1492         sysuse auto, clear
1493
1494         log using mylog1.log, text replace // _mylog1.log_-begin-__
1495             dis "Part I: 统计分析"
1496             sum price weight length
1497         log close                               // _mylog1.log_-over-__
1498
1499             tab rep78 // 这些分析不计入 log 文件
1500             des, detail
1501
1502         log using mylog2.log, text replace // _mylog2.log_-begin-__
1503             tab rep78 foreign
1504             des price rep78 foreign, d
1505         log close                               // _mylog2.log_-over-__
1506     *
1507     *-----记录结束-----
1508
1509     shellout mylog1.log
1510
1511     shellout mylog2.log
1512
1513
1514
1515     *
1516     *-1.8.2 将 log 文件转换为网页
1517
1518     *- -log2html-, -hyperlog-, -autolog-, -logout-, -slog-
1519
1520     *-1.8.2.1 -log2html- 命令: 制作“单页”网页
1521
1522     help log2html
1523
1524     *-示例:
1525         cap log close
1526         log using mylog, replace
1527             sysuse nlsw88, clear
1528             desc
1529             summ
1530             regress wage hours ttl_exp
1531         log close
1532
1533     *-转换为网页
1534         log2html mylog, replace                // 转换 log --> 网页
1535         shellout mylog.html                    // 打开网页
1536         * 你也可以到当前活动目录下打开 mylog.html 文件
1537
1538     *-附加网页标题
1539         log2html mylog, replace title("美国妇女工资影响因素研究")
1540         shellout mylog.html
1541         * 注意:
1542             * 为了能够正确显示中文字符, 请在打开网页后依次点击:
1543             * "查看(V)"-->"编码(D)"-->"简体中文(GB2312)"
1544
1545     *-设定网页风格
1546         log2html mylog, replace input(ff3300) result(003333) bg(grey)
1547         shellout mylog.html
1548
1549
1550     *-1.8.2.2 -hyperlog- 命令: 制作“框架型”网页
1551
1552     help hyperlog
1553
1554     doedit mylog.do
```

---



```

1555      do mylog.do                                // 生成 log 文件
1556
1557      hyperlog mylog.do mylog01.log, replace // 转换为网页
1558
1559      shellout mylog_hlog.html                    // 注意文件名的变化
1560
1561
1562      *-1.8.2.3 其他命令
1563
1564      * -slog-      生成可嵌套的 log 文件，适于程序调试和大型 log 文件的书写
1565      * -logout-    将stata命令结果输出至Word, Excel, TeX中，随后介绍
1566      * -autolog-   更为快捷定义 log 文件，用于定义 profile.do 启动文件
1567      *             我自己定义的 profile.do 文件中，
1568      *             已经涵盖了这个功能，故不再介绍
1569
1570
1571
1572
1573
1574
1575
1576      *=====
1577      *      计量分析与STATA应用
1578      *=====
1579
1580      *      主讲人：连玉君 博士
1581
1582      *      单   位：中山大学岭南学院金融系
1583      *      电   邮：arlionn@163.com
1584      *      主   页：http://blog.cnfol.com/arlion
1585
1586      *      ::第一部分::
1587      *      Stata 操作
1588      *      =====
1589      *      第一讲 STATA简介
1590      *      =====
1591      *      -1.9- do 文档
1592
1593      cd `c(sysdir_personal)'Net_course_A\A1_intro
1594
1595
1596      *-----
1597      *-> 1.9 do 文档：高效快捷地执行命令
1598      *-----
1599
1600      *      ==本节目录==
1601
1602      *      1.9.1 do 文档简介
1603      *          1.9.1.1 打开 do 文档编辑器
1604      *          1.9.1.2 保存和关闭
1605      *          1.9.1.3 执行 do 文档
1606      *      1.9.2 合理规划你的do文档
1607      *          1.9.2.1 一些基本规则
1608      *          1.9.2.2 注释语句
1609      *          1.9.2.3 断行
1610      *          1.9.2.4 大型 do 文档的设定
1611      *      1.9.3 列印文字
1612      *          1.9.3.1 -display-命令
1613      *          1.9.3.2 列印的颜色
1614      *          1.9.3.3 列印的位置
1615      *      1.9.4 关于编辑器
1616      *      1.9.5 do 文件的转换(制作网页教程)
1617
1618
1619
1620      *
1621      *-1.9.1 do 文档简介
1622
1623
1624      *-> ==概览==
1625
1626      *- do 文档实际上是Stata命令的集合，方便我们一次性执行多条stata命令；
1627
1628      *- do 文档的使用使我们的分析工作具有可重复性；

```

```

1629
1630     *- 在一篇文章的实证分析过程中，我们通常将数据的分析工作写在 do 文档中
1631
1632
1633     *-1.9.1.1 打开 do 文档编辑器
1634
1635         *- 方法 1:
1636             doedit                // 打开 do-editor
1637             doedit mylog.do        // 打开一个已存在的 do 文档，可指定完整路径
1638
1639         *- 方法 2:
1640         * 点击Results窗口上方倒数第六个按钮
1641
1642         *- 设置属性:
1643             * Edit --> Preferences
1644             * 建议选中 [Auto-indent] 和 [Save before do/run]
1645
1646
1647     *-1.9.1.2 保存和关闭
1648
1649
1650     *-1.9.1.3 执行 do 文档
1651
1652         *-Case1: 执行一部分命令
1653             * 选中需要执行的命令，点击doedit窗口中第二行倒数第一个图标。
1654             * 【快捷键】: Ctrl+D
1655
1656             doedit L1_intro_do.do
1657
1658         *-Case2: 整体执行
1659             do L1_intro_do.do
1660
1661
1662
1663     *
1664     *-1.9.2 合理规划你的do文档
1665
1666     *-1.9.2.1 一些基本规则
1667
1668         *-A. 提高 do 文档的可读性
1669         *
1670         * gen z = z + y      is better than    gen z=z+y
1671         *
1672         * gen z = x^2        is better than    gen z = x ^ 2
1673         *
1674         * gen t = hours + minutes/60 + seconds/3600
1675         * is better than
1676         * gen t = hours + minutes / 60 + seconds / 3600
1677         *
1678         * list price if (foreign==1) & (rep78>3)
1679         * is better than
1680         * list price if foreign==1&rep78>3
1681
1682         *-B. 断句和断行
1683         *
1684         * 每一行的语句不要过长，不用拖动下方导引条即可阅读；
1685         * 各段代码采用一个或多个空行加以分隔；
1686
1687
1688     *-1.9.2.2 注释语句
1689
1690         help comments
1691
1692         *-示例:
1693             * 第一种注释方式
1694             sum price weight      /* 第二种注释方式 */
1695             gen x = 5              // 第三种注释方式
1696
1697
1698     *-1.9.2.3 断行
1699
1700         *-三种方式:  "///" 、  "/* */" 、  #delimit 命令
1701
1702         *-第一种断行方式:  ///

```

```

1703         sysuse auto, clear
1704         twoway (scatter price weight)          ///
1705                (lfit price weight),           ///
1706                title("散点图和线性拟合图")
1707
1708     *-第二种断行方式: /* */
1709         twoway (scatter price weight)          /*
1710                */ (lfit price weight),         /*
1711                */ title("散点图和线性拟合图")
1712
1713     *-第三种断行方式: #delimit 命令
1714         #delimit ;
1715         twoway (scatter price wei)
1716                (lfit price wei),
1717                title("散点图和线性拟合图");
1718         #delimit cr
1719
1720     *-另一种习惯:
1721         sysuse auto, clear
1722         #delimit ;
1723         des price wei; sum price wei len; reg price wei;
1724         #delimit cr
1725
1726
1727     *-1.9.2.4 大型 do 文档的设定
1728
1729     * 设定一个主文件, 下设 N 个子文件, 分别处理某一部分分析工作
1730     * 保存在同一个文件夹下
1731
1732         doedit L1_main.do
1733
1734
1735     *
1736     *-1.9.3 列印文字
1737
1738     *-1.9.3.1 -display-命令
1739
1740         dis 3 + 5*7 + sqrt(20)
1741
1742         dis in g sin(_pi*0.5) + cos(0.9)
1743
1744         dis _n(2) _dup(3) "I Love This GAME! "
1745
1746     * 将文字置于 " " 或 `"' 之间
1747         display "This is a pretty girl!"
1748         dis      `"This is a "pretty" girl!"'
1749
1750     *-1.9.3.2 列印的颜色
1751
1752     * 颜色1: red green yellow white
1753         dis in green "I like stata!"
1754         dis in w      "This " in y "is " in g "a " in red "pretty" in g " girl"
1755
1756     * 颜色2: as text(绿色)| as result(黄色)| as error(红色)| as input(白色)
1757         dis as result "Stata is Good !"
1758
1759
1760     *-1.9.3.3 列印的位置
1761
1762     * -----
1763     * 副命令 | 定义
1764     * -----
1765     * _col(#) | 从第 # 格开始列印
1766     * _s(#)   | 跳过 # 格开始列印
1767     * _n(#)   | 从第 # 行开始列印
1768     * _c      | 下次列印接着列印而无须从起一行
1769     * _dup(#) | 重复列印 # 次
1770     * -----
1771
1772         display "Stata is good"
1773         display _col(12) "Stata is good"
1774         display "Stata is good" _s(8) "I like Stata"
1775         display _dup(3) "Stata is good! "
1776         display "Stata is good","I like it"

```

```
1777      display "Stata is good",, "I like it"
1778      display _n(3) "Stata is good"
1779
1780      * 更精美的列印方式
1781      help smcl           // 我们在高级部分会对此作详细介绍
1782
1783      * -display-的一个妙用: 清屏
1784      display _newline(100)
1785
1786
1787      *
1788      *-----1.9.4 关于编辑器-----
1789
1790      * 如下文档详细介绍了如何把外部编辑器与stata联系起来
1791      * http://fmwww.bc.edu/repec/bocode/t/textEditors.html#disclaim
1792      * stata11
1793      * 高亮功能(与LaTeX相仿)
1794
1795
1796      *
1797      *-----1.9.5 do 文件的转换(制作网页教程)-----
1798
1799      * -do2htm- 优点在于可以自动插入图片
1800
1801      doedit L1_do2htm_test.do           // 无需执行
1802      do2htm L1_do2htm_test, replace
1803      // 将 do 文件及其 log 结果转换为 html 网页
1804
1805      * 打开网页
1806      dir *.htm
1807      shellout L1_do2htm_test.htm
1808      * 注意:
1809      * 为了能够正确显示中文字符, 请在打开网页后依次点击:
1810      * "查看(V) "-->"编码(D) "-->"简体中文(GB2312)"
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826      *=====
1827      *      计量分析与STATA应用
1828      *=====
1829
1830      *      主讲人: 连玉君 博士
1831
1832      *      单 位: 中山大学岭南学院金融系
1833      *      电 邮: arlionn@163.com
1834      *      主 页: http://blog.cnfol.com/arlion
1835
1836      *      ::第一部分::
1837      *      Stata 操作
1838      *      =====
1839      *      第一讲 STATA简介
1840      *      =====
1841      *      1.10 stata与Excel、Word、LaTeX的亲密接触
1842
1843
1844      cd `c(sysdir_personal)'Net_course_A\A1_intro
1845
1846
1847      *-----
1848      *-> 1.10 stata与Excel、Word、LaTeX的亲密接触
1849      *-----
1850
```

```

1851      *      ==本节目录==
1852
1853      *      1.10.1 统计表格、矩阵的输出
1854      *          1.10.1.1 输出基本统计量
1855      *          1.10.1.2 输出相关系数矩阵
1856      *          1.10.1.3 输出矩阵
1857      *          1.10.1.4 其它说明
1858      *      1.10.2 估计结果的输出
1859      *          1.10.2.1 esttab : 回归结果的呈现
1860      *          1.10.2.2 logout : 输出 【Excel、Word、TeX文档】
1861      *          1.10.2.3 xml_tab: 专业输出 【Excel 文档】
1862      *          1.10.2.4 outreg2: 专业输出 【Word、Excel文档】
1863
1864
1865
1866      *
1867      *-1.10.1 统计表格、矩阵的输出 -logout-
1868
1869      *-基本设定
1870      *      logout, save(filename) word(excel,tex) [options]: ///
1871      *          输出统计表格或列示矩阵的命令
1872
1873
1874      *-1.10.1.1 输出基本统计量
1875
1876          sysuse auto, clear
1877          tabstat price wei len mpg rep78, ///
1878          stats(mean sd min p50 max) c(s) f(%6.2f)
1879
1880      *- 【Word】 文档
1881      logout, save(mytable) word replace: ///
1882          tabstat price wei len mpg rep78, ///
1883          stats(mean sd min p50 max) c(s) f(%6.2f)
1884
1885      *- 【Excel】 文档
1886      logout, save(mytable) excel replace: ///
1887          tabstat price wei len mpg rep78, ///
1888          stats(mean sd min p50 max) c(s) f(%6.2f)
1889
1890
1891      *-1.10.1.2 输出相关系数矩阵
1892
1893      logout, save(mytable) word replace: ///
1894          pwcorr price wei len mpg rep78
1895      logout, save(mytable) word replace: ///
1896          pwcorr_a price wei len mpg rep78
1897      *-说明: -pwcorr_a-命令由 Arlion 编写
1898
1899
1900      *-1.10.1.3 输出矩阵
1901
1902      mat a = I(10)
1903      mat list a
1904      logout, save(mytable) word replace: ///
1905          mat list a, nohalf
1906
1907
1908      *-1.10.1.4 其它说明
1909
1910      *- -logout- 偶尔会有点小问题(空格)
1911          sysuse nlsw88, clear
1912          logout, save(mytable) word replace: tab occup
1913
1914      *- 其他命令
1915      *      tabout 比较灵活, 但输出后的word文档为-tab-分隔,
1916      *          尚需使用表格自动套用功能
1917      *      tabexport, mkcorr, tabform, tablemat, tabone
1918      *          都不是很好用
1919
1920
1921
1922      *
1923      *-1.10.2 估计结果的输出
1924

```

```

1925 * -esttab-, -logout-, -xml_tab-, -outreg2-
1926
1927
1928 *-1.10.2.1 -esttab- 命令：回归结果的呈现
1929
1930 sysuse auto, clear
1931 reg price wei
1932 est store m1
1933 reg price wei len
1934 est store m2
1935 reg price wei len mpg foreign
1936 est store m3
1937
1938 *-基本用法
1939 esttab m1 m2 m3
1940
1941 *-修改显著水平，紧凑的方式呈现结果
1942 esttab m1 m2 m3, ar2 compress nogap ///
1943 star(* 0.1 ** 0.05 *** 0.01)
1944
1945 *-呈现 p-value，置于 "[]" 中
1946 esttab m1 m2 m3, ar2 compress nogap ///
1947 star(* 0.1 ** 0.05 *** 0.01) ///
1948 b(%6.3f) brackets p
1949
1950 *-呈现标准化系数
1951 esttab m1 m2 m3, beta
1952
1953 *-显示变量的标签，而非变量名
1954 label var weight "汽车重量"
1955 esttab m1 m2 m3, label
1956
1957 *-呈现弹性系数
1958 esttab m1 m2 m3, margin // 默认情况下，略去 Constant
1959 esttab m1 m2 m3, margin constant
1960
1961 *-输出文件的其它格式
1962 esttab m1 m2 m3 using myout.html, replace // 网页
1963
1964 esttab m1 m2 m3 using myout.tex, replace // TeX 文档
1965 * 这个文档可以直接插入 TeX 中，采用 \input{}
1966 shellout mypdf.tex // 一个模板
1967
1968 * 其它输出类型：smcl, fixed, tab, csv, scsv,
1969 * rtf, html, tex, and booktabs
1970
1971
1972 *-输出至 Excel
1973 esttab m1 m2 m3 using myout.csv, replace
1974 esttab m1 m2 m3 using myout.csv, replace ///
1975 compress nogap nonotes ///
1976 addnotes("*** 1% ** 5% * 10%" "" "")
1977
1978 * 说明：
1979 * (1) -esttab- 在输出Excel文档时，标注的限制水平不好看，故修改之
1980 * (2) -addnotes()- 选项中的后两个 "" 是空两行的意思，便于后续追加
1981
1982 * 在已有文件的基础上追加新结果
1982 reg price wei, robust
1983 est store rob01
1984 reg price wei len, robust
1985 est store rob02
1986 reg price wei len mpg foreign, robust
1987 est store rob03
1988
1989 esttab rob01 rob02 rob03 using myout.csv, append ///
1990 compress nogap b(%6.3f) scalars(r2_a N F) ///
1991 star(* 0.1 ** 0.05 *** 0.01) obslast ///
1992 title(Robust check of the main results) ///
1993 addnotes("The White(1980) robust regression" "" "")
1994
1995 * 说明：
1996 * (1) 如果你的研究分成多个部分，你可以依次追加；
1997 * (2) 输出后的结果从Excel中粘贴到Word，仅需简单调整即可
1998 * (3) using file.csv 可以指定文件存储的具体路径

```

```

1999
2000
2001
2002 *-1.10.2.2 -logout- 命令: 输出 【Excel、Word、TeX文档】
2003
2004 *-基本设定
2005 * logout, save(filename) word(excel,tex) [options]: ///
2006 * esttab .....
2007
2008 *-示例
2009 sysuse auto, clear
2010
2011 * Excel 文档
2012 logout, save(myreg) excel dec(3) replace: ///
2013 reg price weight mpg rep78 foreign
2014
2015 * Word 文档
2016 logout, save(myreg) word dec(3) replace: ///
2017 reg price weight mpg rep78 foreign
2018
2019 * _____ 一个完整的例子 _____
2020 *
2021 *-Step1: 估计模型并存储结果
2022 sysuse auto, clear
2023 reg price wei
2024 est store m1
2025 reg price wei len
2026 est store m2
2027 reg price wei len mpg foreign
2028 est store m3
2029
2030 *-Step2: logout—结果直接输出到Word文档中
2031 logout, save(mylogout) word replace fix(3): /// //注意冒号
2032 esttab m1 m2 m3, mtitle(模型1 模型2 模型3) ///
2033 b(%6.3f) se(%6.2f) /// //系数、标准误
2034 star(* 0.1 ** 0.05 *** 0.01) /// //显著水平的标注
2035 scalar(r2 r2_a N F) compress nogap
2036
2037 *
2038
2039 * 说明:
2040 * (1) -fix(#)- 选项决定了转换的敏感度, 本例中, fix(3)效果最佳
2041 * (2) 更改 -word- 选项, 可以输出到 Excel(-excel-) 或 LaTeX 中(-tex-)
2042 * (3) 优势: 输出的-Word-文档比较美观
2043 * (4) 缺陷: 无法追加新的结果,
2044 * 需要多个结果分别存储到不同的-Word-文件中。
2045
2046 *-例: 输出 TeX 文档□
2047 logout, save(mylogout) tex replace fix(3): ///
2048 esttab m1 m2 m3, mtitle(模型1 模型2 模型3) ///
2049 b(%6.3f) se(%6.2f) ///
2050 star(* 0.1 ** 0.05 *** 0.01) ///
2051 scalar(r2 r2_a N F) compress nogap
2052
2053 *-1.10.2.3 -xml_tab- 命令: 专业输出 【Excel 文档】
2054
2055 sysuse nlsw88, clear
2056 reg wage hours married
2057 est store m1
2058 reg wage hours married ttl_exp south
2059 est store m2
2060 xi:reg wage hours married ttl_exp south i.race
2061 est store m3
2062 xi:reg wage hours married ttl_exp south i.race i.occupation
2063 est store m4
2064
2065 *-基本设定
2066 xml_tab m1 m2 m3 m4, replace
2067
2068 * 说明:
2069 * (1) 默认存储于当前活动目录下, 名称为 stata_out.xml;
2070 * (2) 默认显示变量标签, 而非变量名称, 变量标签不支持中文

```



```

2071
2072     *-稍作美化
2073     xml_tab m1 m2 m3 m4, save(result) sheet(OLS) replace ///
2074     tstat below stats(r2 r2_a N)
2075
2076
2077     *-进一步美化
2078     xml_tab m1 m2 m3 m4, save(result) sheet(OLS) replace ///
2079     tstat below stats(r2 r2_a N) ///
2080     drop(_Ioccup*) font("Times New Roman" 10) ///
2081     title(Table 1 Basic Regression of US women wage) ///
2082     tblank(1) format(NCCR3) ///
2083     note("Occupation dummies are not presented")
2084
2085     * 说明:
2086     * (1) 若部分变量有中文标签, 需要事先修改, 或附加 -nolabel- 选项;
2087     * (2) 有关 -format()- 选项的填写, 请参阅帮助文件;
2088     * (3) save() 选项中可填写具体的存储路径
2089     * (4) 优势: 可以用一个-Excel-文件存储多个-sheet-
2090
2091
2092     *-输出结果的追加
2093     * 分析妇女是否加入工会的影响因素
2094     logit union wage ttl_exp
2095     est store a1
2096     xi: logit union wage ttl_exp i.race i.occupation
2097     est store a2
2098     xml_tab a1 a2, save(result) sheet(Logit) append /// //注意此处的变化
2099     tstat below stats(r2 r2_a N) ///
2100     drop(_Ioccup*) font("Times New Roman" 10) ///
2101     title(Table 2 Determinants of being a Union member) ///
2102     tblank(1) format(NCCR3) ///
2103     note("Occupation dummies are controlled, but not presented")
2104
2105     * 说明:
2106     * (1) 不同类别或不同阶段的回归结果, 可以分别放入不同的 sheet () 中;
2107     * (2) 除第一个 sheet 使用 -replace- 选项外,
2108     * 后续追加的 sheet 使用 -append- 选项
2109     * (3) 上述结果稍作整理即可贴入-Word-,
2110     * 建议使用-Word-表格自动调整功能
2111
2112
2113
2114     *-1.10.2.4 -outreg2- 命令: 专业输出【Word、Excel文档】
2115
2116     sysuse nls88, clear
2117     tab race, gen(d_race)
2118     drop d_race1
2119     tab occu, gen(d_occu)
2120     drop d_occu1
2121     reg wage hours ttl_exp married
2122     est store m1
2123     reg wage hours ttl_exp married d_race*
2124     est store m2
2125     reg wage hours ttl_exp married d_race* d_occu*
2126     est store m3
2127
2128     *-基本用法: 在数据窗口中呈现结果
2129     outreg2 [m1 m2 m3] using tab01, seeout replace
2130
2131     *-输出 Word 或 Excel 文档
2132     outreg2 [m1 m2 m3] using tab01, word replace
2133     outreg2 [m1 m2 m3] using tab01, excel replace
2134
2135     *-同时输出Word和Excel文档(亦可增加 tex 选项, 输出 tex 文档)
2136     outreg2 [m1 m2 m3] using tab01, word excel replace
2137
2138     *-使用变量标签
2139     label var hours "每周工作时数"
2140     label var married "已婚==1, 未婚==0"
2141     outreg2 [m1 m2 m3] using tab01, word replace label
2142     outreg2 [m1 m2 m3] using tab01, word replace label(insert)
2143     // 同时呈现变量和标签
2144

```

```

2145      *-s.e., t值, 与 p值
2146      outreg2 [m1 m2 m3] using tab01, word replace tstat
2147              // 呈现 t-value
2148      outreg2 [m1 m2 m3] using tab01, word replace pvalue
2149              // 呈现 p-value
2150
2151      *-小数的显示方式 -tdec()-, -rdec()- 选项
2152      outreg2 [m1 m2 m3] using tab01, word replace tstat tdec(2)
2153              // t-value小数点后两位
2154      outreg2 [m1 m2 m3] using tab01, word replace tstat rdec(3)
2155              // R2小数点后三位
2156
2157      *- "()", "[]", 与 " "
2158      outreg2 [m1 m2 m3] using tab01, word replace pvalue bracket tdec(3)
2159      outreg2 [m1 m2 m3] using tab01, word replace tstat tdec(2) noparen
2160
2161      *-新结果的追加
2162      logit union wage married wage d_race* d_occu*
2163      est store logit
2164      outreg2 [logit] using tab01, word append
2165
2166      *-弹性系数、标准化系数和边际效果
2167      reg wage hours ttl_exp married
2168      mfx, eyex // 计算弹性系数
2169      outreg2 using tab02_mfx, word replace // -replace- 新建word文档
2170
2171      reg wage hours ttl_exp married d_race*
2172      mfx, eyex
2173      outreg2 using tab02_mfx, word append // 追加结果
2174
2175      reg wage hours ttl_exp married d_race* d_occu*
2176      mfx, eyex
2177      outreg2 using tab02_mfx, word append // 进一步追加结果
2178
2179      *-有选择地呈现变量
2180      outreg2 [m1 m2 m3] using tab01, word replace ///
2181              drop(d_occu*)
2182      * 说明:
2183      *   (1) 亦可使用 keep() 选项筛选需要呈现的变量;
2184      *   (2) 使用 order() 选项可以改变变量的先后顺序
2185
2186      *-表格的标题
2187      outreg2 [m1 m2 m3] using tab01, word replace ///
2188              title("表1: 美国妇女工资决定因素估计结果")
2189
2190      *-最后一行的统计量: adj-R2, F值
2191      outreg2 [m1 m2 m3] using tab01, word replace ///
2192              title("表1: 美国妇女工资决定因素估计结果") ///
2193              drop(d_occu*) ///
2194              adjr2 e(F ll)
2195
2196      *-重新定义注释
2197      outreg2 [m1 m2 m3] using tab01, word replace ///
2198              title("表1: 美国妇女工资决定因素估计结果") ///
2199              drop(d_occu*) nonote ///
2200              addnote("注: (1)***, **, *分别表示在1%, 5%和10%水平上显著;", ///
2201                      "(2) 括号中为标准误;", ///
2202                      "(3) m3中控制了职业虚拟变量 d_occu2-d_occu13。")
2203      * 说明:
2204      *
2205      *   (1) -nonote- 选项:
2206      *       不显示原有英文注释 "Standard errors in parentheses"
2207      *       和 "*** p<0.01, ** p<0.05, * p<0.1"
2208      *
2209      *   (2) -addnote- 选项: addnote("注释1", "注释2", "注释3")
2210
2211
2212
2213      * __-<-<-<-<- 【一个模板】 ->->->->-__
2214      *
2215      *- 特征:
2216      *   (T1) 附加表格标题;
2217      *   (T2) 调整变量的显示顺序和多寡 -drop()-, -sortvar()-
2218      *   (T3) t-value 小数点后显示两位; adj-R2 小数点后显示三位;

```

```

2219      *      (T4) 修改表格注释;
2220      *
2221      *      -----outreg2模板-----
2222      outreg2 [m1 m2 m3] using tab01, word replace ///
2223      title("表1: 美国妇女工资决定因素估计结果") /// // (T1)
2224      drop(d_occu*) sortvar(married hours) /// // (T2)
2225      tdec(2) rdec(3) adjr2 e(F) /// // (T3)
2226      nonote /// // (T4)
2227      addnote("注: (1)***, **, *分别表示在1%, 5%和10%水平上显著;", ///
2228              "(2) 括号中为标准误;", ///
2229              "(3) 其它注释语句。")
2230
2231
2232
2233      *-多方程模型结果的呈现
2234      *-示例1: SUR模型
2235      use invest2.dta, clear
2236      sureg (invest1 market1 stock1) ///
2237      (invest2 market2 stock2) ///
2238      (invest3 market3 stock3) ///
2239      (invest4 market4 stock4) ///
2240      (invest5 market5 stock5), corr
2241      outreg2 using table2, word replace
2242      // 单个模型的呈现, 无需est store
2243      outreg2 using table2, word replace long // 长条形显示结果
2244
2245      *-示例2: Multinomial Logit 模型 -mlogit-
2246      use fullauto, clear
2247      replace wei = wei/1000
2248      replace price = price/1000
2249      mlogit rep77 mpg wei price rseat foreign
2250      outreg2 using table2, word replace
2251
2252
2253      *- 评述:
2254
2255      *      (1) 整体而言, -outreg2- 命令最为好用,
2256      *      可以同时实现对 Word, Excel, LaTeX 的支持
2257
2258      *      (2) -esttab-, -xml_tab- 用起来也比较方便
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271      *=====
2272      *      计量分析与STATA应用
2273      *=====
2274
2275      *      主讲人: 连玉君 博士
2276
2277      *      单 位: 中山大学岭南学院金融系
2278      *      电 邮: arlionn@163.com
2279      *      主 页: http://blog.cnfol.com/arlion
2280
2281      *      ::第一部分::
2282      *      Stata 操作
2283      *      =====
2284      *      第一讲 STATA简介
2285      *      =====
2286      *      -1.11- Stata 设定
2287
2288      cd `c(sysdir_personal)'Net_course_A\A1_intro
2289
2290

```

```

2291 *-----
2292 *-> 1.11 Stata 设定
2293 *-----
2294
2295 *      ==本节目录==
2296
2297 *      1.11.1 Stata帮助
2298 *      1.11.2 文件目录
2299 *      1.11.3 Stata 外部命令的获取
2300 *          1.11.3.1 外部命令的存储路径
2301 *          1.11.3.2 外部命令的获取方式
2302 *          1.11.3.3 外部命令的管理和更新
2303 *      1.11.4 Stata 的系统参数
2304 *      1.11.5 文件和文件夹的操作
2305 *          1.11.5.1 文件的基本操作：查找、复制、编辑和删除
2306 *          1.11.5.2 使用stata打开txt, Word, Excel, 网页文件
2307 *          1.11.5.3 文件夹的操作
2308 *      1.11.6 每次启动时均需执行的命令(profile)
2309 *      1.11.7 常用快捷键
2310 *      1.11.8 退出stata(exit)
2311
2312
2313 *
2314 *-----1.11.1 Stata帮助----- -help-, -search-, -hsearch-, -findit-
2315
2316 * -help-命令
2317 * -search-命令  searches the [keywords] of the help files;
2318 * -hsearch-命令 searches the help files [themselves].
2319 * -findit-命令  类似-search-命令，但可以进一步搜索网络上的信息
2320
2321 help regress
2322 search panel data, net
2323 hsearch "fixed effect"
2324 findit panel unit root
2325
2326 * -view- 命令  新开口显示
2327
2328 view search panel data, net          // 新开口显示检索结果
2329 view news                          // 显示stata的最近动态
2330 view browse http://www.baidu.com    // 打开网页
2331 viewsource winsor.ado               // 查看 ado 文件源文件，只读
2332 viewsource xtreg_fe.ado
2333 viewsource xtbalance.ado
2334
2335
2336 *-更多的帮助和讨论
2337
2338 *- 常见问题解答：FAQ
2339 view browse http://www.stata.com/support/statalist/faq
2340
2341 *- 加入STATA用户邮件列表
2342 view browse http://www.stata.com/statalist/
2343
2344 *- 人大经济论坛【stata专版】
2345 view browse http://www.pinggu.org/bbs/forum-67-1.html
2346
2347 *- 人大经济论坛【VIP答疑专区】
2348 view browse http://www.pinggu.org/bbs/forum-114-1.html
2349
2350
2351 *
2352 *-----1.11.2 文件目录----- -help sysdir-
2353
2354
2355 *-1.11.2.1 stata 系统目录的设定
2356
2357 sysdir          // 显示当前系统目录的设定
2358
2359 *- 释义：
2360 *      STATA:   D:\stata11\          stata 安装根目录
2361 *      UPDATES: D:\stata11\ado\updates\ 【更新文件】的存储地址
2362 *      BASE:    D:\stata11\ado\base\   【官方命令】存储地址
2363 *      SITE:    D:\stata11\ado\site\   【自编命令】存储地址
2364 *      PLUS:    D:\stata11\ado\plus\   【外部命令】的储存地址

```

---

```

2365      * PERSONAL:  D:\stata11\ado\personal\ 【自有文件夹】首次安装时，需要自建
2366
2367      *- 查看
2368          pwd                // 当前工作路径
2369          personal           // 显示路径(个人文件夹)
2370          personal dir       // 查看详情
2371
2372      *- 设定  help sysdir
2373          sysdir set PLUS     "D:\stata11\ado\plus"      // 外部命令的存放地址
2374          sysdir set PERSONAL "D:\stata11\ado\personal" // 个人文件夹
2375
2376          adopath + "D:\mypaper\my_ado"      // 增加新的查询目录
2377          adopath - "D:\mypaper\my_ado"      // 取消特定查询目录
2378
2379
2380
2381      *
2382      *-1.11.3 Stata 外部命令的获取
2383
2384      *          -findit-, -ssc-, -net-, -adoupdate-, -mypkg-
2385
2386      *-1.11.3.1 外部命令的存储路径
2387
2388      *-说明:
2389      *   (1) 默认情况下, stata会在 "... \stata11\ado\plus" 文件夹下存储外部命令
2390      *   (2) 可通过 -sysdir set- 命令更改之
2391      *   (3) 第一次下载外部命令时, stata会自动建立 \plus 文件夹
2392
2393      sysdir
2394
2395
2396      *-1.11.3.2 外部命令的获取方式
2397
2398      *-findit-命令: 模糊查询
2399          findit panel data
2400          findit normal test
2401
2402      *-ssc-命令: 安装(卸载)来源于 ssc 的命令
2403      *          ssc: Statistical Software Components
2404          help ssc                // http://www.repec.org/
2405          ssc whatsnew
2406      * 查看来源于 ssc 的外部命令列表
2407          ssc describe b         // 列示以 -b- 开头的所有命令, 可为 a-z, 以及 "_"
2408          ssc describe x
2409          ssc des winsor
2410      * 下载安装 ssc 命令
2411          ssc install winsor, replace
2412
2413      *-net-命令
2414          help net
2415      *
2416      *-示例
2417          net search hausman test
2418          view net search hausman test
2419          net from http://fmwww.bc.edu/RePEc/bocode/m/
2420                                     // [result]窗口显示SSC命令
2421          view net from http://fmwww.bc.edu/RePEc/bocode/m/
2422                                     // 新窗口显示
2423      *
2424      *-Stata Journal(SJ) 相关文档
2425          view net from "http://www.stata-journal.com/"
2426          view net from "http://www.stata-journal.com/software/"
2427          net cd software // 网络不好时, 可能无法连接
2428          net cd sj9-2
2429      *
2430      *-Stata Technical Bulletin(STB) 相关文档
2431          net from "http://www.stata.com/stb/"
2432
2433
2434      *-1.11.3.3 外部命令的管理和更新
2435
2436      *-查询已安装的外部命令  -ado-, -mypkg-, -which-
2437          ado
2438          ado, find(winsor)

```

---

```
2439      ado, find(panel unit)
2440      mypkg                                // 呈现本机上已安装的外部命令 net findit ssc
2441      mypkg xt*
2442      mypkg *lorenz*
2443      mypkg xtbalance
2444      which xtbalance
2445      which outreg2                        // 列示命令的基本信息
2446
2447      *-外部命令的更新      -adoupdate-
2448      adoupdate                        // 更新本机上的外部 ado 命令
2449      adoupdate outreg2, update // 更新特定的命令
2450
2451      *-发布自己的 stata 命令
2452      help usersite
2453
2454
2455
2456      *
2457      *-1.11.4 Stata 的系统参数
2458
2459      query                            // 呈现当前系统参数的设定情况
2460
2461      * 关于版本
2462      about
2463
2464      * 验证是否正确安装
2465      verinst
2466
2467      * 系统参数范围
2468      help limits
2469
2470      * 一些常用的设定
2471      clear
2472      set obs 200                        // 设定观察值的个数
2473      set memory 40m
2474      *-----
2475      set more on                        // 开启 分屏显示
2476      sysuse auto, clear
2477      list price
2478      set more off                      // 禁止 分屏显示
2479      list price
2480      *-----
2481      clear
2482      set memory 40m                    // 设定内存的大小
2483      set matsize 3000                  // 设定矩阵的最大维度
2484      *-----
2485      set trace on                      // 跟踪调试
2486      sysuse auto, clear
2487      reg price wei
2488      set trace off
2489      *-----
2490      set seed 1357923                  // 产生随机数时的种子
2491      matrix a = matuniform(2,2)
2492      matrix list a
2493      *-----
2494      help set_defaults                 // 恢复系统参数的默认值
2495      set_defaults memory               // 仅恢复 memory 项
2496      set_defaults _all                 // 全部恢复
2497
2498
2499      *
2500      *-1.11.5 文件和文件夹的操作
2501
2502      * 相关命令: shell, shellout, findfile, erase,
2503      *           mkdir, rmdir, copysource, winexec
2504
2505      *-1.11.5.1 文件的基本操作: 查找、查看、复制、编辑和删除
2506
2507      findfile xtreg_fe.ado             // 查找文件
2508      copysource xtreg_fe.ado           // 在adopath路径下查找,复制到当前工作目录下
2509      dir xt*.ado                       // 显示当前工作目录下的文件
2510      viewsource xtreg_fe.ado           // 查看指定的 ado 文档(只读)
2511      doedit `c(pwd)'\xtreg_fe.ado     // 编辑指定的 ado 文档
2512      erase `c(pwd)'\xtreg_fe.ado       // 删除文件
```

```
2513
2514     copysource xtreg_fe.ado
2515     shell rename xtreg_fe.ado FE.do // 文件更名
2516     dir *.do
2517     shell // 在 dos 环境下操作
2518
2519     copy d1.txt new_d1.txt,replace // 复制文件
2520     dir *d1.txt
2521     copy http://www.stata.com/examples/simple.dta simple.dta, replace
2522     dir *.dta
2523     erase new_d1.txt
2524     erase simple.dta
2525
2526
2527 *-1.11.5.2 使用stata打开-.txt-, -Word-, -Excel-, -iexplorer- 文件
2528
2529 * 语法:
2530 * shellout 完整文件名 // help shellout
2531
2532 *-打开记事本
2533     shellout d1.txt
2534
2535 *-打开-Word-文档
2536     shellout mypaper.doc
2537
2538 *-打开-Excel-文档
2539     shellout d1.xls
2540
2541 *-打开网页
2542     shellout myhome.mht
2543     shellout my_log.html
2544
2545 *-打开-PPT-文档 // 自娱自乐一下吧
2546 *-打开-PDF-文档
2547
2548
2549 *-把帮助文件转换为 pdf 格式
2550     help hlp2winpdf
2551     hlp2winpdf, cdn(xtreg)
2552     shellout xtreg.pdf
2553
2554     hlp2winpdf, cdn(xtbalance xtabond) replace
2555     shellout xtbalance.pdf
2556     shellout xtabond.pdf
2557
2558 *-说明: 需要安装 Ghostscript 或 WinEdt 套装
2559 * 可到如下网址下载:
2560 * http://www.ctex.org/HomePage
2561
2562
2563
2564 *-1.11.5.3 文件夹的操作
2565
2566 *-stata官方命令 -dir-, -mkdir-, -rmdir-
2567
2568     dir // 显示当前目录下的所有文件
2569     dir *.txt // 显示后缀为 ".txt" 的所有文件
2570     dir xt* // 显示以 "xt" 开头的文件
2571
2572     mkdir `c(pwd)'\mystata // 新建文件夹
2573     rmdir mystata // 删除文件夹
2574
2575
2576 *-dirtools- 命令: 高效管理文件的外部命令
2577
2578     cd `c(sysdir_personal)'Net course_A
2579     lall // 列示所有文件
2580     cd A1_intro
2581     ldta // 列示 .dta 数据文件
2582     cd `c(sysdir_stata)'ado\base\x
2583     lado // 列示 .ado 文件
2584
2585
2586 *-cdout- 命令: 打开当前工作路径所在的文件夹
```



```

2587      cd D:\stata11\utilities
2588      cdout
2589      cd `c(sysdir_personal)'Net_course_A
2590      cdout
2591
2592
2593
2594      *
2595      *-1.11.6 每次启动时均需执行的命令      -profile-
2596
2597      help profile
2598
2599      * 建立一个 profile.do 文档，存于 D:\stata11\ 下
2600
2601      * -----begin profile.do-----
2602      *
2603      * 基本参数设定
2604      set type double
2605      set memory 50m
2606      set matsize 2000
2607      set scrollbufsize 50000 // 设定屏幕的最大显示行数
2608      set more off,perma
2609
2610      * log 文件设定
2611      log using D:\stata11\ado\personal\stata.log, text replace
2612      cmdlog using D:\stata11\ado\personal\command.log, append
2613
2614      * 文件目录设定
2615      sysdir set PLUS "D:\stata11\ado\plus" //外部命令的存放地址
2616      sysdir set OLDPLACE "D:\ado"
2617      sysdir set PERSONAL "D:\stata11\ado\personal" //个人文件夹
2618
2619      * ado文档查找路径
2620      adopath + "D:\stata11\ado\personal"
2621      adopath + "D:\stata11\ado\personal\_Myado"
2622
2623      * 当前工作路径
2624      cd D:\stata11\ado\personal
2625
2626      * -----end profile.do-----
2627
2628
2629      *- Arlion 的 profile.do 文档
2630
2631      *doedit D:\stata11\profile.do
2632      doedit `c(sysdir_stata)'profile.do
2633
2634      *-我的日志文件
2635      cd D:\stata11\do
2636      cdout
2637
2638
2639      *
2640      *-1.11.7 常用快捷键
2641
2642      /*
2643      F-key      Definition
2644      -----
2645      F1         help
2646      F2         #review;
2647      F3         describe;  (*)
2648      F7         save
2649      F8         use
2650      -----
2651
2652      Ctrl-key    Definition
2653      -----
2654      Ctrl+D      执行 (Do) 选中的命令  (*)
2655      Ctrl+R      运行程序 (Run)  (*)
2656      Ctrl+F      在do-editor中搜索特定的关键词
2657      Ctrl+O      打开do文档
2658      Ctrl+N      新建do文档
2659      Ctrl+S      保存do文档  (*)
2660

```



---

```
2661      Ctrl+G      跳转到第#行      (*)
2662      Ctrl+Shift+Y 选中光标所在的行
2663      Ctrl+Y      删除光标所在的行
2664      Ctrl+F2     定义小节标签
2665      Shift+F2    跳转到上一个小节标签
2666      F2         跳转到下一个小节标签
2667      -----
2668      注：上述快捷键仅适用于do-editor
2669
2670                               */
2671
2672
2673      *
2674      *-1.11.8 退出stata:      -exit-
2675
2676      *-几个需要注意的事项：
2677
2678      *- 常规方法
2679          * 点击叉号关闭stata，多数情况下都无需保存；
2680
2681      *- 命令方法
2682          exit
2683          exit, clear
2684
2685
2686
2687
```

```

1
2
3
4
5      *      =====
6      *      计量分析与STATA应用
7      *      =====
8
9      *      主讲人：连玉君 博士
10
11     *      单   位：中山大学岭南学院金融系
12     *      电   邮：arlionn@163.com
13     *      主   页：http://blog.cnfol.com/arlion
14
15     *      ::第一部分::
16     *      Stata 操作
17     *      =====
18     *      第二讲  数据处理
19     *      =====
20
21     * cd D:\stata10\ado\personal\Net_course_A\A2_data
22
23     cd `c(sysdir_personal)'Net_course_A\A2_data
24
25
26 *-----
27 *   本讲目录
28 *-----
29
30 *   2.1  变量转换的更多技巧
31 *   2.2  分位数
32 *   2.3  重复样本值的处理
33 *   2.4  缺漏值的处理
34 *   2.5  离群值的处理
35 *   2.6  资料的合并和追加
36 *   2.7  重新组合样本
37 *   2.8  文字变量的处理
38 *   2.9  类别变量的分析
39 *   2.10 时间序列资料的处理
40 *   2.11 面板资料的处理
41 *   2.12 数据的查验和比较
42
43
44
45
46     *      =====
47     *      计量分析与STATA应用
48     *      =====
49
50     *      主讲人：连玉君 博士
51
52     *      单   位：中山大学岭南学院金融系
53     *      电   邮：arlionn@163.com
54     *      主   页：http://blog.cnfol.com/arlion
55
56     *      ::第一部分::
57     *      Stata 操作
58     *      =====
59     *      第二讲  数据处理
60     *      =====
61     *      2.1 创建变量的更多技巧
62
63
64 *-----
65 *-2.1 创建变量的更多技巧
66 *-----
67
68 *      ==本节目录==
69
70 *      2.1.1   $\bar{n}$  和  $\bar{N}$ 
71 *      2.1.1.1  $\bar{n}$  与  $\bar{N}$ 
72 *      2.1.1.2  $\bar{n}$  与  $\bar{N}$  的应用
73 *      2.1.2  虚拟变量的产生
74 *      2.1.2.1 基本方式

```

```

75      *      2.1.2.2 基于类别变量生成虚拟变量: -tab-命令
76      *      2.1.2.3 基于类别变量生成虚拟变量: -xi-命令
77      *      2.1.2.4 因子变量 (stata11 的一大亮点)
78      *      2.1.2.5 将连续变量转换为类别变量
79      *      2.1.2.6 利用条件函数产生虚拟变量
80      *      2.1.3 交乘项的产生
81      *      2.1.4 -egen- 命令
82      *      2.1.4.1 egen 与 gen 的区别
83      *      2.1.4.2 产生等差数列: seq() 函数
84      *      2.1.4.3 填充数据: fill() 函数
85      *      2.1.4.4 产生组内均值和中位数
86      *      2.1.4.5 跨变量的比较和统计
87      *      2.1.4.6 变量的标准化
88      *      2.1.4.7 变量的平滑化 (Moving Average)
89      *      2.1.4.8 更多的 egen() 函数
90
91
92
93      *      =本节命令=
94      *      =====
95      *      _n, _N, tsset, egen, display, list, tabulate
96      *      xi, fvset fvvarlist, recode, recode(), irecode()
97      *      cond(), inlist(), inrange(), egenmore,
98      *      =====
99
100
101      *
102      *-2.1.1 _n 和 _N
103
104      *-2.1.1.1 _n 和 _N 的含义
105
106      *-定义:
107      *   _n "样本序号变量", 是一个变量, 内容为 1,2,3,...,n
108      *   _N "样本数指标", 是一个单值, 内容为 样本数
109
110      *-说明:
111      *   _n 是一个永远存在, 但却不能 list 出来的特殊变量
112      *   _n 的取值会随样本排序的变化而变化
113
114      sysuse nls88.dta, clear
115      list age wage in 1/10      // 最左边的1,2,...就是 _n 中的内容
116      list _n                    // 错误
117
118      sort hours
119      gen nid_1 = _n              // 第一个 _n 的内容
120      list nid_1 hours race in 1/10
121      sort wage
122      gen nid_2 = _n              // 第二个 _n 的内容
123      list nid_1 nid_2 hours race in 1/10
124
125      dis _N                      // _N 是一个单值
126      scalar obs = _N
127      quietly sum wage
128      dis r(mean)*_N
129      dis r(mean)*obs
130
131
132      *-2.1.1.2 _n 和 _N 的应用
133
134      sysuse sp500.dta, clear
135      sort open
136      sum open
137      dis r(max)
138      gen o_max = open[_N]        // 最大值
139      gen o_diff = open[_n] - open[_N] // 与最大值的差
140      gen b_diff = open[_N] - open[1] // range
141      list open o_max o_diff b_diff in 1/20
142
143      *-差分
144      sort date
145      gen d_open = open[_n] - open[_n-1]
146
147      *-对数差分
148      gen dln_open = ln(open[_n]) - ln(open[_n-1])

```

```

149
150     *-移动平均
151     gen mv3_open = (open[_n-1] + open[_n] + open[_n+1]) / 3
152     list open o_max o_diff dln_open mv3_open in 1/10
153
154     *-滞后项、前推项、差分
155     tsset date /*声明数据为时间序列*/
156     gen open_lag = L.open
157     gen open_lag2 = L2.open
158     gen open_forward = F.open
159     gen open_diff = D.open
160     gen open_diff2 = D2.open
161     list open* in 1/10
162     reg close L(1/3).(close open)
163
164     *-增长率
165     qui tsset date
166     gen r1 = D.close/L.close
167     gen lnclose = ln(close)
168     gen r2 = D.lnclose // 第二种计算方法
169     list date r1 r2 in 1/10
170
171     *-分组进行
172     sysuse nls88.dta, clear
173     bysort industry: gen gid = _n
174     list gid industry in 1/50, sepby(industry)
175
176
177
178     *
179     *-2.1.2 虚拟变量的产生
180
181     *-2.1.2.1 基本方式
182
183     *-使用-generate-和-replace-产生虚拟变量
184     sysuse nls88.dta, clear
185
186     gen dum_race2=0
187     replace dum_race2=1 if race==2
188     gen dum_race3 = 0
189     replace dum_race3=1 if race==3
190
191     list race dum_race* in 1/100, sepby(race)
192
193
194     *-2.1.2.2 基于类别变量生成虚拟变量: -tab-命令
195
196     sysuse nls88.dta, clear
197     tab race, gen(dum_r)
198     list race dum_r1-dum_r3 in 1/100, sepby(race)
199
200
201     *-2.1.2.3 基于类别变量生成虚拟变量: -xi-命令
202
203     xi i.race //自动定义虚拟变量的名称, 并附加标签
204
205     list race _Irace_2 _Irace_3 in 1/100, sepby(race)
206
207     *-特别注意: 再次使用-xi-命令时, 此前生成的虚拟变量会被覆盖
208     xi i.occupation /**_Irace_2和_Irace_3变量不复存在
209
210     *-解决方法-1-: 使用 prefix(str) 选项,
211     sysuse nls88, clear
212     xi i.race, prefix(dr_) // 前缀不能超过四个字符
213     xi i.occu, prefix(do_) // 不同的类别变量采用不同的前缀
214
215     *-解决方法-2-: 事先修改变量名称: -renvars- (SJ 5-4)
216     help renvars
217     sysuse nls88.dta, clear
218     xi i.race
219     renvars _Irace* \ dum_race_2 dum_race_3 // 外部命令, 批量修改变量名
220     xi i.occupation
221     renvars _Ioccu*, prefix(dum) // 批量修改变量名称的前缀
222

```

```

223      *-优点：所有虚拟变量的前缀都可以是 "dum_"
224
225      *-noomit- 选项
226      sysuse nlsw88, clear
227      tab race
228      xi i.race    // 只生成了两个虚拟变量, 如何生成三个虚拟变量?
229      des _I*
230
231      xi i.race, prefix(dum_) noomit
232      des dum*
233
234
235      *-2.1.2.4 因子变量 (stata11 的一大亮点)
236
237      help fvvarlist    // 基本语法规则
238      help fvset        // 对照组的设定
239
240      *-简介
241      sysuse nlsw88, clear
242
243      list race i.race in 500/525, sep(0)
244      list race#married in 1/50 , sep(0)    // 4 组
245      list race#married in 1/100, sep(0)    // 6 组, why?
246
247      list i.union i.married union#married in 1/50, sep(0)
248      list union##married in 1/50, sep(0)    // 与上面的命令等价
249
250      *-应用
251      reg wage i.race
252      reg wage i.race i.married race#married
253      reg wage race##married    // 与上面的命令等价
254
255
256      *-对照组的選擇
257      view help fvvarlist##bases
258
259      *-选择 race=other 作为对照组
260      label list racelbl    // race=1 (Min) 是stata默认的对照组
261      reg wage ib3.race
262
263      *-选择 race=other, married=1 作为对照组
264      label list marlbl
265      reg wage ib3.race ib1.married
266      reg wage ib3.race##ib1.married    // 加入交乘项
267
268      *-永久设定对照组
269      help fvset
270      fvset base 3 race    // 在后续使用 i.race 过程中, race=3都是对照组
271      reg wage i.race
272
273
274      *-连续变量的设定
275      help fvvarlist
276      reg wage i.married hours i.married#c.hours
277      reg wage i.married##c.hours    // 等价于上述命令
278
279      reg wage i.married##c.hours    /// // 婚否
280      i.union##c.hours    /// // 是否工会成员
281      i.collgrad##c.hours    // 是否大学毕业
282
283      reg wage hours c.hours#c.hours    // 增加平方项
284      reg wage c.hours##c.hours    // 等价于上述命令
285
286      reg wage c.hours##c.hours##c.hours    // 增加三次方
287
288
289
290      *-2.1.2.5 将连续变量转换为类别变量
291
292      *- 等分样本          -group()-
293      sysuse nlsw88.dta, clear
294      sort wage
295      gen g_wage = group(5)    // 这一步很重要
296      tab g_wage

```

```

297      tabstat wage, stat(N mean med min max) by(g_wage) f(%4.2f)
298
299      *- 指定分界点的转换方式 -recode-
300      sum age
301      recode age (min/39 = 1) (39/42 = 2) (42/max = 3), gen(g_age)
302      * 1   if age<=39           右封闭区间
303      * 2   if 39<age<=42
304      * 3   if age>42
305      list age g_age in 1/50, sepby(g_age)
306
307      *-Q: 如果希望将 39 岁女员工归入第 2 类, 该如何下达命令?
308      recode age (39/42 = 2) (min/39 = 1) (42/max = 3), gen(g1_age)
309
310      *- 利用irecode() 和 recode() 函数进行转换
311
312      * -irecode()- 函数
313      gen g2_age = irecode(age, 39, 42)
314      ttest g_age = g2_age
315
316      * -recode()- 函数
317      gen g3_age = recode(age, 39, 42)
318      list age g_age g2_age g3_age in 1/10, sepby(g_age)
319
320
321      *-2.1.2.6 利用条件函数产生虚拟变量
322
323      *- cond() 函数
324
325      * 基本语法: cond(s,a,b) | cond(s,a,b,c)
326      * 取值:
327      * a     if 表达式 s 为真;
328      * b     if 表达式 s 为假;
329      * c     if 表达式 s 为缺漏值
330      * 示例:
331      sysuse nlsw88, clear
332      gen dum1 = cond(hours>40, 1, 0, .)
333      list hours dum1 in 1/20
334      gen dum2 = cond(hours>40&hours!=., 1, 0, .)
335      list hours dum1 dum2 in 1/20 // 注意此处的区别
336
337      gen dum_ratio = cond(wage/hours>0.5, 1, 0)
338      list wage hours dum_ratio in 1/20
339
340
341      *- inlist() 函数
342
343      * 基本语法: inlist(x, a,b,c,...)
344      * 取值:
345      * 1     if x = a,b,c,...中的任何一个
346      * 0     otherwise
347      * 规则:
348      * 若x为实数, 则后续取值必须介于2-255
349      * 若x为字符, 则后续填项的个数必须介于2-10
350      *
351      * 示例 1:
352      label list occ1b1
353      gen dum_occu = inlist(occu, 1,2,7,12)
354      list occu dum_occu in 1/20
355      * 等价于
356      gen dum_occu1 = (occ==1|occ==2|occ==7|occ==12)
357      *
358      * 示例 2:
359      use gdp_China.dta, clear
360      sort Y
361      list in 1/10 // 如何产生地区虚拟变量?
362      *egen pvname = msub(prov), f(" ") //去掉省名中的空格
363      gen east = inlist(prov,"北京","福建","广东","江苏", ///
364      "辽宁","山东","上海","天津","浙江")
365      sort east prov
366      browse if year ==2003
367
368
369      *- inrange() 函数
370

```

```

371      * 基本语法:  inrange(x, a,b)
372      * 取值:
373      * 1      if  a<= x <= b;
374      * 0      otherwise
375
376      * 示例:
377      sysuse nlsw88, clear
378      gen dum_h2 = inrange(hours, 30,40)
379
380      * 等价于
381      gen dum_h3 = (hours>=30 & hours<=40)
382      list hours dum_h2 dum_h3 in 1/20
383
384
385      *- clip() 函数
386
387      * 基本语法:  clip(x, a,b)
388      * 取值:
389      * a      if x<=a;    // 截尾
390      * x      if a<x<b;    // 原始值
391      * b      if x>=b      // 截尾
392
393      gen g_h4 = clip(hours, 30, 40)
394      list hours g_h4 in 1/100
395
396      *-以此为基础, 可进一步产生虚拟变量
397
398
399
400
401
402
403
404
405
406
407
408      *      =====
409      *      计量分析与STATA应用
410      *      =====
411
412      *      主讲人: 连玉君 博士
413
414      *      单 位: 中山大学岭南学院金融系
415      *      电 邮: arlionn@163.com
416      *      主 页: http://blog.cnfol.com/arlion
417
418      *      ::第一部分::
419      *      Stata 操作
420      *      =====
421      *      第二讲 数据处理
422      *      =====
423      *      2.1 创建变量的更多技巧(续)
424
425
426      *
427      *-2.1.3 交乘项的产生
428
429      *-stata11用户: 参见"-2.1.2.4- 因子变量" 小节
430
431      *-基本方法  -generate- 命令
432
433      sysuse nlsw88, clear
434
435      gen ttl_exp_x_marry = ttl_exp*married
436
437      reg wage married ttl_exp ttl_exp_x_marry
438
439
440      *-批量产生虚拟变量  -xi- 命令
441
442      *-如何得到"种族" 与"是否已婚"的交乘项
443      * 两个类别变量交乘 i.v1*i.v2
444      xi:reg wage married ttl_exp i.race*i.married

```

```
445
446     *-如何得到“种族” 与 “工作经验”的交乘项 ?
447     * 一个类别和一个连续变量交乘 i.v1*v2
448     xi:reg wage married i.race*ttl_exp // ttl_exp会被自动加入
449
450
451
452 *
453 *-2.1.4 -egen- 命令
454
455     * extended generate 的缩写
456     help egen
457
458     *-2.1.4.1 egen 与 gen 的区别
459
460     *-基本差异
461     sysuse sp500, clear
462     gen sum_close0 = sum(close) // 累加
463     egen sum_close1 = sum(close) // 总体加总
464     list close sum_close0 sum_close1 in 1/10
465
466     *-对于缺漏值的处理也有差异
467     clear
468     input v1 v2
469           1 5
470           2 .
471           . 3
472           2 4
473           4 .
474           . 6
475     end
476     gen mean = (v1+v2)/2
477     egen mean_egen = rmean(v1 v2)
478     list
479
480
481     *-2.1.4.2 产生等差数列: seq() 函数
482     clear
483     set obs 100
484     egen x1 = seq(), from(-1)
485     list x1 in 1/10
486     egen year = seq(), from(2000) to(2004)
487     list year in 1/20
488     egen code = seq(), from(1) block(5)
489     list code in 1/20
490     list code year in 1/20
491
492
493     *-2.1.4.3 填充数据: fill() 函数
494
495     egen r2 = fill(2 4) // 间隔 2 的递增数列
496     egen r3 = fill(6 3) // 间隔 -3 的递减数列
497     egen r4 = fill(1990 1991 1992 1990 1991 1992) // 分块重复数列
498     list r2-r4 in 1/20
499
500
501     *-2.1.4.4 产生组内均值和中位数
502
503     sysuse nlsw88.dta, clear
504     egen avg_w_r = mean(wage), by(race)
505     egen med_w = median(wage), by(race)
506     list wage race avg_w_r med_w in 1/20
507
508     use xtcs.dta, clear // 中国上市公司资本结构数据
509     egen msize = mean(size), by(code) // 这样可以保证每家公司的组别一致
510     sort msize
511     gen gsize = group(3) // 根据公司规模分组
512     bysort gsize year: egen mtl = mean(tl) // 注意 -bysort- 的使用方法
513     sort gsize year
514     list code year gsize tl mtl in 1/40, sep(0)
515     list code year gsize tl mtl in 2500/2540, sep(0)
516
517     *-应用举例
518     xtreg tl size fr ndts tobin tang, fe
```



```
519         est store full
520         xtreg tl size fr ndts tobin tang if gsize==1, fe
521         est store small
522         xtreg tl size fr ndts tobin tang if gsize==2, fe
523         est store mid
524         xtreg tl size fr ndts tobin tang if gsize==3, fe
525         est store large
526         local m "full small mid large"
527         esttab `m', mtitle(`m') s(N r2) b(%6.3f) ///
528             nogap compress
529
530     *-说明: 利用 egen 提供的函数, 尚可计算组内s.d., Max, Min 等指标
531
532
533 *-2.1.4.5 跨变量的比较和统计
534
535     sysuse sp500.dta, clear
536
537     egen avg_price = rmean(open close)
538     list open avg_price close in 1/10
539
540     replace open = int(open)
541     replace close= int(close)
542     egen diff = diff(open close)
543     sort diff
544     list open diff close in 1/10
545
546
547 *-2.1.4.6 变量的标准化
548
549     *-定义:  $x_s = (x - x_m) / x_{sd}$ 
550     *- $x_s$  的均值将为 0; 标准差将为 1
551     *-线性转换, 并不改变变量间的相对大小
552
553     sysuse sp500.dta, clear
554     egen s_change1 = std(change)
555     egen s_change2 = std(change), mean(20) std(3)
556     sum change s_change*
557
558     do A2_egen_std.do
559
560
561 *-2.1.4.7 变量的平滑化 (Moving Average)
562
563     sysuse sp500, clear
564     tsset date
565     egen mv3_open      = ma(open)
566     egen mv5_open      = ma(open), t(5)
567     egen mv5_open_nomiss = ma(open), t(5) nomiss
568     list *open* in 1/10
569     dis (1320.28+1283.27+1347.56)/3 // 第一个观察值
570     dis (1320.28+1283.27+1347.56+1333.34)/4 // 第二个观察值
571
572
573 *-2.1.4.8 更多的 egen() 函数
574
575     help egenmore // 外部命令
576
577     *-ntos() 函数
578     sysuse auto, clear
579     tab rep78
580     egen grade = ntos(rep78), from(1/5) to("优秀" 好 较好 较差 差)
581     browse rep78 grade
582     *-ston() 函数的用法与此相似
583
584
585     *-nvals() 函数
586
587     *-数据描述
588     use bank_number.dta, clear // 银企关系数目
589     browse
590     tab objbank, sort // 任务: 统计出各个年度每家公司的银企关系数目
591     drop if strmatch(objbank, "**公司*")
592     drop if strmatch(objbank, "银行")
```

```
593         tab objbank, sort
594
595     *-统计方法
596         egen banknum = nvals(objbank), by(id year)
597
598     *-结果
599         list, sepby(id year)
600
601
602     *-另一种解决方法(stata内部命令)
603         use bank_number.dta, clear
604         egen tag = tag(id year objb) //第一个非重复值标记为1
605         list, sepby(id year)
606         bysort id year: egen banknum = total(tag) ///
607             if strmatch(objbank,"*银行*")
608         list, sepby(id year)
609         drop if banknum == .
610         list, sepby(id year)
611
612
613     *-incss() 函数
614         use bank_number.dta, clear //删除非银行金融结构
615         egen isbank = incss(objbank), substr("银行")
616         list, sep(0)
617         drop if isbank==0
618
619
620     *-gmean() 函数 [geometric mean] 几何平均数
621
622     *-定义:  $G = [x_1 \times x_2 \times \dots \times x_n]^{1/n}$ 
623
624         sysuse auto, clear
625         egen g_mpg = gmean(mpg), by(rep78) // 几何平均数
626         egen m_mpg = mean(mpg), by(rep78) // 算术平均数
627         sort rep78
628         list rep78 *mpg, sepby(rep78)
629
630
631     *-hmean() 函数 [] 调和平均数
632
633         
$$H = \frac{n}{1/x_1 + 1/x_2 + \dots + 1/x_n}$$

634     *-定义:
635
636
637
638     *-semean(), var(), sumoth(), xtile() 函数
639
640
641     *-其它函数
642     * -egenms- create a moving sum.
643     * -egenmsd- create a moving standard deviation.
644     * -egenmmed- create a moving median.
645
646
647
648
649
650
651
652
653
654
655     * =====
656     * 计量分析与STATA应用
657     * =====
658
659     * 主讲人: 连玉君 博士
660
661     * 单 位: 中山大学岭南学院金融系
662     * 电 邮: arlionn@163.com
663     * 主 页: http://blog.cnfol.com/arlion
664
665     * ::第一部分::
666     * Stata 操作
```

```

667      *      =====
668      *      第二讲 数据处理
669      *      =====
670      *      2.2 分位数
671
672
673      *-----
674      *-> 2.2 分位数
675      *-----
676
677      *      ==本节目录==
678
679      *      2.2.1 分位数的基本概念
680      *      2.2.2 -pctile- 命令
681      *      2.2.3 -xtile- 命令
682      *      2.2.4 -_pctile- 命令
683
684
685      *      =本节命令=
686      *      =====
687      *      -pctile- -xtile- -_pctile-
688      *      =====
689
690
691      *-----
692      * -2.2.1 分位数的基本概念
693
694      * 示例 1:
695      clear
696      set obs 100
697      gen x = _n
698      sum x, detail
699
700      * 示例 2:
701      clear
702      set obs 101
703      gen x = _n
704      sum x, detail
705
706      * 定义和公式
707
708      * 第 p_th 百分位数值, 记为 x_|p|, 设 p = N_p/100,
709      *
710      * 例如, 若求取第 25 百分位的数值, 则 N_p=25, p=0.25
711      *
712      * 每个观察值的权重为 w(i)=1/N (N为样本数),
713      *
714      * 前 i 个观察值的权重之和为 W(i)=SUM_w(i) = i/N
715      *
716      * 则第 p_th 百分位的数值定义为, 第一个满足 W(i)>p 的观察值, 即
717      *
718      *      { {x[i-1]+x[i]}/2      if W(i-1)=p
719      *      x_|p| = {
720      *      {      x[i]              otherwise
721      *
722      dis 5/101      // 条件 W(5)=0.495>0.05 不满足
723      dis 6/101      // 这是第一个满足 W(6)=0.0594>0.05 的观察值的序号
724      dis x[6]       // 第 5 百分位数值
725
726      * 简单的处理方法:
727      dis ceil(101*0.25)
728      dis x[26]      // 这就是第25百分位数值, 多数情况都是"otherwise"
729
730      * 示例 1(回顾):
731      clear
732      set obs 100      // 此例中, 仅有100个观察值
733      gen x = _n
734      sum x, detail
735      dis 5/100      // 0.05 = w(6-1) ==> i=6
736      dis (x[5]+x[6])/2 // 第5百分位数值
737
738
739      *-----
740      * -2.2.2 -pctile- 命令

```

```

741
742     sysuse auto, clear
743     pctlile p_price = price, nq(10)
744             // nq(#) 指定分9个百分位数,把样本切割为10组
745     list p_price in 1/12, sep(0)
746     sum price, detail
747
748     pctlile p_price2 = price, nq(10) gen(percent)
749             // gen() 选项用于生成对应的百分位标识
750     list percent p_price2 in 1/12, sep(0)
751
752
753 *
754 *-2.2.3 -xtile- 命令 // 根据指定的百分位数定义类别变量
755
756     use bp1.dta, clear
757     xtile x_bp = bp, nq(4)
758     list, sepby(x_bp)
759
760     *-解释: nq(4) 创建规则如下
761
762     * (-00,x25], (x25,x50], (x50,x75], (x75,+00) // 右封闭区间
763
764     * 上述分位数的生成过程
765     pctlile xp_bp = bp, nq(4) genp(percent)
766     list bp xp_bp percent
767
768     * cutpiont() 选项
769     input class
770         100
771         110
772         120
773         130
774     end
775     xtile c_bp = bp, cutpoints(class)
776     list bp c_bp class, sepby(c_bp)
777
778
779 *
780 *-2.2.4 -_pctlile- 命令
781
782     *-类似于-pctlile-命令, 但能够提供各个分位值的返回值
783
784     sysuse auto, clear
785     _pctlile length, nq(10)
786     return list
787     sum price if (length>r(r9))
788
789     _pctlile price, p(33.3 72 90 99)
790     return list
791     *-自行指定分为点, 这是该命令的最大优势
792
793
794
795
796
797
798
799
800     * =====
801     * 计量分析与STATA应用
802     * =====
803
804     * 主讲人: 连玉君 博士
805
806     * 单 位: 中山大学岭南学院金融系
807     * 电 邮: arlionn@163.com
808     * 主 页: http://blog.cnfol.com/arlion
809
810     * ::第一部分::
811     * Stata 操作
812     * =====
813     * 第二讲 数据处理
814     * =====

```

```

815          *          2.3 重复样本值的处理
816
817
818  *-----
819  *-2.3 重复样本值的处理
820  *-----
821
822  *-类别变量中样本的重复非常普遍，也具有特殊的含义
823  *-连续变量中的重复样本往往因为资料谬误所致
824
825
826          *          ==本节目录==
827
828          *          2.3.1 检查重复的样本组
829          *          2.3.2 标记和删除重复的样本组合
830
831
832          *          =本节命令=
833  * =====
834  * isid, duplicates report/examples/list/tag/drop
835  * egen group()
836  * =====
837
838
839  *-----
840  *-2.3.1 检查重复的样本组合
841
842          sysuse nlsw88.dta, clear
843
844  *-isid- 命令 学号和姓名
845          isid race age
846          isid idcode
847
848  *-duplicates list- 命令
849          duplicates list race married in 1/20
850
851  *-duplicates report- 命令
852          duplicates report race
853          duplicates report race married
854          duplicates report race married occupation
855
856  *-duplicates example- 命令
857          duplicates example race married
858          tab race married
859
860
861  *-----
862  *-2.3.2 标记和删除重复的样本组合
863
864  *-标记重复的样本组合
865
866  *-使用 group() 函数
867
868          sysuse nlsw88.dta, clear
869
870          egen rm = group(race married)
871          tab rm, gen(dum_rm) // 可以进一步用此变量创造虚拟变量
872
873          egen rm_lb = group(race married), label
874          label list rm_lb
875          list rm rm_lb in 1/10
876          browse race married rm_lb rm
877
878  *-使用 tag() 函数，第一个非重复样本为1，其他为零
879
880          egen rm_tag = tag(race married)
881          list rm* in 1/20
882
883  *-使用 -duplicates tag- 命令
884
885          duplicates tag race married, gen(rm_dtag) //重复值的个数
886          list rm* in 1/20
887
888

```

```
889      *-删除重复的样本组合
890
891      duplicates drop race married, force
892
893      *-对于 Panel Data 而言，我们可以使用如下命令删除重复的样本公司
894      * duplicates drop id year, force
895      * 详见：第 2.11.1 小节
896
897
898
899
900
901
902
903      *      =====
904      *      计量分析与STATA应用
905      *      =====
906
907      *      主讲人：连玉君 博士
908
909      *      单 位：中山大学岭南学院金融系
910      *      电 邮：arlionn@163.com
911      *      主 页：http://blog.cnfol.com/arlion
912
913      *      ::第一部分::
914      *      Stata 操作
915      *      =====
916      *      第二讲 数据处理
917      *      =====
918      *      2.4 缺漏值的处理
919
920
921      *-----
922      *-2.4 缺漏值的处理
923      *-----
924
925      *      ==本节目录==
926
927      *      2.4.1 缺漏值简介
928      *      2.4.2 缺漏值的标记
929      *      2.4.3 查找/删除缺漏值
930      *      2.4.3.1 缺漏值的形态
931      *      2.4.3.2 删除缺漏值
932      *      2.4.4 填补空缺(gap)
933      *      2.4.5 多重补漏分析(multiple-imputation)
934      *      2.4.5.1 MI 简介
935      *      2.4.5.2 实例分析
936      *      2.4.5.3 MI impute regress 的假设条件
937      *      2.4.5.4 其它补漏方法
938      *      2.4.5.5 假设检验
939
940
941      *      =本节命令=
942      *      =====
943      *      missing, mi(), mvencode, mvdecode, mistable
944      *      rmiss(), dropmiss,
945      *      mi set, mi impute, miestimate,
946      *      =====
947
948
949      *
950      *-2.4.1 缺漏值简介
951
952      help missing
953
954      *- "." 大于任何自然数
955
956      sysuse auto, clear
957      sort rep78
958      list rep78
959      sum rep78 if rep78>4      // obs=11
960      count if rep78>4         // obs=16, why?
961      keep if rep78>4
962      list rep78
```

```
963
964  *说明:
965  *有些命令, 如 sum, regress, generate 等, 会自动忽略缺漏值;
966  *有些命令, 如 count, keep 等, 则会将 "." 视为一个无穷大的数值
967  *使用过程重要特别注意
968
969
970  *
971  *-----2.4.2 缺漏值的标记-----
972
973  *数值型缺漏值
974  shellout d_miss.txt
975  insheet using d_miss.txt, clear
976  sum
977  mvdecode x1, mv(-97) // 重新定义某个变量的缺漏值
978  list
979  sum
980  insheet using d_miss.txt, clear
981  mvdecode _all, mv(-97 -999)
982  sum
983
984  *文字型缺漏值
985  type d201.txt
986  insheet using d201.txt, clear
987  replace x1 = . if x1== "N/A" // 错误方式
988  replace x1 = "." if x1== "N/A"
989  des
990  gen x1_new = real(x1)
991
992
993
994  *
995  *-----2.4.3 查找/删除缺漏值----- -misstable- stata11新增功能
996
997  *2.4.3.1 缺漏值的形态
998
999  *最简单的命令: -summarize-
1000  sysuse nlsw88.dta, clear
1001  sum
1002
1003  *misstable summarize-命令: 缺漏值的基本统计
1004  sysuse nlsw88.dta, clear
1005  misstable summarize // 所有变量
1006  misstable sum age-union // 指定变量
1007
1008  *misstable pattern-命令: 列示缺漏值的模式
1009  misstable pattern
1010  misstable pattern, bypat
1011
1012  *misstable tree-命令: 详细列示缺漏值的模式
1013  misstable tree union tenure in 1/1000, freq
1014  *对照解释
1015  misstable summ union tenure in 1/1000
1016
1017
1018  *2.4.3.2 删除缺漏值
1019
1020  *rmiss()函数
1021  sysuse nlsw88.dta, clear
1022  egen miss = rmiss(wage industry occupation)
1023  list wage industry occupation miss if miss!=0
1024  drop if miss!=0
1025
1026  *missing()函数
1027  sysuse nlsw88.dta, clear
1028  sum
1029  drop if missing(grade, indus, occup, union, hours, tenure)
1030  sum
1031
1032
1033  *更为简洁的命令: -dropmiss- (外部命令)
1034
1035  help dropmiss
1036
```

```
1037 sysuse nlsw88.dta, clear
1038 sum
1039 misstable sum
1040 dropmiss, obs // 以观察值为单位
1041 sum
1042 misstable sum // nothing dropped
1043
1044 sysuse nlsw88.dta, clear
1045 sum
1046 dropmiss, any // 以变量为单位
1047 sum // 6 variables dropped
1048 misstable sum
1049
1050 sysuse nlsw88.dta, clear
1051 dropmiss, any obs // 这或许是我们所需要的
1052 sum
1053
1054
1055 *-另一种巧妙的方法 -regress- 命令
1056
1057 sysuse nlsw88.dta, clear
1058 sum
1059
1060 reg wage industry occup tenure hours
1061
1062 gen byte nomis = e(sample) // 标示样本的虚拟变量
1063 sum wage industry occup tenure hours if (nomis==1)
1064 keep if nomis
1065
1066
1067
1068 *
1069 *-2.4.4 填补空缺(gap)
1070
1071 * 对于Panel Data或一些特殊的资料, 可采用前向或后向非缺漏值填补
1072 * http://www.stata.com/support/faqs/data/missing.html
1073
1074 * case1: 单一缺漏值之填补
1075 use d_miss01.dta, clear
1076 list
1077 sort t
1078 replace x = x[_n-1] if x==.
1079 list
1080
1081 use d_miss01.dta, clear
1082 list
1083 sort t
1084 replace x = x[_n+1] if missing(x) // help missing()
1085 list
1086
1087 * case2: 多个缺漏值之填补
1088 use d_miss02.dta, clear
1089 list
1090 sort t
1091 replace x = x[_n-1] if mi(x)
1092 list
1093 * 解释: 依次进行替换
1094 * 后向替换
1095 use d_miss02.dta, clear
1096 list
1097 gsort -t
1098 list
1099 replace x = x[_n-1] if mi(x)
1100 sort t
1101 list
1102
1103 * case3: Panel Data缺漏值之填补
1104 use d_miss03.dta, clear
1105 list , sep(4)
1106 tsset id year
1107 by id: replace x = L.x if mi(x)
1108 list, sep(4)
1109
1110
```



```
1111  *--进一步阅读的资料:
1112  *
1113  * [1] How can I drop spells of missing values at
1114  *      the beginning and end of panel data?
1115  view browse http://www.stata.com/support/faqs/data/dropmiss.html
1116  * [2] How can I replace missing values with previous or
1117  *      following nonmissing values or within sequences?
1118  view browse http://www.stata.com/support/faqs/data/missing.html
1119
1120
1121
1122  *
1123  *--2.4.5 多重补漏分析 (multiple-imputation) -mi-
1124
1125  help mi          // stata11 的新功能
1126
1127  *--2.4.5.1 MI 简介
1128
1129  *--缺漏值的产生: 随机 v.s. 非随机
1130
1131  *--缺漏值的影响:
1132  *   若缺漏值是非随机的, 则相应的统计推断会存在严重偏误
1133  *   换言之, 非缺漏样本不能很好的体现母体特征
1134
1135  *--多重补漏的基本思想
1136
1137  *--多重补漏的特征
1138  *   [1] 不对缺漏值的产生机制做任何假设 (不假设其为 Random)
1139  *   [2] 采用 Bayesian 或 MCMC 模拟分析
1140
1141  *-----
1142  *--MI 分析的步骤
1143
1144  *--step1: 声明数据结构
1145  *
1146  * stata命令:
1147  *   mi set
1148  *   mi register
1149
1150  *--step2: 补漏估计 (imputation step) [-imputation model-]
1151  *   在给定假设下, 使用某种模型进行 M 次插值 (imputation)
1152  *
1153  * stata命令:
1154  *   mi impute
1155
1156  *--step3: 目标模型估计 (Pooling step)
1157  *   (completed-data analysis step) [-analysis model-]
1158  *
1159  *   利用第二步中的 M 组数据进行 M 次回归分析
1160  *   并将 M 次估计结果整合起来, 得到最终的结果
1161  *
1162  * stata命令:
1163  *   mi estimate
1164  *-----
1165
1166  *--参考资料
1167
1168  *--stata 手册 [MI]
1169  view browse help mi\_intro\_substantive //基本概念介绍
1170
1171  *--一个精辟的介绍
1172  view browse http://www.stat.psu.edu/~jls/mifaq.html
1173
1174
1175  *--2.4.5.2 实例分析
1176  *
1177  *-----实例开始-----
1178
1179  *--E1- 构建数据
1180
1181  *--Case-1-: 随机缺漏 (missing at random, MAR)
1182  sysuse auto, clear
1183  gen price_R = price
1184  set seed 13579
```

```
1185     gen random = uniform()
1186     sum random
1187     replace price_R = . if random>0.9
1188
1189     *-Case-2-: 非随机缺漏(missing not at random, MNAR)
1190     gen price_U = price
1191     replace price_U = . if price>11500
1192     sum price price_
1193     sum weight mpg length foreign if price_R ~=.
1194     sum weight mpg length foreign if price_U ~=.
1195
1196     save mi_auto, replace // 保存数据以备后续分析
1197
1198
1199     *-E2- 多重补漏分析
1200
1201     *-2-1- mi set style 基本设定
1202     view help mi_set##style
1203
1204     use mi_auto, clear
1205     mi set wide // 设定
1206
1207     *-2-2- mi register 声明包含缺漏值的变量
1208     view help mi_set##register
1209     mi register imputed price_R
1210     mi describe
1211
1212     *-2-3- mi impute method 补漏
1213     help mi impute
1214     mi impute regress price_R wei len mpg turn forei, ///
1215         add(20) rseed(1357)
1216
1217     *-补漏效果
1218     egen pav_R_im = rowmean(_*price_R)
1219     list price pav_R_im if price_R==., sep(0)
1220
1221     *-E3- mi estimate 估计模型
1222     mi estimate: logit foreign price_R wei mpg turn
1223     est store mi
1224     *-对比结果
1225     logit foreign price wei mpg turn, nolog noheader
1226
1227     *-E4-小结: 完整过程
1228     use mi_auto, clear
1229     mi set wide
1230     mi register imputed price_R
1231     mi impute regress price_R wei len mpg turn forei, add(20)
1232     mi estimate: logit foreign price_R wei mpg turn
1233
1234     *-----实例结束-----
1235
1236     *-说明: 对结果的详细解释, 参见 [MI]手册 p.46
1237
1238     *- (1) average RVI: average relative variance increase
1239     * 缺漏值的存在会导致模型的var增加
1240     * 该指标衡量了缺漏值的影响程度, RVI越小表明影响越小
1241     sysuse auto, clear
1242     replace price = . in 1/1 // 只有一个缺漏值
1243     mi set wide
1244     mi register imputed price
1245     mi impute regress price wei len mpg turn forei, add(20)
1246     mi estimate: logit foreign price wei mpg turn
1247
1248     *- (2) 估计过程的公式, 参见 [MI]手册, p.56
1249
1250
1251     *-2.4.5.3 MI impute regress 的假设条件
1252
1253     *-要求满足正态分布假设
1254
1255     use mi_auto, clear
1256
1257     gen ln_price_U = ln(price_U) // 对数转换
1258     sum price_U ln_price_U, d
```

```
1259
1260     mi set wide
1261     mi register imputed ln_price_U
1262
1263     mi impute regress ln_price_U wei len mpg turn forei, ///
1264             add(20) rseed(2468)
1265
1266     *-后续分析中需要使用 price_U, 而非 ln_price_U
1267     *-由于 price_U 是 ln_price_U 的函数,
1268     *-而 ln_price_U 是一个 "imputed variable",
1269     *-因此, price_U 是一个 "passive variable"
1270         mi register passive price_U
1271     qui mi passive: replace price_U = exp(ln_price_U) //返回
1272
1273     *-估计MI模型
1274     mi estimate: logit foreign price_U wei mpg turn
1275     * 对比真实数据
1276     logit foreign price wei mpg turn, nolog noheader
1277     * 对比直接删除数据的情形
1278     logit foreign price_U wei mpg turn, nolog noheader
1279
1280     *-说明: (1)主要的差别在于变量 -turn-
1281     *         (2)本例中, MI的结果更接近直接删除的结果
1282
1283
1284     *-2.4.5.4 其它补漏方法
1285
1286     *-mi impute pmm
1287         help mi impute pmm
1288     *-对模型设定和分布假设不敏感, 稳健性较高
1289
1290     *-mi impute logit
1291         help mi impute logit
1292     *-用于填补 {0/1} 变量的缺漏值
1293
1294     *-mi impute mlogit
1295         help mi impute mlogit
1296     *-用于填补 {0/1/2/3...} 序别变量的缺漏值
1297
1298     *-其它
1299         help mi impute
1300
1301
1302     *-2.4.5.5 假设检验
1303
1304         help mi estimate postestimation
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314     *      =====
1315     *      计量分析与STATA应用
1316     *      =====
1317
1318     *      主讲人: 连玉君 博士
1319
1320     *      单 位: 中山大学岭南学院金融系
1321     *      电 邮: arlionn@163.com
1322     *      主 页: http://blog.cnfol.com/arlion
1323
1324     *      ::第一部分::
1325     *      Stata 操作
1326     *      =====
1327     *      第二讲 数据处理
1328     *      =====
1329     *      2.5 离群值的处理
1330
1331
1332     *-----
```

```

1333  *--2.5 离群值的处理
1334  *-----
1335
1336      *      ==本节目录==
1337
1338      *      2.5.1 离群值的影响
1339      *      2.5.2 查找离群值
1340      *      2.5.3 离群值的处理
1341      *          2.5.3.1 删除
1342      *          2.5.3.2 对数转换
1343      *          2.5.3.3 缩尾处理
1344      *          2.5.3.4 截尾处理
1345
1346
1347      *      =本节命令=
1348      *      =====
1349      *      histogram, winsor, hadimvo, egen outside()
1350      *      qr, adjacent, fsreg, lv
1351      *      =====
1352
1353
1354      *
1355  *--2.5.1 离群值的影响
1356
1357      *--例：离群值对回归结果的影响
1358
1359      sysuse auto, clear
1360      histogram price
1361      count if price>13000
1362
1363      reg price weight length foreign
1364      est store r1
1365      reg price weight length foreign if price<13000
1366      est store r2
1367
1368      esttab r1 r2, mtitle("with" "without")
1369
1370      *--结论：虽然离群值只有4个，但对回归结果的影响却很大
1371
1372
1373      *
1374  *--2.5.2 查找离群值
1375
1376      *      -----
1377      *      基本概念
1378      *      -----
1379      *
1380      * 第25、50、75百分位上的数值分别称为第1、2、3四分位
1381      * 四分位间距(interquartile range): iqr = p75-p25
1382      * 上界(upper adjacent) = p75 + 1.5*iqr
1383      * 下界(lower adjacent) = p25 - 1.5*iqr
1384      *-----
1385
1386      *--adjacent- 命令
1387      sysuse auto, clear
1388      adjacent price
1389      adjacent price, by(foreign)
1390
1391      *--egenmore 提供的 outside() 函数
1392      egen out = outside(price)
1393      egen out2 = outside(price), factor(2)
1394      egen outby= outside(price), by(foreign) factor(2)
1395      list price out*
1396      keep if outby==. // 删除离群值
1397
1398      *--箱形图
1399      help graph box
1400      graph box price
1401      graph box price, by(foreign)
1402      graph box weight, by(foreign)
1403
1404
1405
1406      *

```

```
1407  *-2.5.3 离群值的处理
1408
1409      *-2.5.3.1 删除
1410
1411          sysuse auto, clear
1412          adjacent price, by(foreign)
1413          drop if (price>8814&foreign==0) | (price>9735&foreign==1)
1414
1415      *-or          // 需要提前安装-egenmore-相关命令
1416          sysuse auto, clear
1417          egen outby = outside(price), by(foreign) factor(2)
1418          drop if outby != .
1419
1420
1421  *-2.5.3.2 对数转换
1422
1423          sysuse nls88, clear
1424          gen ln_wage = ln(wage)
1425
1426          twoway (histogram wage,color(green))          ///
1427                  (histogram ln_wage,color(yellow))
1428
1429          sum wage ln_wage, d
1430
1431          graph box wage
1432          graph box ln_wage
1433
1434
1435  *-2.5.3.3 缩尾处理
1436
1437          sysuse nls88.dta, clear
1438          histogram wage
1439
1440      *-双边缩尾
1441          winsor wage, gen(wage_w2) p(0.025)
1442
1443      *-图示
1444          twoway (histogram wage,color(green))          ///
1445                  (histogram wage_w2,color(yellow)), ///
1446                  legend(label(1 "wage") label(2 "wage_winsor2"))
1447
1448      *-单边缩尾
1449          winsor wage, gen(wage_h) p(0.025) highonly
1450      *-图示
1451          twoway (histogram wage,color(green))          ///
1452                  (histogram wage_h,color(yellow)), ///
1453                  legend(label(1 "wage") label(2 "wage_winsorH"))
1454
1455
1456      *-若无法下载 winsor 命令，可以采用如下处理方法：
1457
1458          _pctile wage, percentile(1 99)
1459          replace wage = r(r1) if wage<r(r1)
1460          replace wage = r(r2) if wage>r(r2)
1461
1462      *-亦可采用 clip() 函数 (参见第2.1.2.6小节，第385行)
1463          gen wage_w = clip(wage, r(r1), r(r2))
1464          sum wage wage_w, detail
1465
1466
1467
1468  *-2.5.3.4 截尾处理
1469
1470          sysuse nls88, clear
1471          _pctile wage, percentile(1 99)
1472          return list
1473          drop if wage<r(r1) // 删除小于第1百分位的样本
1474          drop if wage>r(r2) // 删除大于第99百分位的样本
1475
1476      *-说明：
1477      * (1) 可以先绘制直方图，进而根据分布情况选择左截尾、
1478      *      右截尾还是双边截尾
1479      * (2) 相比于ln()处理和winsor处理，该处理会损失样本
1480      *      但对于大样本而言，该方法比较“干净”
```

```

1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493      *      =====
1494      *      计量分析与STATA应用
1495      *      =====
1496
1497      *      主讲人：连玉君 博士
1498
1499      *      单   位：中山大学岭南学院金融系
1500      *      电   邮：arlionn@163.com
1501      *      主   页：http://blog.cnfol.com/arlion
1502
1503      *      ::第一部分::
1504      *      Stata 操作
1505      *      =====
1506      *      第二讲  数据处理
1507      *      =====
1508      *      2.6 资料的合并和追加
1509
1510
1511 *-----
1512 *-2.6 资料的合并和追加
1513 *-----
1514
1515      *      ==本节目录==
1516
1517      *      2.6.1 横向合并：增加变量
1518      *          2.6.1.1 一对一合并
1519      *          2.6.1.2 多对一合并
1520      *          2.6.1.3 一对多合并
1521      *          2.6.1.4 一个例子
1522      *      2.6.2 横向关联：-joinby-
1523      *      2.6.3 纵向合并：追加样本
1524      *      2.6.4 大型数据的处理
1525      *      2.6.5 一些有用的外部命令
1526
1527
1528      *      =本节命令=
1529      *      =====
1530      *      -merge-  -joinby-  -append-
1531      *      =====
1532
1533
1534      cd `c(sysdir_personal)'Net_course_A\A2_data
1535
1536
1537      *
1538 *-2.6.1 横向合并：增加变量      -merge-
1539
1540      do L2_data_gr_merge.do
1541
1542 *-2.6.1.1 一对一合并  [ 1:1 ]
1543
1544      view help merge##ii
1545
1546 *-待合并的数据
1547      use merge_u.dta, clear
1548      browse
1549      use merge_m.dta, clear
1550      browse
1551
1552 *-合并方法：
1553      use merge_m.dta, clear
1554      merge 1:1 date using merge_u

```

```
1555
1556     *- _merge 变量的含义:
1557     *
1558     *   _merge==1   obs. from master data
1559     *   _merge==2   obs. from only one using dataset
1560     *   _merge==3   obs. from at least two datasets, master or using
1561
1562     *- 【说明】 在stata11以前, 横向合并的过程要复杂一些
1563     *           在合并前, 必须先依据 date 变量对两组数据进行排序
1564
1565     *-其它选项
1566
1567         help merge
1568
1569     *-keepusing(varlist) 选项 (仅合并部分数据)
1570         use merge_m.dta, clear
1571         merge 1:1 date using merge_u, keepusing(close)
1572
1573     *-generate() 选项 v.s. nogenerate 选项
1574         use merge_m.dta, clear
1575         merge 1:1 date using merge_u, gen(m1)
1576
1577     *-nolabel, nonotes 选项 (不拷贝被合并数据的"数字-文字对应表")
1578
1579     *-update 选项 (更新主数据集中的缺漏值)
1580     *-问题
1581         use merge_u.dta, clear
1582         browse
1583         use merge_m.dta, clear
1584         gen close = . // merge_u.dta 中也有该变量, 但取值不同
1585         browse
1586     *-合并方法
1587         merge 1:1 date using merge_u, update
1588         browse
1589
1590     *-replace 选项 ()
1591         use merge_m.dta, clear
1592         gen close = 0 // 注意, 此例中 close=0
1593         browse
1594         merge 1:1 date using merge_u, update
1595         browse // close=0 并未发生变化
1596
1597         drop _merge
1598         merge 1:1 date using merge_u, update replace // 正确做法
1599         browse
1600
1601
1602     *-2.6.1.2 多对一合并 [ m:1 ]
1603
1604         view help merge##mi
1605
1606     *-数据形态
1607         use GTA_FS.dta,clear // 上市公司财务资料
1608         browse
1609         use GTA_basic.dta,clear // 上市公司基本资料,只有 id 没有 year
1610         browse
1611
1612     *-合并方法
1613         use GTA_FS.dta, clear
1614         merge m:1 id using GTA_basic.dta, nogen
1615
1616
1617     *-2.6.1.3 一对多合并 [ 1:m ]
1618
1619         view help merge##im // 其实就是 m:1 的逆向操作
1620
1621     *-数据形态
1622         use GTA_FS.dta,clear // 上市公司财务资料
1623         browse
1624         use GTA_basic.dta,clear // 上市公司基本资料,只有 id 没有 year
1625         browse
1626
1627     *-合并方法
1628         use GTA_basic.dta, clear
```

```
1629     merge 1:m id using GTA_FS.dta, nogen
1630     browse
1631     order id year
1632     tsset id year
1633
1634
1635     *-2.6.1.4 一个例子
1636
1637     *-数据形态:
1638
1639         use GTA_FS.dta, clear      // 上市公司财务资料
1640         browse
1641         use GTA_GC.dta, clear      // 上市公司治理结构信息
1642         browse
1643         use GTA_div.dta, clear     // 上市公司股利分配、增发配股
1644         browse
1645         use GTA_basic.dta, clear   // 上市公司基本资料, 只有 id 没有 year
1646         browse
1647
1648     *-合并上述数据
1649
1650     *-基本思路:
1651     * (1) 先根据 id year 把前三个数据一次性合并起来: [1:1]
1652     * (2) 再根据 id      把GTA_basic数据合并进来      [m:1]
1653
1654         use GTA_FS.dta, clear
1655         merge 1:1 id year using GTA_GC.dta , nogen
1656         merge 1:1 id year using GTA_div.dta, nogen
1657         merge m:1 id      using GTA_basic, nogen
1658
1659         tsset id year
1660         save GTA_merge.dta, replace    // 保存合并后的数据
1661
1662
1663
1664     *
1665     *-2.6.2 横向关联      -joinby-
1666
1667     *-应用背景: 我们只需要保留两份数据中有对应关系的数据
1668         use child.dta, clear
1669         list, sepby(family_id)
1670         sort family_id
1671         save, replace
1672         use parent.dta, clear
1673         sort family_id      // 这一步很重要!
1674         list, sepby(family_id)
1675         joinby family_id using child.dta
1676         sort family_id parent_id child_id
1677         order family_id parent_id child_id
1678         list, sepby(fam)
1679
1680
1681     *-与-merge- 命令的对比
1682         use parent, clear
1683         sort fam*id
1684         merge m:m fam using child
1685         sort family_id parent_id child_id
1686         order family_id parent_id child_id
1687         list, sepby(fam)
1688
1689
1690     *
1691     *-2.6.3 纵向合并: 追加样本      -append-
1692
1693         do L2_data_gr_append.do    // 基本原理
1694
1695     *-两个数据库中的"同名变量"会自动对应累叠
1696
1697     *-数据形态
1698         use append_m.dta, clear
1699         browse
1700         tsset date
1701         use append_u.dta, clear
1702         browse
```



```
1703         tsset date
1704
1705         use append_m.dta, clear
1706         append using append_u.dta
1707         browse
1708         tsset date
1709
1710     *-generate() 选项
1711     use append_m.dta, clear
1712     append using append_u.dta, gen(append_id)
1713     browse
1714
1715     *-nolabel, nonotes 选项
1716
1717     *- 几个注意事项:
1718     *   (1) 两个数据库中的变量名称要相同
1719     *       PRICE 和 price 是不同的变量
1720     *   (2) 两个数据库中的同名变量要具有相同的存储类型
1721     *       同为文字变量或同为数值变量
1722
1723     *-问题(2) 示例: 两个数据集中的变量存储类型不同
1724
1725     sysuse auto, clear
1726     keep foreign
1727     keep if !foreign      // keep if foreign==0
1728     save auto_dom, replace // 数据集1: 变量foreign为数值类型
1729
1730     sysuse auto, clear
1731     keep foreign
1732     keep if foreign
1733     rename foreign s
1734     gen foreign = "foreign" if s
1735     drop s
1736     save auto_for, replace // 数据集2: 变量foreign为文字类型
1737
1738     use auto_dom, clear
1739     browse
1740     append using auto_for
1741     browse
1742
1743     *-更换合并的先后顺序: 于事无补!
1744     use auto_for, clear
1745     browse
1746     append using auto_dom
1747     browse
1748
1749     *-增加 -force- 选项, 并无实质性改进
1750     use auto_dom, clear
1751     append using auto_for, force
1752     browse
1753
1754     *-如何解决?
1755     use auto_for, clear
1756     rename foreign ss
1757     gen byte foreign=1
1758     drop ss
1759
1760     append using auto_dom
1761     browse
1762
1763
1764
1765     *
1766     *-2.6.4 大型数据的处理
1767
1768     *-范例: 构建上市公司研究数据库
1769
1770     *-数据特征描述:
1771     *   *-样本区间: 1990-2008
1772     *   *-指标范围: 上市公司财务资料、基本信息、治理信息、股利分配、增发配股等
1773     *   *-数据来源: CCER、GTA (每个数据库都分成了若干个字库)
1774
1775     *-任务: 把不同来源的各项数据合并起来, 整合成一个完成的数据集合
1776
```

---

```
1777      doedit GTA_2008.do                // 该文件历时 3 天完成
1778
1779      shellout 连玉君_GTA2008说明书.pdf // 说明书
1780
1781
1782
1783      *
1784      *-----2.6.5 一些有用的外部命令-----
1785
1786      * -nearmrg- performs nearest match merging of two datasets
1787
1788      * -mmerge- 一个更灵活的合并命令
1789
1790      * -reclink- module to probabilistically match records
1791
1792      * -xmerge-, -xmerged-, -nmerge- 批量合并命令
1793
1794      * -mergein- 自动排序后合并 // stata11 用户已经不需要了
1795
1796      * -mergedct- 直接将 .raw 文件合并至已有 .dta 文件中
1797
1798      * -addinby- 对-merge-做了改进，不会生成_merge，并检查合并的配对情况
1799
1800      * -kcountry- standardize country names across various datasets (SJ8-3)
1801
1802      * -tvc_merge- merge two files which each contain time varying covariates
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813      *      =====
1814      *      计量分析与STATA应用
1815      *      =====
1816
1817      *      主讲人：连玉君 博士
1818
1819      *      单 位：中山大学岭南学院金融系
1820      *      电 邮：arlionn@163.com
1821      *      主 页：http://blog.cnfol.com/arlion
1822
1823      *      ::第一部分::
1824      *      Stata 操作
1825      *      =====
1826      *      第二讲 数据处理
1827      *      =====
1828      *      2.7 重新组合样本
1829
1830
1831      *-----
1832      *-2.7 重新组合样本
1833      *-----
1834
1835      *      ==本节目录==
1836
1837      *      2.7.1 样本的转置
1838      *      2.7.2 数据的横纵变换
1839      *      2.7.3 样本的交叉组合
1840      *      2.7.3.1 -fillin- 命令
1841      *      2.7.3.2 -cross-命令
1842      *      2.7.4 样本的堆砌
1843
1844
1845      *      =本节命令=
1846      *      =====
1847      *      -xpose- -reshape- -fillin- -stack- -cross-
1848      *      =====
1849
1850
```

---

```
1851 *
1852 *-2.7.1 样本的转置 -xpose-
1853
1854     use d205.dta, clear
1855     list v1-v7
1856
1857     xpose, clear           // clear 选项必须加
1858
1859     rename v1 date
1860     rename v2 open
1861     rename v3 close
1862     save d204.dta, replace // 另存一份数据，因为原始数据已被修改
1863
1864
1865 *
1866 *-2.7.2 数据的横纵变换 -reshape-
1867
1868     *- 问题描述
1869         shellout reshape0.txt // -xpose- 命令不奏效
1870
1871     *- wide --> long
1872         use reshape1.dta, clear
1873         list
1874         reshape long inc ue, i(id) j(year) // sex 不发生变化，无需转换
1875                                             // j() 选项中填写新的变量名称
1876         list, sepby(id)
1877         replace year = real("19" + string(year))
1878         list, sepby(id)
1879
1880     *- long --> wide
1881         reshape wide inc ue, i(id) j(year)
1882
1883 *-示例：
1884 *-World Development Indicator 转换
1885     view browse ///
1886         http://dss.princeton.edu/online_help/analysis/reshape_wdi.htm
1887
1888 *-进一步的参考资料
1889     view browse ///
1890         http://www.stata.com/support/faqs/data/reshape3.html#
1891
1892
1893
1894 *
1895 *-2.7.3 样本的交叉组合 -fillin- -cross-
1896
1897     *-2.7.3.1 -fillin- 命令
1898
1899     *-例1: Nlogit模型中的选择行为
1900     *-see SJ 5-1, p.135, Filling in the gaps
1901     clear
1902     input id choice
1903         1    -1
1904         2     1
1905         3     0
1906         4    -1
1907     end
1908     list
1909     fillin id choice
1910     list, sepby(id)
1911     *-可见，-fillin-的作用在于对原始观察值进行“组合”
1912
1913     *-例2: 建立一个Panel Data 的 id, year
1914     * i = 1,2,20; t = 2000-2008
1915     clear
1916     set obs 20
1917     gen id = _n
1918     gen tt = _n + 2000
1919     list
1920
1921     fillin id tt
1922
1923     list, sepby(id)
1924     drop if (tt>2008)
```

```
1925         drop _fillin
1926         list, sepby(id)
1927
1928
1929     *-2.7.3.2 -cross-命令
1930
1931         clear
1932         input str6 sex
1933             male
1934             female
1935         end
1936         save sex, replace
1937
1938         clear
1939         input age
1940             20
1941             40
1942             50
1943         end
1944
1945         cross using sex.dta
1946
1947         list, sep(0)
1948
1949     *-亦可用 -fillin- 命令加以解决
1950         clear
1951         input str6 sex      age
1952             male      20
1953             female    40
1954             .         60
1955         end
1956         fillin sex age
1957         list, sep(0)
1958         drop if sex == "."
1959         list, sep(0)
1960
1961     *-说明: -cross-命令较少使用, SJ 6-1, p.147 提供了一个妙用
1962
1963     *-相关阅读: 第 16 楼
1964         view browse ///
1965             http://www.pinggu.org/bbs/thread-436189-1-1.html
1966
1967
1968
1969     *
1970     *-2.7.4 样本的堆砌
1971
1972     *-简介
1973     * 基本思想: 向量化
1974
1975         use stackxmpl.dta, clear
1976         list
1977         stack a b c d, into(x) clear      // 堆砌成一行
1978         list, sepby(_stack)
1979
1980         use stackxmpl.dta, clear
1981         list
1982         stack a b c d, into(x1 x2) clear  // 堆砌成两行
1983         list, sepby(_stack)
1984
1985
1986     *-范例
1987
1988     *-原始样本
1989         use stack_lnxp.dta, clear
1990         browse
1991
1992     *-堆砌样本
1993         stack china-unitedstates, into(lnxp) clear
1994         browse
1995
1996     *-进一步美化 I: 增加年度变量
1997         rename _stack id
1998         egen year = seq(), from(1998) to(2007)
```

```

1999      order id year
2000      tsset id year
2001      save lnexp_temp, replace // 后续还要做进一步处理
2002
2003      *-进一步美化 II: 增加国名
2004
2005      *-取出国名
2006      use stack_lnexp.dta, clear
2007      drop lnexp
2008      browse
2009
2010      mkmat _all, mat(a) // 矩阵的列名(国名)就是我们需要的
2011      mat list a
2012
2013      global vn: colnames a // 将国名存储于暂元 vn 中
2014      dis "$vn"
2015
2016      *-处理国名
2017      use lnexp_temp, clear
2018      rename id id123
2019      gen id = ""
2020      local i = 1
2021      foreach nn of global vn{ // 1 --> "china"
2022          qui replace id="`nn'" if id123=="`i++'"
2023      }
2024
2025      sencode id, replace // id 是一个文字变量，现转化为数值变量
2026      labelbook
2027
2028      order id year id123
2029      tsset id year
2030
2031      browse
2032
2033      save lnexp.dta, replace
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043      *      =====
2044      *      计量分析与STATA应用
2045      *      =====
2046
2047      *      主讲人: 连玉君 博士
2048
2049      *      单 位: 中山大学岭南学院金融系
2050      *      电 邮: arlionn@163.com
2051      *      主 页: http://blog.cnfol.com/arlion
2052
2053      *      ::第一部分::
2054      *      Stata 操作
2055      *      =====
2056      *      第二讲 数据处理
2057      *      =====
2058      *      2.8 文字变量的处理
2059
2060
2061      *-----
2062      *-2.8 文字变量的处理
2063      *-----
2064
2065      *      ==本节目录==
2066
2067      *      2.8.1 将文字转换为数字
2068      *          2.8.1.1 以文字类型存储的数字之转换
2069      *          2.8.1.2 纯文字类别变量之转换
2070      *      2.8.2 将数字转换成文字
2071      *      2.8.3 文字样本值的分解
2072      *      2.8.4 处理文字的函数

```

```

2073 *      2.8.4.1 文字函数简介
2074 *      2.8.4.2 例-1-: 上市公司日期、行业代码和所在地的处理
2075 *      2.8.4.3 例-2-: 银企关系数据中银行名称的提取
2076 *      2.8.4.4 例-3-: 处理不规则的日期
2077
2078
2079 *      =本节命令=
2080 *      =====
2081 *      -destring- -encode- -sdecode- -real()-
2082 *      -tostring- -decode- -rdecode- -redecodeall-
2083 *      -substr()- -strmatch()- -split-
2084 *      -regexm()- -regexs()- -regexr()-
2085 *      =====
2086
2087
2088 *
2089 *-2.8.1 将文字转换为数字-
2090
2091 *-2.8.1.1 以文字类型存储的数字之转换 -destring-
2092
2093 *- 说明:
2094 *- 从 .txt 文档中读入数值变量之所以会以文字值方式存储,
2095 *- 主要原因是变量中可能包含了如下特殊符号:
2096 *- 金额`$'、逗号`,`、斜线`/'、百分比`%'、破折号`-'
2097
2098 shellout d202.txt
2099 insheet using d202.txt, clear names
2100 save d202.dta, replace
2101 des
2102 sum
2103
2104 *-说明: 虽然 code 变量由数字组成, 但其类型为 str7, 即为文字型变量
2105 *      leverage, size, date 都存在类似的问题
2106
2107 use d202.dta, clear
2108 destring code, gen(code1) ignore(" ")
2109 destring leverage, gen(lev) percent
2110 destring year date size lev, replace ignore("-/,%")
2111
2112
2113 *-2.8.1.2 纯文字类别变量之转换 -encode-, -rdecode-
2114
2115 use d202.dta, clear
2116 encode gov, gen(gov1)
2117 labelbook
2118
2119 *-说明:
2120 *
2121 *- encode 命令会自动根据文字类别编号,
2122 *      并设定相应的[数字-文字对应表]
2123 *
2124 * [数字-文字对应表] 按“字母顺序排列”
2125 *      sysuse auto, clear
2126 *      encode make, gen(make_num)
2127 *      order make make_num
2128 *      labelbook
2129
2130 *_Q: 如何根据出现的先后顺序设定[数字-文字对应表]? [-sdecode-]
2131
2132 *- 缺陷:
2133 * (1) 没有 -replace- 选项 [-rdecode-]
2134 * (2) 每次只能转换一个变量, 无法实现批量转换 [-redecodeall-]
2135
2136
2137 *-rdecode- 命令: 附加 replace 选项
2138
2139 use d202.dta, clear
2140 rencode gov, replace
2141 labelbook
2142
2143 *-说明:
2144 * (1) 与该命令功能相似的还有 -sencode- 命令
2145 * (2) 使用 -redecodeall- 命令可以同时转换多个变量
2146

```

```
2147
2148     *-encode 命令与 -destring- 的区别
2149     *
2150     *-(1) 若数字“误存”为文字型变量，使用-destring-命令或 real() 函数
2151     *
2152     *-(2) 若观察值均为“文字值”，则需使用-encode-或-recode-命令，
2153     *      这些命令会自动产生【数字-文字对应表】
2154
2155
2156 *
2157 *-----2.8.2 将数字转换成文字-----
2158
2159     *-某些情况下，先把数字转换成文字，
2160     *-然后利用处理文字的函数进行处理比较方便
2161
2162     *-eg01: 年月日的组合
2163
2164         use tostring.dta, clear
2165         tostring year day, replace
2166         gen date = year + "-" + month + "-" + day
2167         gen edate = date(date, "YMD")
2168         format edate %td
2169         browse
2170
2171
2172     *-eg02: 年月日的分离
2173
2174         use tostring2.dta, clear
2175         browse
2176         tostring date_pub, gen(date1)
2177         gen year = substr(date1, 1, 4)
2178         gen month = substr(date1, 5, 2)
2179         gen day = substr(date1, 7, 2)
2180         browse
2181         destring year month day, replace
2182         browse
2183
2184     *-说明:
2185     *      -decode-命令的缺陷同样在于没有 -replace- 选项，
2186     *      可以采用外部命令 -rdecode- 或 -sdecode- 代替之。
2187
2188
2189 *
2190 *-----2.8.3 文字样本值的分解-----
2191
2192         use d202.dta, clear
2193         list
2194
2195     *-从 year 变量中提取年份    -split-
2196         split year, parse(-)
2197         order year year1 year2
2198         list
2199         browse
2200         gen year3 = real(year1)    // year1中全为数值，但以文字型存储
2201
2202         * destring year1, replace // 另一种方式
2203
2204     *-从 date 变量中提取年份、月度和日期,并转化为数值
2205         split date, parse(/) destring ignore("/")
2206         order date date*
2207         edit
2208
2209     *-Also see 一个比较复杂的例子
2210         view browse http://www.stata.com/support/faqs/data/splitstr.html
2211
2212
2213 *
2214 *-----2.8.4 处理文字的函数-----
2215
2216         help string functions
2217
2218     *-2.8.4.1 文字函数简介
2219
2220         dis lower("AbCDef")
```

```

2221     dis length("price weight length mpg")
2222     dis wordcount("price weight length mpg") //统计变量的个数
2223     dis proper("mR. joHn a. sMitH") // 规整人名
2224     dis strmatch("C51", "C")
2225     dis strmatch("C51", "C*") // 寻找制造业公司
2226     dis trim(" I love STATA ") // 去掉两端的空格
2227     dis ltrim(" I love STATA ") // 去掉左边的空格
2228     dis rtrim(" I love STATA ") // 去掉右边的空格
2229     dis itrim(" I love STATA ") // 去掉中间的空格
2230     dis itrim("内 蒙 古 自 治 区") // 去掉中间的空格，不奏效？
2231     dis substr("内 蒙 古 自 治 区", " ", "", .)
2232     *-释义：
2233     * substr(s, s1, s2, n)
2234     * s 原始字符串
2235     * s1 “将被替换”的字符串
2236     * s2 “替换成”的字符串
2237     * n 前n个出现的目标字符，若为“.”则表示全部替换
2238     dis substr("内 蒙 古 自 治 区", " ", "", 1)
2239     dis substr("内 蒙 古 自 治 区", " ", "", 3)
2240
2241     *-说明：上述函数都可以用于 -generate- 命令来生成新的变量
2242
2243
2244     *-2.8.4.2 例-1-：上市公司日期、行业代码和所在地的处理
2245
2246     *-a 待处理的数据
2247     shellout d203.txt
2248     insheet date sic location using d203.txt, clear
2249     save d203.dta, replace
2250     browse
2251
2252     *-b 从date中分离出年、月、日
2253     gen year = int(date/10000)
2254     tostring date, gen(date1)
2255     gen year1 = substr(date1,1,4)
2256     gen year2 = real(year1)
2257     gen month = substr(date1,5,2)
2258     gen month1= real(month)
2259     gen day = substr(date1,7,2)
2260     gen day1 = real(day)
2261     browse
2262
2263     *-更为简洁的命令
2264     use d203.dta, clear
2265     gen sdate = string(date,"%10.0g") // help string()
2266     gen year = real(substr(string(date,"%10.0g"), 1, 4))
2267     gen month = real(substr(sdate, 5, 2))
2268     gen day = real(substr(sdate, 7, 2))
2269     browse
2270     drop sdate
2271
2272     *-c 从行业大类sic中分离出行业门类
2273     gen sic_men0 = substr(sic,1,1)
2274     encode sic_men0, gen(sic_men)
2275     tab sic_men
2276     label list sic_men
2277
2278     *-d 从地点中分离出省份和城市
2279     use d203.dta,clear
2280     list
2281     gen province1 = substr(location, 1,2)
2282     gen city1 = substr(location, 4,4)
2283     list location province1 city1
2284     gen province2 = word(location, 1)
2285     gen city2 = word(location, 2)
2286     list location province1 city1 province2 city2
2287
2288     *-注意：每个英文字母占一位，但每个中文字符占两位
2289
2290
2291
2292     *-2.8.4.3 例-2-：银企关系数据中银行名称的提取
2293
2294     *-数据描述

```



```
2295     use bankname.dta, clear
2296     tab objnm
2297     list in 1/15
2298
2299     *-任务：提取出关联银行总部的名称
2300     keep in 1/15
2301     gen bank = objnm
2302     replace bank="中国农业银行" if strmatch(bank,"*农业银行*")
2303     replace bank="招商银行" if strmatch(bank,"*招商*")
2304     replace bank="中国银行" if strmatch(bank,"*中国银行*")
2305     replace bank="中国工商银行" if strmatch(bank,"*工商*")
2306     replace bank="兴业银行" if strmatch(bank,"*兴业*")
2307     replace bank="光大银行" if strmatch(bank,"*光大*")
2308     replace bank="交通银行" if strmatch(bank,"*交通*")
2309     replace bank="北京银行" if strmatch(bank,"*北京*")
2310     compress
2311     browse
2312
2313
2314     *-2.8.4.4 例-3-：处理不规则的日期
2315
2316     *- regexm(), regexs(), regexr() 函数
2317
2318     help regexm()
2319
2320     *-基本语法规则
2321     view browse http://www.stata.com/support/faqs/data/regex.html
2322
2323     *-示例：处理不规则的日期
2324     clear
2325     input str18 date
2326         20jan2007
2327         16June06
2328         06sept1985
2329         21june04
2330         4july90
2331         9jan1999
2332         6aug99
2333         19august2003
2334     end
2335
2336     *-如何规整之？
2337     gen day = regexs(0) if regexm(date, "^[0-9]+")
2338     gen month = regexs(0) if regexm(date, "[a-zA-Z]+")
2339     gen year = regexs(0) if regexm(date, "[0-9]*$")
2340     browse
2341     replace year = "20"+regexs(0) if regexm(year, "^[0][0-9]$")
2342     replace year = "19"+regexs(0) if regexm(year, "^[1-9][0-9]$")
2343     gen date2 = day+month+year
2344     browse
2345
2346     *-释义：
2347     * (1) "^[0-9]+"
2348     * ^ 表示字符串的开头部分
2349     * [0-9] 表示属于自然数0-9的任何一个
2350     * + 表示有至少一个对象符合匹配条件(*任何一个；?只有一个)
2351     * (2) "[a-zA-Z]+"
2352     * [a-zA-Z] 表示阿拉伯字母中的a-z或A-Z
2353     * (3) "[0-9]*$"
2354     * $ 表示字符串的结尾部分
2355
2356     *-更多示例：(1) 如何从地址中提取“邮编”？
2357     * (2) 如何规整人名？
2358     view browse http://www.ats.ucla.edu/stat/stata/faq/regex.htm
2359     view browse http://www.stata.com/support/faqs/data/regex.html
2360
2361
2362     *-Also see:
2363
2364     * Cox, N., 2002, Speaking Stata: On getting functions to do the work,
2365     * STATA JOURNAL, 2 (4): 411-427. (p.414)
2366
2367     * 外部命令 -egenmore- 提供了大量的文字处理函数，可供参考
2368
```

```
2369      help egenmore
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379      *      =====
2380      *      计量分析与STATA应用
2381      *      =====
2382
2383      *      主讲人：连玉君 博士
2384
2385      *      单   位：中山大学岭南学院金融系
2386      *      电   邮：arlionn@163.com
2387      *      主   页：http://blog.cnfol.com/arlion
2388
2389      *      ::第一部分::
2390      *      Stata 操作
2391      *      =====
2392      *      第二讲 数据处理
2393      *      =====
2394      *      2.9 类别变量的分析
2395
2396
2397      *-----
2398      *-2.9 类别变量的分析
2399      *-----
2400
2401      *      ==本节目录==
2402
2403      *      2.9.1 类别数的统计
2404      *      2.9.2 交叉类别变量的生成
2405      *      2.9.3 分组统计量
2406      *      2.9.3.1 单层分组统计量
2407      *      2.9.3.2 二层次和三层次分组统计量
2408      *      2.9.3.3 多层次分组统计量
2409      *      2.9.4 计算分组统计量的其它方法
2410      *      2.9.4.1 -egen-命令
2411      *      2.9.4.2 转换原资料为分组统计量：-collapse-命令
2412      *      2.9.5 图示分组统计量
2413      *      2.9.5.1 柱状图
2414      *      2.9.5.2 箱形图
2415
2416
2417      *      =本节命令=
2418      *      =====
2419      *      -tab- -distinct- -xgroup- -bysort-
2420      *      -tabstat- -collapse- -graph bar- -graph box-
2421      *      =====
2422
2423
2424      *
2425      *-2.9.1 类别数的统计
2426
2427      *-简单方法：-tab- 命令
2428
2429      sysuse nlsw88, clear
2430      tab race
2431      tab occupation // 局限：无法直接看到类别数目
2432
2433
2434      *-统计非重复值的个数
2435
2436      distinct occupation
2437      ret list
2438      distinct married race
2439      distinct married race, joint // 组合个数
2440      distinct married race occupation, joint
2441
2442
```

```

2443 *
2444 *-2.9.2 交叉类别变量的生成
2445
2446 sysuse nlsw88, clear
2447 tab race
2448 tab married
2449
2450 *-Q: 如何生成一个新的类别变量, 取值为1-6, 是race和married的组合
2451
2452 xgroup race married, gen(race_marr)
2453     browse race married race_marr in 1/20
2454 xgroup race married, gen(race_marr2) label lname(race_marr_lab)
2455     browse race married race_marr in 1/20
2456     label list race_marr_lab
2457
2458 *-说明:
2459 * (1) 可同时基于多个类别变量生成它们的组合类型;
2460 * (2) 基于新生成的类别变量, 可以进一步创建虚拟变量
2461 * (3) 参见 -xi-, -fvvarlist- 帮助文件 (2.1.2小节)
2462
2463
2464
2465 *
2466 *-2.9.3 分组统计量
2467
2468 *-2.9.3.1 单层分组统计量
2469
2470 *-bysort, sum
2471     sysuse nlsw88.dta, clear
2472     bysort race: sum wage
2473
2474 *-tabstat 命令
2475     tabstat wage, by(race) stat(mean sd med min max)
2476     tabstat wage hours ttl_exp, by(race)          ///
2477             stat(n mean sd med min max)          ///
2478             format(%6.2f) columns(statistics)
2479
2480 *-tabulate 命令
2481     tabulate industry
2482     tab industry, sort    // 可简写为 -tab-
2483     tab industry, summarize(wage)
2484
2485
2486 *-2.9.3.2 二层次和三层次分组统计量
2487
2488     bysort race married: sum wage
2489     bysort race married: tabstat wage,          ///
2490             by(union) s(n mean sd p50 min max)
2491     tabstat wage, by(race married union)        ///
2492             s(n mean sd p50 min max) // 错误方式
2493     bysort race married: tab union, sum(wage)
2494
2495
2496 *-2.9.3.3 多层次分组统计量
2497
2498 *-基本架构: table var1 var2 var3, by(var4) contents(...)
2499
2500     table race married union,          ///
2501             by(collgrad) c(mean wage) format(%4.2f)
2502     table race married union,          ///
2503             by(collgrad) c(mean wage freq) format(%4.2f)
2504
2505
2506 *
2507 *-2.9.4 计算分组统计量的其它方法
2508
2509 *-2.9.4.1 egen 命令
2510
2511     bysort industry: egen wage_ind = mean(wage)
2512     bysort industry: egen wage_p50 = pctlile(wage), p(50)
2513     list wage indust wage_ind wage_p50 in 1/30
2514
2515
2516 *-2.9.4.2 转换原资料为分组统计量: collapse 命令

```

```

2517
2518     help collapse
2519
2520     *-语法: collapse (统计量1) 新变量名=原变量名 (统计量2) ...
2521
2522     sysuse nlsw88.dta, clear
2523     collapse (mean) wage hours          ///
2524               (count) n_w=wage n_h=hours, ///
2525               by(industry)
2526     browse
2527
2528     sysuse nlsw88.dta, clear
2529     collapse (mean) wage hours          ///
2530               (count) n_w=wage n_h=hours, ///
2531               by(industry race)
2532     browse
2533
2534     * 几点说明:
2535     *   (1) 经常保存do文档, 但不要轻易选择保存数据文件
2536     *   (2) by() 选项是必填选项, 不可省略
2537
2538     * collapse 后, 原始变量的标签会丢失, 处理方法如下:
2539     view browse ///
2540     "http://www.stata.com/support/faqs/data/variables.html#"
2541
2542
2543
2544     *
2545     *-2.9.5 图示分组统计量
2546
2547     *-2.9.5.1 柱状图
2548
2549     *-纵向柱状图
2550     sysuse nlsw88.dta, clear
2551     graph bar (mean) wage, over(smsa) over(married) over(collgrad)
2552     do L2_data_gr_bar1.do // 更完整的图示
2553     doedit L2_data_gr_bar1.do
2554     *- 说明: over() 选项的呈现顺序是从内到外
2555
2556     *-横向柱状图
2557     graph hbar (mean) hours, over(union) over(industry)
2558     *-note: over() 选项的顺序决定了分组的层次关系,
2559     graph hbar (mean) hours, over(union) over(industry) asyvars
2560     //asyvars-把第一个over()选项中的变量视为纵轴
2561     graph hbar (mean) hours, over(union) over(married) ///
2562     over(race) percent asyvars
2563
2564     *-多变量柱状图
2565     graph bar wage hours, over(race) over(married)
2566     graph bar wage hours, over(race) over(married) stack
2567
2568     *-over() 选项的子选项
2569     graph bar wage hours, stack          ///
2570     over(race, relabel(1 "白人" 2 "黑人" 3 "其他")) ///
2571     over(married, relabel(1 "单身" 2 "已婚"))      ///
2572     legend(label(1 "工资水平") label(2 "工作时数"))
2573
2574
2575     *-2.9.5.2 箱形图
2576
2577     *-箱形图能较清晰的呈现各组样本值的分布情况
2578
2579     sysuse nlsw88, clear
2580
2581     graph box wage, over(race)
2582     graph box hours, over(race) over(married)
2583     graph box hours, over(race) over(married) nooutsides
2584
2585
2586     *-Also see:
2587     *
2588     * Cox, N., 2003, Speaking Stata: Problems with tables, Part I,
2589     * SJ, 3(3):309-324.
2590     * Cox, N., 2003, Speaking Stata: Problems with tables, Part II,

```

```

2591      *      SJ,3(4):420-439.
2592      *      Cox,N., D.City,2007,Stata tip 52:
2593      *      Generating composite categorical variables,
2594      *      SJ,7(4):582-583. (复杂类别变量的产生)
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604      *      =====
2605      *      计量分析与STATA应用
2606      *      =====
2607
2608      *      主讲人：连玉君 博士
2609
2610      *      单 位：中山大学岭南学院金融系
2611      *      电 邮：arlionn@163.com
2612      *      主 页：http://blog.cnfol.com/arlion
2613
2614      *      ::第一部分::
2615      *      Stata 操作
2616      *      =====
2617      *      第二讲 数据处理
2618      *      =====
2619      *      2.10 时间序列资料的处理
2620
2621
2622      *-----
2623      *-2.10 时间序列资料的处理
2624      *-----
2625
2626      *      ==本节目录==
2627
2628      *      2.10.1 简介
2629      *      2.10.1.1 声明时间序列：tsset 命令
2630      *      2.10.1.2 检查是否有断点
2631      *      2.10.1.3 填充缺漏的日期
2632      *      2.10.1.4 追加样本
2633      *      2.10.2 时序变量的生成
2634      *      2.10.2.1 滞后项、超前项和差分项
2635      *      2.10.2.2 产生增长率变量：对数差分
2636      *      2.10.2.3 日期变量的处理
2637
2638
2639      *      =本节命令=
2640      *      =====
2641      *      -tsset- -tsreport- -tsappend- -tsvarlist-
2642      *      -
2643      *      =====
2644
2645
2646
2647      *
2648      *-2.10.1 简介
2649
2650      *-2.10.1.1 声明时间序列：tsset 命令
2651
2652      use gnp96.dta, clear
2653      list in 1/20
2654      gen Lgnp = L.gnp    // 错误
2655
2656      tsset date
2657      list in 1/20
2658      gen Lgnp = L.gnp
2659      *-说明：若希望清除时间标示，可采用
2660      tsset, clear
2661
2662
2663      *-2.10.1.2 检查是否有断点
2664

```

```
2665     use gnp96.dta, clear
2666     tsset date
2667     tsreport, report
2668     drop in 10/10
2669     list in 1/12
2670     tsreport, report
2671     tsreport, report list // 列出存在断点的样本信息
2672
2673
2674     *-2.10.1.3 填充缺漏的日期
2675
2676     list in 1/12           // 缺少 1969q2
2677     tsfill                // 填充之
2678     tsreport, report list
2679     list in 1/12          // 参见 2.4 小节
2680
2681
2682     *-2.10.1.4 追加样本
2683
2684     use gnp96.dta, clear
2685     tsset date
2686     list in -10/-1
2687     sum
2688     tsappend , add(5)      // 追加5个观察值
2689     list in -10/-1
2690     sum
2691
2692     *-应用：样本外预测
2693     reg gnp96 L.gnp96
2694     predict gnp_hat
2695     list in -10/-1
2696
2697
2698     *
2699     *-2.10.2 时序变量的生成
2700
2701     *-2.10.2.1 滞后项、超前项和差分
2702
2703     help tsvarlist
2704
2705     use gnp96.dta, clear
2706     tsset date
2707
2708     gen Lgnp = L.gnp96     // 一阶滞后
2709     gen L2gnp = L2.gnp96   // 二阶滞后
2710     gen Fgnp = F.gnp96
2711     gen F2gnp = F2.gnp96
2712     gen Dgnp = D.gnp96
2713     gen D2gnp = D2.gnp96
2714     list in 1/10
2715     list in -10/-1
2716
2717
2718     *-2.10.2.2 产生增长率变量：对数差分
2719
2720     gen lngnp = ln(gnp96)
2721     tsset date
2722     gen growth = D.lngnp
2723     gen growth2 = (gnp96-L.gnp96)/L.gnp96
2724     gen diff = growth - growth2
2725     list date gnp96 lngnp growth* diff in 1/10
2726
2727
2728     *-2.10.2.3 日期变量的处理
2729
2730     help dates_and_times
2731
2732     *- 参见 stata高级视频 B6_TimeS
2733     *- Also see
2734     *   Cox, N., D. City, 2006,
2735     *       Speaking Stata: Time of day, SJ, 6(1): 124-137.
2736
2737
2738
```

```

2739
2740
2741
2742
2743
2744
2745      *      =====
2746      *      计量分析与STATA应用
2747      *      =====
2748
2749      *      主讲人：连玉君 博士
2750
2751      *      单 位：中山大学岭南学院金融系
2752      *      电 邮：arlionn@163.com
2753      *      主 页：http://blog.cnfol.com/arlion
2754
2755      *      ::第一部分::
2756      *      Stata 操作
2757      *      =====
2758      *      第二讲 数据处理
2759      *      =====
2760      *      2.11 面板资料的处理(I)
2761
2762
2763 *-----
2764 *-2.11 面板资料的处理(I)
2765 *-----
2766
2767      *      ==本节目录==
2768
2769      *      2.11.1 声明面板资料：xtset 命令
2770      *      2.11.2 公司数目和年度的统计
2771      *      2.11.2.1 面板资料的基本描述：xtdes 命令
2772      *      2.11.2.2 记录面板的资料形态：xtpattern 命令
2773      *      2.11.2.3 统计公司数目：panels 命令
2774      *      2.11.3 产生连续的公司代码
2775      *      2.11.4 处理为平行面板
2776      *      2.11.5 剔除IPO当年的数据
2777      *      2.11.6 行业发生变更的公司
2778
2779
2780
2781      *      =本节命令=
2782      *      =====
2783      *      -xtset- -xtpattern- -panels- -xtbalance-
2784      *      -panelthin- -enlarge- -paverage- -center-
2785      *      =====
2786
2787
2788
2789      *
2790      *-----2.11.1 声明面板资料：xtset 命令-----
2791
2792      use xtcs.dta, clear
2793      browse // code+year 才能够唯一标示每个观察值
2794      xtset code year
2795      xtides
2796      gen tl_lag = L.tl
2797
2798      *-说明：
2799      *      (1) xtset 与 tsset 等价，但只能用于stata9以上版本
2800      *      (2) 如何处理错误信息"repeated time values within panel"?
2801      use xtcs.dta, clear
2802      replace year=2003 if year==2004 // 伪造一份年度重复的样本
2803      list code year in 1/30, sepby(code)
2804      tsset code year // 错误
2805      duplicates report code year // 查验 code-year 是否能唯一识别样本
2806      duplicates drop code year, force // 删除重复样本
2807      tsset code year // 正确
2808      xtides
2809
2810
2811      *
2812      *-----2.11.2 公司数目和年度的统计----- -xtides- -panels- -xtpattern-

```

```
2813
2814 *--2.11.2.1 面板资料的基本描述: xtdes 命令
2815     use gta_sample.dta, clear
2816     tsset id year
2817     xtdes                                // 默认: 仅呈现9种频率最高的形态
2818     xtdes, patterns(20)
2819     xtdes, p(30)
2820     xtdes if sicmen_str == "C", p(25) // 制造业
2821
2822 *--2.11.2.2 记录面板的资料形态: xtpattern 命令
2823     use gta_sample.dta, clear
2824     tsset id year
2825     xtpattern , gen(pp)
2826     tab pp, sort
2827     browse id year pp
2828 *--应用
2829     drop if year<1999
2830     xtpattern, gen(pat)
2831     tab pat, sort
2832     keep if pat == "1111111111" // 平行面板的简单处理方式
2833     xtdes
2834
2835 *--2.11.2.3 统计公司数目: panels 命令
2836     use gta_sample.dta, clear
2837     tsset id year
2838     panels id
2839
2840     label list province_lab
2841     panels id if province==5 // 广东上市公司
2842
2843     tab province // 以观察值为单位进行统计
2844     panels id: tab province // 以公司为单位进行统计
2845
2846     tabstat size tl roa tobin, ///
2847         format(%6.3f) c(s) stat(N mean sd p50 min max)
2848     panels id: tabstat size tl roa tobin, c(s) stat(N)
2849         // 进一步统计每个变量对应的公司数目
2850
2851
2852 *
2853 *--2.11.3 产生连续的公司代码 -egen- group()
2854
2855     use xtcs.dta, clear
2856     xtset code year
2857     tab code in 1/100 // 公司代码不连续
2858     egen code_123 = group(code) // 产生连续编号的公司代码
2859     list code code_123 year in 1/50, sepby(code)
2860
2861 *--用途: 你可以使用forvalues等循环命令针对每家公司进行分析
2862     xtdes
2863     gen b = .
2864     forvalues i=1/438{
2865         qui reg tl size tang tobin if code_123==`i'
2866         replace b = _b[tobin] in `i'
2867     }
2868     gen i = _n
2869     browse i b in 1/20
2870
2871
2872 *
2873 *--2.11.4 处理为平行面板 -xtbalance-
2874
2875     use gta_sample.dta, clear
2876     xtdes
2877
2878     help xtbalance // Given by Yu-Jun Lian
2879
2880     xtbalance, range(2000 2008)
2881     xtdes
2882
2883 *--缺漏值的处理: miss() 选项
2884     sum id year cflow cash invt tl size
2885     drop if invt==.
2886     xtdes
```



```
2887
2888     xtbalance, r(2000 2008) miss(cflow cash invt tl size)
2889     xtides
2890
2891     *-一次性处理
2892     use gta_sample.dta, clear
2893     keep id year cflow cash invt tl size roa tobin
2894     xtbalance, r(2000 2008) miss(_all)
2895     xtides
2896
2897
2898 *
2899 *-2.11.5 剔除IPO当年的数据
2900
2901     *-由于缺少公司IPO的年份，本例中假设公司首次有记录的年份即为IPO年度
2902
2903     *-方法1: 利用 xtides 命令的返回值 和 egen 命令的 min() 函数
2904     use GTA_sample.dta, clear
2905     tsset id year
2906     xtides
2907     bysort id: egen Tmin = min(year)
2908     list id year Tmin in 1/50, sepby(id)
2909     drop if (year-Tmin==0)
2910     list id year Tmin in 1/50, sepby(id)
2911     xtides
2912
2913     *-方法2: 巧妙使用差分运算和 _n
2914     use GTA_sample.dta, clear
2915     tsset id year
2916     gen Dyear = D.year
2917     list id year Dyear in 700/900, sepby(id)
2918     bysort id (year): drop if (Dyear==. & _n==1)
2919     list id year Dyear in 700/900, sepby(id)
2920
2921 *
2922 *-2.11.6 行业发生变更的公司
2923
2924     use GTA_sample.dta, clear
2925     label list sicmen_lab
2926
2927     *-人为生成行业变更数据(因为我们这份数据有局限)
2928     replace sicmen=4 if (sicmen==5 & year>2006)
2929
2930     *-查找行业发生变更的公司
2931     qui tsset
2932     gen sic_dif = D.sicmen // 若发生变更，则此变量不为零
2933     bysort id: egen sic_change = sum(sic_dif)
2934                                     // 统计变更的次数, 以公司为单位进行标记
2935     order id year sicmen sic_dif sic_change
2936     list id year sicmen sic_dif sic_change if sic_change!=0, sepby(id)
2937
2938     *-删除行业发生变更的公司
2939     drop if sic_change != 0 // 若不发生行业变更，则该值不等于0
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951     *      =====
2952     *      计量分析与STATA应用
2953     *      =====
2954
2955     *      主讲人: 连玉君 博士
2956
2957     *      单 位: 中山大学岭南学院金融系
2958     *      电 邮: arlionn@163.com
2959     *      主 页: http://blog.cnfol.com/arlion
2960
```

```

2961      *      ::第一部分::
2962      *      Stata 操作
2963      *      =====
2964      *      第二讲 数据处理
2965      *      =====
2966      *      2.11 面板资料的处理(II)
2967
2968      *-----
2969      *-2.11 面板资料的处理(II)
2970      *-----
2971
2972      *      ==本节目录==
2973
2974      *      2.11.7 如何删除面板资料首尾的缺漏值?
2975      *      2.11.8 仅保留连续 T 年以上可获得资料的公司
2976      *      2.11.9 面板资料瘦身 I: 每隔 T 年保留一次资料
2977      *      2.11.10 面板资料瘦身 II: 采用 P 年平均值进行估计
2978      *      2.11.11 面板缺漏值的扩充
2979      *      2.11.12 变量的“去均值”和标准化处理
2980      *      2.11.13 面板资料处理的其他主题
2981
2982
2983      *
2984      *-2.11.7 如何删除面板资料首尾的缺漏值?
2985
2986      *-数据
2987      use xtmiss, clear
2988      list, sepby(id)
2989
2990      *-问题: 只删除首尾的缺漏值, 中间的不删(可以采取其他方法插值)
2991
2992      *-S1: 删除“首部”缺漏值
2993      bysort id (year): drop if sum(mi(x))==_n
2994      list, sepby(id)
2995
2996      *-解析
2997      use xtmiss, clear
2998      bysort id (year): gen n123 = _n
2999      gen miss = mi(x)
3000      bysort id: gen summis = sum(miss) // 注意: 是 gen, 而不是 egen
3001      list, sepby(id)
3002      drop if n123==summis
3003      list
3004
3005      *-S2: 删除“尾部”缺漏值
3006      use xtmiss, clear
3007      gen nyear = -year
3008      list, sepby(id)
3009      bysort id (nyear): drop if sum(mi(x))==_n
3010      list, sepby(id)
3011      tsset id year
3012      list, sepby(id)
3013
3014      *-汇总:
3015      use xtmiss, clear
3016      bysort id (year): drop if sum(mi(x))== n
3017      gsort id -year // 注意此行和下一行的变化
3018      bysort id: drop if sum(mi(x))==_n
3019      qui tsset id year
3020      list, sepby(id)
3021
3022      *-egen 命令提供了更为直接的解决办法
3023      use xtmiss, clear
3024      by id, sort: egen firstnonmis = min(cond(!missing(x), year, .))
3025      by id: egen lastnonmis = max(cond(!missing(x), year, .))
3026      drop if (year<firstnonmis) | (year>lastnonmis)
3027      list, sepby(id)
3028
3029      *-解析
3030      help cond()
3031      * cond(x, a, b)
3032      * 若 x=true, 返回 a;
3033      * 若 x=false, 返回 b;
3034      dis cond(1, 5, -5)

```

```

3035         dis cond(0, 5, -5)
3036     * !missing(x)
3037     * 若 x != ., 返回 1
3038     * 若 x == ., 返回 0
3039         dis !missing(.)
3040         dis !missing(9)
3041
3042     *-回顾
3043         use xtmiss, clear
3044         by id, sort: gen conyear = cond(!missing(x), year, .)
3045         list, sepby(id)
3046         by id, sort: egen firstnonmis = min(conyear)
3047         by id, sort: egen lastnonmis = max(conyear)
3048         list, sepby(id)
3049         drop if (year<firstnonmis) | (year>lastnonmis)
3050         list, sepby(id)
3051
3052
3053
3054     *
3055     *-2.11.8 仅保留连续 T 年以上可获得资料的公司
3056
3057     *-问题: 在有些分析中, 需要差分处理, 或需要考察公司行为的延续性
3058     *      此时便需要筛选出连续多年有观察值的公司
3059
3060     *-示例: 保留连续六年有样本的公司
3061
3062         use gta_sample.dta, clear
3063         keep id year t1 cash tobin size
3064         drop if t1>1
3065         tsset id year
3066         xtides
3067         xtpattern, gen(pp)           // 记录每家公司的样本形态
3068         tab pp
3069         *browse
3070
3071         gen p6 = strpos(pp, "111111")
3072         sort p6
3073         *browse
3074         drop if p6==0               // 仅保留连续六年有资料的公司
3075         tab pp
3076         save xtcontinue_temp, replace
3077
3078     *-如何执行如下两种处理方式:
3079     * Q1: 如何删除所有"退市"或部分年度资料缺失的公司?
3080     *      即 "...1111111....." 或 "...11111111..11111"
3081     * Q2: 如何删除某个间断年份以后的所有数据?
3082     *      即 "...11111111..11111" --> "...11111111....."
3083     *      或 "...111111111111.1." --> "...111111111111..."
3084     * 注: 后一种方式有助于保留尽可能多的样本
3085
3086     *-s1: 去掉"退市"或部分年度资料缺失的公司
3087         use xtcontinue_temp, clear
3088         xtides           // 140 家
3089         tab pp
3090         gen pbreak1 = strpos(pp, "1.")
3091         order id year pp p6 pbreak1
3092         browse
3093         drop if pbreak>0
3094         tab pp, sort
3095         xtides           // 123 家
3096     *-说明: 亦可采用 strmatch(), indexnot() 函数完成上述处理
3097
3098     *-s2: 如何删除某个间断年份以后的所有数据
3099         use xtcontinue_temp, clear
3100         tab pp
3101         *-简化数据(便于解释而已)
3102         keep if pp==".....1111111..1" ///
3103             | pp=="...111111111111.1." ///
3104             | pp==".....1111111111.11" ///
3105             | pp=="..1111111111111111"
3106         keep id year pp
3107         qui tsset id year
3108         gen Dyear = D.year

```

```

3109     order id year Dyear
3110     list, sepby(id)      // 第一年的缺漏值并非真正的“间断”
3111     bysort id (year): replace Dyear=1 if _n==1
3112     list, sepby(id)
3113     bysort id: egen firstyear_mis = min(cond(missing(Dyear), year, .))
3114     bysort id: drop if year>=firstyear_mis
3115     xtides
3116
3117     *-----
3118     *-练习1: 你可以把 keep 开头的那两条语句删除后对样本整体进行处理
3119
3120     *-----
3121     *-练习2: 使用 gta_sample.dta 样本, 要求处理后的数据符合如下条件:
3122     *   (1) 样本是连续的
3123     *   (2) 删除负债率大于1的观察值
3124     *   (3) 可以是非平行面板, 但时间跨度至少是 2003-2008 年
3125
3126     *-解答方案 -I-
3127     use gta_sample.dta, clear
3128     tsset id year
3129     drop if t1>1          // 满足第二个条件
3130     drop if year<1997     // 初步满足第三个条件
3131     xtpattern , gen(pp)
3132     tab pp
3133     gen Dyear = D.year
3134     bysort id: replace Dyear=1 if _n==1
3135     bysort id: egen miss = max(cond(Dyear==.), 1, 0)
3136     // 若某家公司存在断点, 则miss变量的所有年度都标示为1
3137     drop if miss==1
3138     tab pp
3139     bysort id: egen Tmin = min(year)    // 每家公司的最小年份
3140     bysort id: egen Tmax = max(year)    // 每家公司的最大年份
3141     drop if Tmin>2003    // 满足第四个条件
3142     drop if Tmax<2008    // 满足第三、四个条件
3143     xtides
3144
3145     *-解答方案 -II-
3146     *-思路: 统计每个公司的最大年度和最小年度之差,
3147     *       然后计算该公司共有多少年的观察值,
3148     *       若二者不一致, 则剔除该公司
3149     use gta_sample.dta, clear
3150     tsset id year
3151     drop if t1>1
3152     drop if year<1997
3153     xtpattern , gen(pp)
3154     cap dropvars Tmin Tmax Tsum Tdif Tmis
3155     bysort id: egen Tmin = min(year)
3156     gen Tmax = 2008
3157     bysort id: egen Tsum = count(year)
3158     // 统计每家公司实际有多少年的观察值
3159     gen Tdif = Tmax-Tmin+1 // 若样本连续, 应该有这么多年的观察值
3160     gen Tmis = Tsum-Tdif
3161     drop if Tmis != 0      // 删除样本不连续的公司
3162     drop if Tmin>2003      // 删除2003年以后上市的公司
3163     order id year pp Tmin Tmax Tsum Tdif Tmis
3164     tab pp
3165     *-----
3166
3167
3168
3169     *
3170     *-2.11.9 面板资料瘦身 I: 每隔 T 年保留一次资料      -panelthin-
3171
3172     *-目的: 若我们想考察某些变量的长期变化,
3173     *       需要拉长时间间隔来分析
3174
3175     *-基本用法
3176     use GTA_sample.dta, clear
3177     tsset id year
3178     xtides
3179     list id year in 1/50, sepby(id)
3180     panelthin, min(3) gen(OK)      // 每隔 3 年保留一次资料
3181     xtides if OK, p(30)
3182     list id year OK in 1/50, sepby(id)

```

```

3183     list id year if OK in 1/50, sepby(id)
3184
3185     *-示例：现金持有权衡理论的检验
3186     use GTA_sample.dta, clear
3187     tsset id year
3188     xtabond cash size tang roa tobin // 动态面板
3189     est store m_0
3190     panelthin, min(2) gen(OK) // 每隔两年瘦身一次
3191     xtabond cash size tang roa tobin if OK
3192     est store m_thin
3193     esttab m_0 m_thin, mtitle(m_0 m_thin) stat(N) // 结果对比
3194     *-解释和评述：
3195     * (1) 记 L.cash 的系数为 b，则 (1-b) 表示“调整速度”；
3196     * (2) 本例结果表明，若以一个年度为考察单位，则调整速度为 0.453
3197     *      若以两个年度为考察单位，则调整速度为 0.809
3198     *      这意味着，当时间跨度较长时，公司有能力向目标值调整
3199     *      从本例来看，公司基本上可以在两个会计年度内完成调整。
3200     * (3) 这种处理方法在日资料和月度资料中更为常用：
3201     *      对于日资料，min(5) 可能比较常用
3202     *      对于月资料，min(3) 或 min(5) 会比较常用
3203
3204
3205
3206     *
3207     *-2.11.10 面板资料瘦身 II：采用 P 年平均值进行估计 -paverage-
3208
3209     *-目的：克服经济周期和衡量偏误的影响
3210     *-适用于平行面板资料
3211
3212     *-基本用法：
3213     use xtcs, clear
3214     drop if year==1998
3215     xtodes // 平行面板
3216     paverage tl-tobin, p(2) ind(code) yr(year)
3217     xtodes
3218
3219
3220     *-示例：两种处理方式结果的对比
3221     use GTA_sample.dta, clear
3222     tsset id year
3223
3224     *-仅保留待分析的变量
3225     keep id year tl tang roa tobin
3226
3227     *-删除缺漏值并处理为平行面板
3228     xtbalance, range(1999 2008) miss(tl tang roa tobin)
3229     xtreg tl tang roa tobin, fe
3230     est store m_0
3231
3232     *-计算 2 年平均值，并估计
3233     preserve
3234     paverage tl tang roa tobin, p(2) ind(id) yr(year)
3235     xtreg tl tang roa tobin, fe
3236     est store m_av2
3237     restore
3238
3239     *-计算 5 年平均值，并估计
3240     preserve
3241     paverage tl tang roa tobin, p(5) ind(id) yr(year)
3242     xtreg tl tang roa tobin, fe
3243     est store m_av5
3244     restore
3245
3246     *-结果对比
3247     esttab m_0 m_av2 m_av5, ///
3248         mtitle(m_0 m_av2 m_av5) stat(N r2_w r2_o)
3249
3250
3251
3252     *
3253     *-2.11.11 面板缺漏值的扩充
3254
3255     *-问题 I：
3256     *-如何生成一个新变量：只要这家公司曾经发放过至少一次股利就标记为 1

```

```

3257     use GTA_sample.dta, clear
3258     tsset id year
3259     list id year div_yes in 100/130, sepby(id)
3260
3261     *-思路：若某家公司曾经发放过股利，则该公司的 div_yes 均值不为零
3262     bysort id: egen div_mean = mean(div_yes)
3263     gen div_s = 0
3264     replace div_s = 1 if div_mean != 0
3265     list id year div_yes div_s in 100/130, sepby(id)
3266
3267
3268     *-问题 II：
3269     *-假设我们只有2005年的行业分类数据，
3270     *     如何扩充以便各个年度都共享这一信息？
3271     *-假设所有公司的行业归属在样本区间内不发生变更
3272
3273     *-S1：伪造一份数据
3274     use GTA_sample.dta, clear
3275     tsset id year
3276     drop if year<2000
3277     keep id year sicda
3278     replace sicda = . if year!=2005
3279     list in 1/100, sepby(id)
3280     clonevar sicda_s1 = sicda // 克隆两份，以备后用
3281     clonevar sicda_s2 = sicda
3282
3283     *-S2：思路：我们可以对这个唯一数据随意排序
3284     sort id sicda_s1
3285     list in 1/70, sepby(id)
3286     bysort id: replace sicda_s1 = sicda_s1[1] if _n>1
3287     list in 1/70, sepby(id)
3288
3289     *-S3：简洁命令 -enlarge-
3290     enlarge sicda_s2, by(id)
3291     list in 1/70, sepby(id)
3292
3293
3294     *
3295     *-2.11.12 变量的“去均值”和标准化处理
3296
3297     help center // 外部命令
3298     use xtcs.dta, clear
3299     bysort code: center tl fr size ndts tang tobin, prefix(c_)
3300
3301     *-应用：估计FE模型
3302     reg c_tl c_fr-c_tobin
3303     est store ols_fe
3304     xtreg tl fr size ndts tang tobin, fe
3305     est store fe
3306     esttab ols_fe fe, nogap compress
3307
3308     *-说明：该命令尚可进行标准化和quasi-demeaning处理，请查阅帮助文件
3309
3310
3311     *
3312     *-2.11.13 面板资料处理的其他主题
3313
3314     *-[s1] How can I identify first and last occurrences
3315     *     systematically in panel data?
3316     *     http://www.stata.com/support/faqs/data/firstoccur.html
3317     *-[s2] How can I generate a variable relating panel data
3318     *     to a reference panel?
3319     *     http://www.stata.com/support/faqs/stat/panelref.html
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3330     *     =====

```

```

3331      *           计量分析与STATA应用
3332      *           =====
3333
3334      *           主讲人：连玉君 博士
3335
3336      *           单 位：中山大学岭南学院金融系
3337      *           电 邮：arlionn@163.com
3338      *           主 页：http://blog.cnfol.com/arlion
3339
3340      *           ::第一部分::
3341      *           Stata 操作
3342      *           =====
3343      *           第二讲 数据处理
3344      *           =====
3345      *           2.12 数据的查验和比较
3346
3347      cd `c(sysdir_personal)'Net_course_A\A2_data
3348
3349      *-----
3350      *-2.12 数据的查验和比较
3351      *-----
3352
3353      *           ==本节目录==
3354
3355      *           2.12.1 查验变量
3356      *               2.12.1.1 计数
3357      *               2.12.1.2 条件确认
3358      *               2.12.1.3 比较变量的大小
3359      *           2.12.2 查验两组数据
3360      *               2.12.2.1 查验两笔数据的观察值是否一致
3361      *               2.12.2.2 查验两笔数据的变量是否一致
3362
3363      *           =本节命令=
3364      *           =====
3365      *           -assert- -count- -compare- -cf- -cfvar-
3366      *           =====
3367
3368
3369
3370      *
3371      *-2.12.1 查验变量 -assert- -count- -compare-
3372
3373
3374      *-2.12.1.1 计数 -count-
3375
3376      sysuse nlsw88, clear
3377      count if (hours<10 | hours>70)
3378      count if race >=2
3379      count if hours == .
3380      list wage race if hours == .
3381
3382
3383      *-2.12.1.2 条件确认 -assert-
3384
3385      sysuse nlsw88, clear
3386      sum wage age
3387      assert wage>0
3388      assert wage<0
3389      assert wage<20
3390      count if wage<20
3391      assert age<40
3392      count if age<40
3393      assert (hours<10 | hours>70)
3394      count (hours<10 | hours>70)
3395      list hours if (hours<10 | hours>70)
3396
3397
3398      *-2.12.1.3 比较变量的大小 -compare-
3399
3400      sysuse sp500.dta, clear
3401      compare open close
3402
3403
3404

```



```
3405 *
3406 *-2.12.2 查验两组数据
3407
3408
3409 *-2.12.2.1 查验两笔数据的观察值是否一致 -cf-
3410
3411 clear
3412 input id str8 name age ht wt income
3413     11      john  23  68 145 23000
3414     12    charlie  25  72 178 45000
3415     13     sally  21  64 135 12000
3416      4      mike  34  70 156  5600
3417     43      paul  30  73 189 15600
3418 end
3419 sort id
3420 save person1, replace
3421
3422 clear
3423 input id str8 name age ht wt income
3424     11      john  28  68 145 23000
3425     12   charles  25  52 178 45000
3426     13     sally  21  64  . 12000
3427      4   michael  34  70 156  5600
3428     43      Paul  30  73 189  5600
3429 end
3430 sort id
3431 save person2, replace
3432
3433 use person1, clear
3434 cf _all using person2
3435 cf _all using person2, verbose // 详细呈现
3436 cf _all using person2, verbose all // 列出所有不一致的cases
3437
3438
3439 *-2.12.2.2 查验两笔数据的变量是否一致 -cfvars-
3440
3441 sysuse xtcs, clear
3442 drop t1
3443 cfvars xtcs.dta
3444 ret list
3445
3446
3447
3448
3449
3450
```



```

1
2
3
4
5      *=====
6      *      计量分析与STATA应用
7      *=====
8
9      *      主讲人：连玉君 博士
10
11     *      单   位：中山大学岭南学院金融系
12     *      电   邮：arlionn@163.com
13     *      主   页：http://blog.cnfol.com/arlion
14
15     *      ::第一部分::
16     *      Stata 操作
17     *      =====
18     *      第三讲  Stata绘图
19     *      =====
20     *      -3.1- 简 介
21
22
23     *      -----
24     *      本讲目录
25     *      -----
26     *      3.1  简 介
27     *      3.2  二维图选项
28     *      3.3  元素代号
29     *      3.4  常用图形示例
30     *      3.5  结 语
31
32
33
34 *-----
35 *-> 3.1  简 介
36 *-----
37
38     *      ==本节目录==
39
40     *      3.1.1 Stata 图形的种类
41     *      3.1.2 二维图命令的基本结构
42     *      3.1.3 几种常用图形的简单示例
43     *      3.1.4 图形的管理
44     *      3.1.4.1 图形的保存
45     *      3.1.4.2 图形的导出
46     *      3.1.4.3 图形的调入
47     *      3.1.4.4 插入 Word
48     *      3.1.4.5 查询
49     *      3.1.4.6 重新显示图形
50     *      3.1.4.7 图形的合并
51     *      3.1.4.8 删除图形
52     *      3.1.5 图形的显示模式(绘图模板)
53     *      3.1.5.1 显示模式种类
54     *      3.1.5.2 中文投稿的黑白图
55     *      3.1.5.3 stata 用户提供的模板
56     *      3.1.5.4 创建自己的图形模板
57
58
59     * cd D:\stata11\ado\personal\Net_course\A3_graph
60     * cd `c(sysdir_personal)'Net_course_A\A3_graph
61
62 *-----
63 *-3.1.1 Stata 图形的种类
64
65                                     /*
66 graph twoway      二维图
67   scatter        散点图
68   line           折线图
69   area           区域图
70   lfit           线性拟合图
71   qfit           非线性拟合图
72   histogram      直方图
73   kdensity       密度函数图
74   function       函数图

```

```

75 -----
76 graph matrix      矩阵图
77 graph bar         条形图
78 graph dot         点图
79 graph box         箱形图
80 graph pie         饼图
81 -----
82         ac         相关系数图
83         pac        偏相关系数图
84         irf        脉冲相应函数图
85 -----
86         ... ..          */
87
88
89
90 *
91 *-3.1.2 二维图命令的基本结构
92
93 *-整体架构
94
95     * twoway (单元图1) (单元图2) (...), 选项1 选项2 ...
96
97     * twoway 单元图1 || 单元图2 || ..., 选项1 选项2 ...
98
99 *-单元图的定义
100
101     * (单元图类型 y1 y2 ... x, 选项1 选项2 ...)
102
103 *-二维图选项的定义
104
105     * 二维图选项标题 (定义内容, 子选项 子选项 ...)
106
107
108 *- 一个标准的实例
109 *-----
110 sysuse sp500, clear
111 twoway (line high date) (line low date)          ///
112         ,                                          ///
113         title("图1: 股票最高价与最低价时序图", box) ///
114         xtitle("交易日期", margin(medsmall))    ///
115         ytitle("股票价格")                      ///
116         ylabel(900(200)1400) ymtick(##5)        ///
117         legend(label(1 "最高价") label(2 "最低价")) ///
118         note("资料来源: Stata公司, SP500.dta")    ///
119         caption("说明: 我做的第一幅Stata图形!")  ///
120         saving(mypig.gph, replace)
121 *-----
122 *- 注意: 逗号后全部为选项, 裸露在外的逗号只有一个
123
124
125
126 *               +-----+
127 *=====本讲导读=====
128 *               +-----+
129
130 * 图形无非是点、线(面)、文字等元素的组合
131
132 * 这些组合的整体“风格”构成了图类: 单元图 (逗号前的部分)
133
134 * 每种图形的具体特征由元素的特征决定: 选项 (逗号后的部分)
135
136 * 因此, 选项的填写是Stata绘图的关键!
137
138 *=====
139
140
141
142 *
143 *-3.1.3 几种常用图形的简单示例
144
145     sysuse sp500, clear
146
147 *-散点图
148     twoway scatter high date

```

```

149
150  *--折线图
151      twoway line change date
152
153  *--柱状图
154      twoway bar open date in 1/50
155
156  *--直方图
157      histogram change
158
159  *--密度函数图
160      kdensity close, normal
161
162  *--数学函数图
163      twoway (function y=sin(x), range(-10 10) lw(*1.5)) ///
164              (function y=cos(x), range(-10 10) lw(*2.0)), ///
165              ytick(-2(0.5)2) ylabel(, angle(0)) ///
166              yline(0, lcolor(black*0.5) lpattern(dash)) ///
167              scheme(slmono)
168
169
170  *
171  *--3.1.4 图形的管理
172
173  *--3.1.4.1 图形的保存
174
175      help graph save
176
177      *--第一种方式
178          sysuse sp500, clear
179          twoway line high date
180          graph save fig1.gph, replace
181          graph use fig1.gph          // 重现图形
182
183      *--第二种方式
184          twoway line high date, saving(A3_price.gph, replace)
185
186      *--手动方式：右击 -> Save graph ...-> 填入图形名称，选择保存类型
187
188
189  *--3.1.4.2 图形的导出
190
191      help graph export
192
193          sysuse sp500, clear
194          twoway line high low date
195          graph export A3_price.wmf, replace
196          graph export "D:\mypaper\A3_price.wmf", replace
197
198      *--注：相当于另存为其他格式的图形
199
200      *      后缀          附加选项          输出格式
201      *      -----
202      *      .ps            as(ps)           PostScript
203      *      .eps            as(eps)           Encapsulated PostScript
204      *      .wmf            as(wmf)           Windows Metafile
205      *      .emf            as(emf)           Windows Enhanced Metafile
206      *      .pict           as(pict)          Macintosh PICT format
207      *      .png            as(png)           PNG (Portable Network Graphics)
208      *      .tif            as(tif)           TIFF
209      *      other          must specify as()
210      *      -----
211
212  *--调整输出图片的分辨率
213      twoway line high low date
214      graph export A3_price2.tif, width(3160) height(1800) replace
215      shellout A3_price2.tif
216      *--注意：仅适用于 .png 和 .tif 格式的图片
217
218
219  *--3.1.4.3 图形的调入
220
221      help graph use
222

```

```
223     graph use fig1.gph
224     graph use fig1, scheme(slmono)
225     graph use fig1, scheme(economist)
226
227
228     *-3.1.4.4 插入 Word
229
230     shellout mypaper.doc
231
232     * 右击 -> copy -> 粘贴到Word中
233     * 若图片太大，可以右击图片->设置图片格式->大小，进行相应的调整
234     * 建议先将图形输出为 wmf 格式，然后再贴入 word
235
236
237     *-3.1.4.5 查询
238     graph dir
239
240
241     *-3.1.4.6 重新显示图形
242
243     twoway line high low date
244     graph display, scheme(sj)
245     graph save A3_price_sj.gph, replace
246
247
248     *-3.1.4.7 图形的合并
249
250     help graph combine
251
252     sysuse sp500, clear
253     twoway line high low date
254     graph save A3_price.gph, replace
255     twoway line high low date, scheme(slmono)
256     graph save A3_price_sj.gph, replace
257     graph combine A3_price.gph A3_price_sj.gph
258
259
260     *-3.1.4.8 删除图形
261     erase A3_price.gph
262     graph dir
263
264
265     *
266     *-3.1.5 图形的显示模式 (绘图模板)
267
268     *-3.1.5.1 显示模式种类
269
270     help schemes // Stata 提供的显示模式
271
272     *-两种设定方式
273     * set scheme schemename [, permanently]
274     * graph ... [, ... scheme(schemename) ...]
275
276     *-----实例-----
277     sysuse auto, clear
278
279     twoway scatter price weight, scheme(sj)
280     graph save A3_gr1.gph, replace
281     graph use A3_gr1.gph, scheme(s2color)
282
283     set scheme economist
284     twoway scatter price weight
285
286
287     *-各种显示模式一览
288     graph use A3_scheme1.gph
289     doedit A3_scheme1.do
290
291
292     *-3.1.5.2 中文投稿的黑白图
293
294     set scheme slmono
295
296     sysuse auto, clear
```

---

```
297 twoway scatter price weight
298 graph bar price, over(foreign)
299 graph bar price, over(rep78) over(foreign)
300
301 sysuse sp500, clear
302 twoway (connect high date, sort msymbol(D)) ///
303       (connect low date, msymbol(+)) in 1/20 ///
304       , scheme(slmono)
305
306
307 *-3.1.5.3 stata 用户提供的模板
308
309 *-Mitchell 提供的模板
310 * Mitchell, M.
311 *   A visual guide to Stata graphics.
312 *   Stata Press, 2008.
313
314 view browse "http://www.stata-press.com/data/vgsg.html"
315
316 * net from http://www.stata-press.com/data/vgsg2/
317 * net install vgsg2 // 安装外部模式插件
318 * net get vgsg2 // 下载相关数据
319
320 *-e.g
321 use allstates.dta, clear
322 twoway scatter propval100 rent700 ownhome, ///
323       scheme(vg_slc) // vg_slc 黑白底, 彩色图
324
325 twoway scatter propval100 rent700 ownhome, ///
326       scheme(vg_slm) // vg_slc 黑白底, 黑白图
327
328 *-其它模板(10余种): 参见 Mitchell(2008, section 1.3)
329
330
331 *-Roger Newson 提供的模板
332
333 help scheme_rbnlmono
334
335 use allstates.dta, clear
336
337 *-stata默认黑白图
338 twoway scatter propval100 rent700 ownhome, ///
339       scheme(slmono)
340
341 *-rbnlmono 模式 比较紧凑
342 twoway scatter propval100 rent700 ownhome, ///
343       scheme(rbnlmono)
344 *-需要适当修改
345 twoway scatter propval100 rent700 ownhome, ///
346       scheme(rbnlmono) ///
347       xlabel(,angle(0)) legend(row(1))
348
349 *-其它模板
350
351 findit scheme
352
353
354 *-3.1.5.4 创建自己的图形模板
355
356 help scheme_files
357
358 viewsource scheme-rbnlmono.scheme // rbnlmono 模板
359
360
361
362
363
364
365
366
367
368 *=====
369 * 计量分析与STATA应用
370 *=====
```

```

371
372      *      主讲人：连玉君 博士
373
374      *      单  位：中山大学岭南学院金融系
375      *      电  邮：arlionn@163.com
376      *      主  页：http://blog.cnfol.com/arlion
377
378      *      ::第一部分::
379      *      Stata 操作
380      *      =====
381      *      第三讲 Stata绘图
382      *      =====
383      *      -3.2- 二维图选项
384      *      (I)
385
386
387      *      ==本节目录==
388
389      *      3.2.1 坐标类
390      *      3.2.1.1 坐标轴刻度(tick)及刻度标签(label)
391      *      3.2.1.2 坐标轴标题: ytitle() xtitle()
392      *      3.2.1.3 坐标结构: yscale() xscale()
393      *      3.2.1.4 双坐标系
394      *      3.2.2 标题类
395      *      3.2.2.1 标题的种类
396      *      3.2.2.2 示例
397      *      3.2.2.3 标题的位置
398      *      3.2.3 区域类
399      *      3.2.3.1 Stata图形的区域划分
400      *      3.2.3.2 控制内区和外区的边距
401      *      3.2.3.3 控制图形的纵横比例
402      *      3.2.3.4 绘图区的显示模式
403      *      3.2.3.5 绘图区和全图区背景颜色的控制
404      *      3.2.4 图例类
405      *      3.2.4.1 自动产生的图例
406      *      3.2.4.2 从新定制图例
407      *      3.2.4.3 图例的位置
408      *      3.2.4.4 多个图例的重排
409      *      3.2.4.5 线型的控制
410
411
412
413      *-----
414      *-> 3.2  二维图选项
415      *-----
416
417      help twoway_options
418
419      *-----
420      *-3.2.1 坐标类
421
422      help axis_options
423
424
425      *-3.2.1.1 坐标轴刻度(tick)及刻度标签(label)
426
427      help axis_label_options
428
429      set scheme s2color
430      sysuse auto, clear
431      scatter mpg weight, xlabel(#10)    // 显示出来的刻度标签未必是10个，？
432
433      * 主刻度及标签: ylabel(), xlabel() // 显示刻度标签时，同时显示刻度
434      * 主刻度:      ytick(), xtick() // 按设定显示刻度，仅显示主要刻度的标签
435      * 子刻度及标签: ylabel(), xlabel()
436      * 子刻度:      ymtick(), xmtick()
437
438      *-实例
439
440      scatter mpg weight                // Stata 默认设定，比较宽松
441
442      scatter mpg weight, xlabel(#10) // 在横坐标上列示10个最佳的刻度及其标签
443
444      scatter mpg weight, xtick(#10)

```

```

445
446     scatter mpg weight,      ///
447         ylabel(10(5)45) ///
448         xlabel(1500 2000 3000 4000 4500 5000) // 自行设定刻度标签
449
450     scatter mpg weight, ylabel(##5) xmtick(##10) // 子刻度和子刻度标签
451
452     scatter mpg weight, xlabel(1500 2500 3190 "中位数" 3500 4500)
453         // 刻度标签由`数字`替换为`文字`
454
455     * 参数设定规则:
456     *   rule      example      description
457     *   -----
458     *   #?        #4           4 个最佳值
459     *   ##?       ##10         10-1=9 个子刻度列印于主刻度之间
460     *               仅适用于 mlabel() 和 mtick() 选项
461     *   ?(?)?     10(5)45      在 10 到 45 范围内, 每隔 5 列印一个子刻度
462     *   none      none         不显示刻度标签
463     *   -----
464     *   注: #? 和 ##? 比较常用
465
466     * 刻度标签的角度(详见文字选项部分)
467     scatter mpg weight, xlabel(,angle(45)) ylabel(,angle(-15))
468
469
470
471     *-3.2.1.2 坐标轴标题: ytitle() xtitle()
472
473     help axis_title_options
474
475     sysuse auto, clear
476     scatter mpg weight, ytitle("汽车里数") xtitle("汽车重量")
477
478     *-坐标轴标题的位置
479     scatter mpg weight, ytitle("汽车里数",place(top)) ///
480         xtitle("汽车重量",place(right))
481
482     *-长标题的处理
483     scatter mpg weight, xtitle("汽车里数" "(mpg)")
484
485
486
487     *-3.2.1.3 坐标结构: yscale() xscale()
488
489     help axis_scale_options
490
491     *-显示范围的控制
492     scatter mpg weight
493     scatter mpg weight, xscale(range(0 5000)) xlabel(0(1000)5000)
494     scatter mpg weight, xscale(range(1000 6000))
495     scatter mpg weight, xscale(range(3000 4000)) //为何不奏效?
496     scatter mpg weight if (wei>=3000&wei<=4000) // 局部显示需要用if语句
497
498     *-坐标轴标题间距的控制
499     label var mpg      "汽车里数"
500     label var weight   "汽车重量"
501     scatter mpg weight , xlabel(#14) // 默认设置
502     scatter mpg weight, xscale(titlegap(2)) // 坐标轴与坐标轴标题间距
503     scatter mpg weight, xscale(titlegap(2) outergap(-2)) // 坐标轴标题下边距
504
505
506     *-坐标轴的显示
507
508     *-不显示坐标轴
509     scatter mpg weight, yscale(noline) xscale(noline)
510
511     *-不显示坐标轴和刻度标签
512     scatter mpg weight, yscale(off) xscale(off)
513
514     *-无边距
515     scatter mpg weight, yscale(off) xscale(off) plotregion(style(none))
516
517     *-坐标轴线型
518     scatter mpg weight, xscale(lcolor(red) lwidth(vthick))

```

```

519
520
521
522 *-3.2.1.4 双坐标系
523
524     help axis_choice_options
525
526 *-共用 x 轴
527
528     sysuse sp500, clear
529     twoway line close change date
530     twoway (line close date, yaxis(1)) ///
531            (line change date, yaxis(2))
532
533     twoway (line close date, yaxis(1)) ///
534            (line change date, yaxis(2)), ///
535            ylabel(-50(10)40, axis(2) angle(0) labsize(small))
536
537 *-单独的 y 轴和 x 轴
538
539     twoway (line close date, yaxis(1) xaxis(1)) ///
540            (line change date, yaxis(2) xaxis(2)), ///
541            ylabel(-50(10)40, axis(2)) ///
542            xlabel(15005 15239, axis(2)) ///
543            xtitle("", axis(2))
544
545
546
547 *
548 *-3.2.2 标题类
549
550 *-3.2.2.1 标题的种类
551
552     * 主标题、副标题、注释、说明
553     * title()、subtitle()、note()、caption()
554
555     help title_options
556
557 *-3.2.2.2 示例
558
559     sysuse auto, clear
560
561     scatter mpg weight, title("Mileage and weight")
562
563     scatter mpg weight, title("Mileage and weight", box)
564     scatter mpg weight, title("Mileage and weight", box bexpand)
565
566     scatter mpg weight, title("主标题") subtitle("副标题")
567
568     scatter mpg weight, title("主标题") ///
569                        subtitle("副标题") ///
570                        note("注释内容") ///
571                        caption("进一步的说明")
572
573     scatter mpg weight, title("汽车里数和重量的" "散点图") ///
574                        subtitle("—美国资料实例")
575
576
577 *-3.2.2.3 标题的位置
578
579 *-说明：本节内容同样适用于其它包含 legend() 选项的类目
580
581 * 默认位置
582 * -----
583 * title()                居中
584 * subtitle()             居中
585 * note()                 左对齐
586 * caption()              左对齐
587 * -----
588
589 * 重新定位: position() 的取值
590 *
591 * +-----+
592 * |          11          12          1          |

```



```

593      *
594      *
595      *   10   +-----+
596      *   |   | 10 or 11  12  1 or 2 |   2
597      *   |   |          绘图区          |
598      *   |   | 9          ring=0          |   3
599      *   |   |          |          |
600      *   |   | 7 or 8    6    4 or 5 |   4
601      *   |   |          |          |
602      *   |   |          +-----+
603      *   |   |          7          6          5
604      *   |   |          +-----+
605      *   +-----+
606
607      * 默认相对间距: ring() 的取值
608      * -----
609      * plot region          0          | ring(0) = 绘图区内
610      * {t|b|l|r}1title()   1          |
611      * {t|b|l|r}2title()   2          | ring(k), k>0, 绘图区以外
612      * legend()            3          |
613      * note()              4          |
614      * caption()           5          | ring() 的值越大, 距离绘图区越远
615      * subtitle()          6          |
616      * title()             7          |
617      * -----
618
619      * 示例
620      scatter mpg weight, title("汽车里数和重量",position(5))
621      scatter mpg weight, title("汽车里数和重量",position(3) ring(0))
622      scatter mpg weight, title("汽车里数和重量",position(3) ring(12))
623
624
625
626      *
627      * -3.2.3 区域类
628
629      help region_options
630
631      * -3.2.3.1 Stata图形的区域划分
632      do A3_region.do
633
634      * -3.2.3.2 控制内区和外区的边距
635      twoway function y=x
636      twoway function y=x, plotregion(fcolor(green*0.4)) ///
637      plotregion(ifcolor(white))
638      twoway function y=x, plotregion(margin(0)) // 图形真正从原点开始出发
639      twoway function y=x, graphregion(margin(0))
640      twoway function y=x, plotregion(margin(l+15 r+5 t=10 b+4))
641      /*四个边距可以分别控制*/
642
643      * -3.2.3.3 控制图形的纵横比例
644      twoway function y=x /*如何得到正方形的图形? */
645      twoway function y=x, ysize(5) xsize(5)
646
647      * -3.2.3.4 绘图区的显示模式
648      twoway function y=x, plotregion(style(none))
649      twoway function y=x, plotregion(style(ci2))
650
651      * -3.2.3.5 绘图区和全图背景颜色的控制
652      sysuse auto, clear
653      scatter mpg weight, graphregion(fcolor(green*0.8)) ///
654      graphregion(ifcolor(yellow)) ///
655      plotregion(fcolor(black*0.3)) ///
656      plotregion(ifcolor(white)) ///
657      title("Stata图形分成四个区域")
658
659
660
661      *
662      * -3.2.4 图例类
663
664      help legend_options
665
666      * -3.2.4.1 自动产生的图例

```

```

667
668 * 一张图中同时呈现多个序列，便会自动产生图例
669 * 对于变量而言，其默认图例是它的变量标签
670
671 sysuse sp500, clear
672 twoway (line high date) (line low date) // 如何加入中文图例?
673
674 sysuse auto, clear
675 twoway (scatter price weight if foreign==1) ///
676       (lfit price weight if foreign==1) ///
677       (scatter price weight if foreign==0) ///
678       (lfit price weight if foreign==0)
679 * 此时，图例显得过于繁琐
680
681
682 *-3.2.4.2 从新定制图例
683
684 * 第一种方式：预先定义变量标签
685 sysuse sp500, clear
686 label var high 最高股价
687 label var low 最低股价
688 twoway (line high date) (line low date)
689 *-缺点：会永久改变变量标签
690
691 * 第二种方式：每个图单独加图例
692 sysuse sp500, clear
693 twoway (line high date, legend(label (1 "最高价")))) ///
694       (line low date, legend(label (2 "最低价"))))
695
696 * 第三种方式：整体加图例
697 twoway line high date || line low date, ///
698       legend(label(1 "最高价") label(2 "最低价"))
699
700 * 不显示图例 legend(off)
701 twoway (line high date) (line low date), legend(off)
702
703
704 *-3.2.4.3 图例的位置
705
706 * legend 的默认位置是 ring(3)
707
708 * 绘图区`外'的时钟点上
709 twoway line high date || line low date, ///
710       legend(position(12))
711
712 * 绘图区`内'的时钟点上 ring(0)
713 twoway line high date || line low date, ///
714       legend(ring(0))
715 twoway line high date || line low date, ///
716       legend(position(12) ring(0))
717
718 * 改变legend()的相对位置
719 * note() 的默认位置是 ring(4)
720 * caption()的默认位置是 ring(5)
721 twoway line high date || line low date, ///
722       note("addad") caption(资料来源: Stata 公司)
723 twoway line high date || line low date, ///
724       caption(资料来源: Stata 公司, ring(3)) ///
725       legend(ring(5))
726
727
728 *-3.2.4.4 多个图例的重排 rows(#), cols(#) 选项
729 sysuse uslifeexp.dta, clear
730 line le le_w le_b year
731 line le le_w le_b year, legend(rows(1))
732 line le le_w le_b year, legend(cols(1) size(small))
733
734
735 *-3.2.4.5 线型的控制
736
737 help connect_options
738 help linepatternstyle
739 help linestyle
740

```

```
741      sysuse sp500, clear
742
743      twoway connect open close low  date in 1/10
744
745      twoway connect open close low  date in 1/10, ///
746              lpattern(solid dash longdash)
747
748      twoway connect open close low  date in 1/10, ///
749              lpattern(solid dash longdash) ///
750              scheme(slmono) // 黑白图片
751
752
753
754
755
756
757
758
759
760
761      *=====
762      *      计量分析与STATA应用
763      *=====
764
765      *      主讲人：连玉君 博士
766
767      *      单  位：中山大学岭南学院金融系
768      *      电  邮：arlionn@163.com
769      *      主  页：http://blog.cnfol.com/arlionn
770
771      *      ::第一部分::
772      *      Stata 操作
773      *      =====
774      *      第三讲 Stata绘图
775      *      =====
776      *      -3.2- 二维图选项
777      *      (II)
778
779      cd `c(sysdir_personal)'Net_course_A\A3_graph
780
781
782      *      ==本节目录==
783
784      *      3.2.5 附加线类
785      *          3.2.5.1 选项结构
786      *          3.2.5.2 附加线 <位置>
787      *          3.2.5.3 附加线 <风格>
788      *          3.2.5.4 附加线 <线宽>
789      *          3.2.5.4 附加线 <颜色>
790      *          3.2.5.5 附加线 <线型>
791      *          3.2.5.5 附加线属性的独立性
792      *      3.2.6 文字与文本框
793      *          3.2.6.1 选项类别
794      *          3.2.6.2 文字和文本框的整体风格
795      *          3.2.6.3 文本框属性
796      *          3.2.6.4 文字属性
797      *      3.2.7 图标类
798      *          3.2.7.1 简介
799      *          3.2.7.2 图标的位置
800      *          3.2.7.3 图标的大小
801      *          3.2.7.4 图标的角度
802      *          3.2.7.5 图标的颜色
803      *      3.2.8 其它选项
804      *          3.2.8.1 分组绘图
805      *          3.2.8.2 重新设置变量标签
806      *          3.2.8.3 重新设置变量显示格式
807      *          3.2.8.4 重设图形种类
808
809
810      *
811      *-----
812      *-3.2.5 附加线类
813
814      help added_line_options
```

```
815  *--说明：本节中介绍的附加线属性，适用于所有与线相关的对象
816
817  *-3.2.5.1 选项结构
818
819  * twoway ..., yline(数字, 子选项)
820  * twoway ..., xline(数字, 子选项)
821  *-数字：控制附加线的位置
822  *-子选项：控制附加线的类型、颜色、宽度等
823
824
825  *-3.2.5.2 附加线 <位置>
826
827  sysuse sp500, clear
828  line open date, yline(1100)
829  line open date, yline(1100 1313) xline(15242)
830
831
832  *-3.2.5.3 附加线 <风格>
833
834  * default      决定于显示模式(set scheme)
835  * extended     延伸到绘图外区
836  * unextended   不延伸到绘图外区
837
838  line open date, yline(1100, style(unextended))
839  *-解释
840  line open date, yline(1100, style(unextended)) ///
841  plotregion(fcolor(green*0.3)) ///
842  plotregion(ifcolor(white))
843  line open date, yline(1100) ///
844  plotregion(fcolor(green*0.3)) ///
845  plotregion(ifcolor(white))
846
847
848  *-3.2.5.4 附加线 <线宽>
849
850  help linewidthstyle
851
852  line open date, yline(1100, lwidth(thick)) // 采用代号设定
853  line open date, yline(1100, lwidth(*1.5)) // 设定相对宽度
854
855
856  *-3.2.5.4 附加线 <颜色>
857
858  graph query colorstyle
859
860  line open date, yline(1100, lcolor(blue))
861  line open date, yline(1100, lcolor(blue*0.3))
862
863
864  *-3.2.5.5 附加线 <线型>
865
866  help linepatternstyle
867
868  palette linepalette
869
870  line open date, yline(1100, lpattern(dash) lcolor(black*0.3))
871  line open date, yline(1100, lpattern(dot))
872
873
874  *-3.2.5.5 附加线属性的独立性
875
876  line open date, yline(1100, lp(shortdash_dot) lc(blue*0.6)) ///
877  yline(1313, lw(*2.5) lc(green*0.4)) ///
878  xline(15242, lw(*2) lc(pink*0.4) lp(longdash))
879
880
881
882  *
883  *-3.2.6 文字与文本框
884
885  help textbox_options
886  help textstyle
887  help textboxstyle
888
```

```
889 * 指点迷津：想想 word 中的文本框
890 * 凡是出现文字的地方都可以做下面的设定
891
892
893 *-3.2.6.1 选项类别
894
895 * 文字和文本框的整体风格： 标题、副标题、文本、小号
896
897 * 文本框相关设定：文本框颜色、背景、与文字的边距等
898
899 * 文字相关的设定：大小、颜色、位置、行距
900
901
902 *-3.2.6.2 文字和文本框的整体风格
903
904 *-文字的风格：文字的标准化大小
905     help textstyle
906
907 *-文本框的风格
908     help textboxstyle
909     line open date, title("SP500 开盘价", tstyle(subheading))
910
911 *-文字与文本框的区别：
912 * 文字： 单行，无边框
913 * 文本框：单行或多行，可加边框，是文字的更一般化定义
914
915
916 *-3.2.6.3 文本框属性
917
918 *-显示文本框
919     line open date, title("SP500 开盘价", box)
920
921 *-文本框的相对大小
922     line open date, title("SP500 开盘价", box width(60) height(15))
923
924 *-文本框的背景和边框的颜色
925     line open date, title("SP500 开盘价", box fcolor(blue*0.2)) //仅背景
926     line open date, title("SP500 开盘价", box bcolor(yellow*0.4)) //背景和边框
927     line open date, title("SP500 开盘价", box fc(blue*0.2) lc(red))
928
929 *-边框的粗细、线型
930     line open date, title("SP500 开盘价", box fc(yellow*0.2) ///
931         lc(green) lwidth(*2.5) lpattern(dash))
932
933 *-文字与边框的相对位置
934     line open date, title("SP500 开盘价", box width(60) height(15) ///
935         alignment(middle)) // 纵向定位
936     line open date, title("SP500 开盘价", box width(60) height(15) ///
937         justification(right)) // 横向定位
938
939
940 *-3.2.6.4 文字属性
941
942 *-文字位置
943
944     help compassdirstyle
945
946 * 控制标题等位置： place()
947     line open date, xtitle("交易日期", place(right)) ///
948         ytitle("开盘价格", place(top))
949
950 * 在图形中的特定坐标点添加文字
951     line open date, text(1324.83 15117 "一个波峰")
952
953
954 *-文字的角度
955
956     help anglestyle
957
958     line open date
959     line open date, xlabel(, angle(30)) ylabel(,angle(0))
960     line open date, xlabel(, angle(30)) ylabel(,angle(15)) ///
961         ylabel(##4,angle(15))
962
```

```

963
964     *-文字大小
965
966         help textsizestyle
967
968         line open date, text(1324.83 15117 "一个波峰",size(huge)) // 绝对大小
969         line open date, text(1324.83 15117 "一个波峰",size(*1.6)) // 相对大小
970
971
972     *-文字颜色
973
974         help colorstyle
975
976         line open date, text(1324.83 15117 "一个波峰",color(blue))
977         line open date, text(1324.83 15117 "一个波峰",color(black*0.4))
978
979     *-文字行距
980
981         line open date, ///
982             note("SP500指数的时序图"(在此期间, 股市两次大跌! ), ///
983                 color(blue))
984
985         line open date, ///
986             note("SP500指数的时序图"(在此期间, 股市两次大跌! ), ///
987                 color(blue) linegap(2.5))
988
989
990
991     *
992     *-3.2.7 图标类
993
994         help markerlabelstyle
995         help marker_options
996         help marker_label_options
997
998
999     *-3.2.7.1 简介
1000
1001     *-命令结构:   twoway (单元图) , mlabel(文字变量) 其他选项
1002
1003         sysuse lifeexp, clear
1004
1005         do A3_mlabel.do
1006
1007         list lexp gnppc country2 if region==2
1008         scatter lexp gnppc if region==2, mlabel(country2)
1009
1010
1011     *-3.2.7.2 图标的位置
1012
1013     *-整体设定
1014         scatter lexp gnppc if region==2, ///
1015             mlabel(country2) mlabposition(9)
1016         scatter lexp gnppc if region==2, ///
1017             mlabel(country2) mlabp(3)
1018
1019         help clockposstyle
1020
1021         *           11  12  1
1022         *          10      2
1023         *           9   0   3
1024         *           8      4
1025         *           7   6   5
1026
1027     *-个别设定
1028         gen pos = 3
1029         replace pos = 4 if country2=="美国"
1030         replace pos = 1 if country2=="宏都拉斯"
1031
1032         scatter lexp gnppc if region==2, ///
1033             mlabel(country2) mlabvp(pos)
1034         scatter lexp gnppc if region==2, ///
1035             mlabel(country2) mlabvp(pos) ///
1036             xscale(range(-2000 33000))

```

```
1037
1038
1039 *-3.2.7.3 图标的大小
1040
1041 *-标准化大小
1042
1043     help textstyle
1044
1045     scatter lexp gnppc if region==2,      ///
1046             mlabel(country2) mlabvp(pos) ///
1047             mlabtextstyle(heading)
1048
1049 *-任意大小
1050
1051     help textsizestyle
1052
1053     scatter lexp gnppc if region==2,      ///
1054             mlabel(country2) mlabvp(pos) ///
1055             mlabsize(vsmall)
1056
1057     scatter lexp gnppc if region==2,      ///
1058             mlabel(country2) mlabvp(pos) ///
1059             mlabsize(*0.7) // 推荐采用此法!
1060
1061
1062 *-3.2.7.4 图标的角度
1063
1064 * 可以是任意数值
1065 * 0 水平 90 竖直
1066
1067     help anglestyle
1068
1069     scatter lexp gnppc if region==2,      ///
1070             mlabel(country2) mlabvp(pos) ///
1071             mlabangle(15)
1072
1073     scatter lexp gnppc if region==2,      ///
1074             mlabel(country2) mlabvp(pos) ///
1075             mlabangle(-15)                ///
1076             xscale(range(35000) log)
1077
1078     help axis_scale_options
1079
1080
1081 *-3.2.7.5 图标的颜色
1082
1083     help colorstyle
1084
1085     scatter lexp gnppc if region==2,      ///
1086             mlabel(country2) mlabvp(pos) ///
1087             mlabcolor(green)
1088
1089
1090
1091 *
1092 *-3.2.8 其它选项
1093
1094 *-3.2.8.1 分组绘图
1095
1096     help by_option
1097
1098     sysuse auto, clear
1099     scatter mpg weight, by(foreign)
1100     scatter mpg weight, by(foreign, total)
1101     scatter mpg weight, by(foreign, total rows(1))
1102     scatter mpg weight, by(foreign, total cols(1))
1103     scatter mpg weight, by(foreign, total cols(1) style(compact))
1104
1105
1106 *-----一个复杂的示例-----
1107     use comp2001ts, clear
1108     browse
1109     reshape long price, i(date) j(compname) string
1110     browse
```

```

1111
1112     #delimit ;           // 彩色图形
1113     twoway tsline price ,
1114         by(compname, cols(1) yrescale note("") compact)
1115         ylabel(#2, nogrid)
1116         title(" ", box width(130) height(.001) bcolor(ebblue))
1117         subtitle(, pos(5) ring(0) nobexpand nobox color(red))
1118         scheme(s2color) ;
1119     #delimit cr
1120
1121     #delimit ;           // 黑白图形
1122     twoway tsline price ,
1123         by(compname, cols(1) yrescale note("") compact)
1124         ylabel(#3, nogrid)
1125         title(" ", box width(130) height(.001) bcolor(black*0.3))
1126         subtitle(, pos(5) ring(0) nobexpand nobox color(black))
1127         scheme(slmono) ;
1128     #delimit cr
1129     *-----
1130
1131
1132     *-3.2.8.2 重新设置变量标签
1133
1134     help advanced_options
1135
1136     sysuse sp500, clear
1137
1138     twoway line close date,      ///
1139         yvarlabel("收盘价") xvarlabel("交易日期")
1140
1141     twoway line high low date,    ///
1142         yvarlabel("最高价" "最低价") ///
1143         xvarlabel("交易日期")
1144
1145     *-说明：比 legend() 命令要简洁
1146
1147
1148     *-3.2.8.3 重新设置变量显示格式
1149
1150     help advanced_options
1151
1152     twoway line high date, xvarformat(%tdY-n-d) yvarformat(%6.2f)
1153
1154
1155     *-3.2.8.4 重设图形种类
1156
1157     twoway line change date, recast(area)
1158
1159     twoway area change date
1160
1161     twoway (line change date if change>0, recast(spike)) ///
1162         (line change date if change<0, recast(area))
1163
1164     twoway (line change date, recast(area) color(blue)) ///
1165         (line change date if abs(change)<15, recast(area) color(red)), ///
1166         legend(label(1 " |change|>=15" ) label(2 " |change|<15" ))
1167
1168     twoway function y=normalden(x), range(-4 4)
1169
1170     twoway function y=normalden(x), range(-4 4) recast(spike)
1171
1172     twoway (function y=normalden(x), range(-4 4)) ///
1173         (function y=normalden(x), range(-4 -1.96)) ///
1174         recast(area) color(black*0.4)) ///
1175         (function y=normalden(x), range(1.96 4)) ///
1176         recast(area) color(black*0.4)), ///
1177         legend(off)
1178
1179
1180     *-示例：彩色花瓣
1181
1182     doedit A3_area02.do
1183
1184

```



```

1185
1186
1187
1188
1189
1190
1191
1192
1193      *=====
1194      *      计量分析与STATA应用
1195      *=====
1196
1197      *      主讲人：连玉君 博士
1198
1199      *      单 位：中山大学岭南学院金融系
1200      *      电 邮：arlionn@163.com
1201      *      主 页：http://blog.cnfol.com/arlion
1202
1203      *      ::第一部分::
1204      *      Stata 操作
1205      *      =====
1206      *      第三讲 Stata绘图
1207      *      =====
1208      *      -3.3- 元素代号
1209
1210
1211      *-----
1212      *-> 3.3      元素代号
1213      *-----
1214
1215      *      ==本节目录==
1216
1217      *      3.3.1 颜色代号
1218      *      3.3.2 线 相关的代号
1219      *          3.3.2.1 线型代号
1220      *          3.3.2.2 线宽代号
1221      *          3.3.2.3 连接方式代号
1222      *      3.3.3 标记符号的代号
1223      *          3.3.3.1 符号样式
1224      *          3.3.3.2 符号的边界和填充
1225      *          3.3.3.3 符号代号一览
1226      *      3.3.4 文字相关的代号
1227      *          3.3.4.1 文字大小代号
1228      *          3.3.4.2 文字角度代号
1229      *          3.3.4.3 文字对齐方式的代号
1230      *      3.3.5 边距大小的代号
1231
1232
1233      cd `c(sysdir_personal)'Net_course_A\A3_graph
1234
1235
1236      *-----
1237      * -3.3.1 颜色代号
1238
1239      help colorstyle
1240      graph query colorstyle
1241
1242      * 显示特定的颜色
1243      palette color blue brown
1244      palette color olive dkorange
1245
1246      * 颜色模板
1247      palette_all          // 外部命令
1248      palette_all, b(white) // 指定背景，便于对比
1249      palette_all, b(black)
1250
1251      vgc colormap          // 外部命令,效果更佳
1252
1253      clear
1254      full_palette          // 外部命令，附加 RGB 代码，66 种颜色
1255      browse
1256
1257
1258      * 调制自己喜欢的颜色

```

```

1259 *      代码格式      调色方式
1260 * -----
1261 *      # # #      RGB value; white = "255 255 255"
1262 *      # # # #    CMYK value; yellow = "0 0 255 0"
1263 *      color*#    color with adjusted intensity; yellow*1.2
1264 *      *#        default color with adjusted intensity
1265 * -----
1266
1267 *- 三个基准色:
1268 *      red      =    255    0    0
1269 *      green    =      0   255    0
1270 *      blue     =      0    0   255
1271
1272
1273 *-RGB 与 CMYK 之间的转换
1274     colortrans 255 0 0
1275     colortrans 0 255 255 0
1276     ret list
1277
1278
1279
1280 *
1281 *-3.3.2 线 相关的代号
1282
1283     help lines
1284
1285     help line_options
1286
1287
1288 *-3.3.2.1 线型代号
1289
1290     help linepatternstyle
1291     help linestyle
1292
1293     palette linepalette          // 图示
1294
1295     graph query linepatternstyle // 列示代码
1296
1297     twoway function y=normalden(x), range(-4 4) lpattern(longdash)
1298
1299
1300 *-3.3.2.2 线宽代号
1301
1302     help linewidthstyle
1303
1304     graph query linewidthstyle
1305
1306     twoway function y=normalden(x), range(-4 4) lwidth(vthick)
1307
1308
1309 *-3.3.2.3 连接方式代号
1310
1311     help connectstyle
1312
1313     graph query connectstyle
1314
1315     twoway function y=normalden(x), range(-4 4) n(50) ///
1316                 connect(stepstair)
1317
1318
1319
1320 *
1321 *-3.3.3 标记符号的代号
1322
1323     help symbolstyle
1324     palette symbolpalette
1325
1326 *-3.3.3.1 符号样式
1327
1328     sysuse auto, clear
1329     twoway (scatter price weight if foreign, msymbol(T)) ///
1330           (scatter price weight if !foreign, msymbol(dh)), ///
1331           legend(label(1 "国产") label(2 "进口"))
1332

```

```
1333      * 另一种语法格式
1334      sysuse sp500, clear
1335      twoway scatter high low date, msymbol(oh dh)
1336
1337
1338      *-3.3.3.2 符号的边界和填充
1339
1340      * mlcolor(): 边界颜色;  mfcolor(): 填充颜色
1341
1342      sysuse auto, clear
1343      scatter mpg weight, msymbol(0) mlcolor(green) mfcolor(yellow*0.5)
1344
1345
1346      *-3.3.3.3 符号代号一览
1347
1348      help showmarkers
1349
1350      showmarkers, over(msymbol)
1351      showmarkers, over(msymbol) msize(large)
1352      showmarkers, over(msize)
1353      showmarkers, over(mcolor)      // 边界颜色
1354      showmarkers, over(mfcolor)     // 填充颜色
1355      showmarkers, over(mlcolor) mfcolor(gray) ///
1356                          msize(large) mlwidth(medthick)
1357      showmarkers, over(mlwidth) mfcolor(gray) ///
1358                          msize(large) mlcolor(navy)
1359      showmarkers, over(msymbol) scheme(slmono)
1360      showmarkers, over(msymbol) msize(large) ///
1361                          scheme(slmono)
1362
1363
1364
1365
1366      *
1367      *-3.3.4 文字相关的代号
1368
1369      *-3.3.4.1 文字大小代号
1370
1371      help textsizestyle
1372
1373
1374      *-3.3.4.2 文字角度代号
1375
1376      help anglestyle
1377
1378
1379      *-3.3.4.3 文字对齐方式的代号
1380
1381      help justificationstyle // 左右对齐方式
1382      help alignmentstyle    // 上下对齐方式
1383
1384
1385      *
1386      *-3.3.5 边距大小的代号
1387
1388      help marginstyle
1389
1390
1391      *-----关于代号的一个说明-----
1392      * 多数情况下, Stata都支持相对数值, 为我们提供了一种便捷的设定方式
1393      * 如, text("文字",size(*0.5))
1394      *      color(green*0.3))
1395      *      xline(30, lwidth(*1.5))
1396      *-----
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
```

```

1407      *=====
1408      *      计量分析与STATA应用
1409      *=====
1410
1411      *      主讲人：连玉君 博士
1412
1413      *      单  位：中山大学岭南学院金融系
1414      *      电  邮：arlionn@163.com
1415      *      主  页：http://blog.cnfol.com/arlion
1416
1417      *      ::第一部分::
1418      *      Stata 操作
1419      *      =====
1420      *      第三讲 Stata绘图
1421      *      =====
1422      *      -3.4- 常用图形示例
1423      *      (I)
1424
1425
1426      *-----
1427      *-> 3.4  常用图形示例 (I)
1428      *-----
1429
1430      *      ==本节目录==
1431
1432      *      3.4.1 散点图
1433      *      3.4.2 折线图
1434      *      3.4.3 区域图
1435      *      3.4.4 钉形图
1436      *      3.4.5 直方图
1437      *      3.4.6 密度函数图
1438      *      3.4.7 累积分布函数图
1439
1440      cd `c(sysdir_personal)'Net_course_A\A3_graph
1441
1442
1443
1444      *-导言：帮助文件的使用
1445
1446      * 各类图形的选项分为两类：`专属选项' 和 `公共选项'
1447      * 公共选项可以参考上面的说明进行填写
1448      * 专属选项通常较少，也容易填写
1449
1450      help twoway bar
1451      help twoway lfit
1452      help twoway scatter
1453
1454
1455      *-----
1456      *-3.4.1 散点图
1457
1458      help twoway scatter
1459
1460      sysuse uslifeexp2, clear
1461      #delimit ;
1462      scatter le year,
1463              title("图1：散点图示例")
1464              subtitle("预期寿命，美国")
1465              yvarlabel(预期寿命)
1466              xvarlabel(年份)
1467              note("1")
1468              caption("数据来源： 美国国家重要统计资料报告，
1469                    第5卷-第6期")
1470              scheme(economist);
1471      #delimit cr
1472
1473
1474
1475      *-----
1476      *-3.4.2 折线图
1477
1478      help line
1479
1480      *-注意：需要对 x 变量排序

```

```

1481     sysuse auto, clear
1482     line mpg weight
1483     line mpg weight, sort
1484
1485
1486     *-一个较复杂的例子
1487
1488     do A3_line.do
1489
1490     *-----A3_line.do -----
1491     sysuse uslifeexp, clear
1492     gen diff = le_wmale - le_bmale
1493     label var diff "寿命差异"
1494     #delimit ;
1495     twoway (line le_wmale year, yaxis(1 2) xaxis(1 2))
1496            (line le_bmale year)
1497            (line diff year)
1498
1499            ,
1500            ylabel(0 20(10)80, gmax angle(0))
1501            ylabel(0(5)20, axis(2) gmin angle(0))
1502            xlabel(1918, axis(2))
1503            title("图2: 白人和黑人预期寿命")
1504            subtitle("美国, 1900-1999")
1505            ytitle("预期寿命 (年)")
1506            xtitle("年份")
1507            ytitle("", axis(2))
1508            xtitle("", axis(2))
1509            note("数据来源: 美国国家重要统计资料报告, 第5卷-第6期"
1510                "(1918 巨降: 源于1918年全国性流行感冒)", linegap(1.2))
1511            legend(label(1 "白人男性") label(2 "黑人男性")
1512                  rows(1) size(*0.7));
1512     #delimit cr
1513     *-----
1514
1515
1516
1517     *-----
1518     *-3.4.3 区域图
1519
1520     help twoway area
1521
1522     *-事实上是折线图的变形, 无非是在折线下方的区域内涂上颜色而已!
1523
1524     sysuse gnp96, clear
1525     twoway line d.gnp96 date, yline(0,lc(black*0.4) lp(dash))
1526     twoway area d.gnp96 date
1527
1528     * 一个相对完整的示例
1529     #delimit ;
1530     twoway area d.gnp96 date,
1531            xlabel(36(8)164, angle(45))
1532            ylabel(-100(50)200, angle(0))
1533            ytitle("Billions of 1996 Dollars")
1534            xtitle("")
1535            subtitle("Change in U.S. GNP", position(11))
1536            note("Source: U.S. Department of Commerce,
1537                Bureau of Economic Analysis") ;
1538     #delimit cr
1539
1540
1541
1542     *-----
1543     *-3.4.4 钉形图
1544
1545     help twoway spike // 简单钉形图
1546     help twoway rspike // 区域钉形图
1547
1548     *-多用于股票数据
1549
1550     sysuse sp500, clear
1551     twoway spike high date
1552     twoway rspike high low date
1553     twoway (rspike hi low date) (line close date) in 1/57
1554

```

```

1555  *--完整示例
1556
1557  *--e.g 1
1558      sysuse sp500, clear
1559      replace volume = volume/1000
1560      #delimit ;
1561          twoway (rspike hi low date)
1562              (line close date)
1563              (bar volume date, barw(.25) yaxis(2))
1564              in 1/57
1565      ,
1566      yscale(axis(1) r(900 1400))
1567      yscale(axis(2) r( 9 45))
1568      ylabel(, axis(2) grid)
1569      ytitle("股价 -- 最高, 最低, 收盘", place(top))
1570      ytitle("交易量 (百万股)", axis(2) bexpand just(left))
1571      xtitle(" ")
1572      legend(off)
1573      subtitle("S&P 500", margin(b+2.5))
1574      note("数据来源: 雅虎财经! ");
1575  #delimit cr
1576
1577  *--e.g 2
1578      sysuse sp500, clear
1579      replace volume = volume/10000
1580      twoway (rarea high low date)          ///
1581          (spike volume date, yaxis(2)),    ///
1582          legend(span)
1583  *--改进
1584      twoway (rarea high low date)          ///
1585          (spike volume date, yaxis(2)),    ///
1586          legend(span)                    ///
1587          yscale(range(500 1400) axis(1))    /// // new!
1588          yscale(range(0 5) axis(2))         /// // new!
1589          ylabel(,angle(0))                 /// // new!
1590          ylabel(,angle(0) axis(2))
1591
1592
1593
1594  *-----
1595  *--3.4.5 直方图
1596
1597      help histogram
1598
1599  *--概览
1600      sysuse nlsw88.dta, clear
1601      histogram wage
1602      gen ln_wage = ln(wage)
1603      histogram ln_wage          // 对数转换后往往更符合正态分布
1604
1605  *--图形的纵坐标
1606      histogram wage             // 长条的高度对应样本数占总样本的比例,
1607                                // 总面积为 1
1608      graph save g0.gph, replace
1609      histogram wage, fraction    // 将长条的高度总和限制为 1
1610      graph save g_frac.gph, replace
1611      histogram wage, frequency  // 纵坐标为对应的样本数, 而非比例
1612      graph save g_freq.gph, replace
1613      graph combine g0.gph g_frac.gph g_freq.gph, rows(1)
1614
1615  *--其他选项
1616      histogram ttl_exp, normal  // 附加正态分布曲线
1617      histogram wage, kdensity   // 附加密度函数曲线
1618      histogram wage, addlabels  // 每个长条上方附加一个表示其高度的数字
1619      histogram wage, by(race)
1620
1621  *--离散变量的直方图
1622      histogram grade
1623      graph save d1, replace
1624      histogram grade, discrete  // 离散变量的直方图必须附加 discrete 选项
1625      graph save d2, replace
1626      graph combine d1.gph d2.gph
1627
1628  *--长条的显示

```

```

1629     histogram wage, gap(50)
1630     histogram wage, gap(90) scheme(slmono)
1631     histogram wage, gap(99.9) scheme(slmono) blwidth(thick)
1632
1633     *-分组绘制直方图
1634     sysuse auto, clear
1635     histogram mpg, percent discrete ///
1636         by(foreign, col(1) note(分组指标: 汽车产地) ///
1637             title("图3: 不同产地汽车里数") ///
1638             subtitle("直方图") ///
1639         ) ///
1640     ytitle(百分比) xtitle(汽车里数)
1641
1642
1643     *-一个较复杂的例子
1644
1645     do A3_histogram.do
1646
1647     *-----A3_histogram.do-----
1648     sysuse sp500, clear
1649     #delimit ;
1650     histogram volume, freq normal
1651         addlabels addlabopts(mlabcolor(blue))
1652         xaxis(1 2)
1653         ylabel(0(10)65, grid)
1654         xlabel( 12321 "mean"
1655             9735 "-1 s.d."
1656             14907 "+1 s.d."
1657             7149 "-2 s.d."
1658             17493 "+2 s.d."
1659             20078 "+3 s.d."
1660             22664 "+4 s.d."
1661             ,axis(2) grid gmax
1662         )
1663         subtitle("图4: S&P 500 交易量 (2001年1月-12月)")
1664         ytitle(频数)
1665         xtitle("交易量(千笔)") xscale(titlegap(2))
1666         xtitle("", axis(2))
1667         note("数据来源: 雅虎! 财经数据");
1668     #delimit cr
1669     *-----
1670
1671
1672
1673     *-----
1674     *-3.4.6 密度函数图
1675
1676
1677     *-Kernal 密度函数图
1678
1679     help kdensity
1680
1681     sysuse nlsw88, clear
1682     kdensity wage
1683     kdensity wage, normal
1684
1685     *-把多个变量的核密度函数图绘制在一张图上
1686     sysuse sp500, clear
1687     twoway (kdensity open) (kdensity low)
1688     twoway (kdensity open) (kdensity high) (kdensity low) (kdensity close)
1689
1690     *-比较不同子样本的密度函数
1691     sysuse auto, clear
1692     kdensity weight, nograph generate(x dx)
1693     kdensity weight if foreign==0, nograph generate(dx0) at(x)
1694     kdensity weight if foreign==1, nograph generate(dx1) at(x)
1695     label var dx "all cars"
1696     label var dx0 "Domestic cars"
1697     label var dx1 "Foreign cars"
1698     line dx dx0 dx1 x, sort lw(*2.5 *1.5 *1.5)
1699
1700     *-另一种方法
1701     sysuse auto, clear
1702     kdensity weight , nograph gen(p_x d_x)

```

```
1703      kdensity weight if foreign==0, nograph gen(p_x0 d_x0)
1704      kdensity weight if foreign==1, nograph gen(p_x1 d_x1)
1705      label var d_x  "all cars"
1706      label var d_x0 "Domestic cars"
1707      label var d_x1 "Foreign cars"
1708      twoway (line d_x p_x) (line d_x0 p_x0) (line d_x1 p_x1)
1709
1710      *-附加置信区间      -akdensity- 外部命令 SJ 3(2):148--156
1711      sysuse auto, clear
1712      akdensity length, stdbands(2)
1713
1714
1715      *-双变量联合密度函数图  -kdens2- 外部命令
1716
1717      help kdens2
1718
1719      use grunfeld, clear
1720      gen linv = log(invest)
1721      gen lmkt = log(mvalue)
1722
1723      kdens2 linv lmkt
1724      kdens2 linv lmkt, n(100)          // defaults Min(_N,50)
1725      kdens2 linv lmkt, xw(.5) yw(.5)  // defaults `optimal'
1726
1727
1728      *-Furthur reading:
1729      *   Cox, N., 2005,
1730      *   Speaking Stata: Density probability plots,
1731      *   Stata Journal, 5(2): 259-273.
1732
1733
1734
1735      *-----
1736      *-3.4.7 累积分布函数图
1737
1738      help cumul
1739
1740      *-基本概念
1741      sysuse auto, clear
1742      cumul price, gen(pcum)
1743      line pcum price, sort
1744
1745      sort price
1746      list price pcum in 1/5
1747      dis 1/74
1748      dis 3/74
1749
1750      list price pcum in 70/74
1751      dis 72/74
1752      dis 73/74
1753
1754
1755      *-更为简洁的命令  -displot- (外部命令)
1756
1757      help distplot
1758
1759      sysuse auto, clear
1760      distplot scatter mpg
1761      distplot line mpg, by(foreign)
1762      distplot connected mpg, trscale(ln(@))
1763
1764      *-支持的图形种类
1765      * area bar connected dot dropline line scatter spike
1766
1767      foreach t in area bar connected dot dropline line scatter spike {
1768          distplot `t' mpg, by(foreign)
1769      }
1770
1771
1772      *-cdfplot- 命令
1773
1774      help cdfplot
1775
1776      sysuse auto, replace
```



```
1777      cdfplot length, normal
1778      cdfplot length, by(foreign)
1779      cdfplot length, by(foreign) norm saving(mygraph, replace)
1780
1781      *-示例：对数转换的作用
1782      sysuse nlsw88, clear
1783      cdfplot wage, normal
1784      gen ln_wage = ln(wage)
1785      cdfplot ln_wage, normal
1786
1787
1788      *-Furthur reading:
1789      *   Cox, N., 2004,
1790      *   Speaking stata: Graphing distributions,
1791      *   STATA JOURNAL, 4(1): 66-88.
1792      *   Cox, N., 2004,
1793      *   Speaking Stata: Graphing categorical and compositional data,
1794      *   STATA JOURNAL, 4(1): 190-215.
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805      *=====
1806      *           计量分析与STATA应用
1807      *=====
1808
1809      *           主讲人：连玉君 博士
1810
1811      *           单   位：中山大学岭南学院金融系
1812      *           电   邮：arlionn@163.com
1813      *           主   页：http://blog.cnfol.com/arlion
1814
1815      *           ::第一部分::
1816      *           Stata 操作
1817      *           =====
1818      *           第三讲 Stata绘图
1819      *           =====
1820      *           -3.4- 常用图形示例
1821      *                   (II)
1822
1823      *-----
1824      *-> 3.4   常用图形示例 (II)
1825      *-----
1826
1827      *           ==本节目录==
1828
1829      *           3.4.8  线性/非线性 拟合图
1830      *           3.4.9  矩阵图：显示变量间的相关性
1831      *           3.4.10 柱状图
1832      *                   3.4.10.1 一维柱状图
1833      *                   3.4.10.2 二维柱状图
1834      *           3.4.11 点 图
1835      *           3.4.12 函数图
1836      *           3.4.13 合图示例
1837      *           3.4.14 三维图形
1838      *           3.4.15 地 图
1839
1840
1841      cd `c(sysdir_personal)'Net_course_A\A3_graph
1842
1843
1844      *-----
1845      *-3.4.8 线性/非线性 拟合图
1846
1847      help twoway lfit
1848      help twoway qfit
1849
1850      *-简单示例
```

```

1851     sysuse auto, clear
1852     scatter mpg weight || lfit mpg weight
1853     scatter mpg weight || lfit mpg weight, by(foreign, total row(1))
1854
1855
1856     *-附加置信区间
1857     help twoway lfitci
1858     help twoway qfitci
1859
1860     twoway (lfitci mpg wei, stdf) (scatter mpg wei) // 线性拟合的置信区间
1861
1862     twoway (scatter mpg wei) (lfitci mpg wei, stdf) // 图层的概念
1863
1864     twoway (qfitci mpg wei, stdf) (scatter mpg wei) // 非线性拟合
1865
1866     twoway (qfitci mpg wei, stdf level(99) color(yellow)) ///
1867             (qfitci mpg wei, stdf level(90))           ///
1868             (scatter mpg wei)                          // 置信水准
1869
1870
1871
1872     *-----
1873     *-3.4.9 矩阵图：显示变量间的相关性
1874
1875     help graph matrix
1876
1877     sysuse auto, clear
1878     graph matrix mpg weight length
1879     pwcorr mpg weight length
1880     graph matrix mpg weight length, ///
1881             diag("mpg(汽车里数)" . "length (汽车长度)")
1882
1883     *-整体缩放
1884     graph matrix mpg weight length, scale(1.5)
1885     graph matrix mpg weight length, scale(0.8)
1886
1887     *-图标
1888     sysuse citytemp, clear
1889     sum
1890     graph matrix heatdd-tempjuly
1891     gr mat heatdd-tempjuly, msymbol(point)
1892     help symbolstyle
1893
1894     *-半边显示
1895     gr mat heatdd-tempjuly, ms(p) half
1896
1897     *-坐标刻度和标签
1898     gr mat heatdd-tempjuly, ms(p) half ///
1899             maxes(ylab(#4) xlab(#4))
1900
1901     *-附加网格线
1902     gr mat heatdd-tempjuly, ms(p) half ///
1903             maxes(ylab(#4, grid) xlab(#4, grid))
1904
1905
1906
1907     *-----
1908     *-3.4.10 柱状图
1909
1910
1911     *-3.4.10.1 一维柱状图 (参见第二讲)
1912
1913     help graph bar
1914
1915     *-命令格式1:
1916     *   graph bar yvars ...
1917
1918     *-命令格式2:
1919     *   graph bar (mean) varlist, over(g1) over(g2)... [other options]
1920
1921     *-基本用法: graph bar yvars ...
1922     sysuse nlsw88, clear
1923     graph bar wage, over(race)
1924

```

```

1925
1926     *-组变量的设定
1927
1928     sysuse nlsw88, clear
1929
1930     graph bar (mean) wage, over(race) scheme(slmono)
1931
1932     graph bar (mean) wage, over(smsa) over(married) over(collgrad)
1933
1934     #delimit ;
1935     graph bar (mean) wage, over(smsa) over(married) over(collgrad)
1936         title("Average Hourly Wage, 1988, Women Aged 34-46")
1937         subtitle("by College Graduation, Marital Status,
1938             and SMSA residence")
1939         note("Source: 1988 data from NLS, U.S. Dept. of Labor,
1940             Bureau of Labor Statistics");
1941     #delimit cr
1942
1943     *-柱体的样式
1944     help barlook_options
1945
1946     graph bar (mean) wage hours, over(race) over(married) ///
1947         scheme(slmono) ///
1948         bar(1, bstyle(p1)) ///
1949         bar(2, bstyle(p6))
1950
1951     *-柱体的标签
1952     help blabel_option
1953
1954     graph bar (mean) wage, over(race) over(married) ///
1955         blabel(bar, position(outside) format(%3.1f) color(green))
1956
1957     graph hbar (mean) wage, over(industry) over(married) ///
1958         blabel(bar, position(outside) format(%3.1f) ///
1959             color(blue) size(vsmall))
1960
1961     *-累加柱体
1962     sysuse educ99gdp, clear
1963     graph hbar (mean) public private, over(country)
1964     graph hbar (mean) public private, over(country) stack
1965
1966     *-完整示例
1967     generate total = private + public
1968     #delimit ;
1969     graph hbar public private, stack
1970         over(country, sort(total) descending)
1971         blabel(bar, posi(center) color(white) format(%3.1f))
1972         title("Spending on tertiary education as % of GDP, 1999",
1973             span pos(11))
1974         subtitle(" ")
1975         note("Source: OECD, Education at a Glance 2002", span) ;
1976     #delimit cr
1977
1978     *-进一步美化
1979     generate frac = private/(private + public)
1980     #delimit ;
1981     graph hbar public private, stack percent
1982         over(country, sort(frac) descending)
1983         blabel(bar, posi(center) color(white) format(%3.1f))
1984         title("Public and private spending on tertiary education, 1999",
1985             span pos(11) )
1986         subtitle(" ")
1987         note("Source: OECD, Education at a Glance 2002", span);
1988     #delimit cr
1989
1990
1991     *-重叠柱体
1992     sysuse nlsw88, clear
1993     graph bar (mean) hours wage, over(race) over(married)
1994     graph bar (mean) hours wage, over(race) over(married) bargap(-30)
1995
1996     *-图形的比例
1997     sysuse nlsw88, clear
1998     graph hbar wage, over(ind, sort(1)) over(collgrad)

```

```

1999      graph hbar wage, over(ind, sort(1)) over(collgrad) ///
2000              ysize(4) xsize(8)
2001
2002
2003      *-3.4.10.2 二维柱状图
2004
2005      help twoway bar
2006
2007      sysuse sp500, clear
2008
2009      twoway bar change date in 1/100
2010      twoway bar change date in 1/100, barwidth(0.6)
2011
2012
2013      sysuse pop2000, clear
2014      replace maletotal = -maletotal
2015      twoway bar maletotal agegrp, horizontal || ///
2016              bar femtotal agegrp, horizontal
2017
2018      *-一个较复杂的例子
2019      sysuse pop2000, clear
2020      replace maletotal = -maletotal/1e+6
2021      replace femtotal = femtotal/1e+6
2022      gen zero = 0
2023      #delimit ;
2024      twoway
2025              (bar maletotal agegrp, horizontal xvarlab(Males))
2026              (bar femtotal agegrp, horizontal xvarlab(Females))
2027              (scatter agegrp zero, mlabel(agegrp) mlabcolor(black) msymbol(i))
2028      , xtitle("Population in millions") ytitle("")
2029      plotregion(style(none))
2030      ysca(noline) ylabel(none)
2031      xsca(noline titlegap(-3.5))
2032      xlabel(-12 "12" -10 "10" -8 "8" -6 "6" -4 "4" 4(2)12,
2033              tlength(0) grid gmin gmax)
2034      legend(label(1 Males) label(2 Females))
2035      legend(order(1 2))
2036      title("US Male and Female Population by Age, 2000")
2037      note("Source: U.S. Census Bureau, Census 2000")
2038      ;
2039      #delimit cr
2040
2041      *-解析:
2042      scatter agegrp zero
2043      scatter agegrp zero, mlabel(agegrp) mlabcolor(black) msymbol(i)
2044
2045
2046
2047      *-----
2048      *-3.4.11 点图
2049
2050      help graph dot
2051
2052      *-事实上是柱状图的另一种表示方法, 比较适合中文投稿, 省墨!
2053
2054      sysuse nlsw88, clear
2055      graph dot wage, over(occ) by(collgrad)
2056      graph dot wage, over(occ, sort(1)) by(collgrad)
2057
2058      *-一个相对完整的示例
2059      sysuse nlsw88, clear
2060      #delimit ;
2061      graph dot wage, over(occ, sort(1))
2062              by(collgrad,
2063                  title("Average hourly wage, 1988, women aged 34-46", span)
2064                  subtitle(" "))
2065              note("Source: 1988 data from NLS, U.S. Dept. of Labor,
2066      Bureau of Labor Statistics", span)
2067              );
2068      #delimit cr
2069
2070
2071
2072      *-----

```

```

2073  *--3.4.12 函数图
2074
2075      help twoway function
2076
2077      twoway function y=normalden(x), range(-4 4) n(15)
2078
2079      twoway function y=normalden(x), range(-4 4) dropline(-1.96 1.96)
2080
2081      twoway function y=normalden(x), range(-4 4) xline(-1.96 1.96)
2082
2083      twoway function y=normalden(x), range(-4 4) dropline(-1.96 1.96) horizon
2084
2085      twoway function y=exp(-x/6)*sin(x), range(0 12.57) ///
2086          xlabel(0 3.14 "pi" 6.28 "2 pi" 9.42 "3 pi" 12.57 "4 pi") ///
2087          ylabel(0, lstyle(foreground)) dropline(1.48) ///
2088          plotregion(style(none)) ///
2089          xsca(noline) ytitle(" ") xtitle(" ")
2090
2091      sysuse sp500, clear
2092      twoway (scatter open close, msize(*.35) mcolor(*.8)) ///
2093          (function y=x, range(close) yvarlab("y=x") clwidth(*1.5))
2094
2095
2096  *--综合示例
2097
2098      do A3_function_ci90.do
2099
2100  *-----A3_function_ci90.do-----
2101      #delimit ;
2102      twoway
2103          function y=normden(x), range(-4 -1.96) color(gsl2) recast(area)
2104          || function y=normden(x), range(1.96 4) color(gsl2) recast(area)
2105          || function y=normden(x), range(-1.96 -1.64) color(green) recast(area)
2106          || function y=normden(x), range(1.64 1.96) color(green) recast(area)
2107          || function y=normden(x), range(-4 4) lstyle(foreground)
2108          || ,
2109          plotregion(style(none))
2110          legend(off)
2111          xlabel(-4 "-4 sd" -3 "-3 sd" -2 "-2 sd" -1 "-1 sd" 0 "mean"
2112              1 "1 sd" 2 "2 sd" 3 "3 sd" 4 "4 sd"
2113              , grid gmin gmax)
2114          xtitle("");
2115      #delimit cr
2116  *-----
2117
2118
2119
2120
2121  *-----
2122  *--3.4.13 合图示例 -graph combine-
2123
2124  *--例 1:
2125
2126      do A3_eg1.do
2127
2128  *-----A3_eg1.do-----
2129      sysuse lifeexp, clear
2130
2131      gen loggnp = log10(gnppc)
2132      label var loggnp "人均GNP(Log10)"
2133      label var lexp "期望寿命"
2134
2135      scatter lexp loggnp, ysca(alt titlegap(1.5)) ///
2136          xsca(alt titlegap(0.8)) ///
2137          xlabel(, grid gmax) ///
2138          ylabel(,angle(0)) ///
2139          saving(yx, replace)
2140      histogram lexp, percent xsca(alt reverse titlegap(0.8)) ///
2141          horiz xtitle(占比) ylabel(,angle(0)) ///
2142          saving(hy, replace)
2143      histogram loggnp, percent ysca(alt reverse titlegap(1.5)) ///
2144          ytitle(占比) ylabel(,nogrid angle(0)) ///
2145          xscale(titlegap(2)) xlabel(,grid gmax) ///
2146          saving(hx, replace)

```

```

2147
2148     graph combine hy.gph yx.gph hx.gph, ///
2149         hole(3) imargin(0 0 0 0) ///
2150         graphregion(margin(l=12 r=12)) ///
2151         title("图1: 期望寿命与人均 GNP") ///
2152         note("资料来源: 世界银行小组, 1988")
2153 *-----
2154
2155
2156 *-进一步美化
2157 *-----A3_eg1.do----modify-----
2158     sysuse lifeexp, clear
2159
2160     gen loggnp = log10(gnppc)
2161     label var loggnp "人均GNP(Log10)"
2162     label var lexp "期望寿命"
2163
2164     scatter lexp loggnp, ysca(alt titlegap(1.5)) ///
2165         xsca(alt titlegap(0.8)) ///
2166         xlabel(, grid gmax) ///
2167         ylabel(,angle(0)) ///
2168         saving(yx, replace)
2169     histogram lexp, percent xsca(alt reverse titlegap(0.8)) ///
2170         horiz xtitle(占比) ylabel(,angle(0)) ///
2171         saving(hy, replace) ///
2172         fysize(25) // new! fy
2173     histogram loggnp, percent ysca(alt reverse titlegap(1.5)) ///
2174         ytitle(占比) ylabel(,nogrid angle(0)) ///
2175         xscale(titlegap(2)) xlabel(,grid gmax) ///
2176         saving(hx, replace) ///
2177         fysize(25) // new! fx
2178
2179     graph combine hy.gph yx.gph hx.gph, ///
2180         hole(3) imargin(0 0 0 0) ///
2181         graphregion(margin(l=12 r=12)) ///
2182         title("图1: 期望寿命与人均 GNP" ) ///
2183         subtitle(" ", size(*0.5)) /// new! a blank line
2184         note("资料来源: 世界银行小组, 1988")
2185 *-----
2186 *-解释:
2187 * fysize(#) 仅将 x 轴方向缩小为原始尺寸的 25%
2188 * fysize(#) 仅将 y 轴方向缩小为原始尺寸的 25%
2189
2190
2191 *-例 2:
2192     sysuse sp500, clear
2193     replace volume = volume/1000
2194     twoway rarea high low date, name(hilo, replace)
2195     twoway spike volume date, name(vol, replace)
2196     graph combine hilo vol
2197
2198 *-美化 I
2199     graph combine hilo vol, cols(1)
2200
2201 *-美化 II
2202     twoway rarea high low date, ///
2203         xscale(off) name(hilo, replace) // new! off
2204     graph combine hilo vol, cols(1)
2205     graph combine hilo vol, cols(1) imargin(b=1 t=1)
2206
2207 *-美化 III
2208     twoway spike volume date, name(vol, replace) ///
2209         ylabel(5 15 25) fysize(25) // new! fysize
2210     graph combine hilo vol, cols(1) imargin(b=1 t=1)
2211
2212
2213
2214 *-----
2215 *-3.4.14 三维图形 -surface- 外部命令
2216
2217     clear
2218     set obs 900
2219     gen x = int((_n - mod(_n-1,30) -1 ) /30 )
2220     gen y = mod(_n-1,30)

```

```
2221     gen z = normalden(x,10,3)*normalden(y,15,5)
2222     surface x y z
2223
2224
2225
2226 *-----
2227 *-3.4.15 地图
2228
2229 *-tmap- 命令
2230
2231 *-参考资料
2232
2233 *-查看最新资料
2234     findit tmap
2235
2236 *-说明文档和范例
2237     *-SJ 4(4):361-378
2238     view browse http://www.stata.com/support/faqs/graphics/tmap.html
2239     shellout tmap.mht // 范例网页
2240     shellout tmap2-userguide.pdf // -tmap- 的说明书
2241
2242 *-相关辅助命令
2243     doedit usmaps.do // module to provide US state map coordinates for tmap
2244     findit usmaps2 // module to provide US county map coordinates for tmap
2245
2246 *-范例
2247
2248     use Us-Database.dta, clear
2249
2250     tmap choropleth murder, id(id) map(Us-Coordinates.dta)
2251
2252     tmap cho murder if conterminous, id(id) map(Us-Coordinates.dta)
2253
2254     tmap cho murder if conterminous, id(id) ocolor(white) ///
2255         map(Us-Coordinates.dta) palette(Blues) ///
2256         title("`"Murders per 100,000 population"'"') ///
2257         subtitle("United States 1994")
2258
2259     tmap propsymbol murder if conterminous, ///
2260         x(x_coord) y(y_coord) map(Us48-Coordinates.dta) ///
2261         sshape(o) scolor(edkblue) fcolor(eltblue) ///
2262         title("`"Murders per 100,000 population"'"') ///
2263         subtitle("United States 1994")
2264
2265     tmap deviation murder if conterminous, ///
2266         x(x_coord) y(y_coord) map(Us48-Coordinates.dta) ///
2267         sshape(s) scolor(sienna) fcolor(eggshell) ///
2268         title("`"Murders per 100,000 population"'"') ///
2269         subtitle("United States 1994")
2270
2271     tmap label label if conterminous, ///
2272         x(x) y(y) map(Us48-Coordinates.dta) ///
2273         lc(white) ls(0.9) fc(emerald)
2274
2275     use MilanoPolice-Database.dta, clear
2276     tmap dot, x(x) y(y) map(MilanoOutline-Coordinates.dta) ///
2277         by(type) marker(both) sshape(s d) ///
2278         title("Location of police stations") ///
2279         subtitle("Milano 2004") legtitle("Police force", ///
2280             size(*0.7)) legbox(lc(black))
2281
2282
2283 *-spmap- 命令
2284
2285 *-使用说明:
2286
2287     view browse http://www.stata.com/support/faqs/graphics/spmap.html
2288
2289     shellout spmap_intro.mht
2290
2291     help spmap
2292
2293     use "Italy-RegionsData.dta", clear
2294     spmap relig1 using "Italy-RegionsCoordinates.dta", id(id) ///
```



```

2295         clnumber(20) fc(Greens2) oc(white ..) osize(medthin ..) ///
2296         title("Pct. Catholics without reservations", size(*0.8)) ///
2297         subtitle("Italy, 1994-98" " ", size(*0.8)) ///
2298         legstyle(3) legend(ring(1) position(3)) ///
2299         plotregion(icolor(stone)) graphregion(icolor(stone))
2300
2301     use "Italy-RegionsData.dta", clear
2302     spmap relig1 using "Italy-RegionsCoordinates.dta", id(id) ///
2303     clmethod(stdev) clnumber(5) ///
2304     title("Pct. Catholics without reservations",size(*0.8)) ///
2305     subtitle("Italy, 1994-98" " ", size(*0.8)) area(pop98) ///
2306     note(" " ///
2307         "NOTE: Region size proportional to population", size(*0.75))
2308
2309
2310     *-中国地图
2311
2312     findit china map
2313
2314     use china_label,clear
2315     tab name
2316     replace name = subinstr(name, "省", "", .)
2317     replace name = subinstr(name, "市", "", .)
2318     replace name = subinstr(name, "回族自治区", "", .)
2319     replace name = subinstr(name, "壮族自治区", "", .)
2320     replace name = subinstr(name, "特别行政区", "", .)
2321     replace name = subinstr(name, "自治区", "", .)
2322     replace name = subinstr(name, "维吾尔", "", .)
2323     tab name
2324     gen x = uniform()
2325     format x %9.3g
2326
2327     spmap x using "china_map.dta", id(id) ///
2328     label(label(name) ///
2329         xcoord(x_coord) ycoord(y_coord) size(*.9)) ///
2330     plotregion(icolor(stone)) graphregion(icolor(stone)) ///
2331     clnumber(8) fc(Greens2) oc(white ..) osize(medthin ..)
2332
2333
2334     *-其它命令
2335
2336     findit spgrid
2337     doedit spgrid_example.do // 构建地图网格
2338
2339
2340
2341
2342     *-----
2343     *-3.5 结语
2344
2345     *-学会帮助画天下!
2346
2347     *-一本有用的书
2348
2349     * Mitchell, M.
2350     *   A visual guide to Stata graphics.
2351     *   Stata Press, 2008.
2352
2353     view browse ///
2354         "http://www.stata.com/support/faqs/graphics/gph/statagraphs.html"
2355
2356
2357
2358     *-练习：一个尚未搞定的圆圈
2359
2360     twoway ( function y = sqrt(1-x^2),          ///
2361             plotregion(margin(0))              ///
2362             range(-1.5 1.5) lc(blue) )          ///
2363     ( function y = -sqrt(1-x^2),                ///
2364             plotregion(margin(0))              ///
2365             range(-1.5 1.5) lc(blue) )          ///
2366     ,                                             ///
2367         ysize(2) xsize(2)                      ///
2368         ylabel(-1.5 1.5) xlabel(-1.5 1.5)

```



```
2369
2370     *-方案 1:
2371     twoway ( function y = sqrt(1-(x-1)^2),    ///
2372             plotregion(margin(0))            ///
2373             range(-0 2) lc(blue) )           ///
2374             ( function y =-sqrt(1-(x-1)^2),    ///
2375             plotregion(margin(0))            ///
2376             range(0 2) lc(blue))            ///
2377             ,                                  ///
2378             ysize(3) xsize(3)                ///
2379             ylabel(-1.5 1.5) xlabel(-1 2)
2380
2381     *-方案 2:
2382     clear
2383     set obs 100000
2384     gen z = invnorm(uniform())
2385     gen y = sin(z)
2386     gen x = cos(z)
2387     twoway (scatter y x), ysize(4) xsize(4)
2388     twoway (scatter y x, msymbol(smcircle)), ysize(4) xsize(4)
2389
2390
2391
2392     *-----OVER-----
2393
```

```
1
2
3
4
5      *      -----
6      *      -----  计量分析与STATA应用  -----
7      *      -----
8
9      *      主讲人：连玉君 博士
10
11     *      单   位：中山大学岭南学院金融系
12     *      电   邮：arlionn@163.com
13     *      主   页：http://blog.cnfol.com/arlion
14
15     *      ::第一部分::
16     *      Stata 操作
17     *      =====
18     *      第四讲   矩  阵
19     *      =====
20
21     *cd D:\stata11\ado\personal\Net_course_A\A4_matrix
22     cd `c(sysdir_personal)'Net_course_A\A4_matrix
23
24
25     *-----
26     *      本讲目录
27     *-----
28     * 4.1  矩阵的基本操作
29     * 4.2  矩阵运算
30     * 4.3  矩阵的解析
31     * 4.4  关于矩阵的进一步说明
32
33
34
35     *-----
36     *->4.1  矩阵的基本操作
37     *-----
38
39     *      ==本节目录==
40
41     *      4.1.1  基本定义方式
42     *      4.1.2  矩阵的管理
43     *      4.1.2.1  矩阵的名称
44     *      4.1.2.2  列示矩阵
45     *      4.1.2.3  矩阵的行数和列数
46     *      4.1.2.4  查找/删除矩阵
47     *      4.1.2.5  查验矩阵中是否存在缺漏值
48     *      4.1.3  矩阵的行名和列名
49     *      4.1.4  选取部分矩阵
50     *      4.1.4.1  选取1个元素：1*1矩阵
51     *      4.1.4.2  选取子矩阵
52     *      4.1.4.3  矩阵元素的修改
53     *      4.1.5  更一般化的矩阵定义
54     *      4.1.6  常用矩阵的定义
55     *      4.1.6.1  单位矩阵
56     *      4.1.6.2  常数矩阵
57     *      4.1.6.3  元素为随机数的矩阵
58     *      4.1.6.4  对角矩阵
59     *      4.1.7  变量和矩阵的相互转换
60     *      4.1.7.1  变量->矩阵
61     *      4.1.7.2  矩阵->变量
62     *      4.1.8  用矩阵存储统计结果
63     *      4.1.8.1  以矩阵方式呈现tabstat命令的结果
64     *      4.1.8.2  更一般化的矩阵存储
65     *      4.1.9  采用变量的方式操作矩阵
66     *      4.1.9.1  对矩阵中的各列进行变换和运算
67     *      4.1.9.2  矩阵元素的数学变换
68     *      4.1.10 矩阵的保存和调入
69     *      4.1.10.1 将矩阵保存为 .dta 文档中
70     *      4.1.10.2 将矩阵保存到 txt, word, excel 文档中
71
72
73     *      本节命令
74     *-----
```

```
75 * matrix, matrix dir, matrix list, matrix rename, matrix drop
76 * matmissing(), rowsof(), colsof(), matuniform(), diag(),
77 * rownames, colnames, rownumb(), colnumb()
78 * mat_capp, mat_rapp, mat_order
79 * roweq, coleq, mkmat, svmat, set matsize,
80 * mat accum, mat glsaccum, mat opaccum
81 *-----
82
83 *
84 *-4.1.1 基本定义方式
85
86 *-简介(stata中的数据可以视为矩阵)
87 sysuse auto, clear
88 keep in 1/10
89 keep price mpg weight length
90 list
91
92 *-规则: 逗号分列 反斜线分行
93 matrix a = (1,2,3 \ 4,5,6)
94 mat list a
95 matrix b = (-1.3, 2.6 \ 3.89, 0.42 \ 50.1, -0.634)
96 mat list b
97 matrix c = (-10 \ -5 \ -8 \ 3 \ 5.6 \ 9)
98 mat list c
99 matrix d = (-10,-5,-8,5.6,9)
100 mat list d
101 matrix e = (1,2,3,4,5 \ 2,3,4,5,6 \ 3,4,5,6,7 \ 4,5,6,7,8 \ 5,6,7,8,9)
102 mat list e
103
104
105 *
106 *-4.1.2 矩阵的管理
107
108 *-4.1.2.1 矩阵的名称
109
110 * 可以和内存中的变量同名
111 mat price = (2,3)
112 * 不可以和单值重名, 虽然不会提示错误信息, 但会自动覆盖
113 * 在数学运算中, 如果表达式中出现一个既是变量名称又是矩阵名称的名称,
114 * stata会将其解释为变量名称。
115 clear
116 set obs 100
117 gen x = 5
118 mat x = J(3,3,2)
119 sum x
120
121 *-矩阵更名
122 mat dir
123 matrix rename a MM
124 mat dir
125
126 *-4.1.2.2 列示矩阵
127 mat list MM
128 mat list b // 元素的默认显示格式为: %10.0g
129 mat list b, format(%3.1f)
130 mat list e
131 mat list e, nohalf
132 mat list e, nohalf nonames
133 mat list e, nonames title("一个5*5的对称矩阵")
134
135
136 *-matlist 命令 (更为灵活的设定方式)
137 * 主要用于编程, 呈现结果
138
139 *-eg1--
140 matrix A = (1, 2 \ 3, 4 \ 5, 6)
141 matrix list A
142 matlist A
143 matlist A, border(rows) rowtitle(rows) left(4)
144 matlist 2*A, border(all) lines(none) format(%6.1f) names(rows) ///
145 twidth(8) left(4) title(Guess what, a title)
146
147 *-eg2--
148 #delimit ;
```

```

149     matrix Htest = ( 12.30,  2,  .00044642 \
150                     2.17,  1,  .35332874 \
151                     8.81,  3,  .04022625 \
152                     20.05, 6,  .00106763 ) ;
153     #delimit cr
154     matrix rownames Htest = trunk length weight overall // 定义行名
155     matrix colnames Htest = chi2 df p // 定义列名
156     matrix list Htest
157     matlist Htest // 比较两种结果的差异
158     * 更为细致的呈现方式
159     matlist Htest, title("检验结果") rowtitle("变量名称") ///
160         cspec(o4& %12s | %8.0g & %5.0f & %8.4f o2&) rspec(&-&&-- )
161
162     /* 上述命令的含义
163     -----
164     Element      Purpose      Description
165     -----
166     o4&          before column 1      4 spaces/no vertical line
167     %12s         display format column 1      string display format %12s
168     |           between columns 1 and 2      1 space/vertical line/1 space
169     %8.0g        display format column 2      numeric display format %8.0g
170     &           between columns 2 and 3      1 space/no vertical line/1 space
171     %5.0f        display format column 3      numeric display format %5.0f
172     &           between columns 3 and 4      1 space/no vertical line/1 space
173     %8.4f        display format column 4      numeric display format %8.4f
174     o2&         after column 4              2 spaces/no vertical line
175     &-&&--       首行上方无横线，首行下方有横线，最后一个行上下方均有横线
176     -----
177     */
178
179     * 修改上述表格的呈现方式
180     matlist Htest, title("检验结果") rowtitle("变量名称") ///
181         cspec(o4| %12s | %8.0g | %5.0f | %8.4f o2|) rspec(--&&-- )
182
183     * 进一步修改
184     matlist Htest, title("检验结果(New)") rowtitle("变量名称") ///
185         cspec( o4&o2 %10s | b t %8.0g & %4.0f & i c %7.4f o2& ) ///
186         rspec( & - & & - & )
187     *-说明:
188     * (1) b t %8.0g 第二列 加粗(bold), 绿色(text color)
189     * (2) i c %7.4f 第四列 斜体(italic), 白色(command color)
190
191     *-4.1.2.3 矩阵的行数和列数
192     matrix a = (1,2,3 \ 4,5,6)
193     display colsof(d)
194     display rowsof(c)
195     scalar ra = rowsof(a)
196     scalar ca = colsof(a)
197     dis in g "矩阵 a 的行数是: " in y ra
198     dis in g "矩阵 a 的列数是: " in y ca
199
200
201
202     *-4.1.2.4 查找/删除矩阵
203
204     *-查找矩阵
205     mat dir
206
207     *-删除矩阵 (这个其实没有必要)
208     mat drop MM
209     *mat drop _all
210
211
212     *-4.1.2.5 查验矩阵中是否存在缺漏值
213     mat list e
214     display matmissing(e)
215     mat e[2,3] = .
216     mat list e
217     display matmissing(e)
218
219
220     *
221     *-4.1.3 矩阵的行名和列名
222

```

```
223 mat A = (1,2,3,4,5 \ 2,3,4,5,6 \ 3,4,5,6,7 \ 4,5,6,7,8 \ 5,6,7,8,9)
224 mat rownames A = 1998 1999 2000 2001
225 mat colnames A = y x1 x2 x3
226 mat list A
227
228 mat r = rownumb(A, "2000")
229 mat c = colnumb(A, "x1")
230 mat list r
231 mat list c
232
233
234 *
235 *-4.1.4 选取部分矩阵
236
237 *-4.1.4.1 选取1个元素: 1*1矩阵
238 matrix a = (1,2,3 \ 4,5,6)
239 mat list a
240 mat a1 = a[1,1]
241 mat list a1
242 mat a4 = a[2,1]
243 mat list a4
244
245 *-4.1.4.2 选取子矩阵
246 mat list e, nohalf
247 mat ec3 = e[1..3,3]
248 mat list ec3
249 mat e3c = e[....,3]
250 mat list e3c
251 mat e34 = e[3...,4...]
252 mat list e
253 mat list e34
254
255 *-4.1.4.3 矩阵元素的修改
256 matrix a = (1,2,3 \ 4,5,6)
257 mat list a
258 mat a[1,2] = -10
259 mat list a
260 mat a[2,2] = (-9, 20)
261 mat list a
262
263
264 *
265 *-4.1.5 更一般化的矩阵定义
266
267 * 矩阵中的每一个元素都可以视为一个1*1维矩阵,
268 * 所以矩阵的操作可以分块进行
269
270 mat a1 = (1, 2, 3 \ 42, 50, 63)
271 mat a2 = (-3,-5,-7 \ -9 , -11, -13)
272 mat list a1
273 mat list a2
274
275 mat aa = [a1, a2] // 横向合并两个矩阵
276 mat list aa
277 mat aaa = [a1 \ a2] // 纵向追加两个矩阵
278 mat list aaa
279
280
281 *-更为直观的定义方式
282 mat_capp a1_a2 : a1 a2 // 横向合并
283 mat list a1_a2
284 mat_rapp ala2 : a1 a2 // 纵向追加
285 mat list ala2
286 * 注意: 上述命令中, 冒号前必须有一个空格
287
288
289 *
290 *-4.1.6 常用矩阵的定义
291
292 *-4.1.6.1 单位矩阵
293 mat I = I(5)
294 mat list I
295
296 *-4.1.6.2 常数矩阵
```

```
297     mat r1 = J(5,5,1)
298     mat r2 = J(2,6,-3)
299     mat list r1
300     mat list r2
301
302 * -----
303 * 一个实例：差分矩阵
304 * 构造
305
306     mat B = J(4,5,0)
307     mat B[1,1] = -1*I(4)
308     mat B1 = B
309     mat B = J(4,5,0)
310     mat B[1,2] = I(4)
311     mat B2 = B
312     mat B = B1 + B2
313     mat list B1
314     mat list B2
315     mat list B
316 * 应用
317     mat cc = J(5,5,1) + 2*I(5)
318     mat rownames cc = 1998 1999 2000 2001 2002 // 定义矩阵的行名
319     mat list B, nonames
320     mat list cc, nohalf
321     mat dd = B*cc
322     mat list dd
323     mat rownames dd = 1999 2000 2001 2002
324     mat list dd
325 * -----
326
327 *--一般化定义
328     local T = 10
329     mat B = J(`T'-1,`T',0)
330     mat B[1,1] = -1*I(`T'-1)
331     mat B1 = B
332     mat B = J(`T'-1,`T',0)
333     mat B[1,2] = I(`T'-1)
334     mat B2 = B
335     mat B = B1 + B2
336     mat list B1
337     mat list B2
338     mat list B
339
340
341 *--4.1.6.3 元素为随机数的矩阵
342     *set seed 13699
343     mat r3 = matuniform(10,4)
344     mat list r3
345
346
347 *--4.1.6.4 对角矩阵
348     mat u = J(5,1,-0.5)
349     mat list u
350     mat du = diag(u) // 取出对角元素
351     mat list du
352     mat v = diag(matuniform(5,1)) // 一个任意的5*5对角矩阵
353     mat list v
354
355
356
357 *
358 *--4.1.7 变量和矩阵的相互转换
359
360 *--4.1.7.1 变量->矩阵 -mkmat-
361
362 * 转换单变量为同名列向量
363     sysuse auto,clear
364     mkmat price in 1/10 // 生成一个 10*1 的列向量，矩阵名为 price
365     mat list price
366
367     mkmat price weight length if rep78==4 // 生成三个同名列向量
368     mat list price
369     mat list weight
370     mat list length
```

```

371
372 * 将多个变量合并至一个矩阵
373 mkmat price, matrix(Y)
374 gen cons= 1
375 mkmat weight length foreign cons, mat(X)
376 mat list Y
377 mat list X
378
379 * 应用实例: OLS 系数估计
380 mat b = inv(X'*X)*X'*Y
381 mat list b
382 reg price weight length foreign
383
384 * 缺漏值的处理
385 count if price>10000
386 replace price =. if price>10000
387 count if weight>4000
388 replace weight =. if weight>4000
389 mkmat price wei, mat(pw)
390 dis rowsof(pw)
391 mkmat price wei, mat(pw_no) nomissing // 仅包含非缺漏值
392 dis rowsof(pw_no)
393 list price weight if price==.|wei==.
394 count if price==.|wei==.
395
396
397 *-4.1.7.2 矩阵->变量 -svmat- -xvmat-
398
399 svmat b, names(coff)
400 list coff1 in 1/5
401 svmat X, names(var) // 自行定义统一的变量名
402 drop weight length foreign cons
403 svmat X, names(col) // 用矩阵的列名作为变量的名称
404
405 *-xsvmat 命令 (svmat的拓展)
406 sysuse nls88, clear
407 xi: reg wage hours ttl_exp i.race
408 mat covmat = e(V) // 方差-协方差矩阵
409 mat list covmat
410
411 xsvmat covmat, list(,) // 以变量方式列示矩阵的内容
412 xsvmat covmat, rowname(xvar) rowlab(label) list(, abbr(32))
413
414
415 *
416 *-4.1.8 用矩阵存储统计结果 -makematrix- -tabstatmat-
417
418 *-4.1.8.1 以矩阵方式呈现tabstat命令的结果 -tabstatmat-
419 *-eg1-
420 sysuse auto, clear
421 tabstat price mpg weight length, save
422 tabstatmat A
423 mat list A
424 *-eg2-
425 tabstat price mpg weight length, save ///
426 by(foreign) stat(mean p50 sd min max) format(%6.3f)
427 tabstatmat A
428 mat list A, format(%6.3f)
429
430 *-4.1.8.2 更一般化的矩阵存储 -makematrix-
431 sysuse auto, clear
432 makematrix, from(r(mean) r(sd) r(skewness)) : ///
433 sum price trunk length weight, detail
434
435 makematrix A, from(_b[_cons] _b[mpg] e(r2) e(r2_a)) ///
436 lhs(rep78-foreign) format(%4.3f) : ///
437 regress mpg
438 mat list A
439
440 sysuse nls88, clear
441 makematrix B, ///
442 from(_b[_cons] _b[married] _b[age] _b[south] ///
443 _b[ttl_exp] e(r2) e(r2_a)) ///
444 lhs(wage hours) /// // 被解释变量

```

---

```

445         format(%4.3f) list sep(0) divider:      ///
446         regress married age south ttl_exp      // 解释变量
447         mat B = B'
448         mat colnames B = wage hours
449         mat list B
450
451         use xtcs.dta, clear
452         makematrix, from(_b[_cons] _b[tobin] _se[tobin] ///
453         e(r2) e(r2_w) e(F_f))                  ///
454         lhs(tl sl ll) format(%6.5f) :          ///
455         xtreg fr-tobin, fe
456
457         sysuse auto, clear
458         makematrix, from(r(rho)) :              ///
459         spearman head trunk length displacement weight
460         spearman head trunk length displacement weight // 对比一下
461
462         *- from()选项中可以执行数学运算
463         makematrix, from(r(rho)^2) format(%4.3f) : ///
464         spearman head trunk length displacement weight
465
466
467     *
468     *-4.1.9 采用变量的方式操作矩阵      -mgen-
469
470     *-4.1.9.1 对矩阵中的各列进行变换和运算，如加总、相除等
471     clear
472     mat drop _all
473     matrix a = (1,2,3 \ 4,5,6)
474     mat list a
475     mgen v1=c1+c2 v2=c2+c3, in(a) out(z)
476     mat list z
477
478     *-4.1.9.2 矩阵元素的数学变换
479     mgen ln_c1=ln(c1) exp_c2=exp(c2), in(a) out(c)
480     mat list c
481     *- 基于这一思路，我们可以对矩阵中的元素进行数学变换
482     *- 如下数学函数都可以使用：
483     help math functions
484
485
486     *
487     *-4.1.10 矩阵的保存和调入      -matsave-, -matload-, -mat2txt-
488
489     *-4.1.10.1 将矩阵保存为 .dta 文档中
490
491     *- 基本思路：
492     *   matsave
493     *   把矩阵转换为变量(参见4.1.7.2小节),然后保存为 .dta 文件
494     *   matload
495     *   把 .dta 文件调入，然后将变量转换为矩阵(参见4.1.7.1小节)
496
497     *-说明：
498     * (1) 多数情况下，我们都无需保存矩阵，只需保存do文档即可；
499     * (2) 极少数情况下，要通过非常耗时的计算才能得到某个矩阵，
500     *     而这个矩阵可能还会参与后续运算，此时需要保存；
501
502     *- 矩阵的保存: matsave
503     sysuse auto, clear
504     reg price weight length mpg
505     eret list
506     mat COV = e(V)
507     *-基本用法
508     matsave COV          // 错误命令
509     matsave COV, dropall replace // 正确命令
510
511     *- 矩阵的调入: matload
512     mat dir              // 当前内存中已经有一个 COV 矩阵
513     matload COV, overwrite dropall // 覆盖当前内存中的同名矩阵
514
515
516     *-4.1.10.2 将矩阵保存到 txt, word, excel 文档中      -mat2txt-, -dataout-
517
518     sysuse nls88, clear

```

---



```

519
520    *-基本统计量
521    tabstat wage age ttl_exp hours, stats(N mean sd min max) c(s) save
522    tabstatmat A
523    mat A = A' // 使结果与tabstat一致
524    *-保存为txt格式
525    mat2txt, matrix(A) saving(mytable01) replace ///
526    title("Table 1: statistics of key variables")
527    shellout mytable01.txt
528
529    *-相关系数矩阵
530    makematrix R, from(r(rho)) : spearman wage age ttl_exp hours
531    *-追加结果到 mytable01.txt 文档中
532    mat2txt, matrix(R) saving(mytable01) append ///
533    title("Table 2: correlation of key variables")
534    dataout using mytable01.txt, word excel replace // 转换为word,excel格式
535
536    *-练习：请进一步将回归结果追加到上述文件中
537
538    *-其它处理方式：
539    * 参见 A1_intro 第【10.1.1小节】 输出基本统计量
540
541
542
543
544
545
546
547
548
549
550    * -----
551    * ----- 计量分析与STATA应用 -----
552    * -----
553
554    * 主讲人：连玉君 博士
555
556    * 单 位：中山大学岭南学院金融系
557    * 电 邮：arlionn@163.com
558    * 主 页：http://blog.cnfol.com/arlion
559
560    * ::第一部分::
561    * Stata 操作
562    * =====
563    * 第四讲 矩阵操作
564    * =====
565    * -4.2- 矩阵的运算
566
567
568    *-----
569    *-4.2 矩阵的运算
570    *-----
571
572    help matrix operators
573
574    * ==本节目录==
575
576    * 4.2.1 矩阵的基本运算
577    * 4.2.1.1 加、减、乘
578    * 4.2.1.2 直乘
579    * 4.2.1.3 哈式乘法
580    * 4.2.1.4 矩阵元素的数学变换
581    * 4.2.1.5 矩阵与单值的运算
582    * 4.2.2 矩阵的转置
583    * 4.2.3 矩阵的逆矩阵
584    * 4.2.3.1 矩阵的行列式
585    * 4.2.3.2 矩阵求逆
586    * 4.2.4 矩阵的向量化
587    * 4.2.5 矩阵的对角值
588    * 4.2.6 交乘矩阵的定义
589    * 4.2.6.1 简单交乘矩阵
590    * 4.2.6.2 加权交乘矩阵
591    * 4.2.6.3 用户自行设定的权重
592    * 4.2.6.3 特殊加权交乘矩阵

```

```

593
594
595 *                               本节命令
596 *-----
597 * hadamard(), inv(), issym(), det(), trace(), vecdiag()
598 * diag(), math(), vec(), mgen(), + - * / #
599 *-----
600
601 *           Operator           Symbol
602 *           -----
603 *           parentheses        ( )
604 *           transpose          '
605 *           negation            -
606 *           Kronecker product  #
607 *           division by scalar /
608 *           multiplication     *
609 *           subtraction        -
610 *           addition           +
611 *           column join        ,
612 *           row join           \
613 *           -----
614
615
616 *
617 *-4.2.1 矩阵的基本运算
618
619   *-4.2.1.1 加(+)、减(-)、乘(*)
620
621   matrix e   = J(5,5,3)
622   matrix I5  = 5*I(5)
623   mat list e, nohalf
624   mat list I5
625
626   * 加法
627   mat add = e + I5
628   mat list add, nohalf
629
630   mat add1 = e + 2           // 错误方式
631   mat add1 = e + J(5,5,2)
632   mat list add1
633
634   * 减法
635   mat sub = e - I5
636   mat list sub, nohalf
637
638   * 乘法
639   mat prod= e*I5
640   mat list prod
641
642
643   *-4.2.1.2 直乘
644
645   *-定义:
646
647   *           [ a11*B  a12*B  ... a1k*B ]
648   *           [ a21*B  a22*B  ... a2k*B ]
649   *   A # B = [   .      .      ...   . ]
650   *           [   .      .      ...   . ]
651   *           [ an1*B  an2*B  ... ank*B ]
652
653   *--eg1-----
654   mat one = J(4,1,1)
655   mat I1  = I(5)
656   mat kro = I1 # one
657   mat list one
658   mat list I1
659   mat list kro
660
661   *--eg2-----
662   mat xx = J(3,3,-1)
663   mat kro2 = I1 # xx
664   mat list xx, nonames nohalf
665   mat list I1, nonames nohalf
666   mat list kro2, nohalf

```

```

667
668      *--eg3-----
669      mat a = (1,2 \ 3,4 \ 5, 6)
670      mat kro3 = a # xx
671      mat list a
672      mat list xx, nohalf
673      mat list kro3
674
675      *-直乘的性质:
676      *
677      * (1)  (A # B)' = A' # B'
678      *
679      * (2)  inv(A # B) = inv(A) # inv(B)
680      *
681      * (3)  |A # B| = |A|^k * |B|^n  (A是nXn矩阵, B是kXk 矩阵)
682      *
683      * (4)  tr(A # B) = tr(A)*tr(B)
684      *
685      * (5)  a*b' = a # b' = b' # a
686      *
687      * (6)  (a # B)*C = a # B*C
688      *
689      * (7)  A*(b'#C) = b'#AC
690      *
691      * (8)  (A#b)*C = AC#b
692      *
693      * (9)  A(B#c') = AB#c'
694      *
695      * (10) a'b*CD = (a'#C)*(b#D)
696
697      * 练习: 请使用stata命令验证上述性质。
698
699
700      *-4.2.1.3 哈式乘法: 元素对元素的乘法
701
702      mat a = (1,2 \ 3,4 \ 5, 6)
703      mat b = (-1,4 \ 0,1 \ -3,12)
704      mat aHb = hadamard(a,b)
705      * 呈现结果
706      mat m = J(3,1,.)
707      mat R = (a, m, b, m, aHb)
708      mat list R
709
710
711      *-4.2.1.4 矩阵元素的数学变换
712
713      *-整体变换
714      help math          // arlion 自行编写的程序
715
716      mat a = J(4,5,8)
717      math ln_a = ln(a)  // 矩阵元素取对数
718      mat list a
719      mat list ln_a
720
721      math exp_a = exp(a) // 矩阵元素取幂
722      mat list exp_a
723
724      sysuse auto, clear
725      reg price wei len foreign
726      mat V = e(V)
727      mat list V
728      mat se2 = vecdiag(e(V))
729      math se = sqrt(se2) // Arlion 自编程序
730      mat se0 = vecdiag(cholesky(diag(vecdiag(e(V)))))
731      mat list se
732      mat list se0
733
734      viewsource math.ado
735
736      *-操作过程详解:
737      view browse http://www.ats.ucla.edu/stat/stata/faq/elemmatrix.htm
738
739      *-可供调用的函数如下:
740      help math functions

```

```
741
742
743     *-分列变换
744     help mgen    // 详见 4.1.9.1 小节
745
746     mgen v1=ln(c1) v2=exp(c2) v3=sin(c3), in(a) out(b)
747     mat list a
748     mat list b
749
750     * 特别注意: mgen后的各项表达式以空格区分,
751     *           所以, "v1=ln(c1)" 不可以写为 "v1 = ln(c1)"
752
753
754     *-4.2.1.5 矩阵与单值的运算
755
756     scalar c = 5
757     mat D = J(4,4,1)
758     mat list D
759
760     mat Dc = D*c
761     mat list Dc
762
763     mat cD = c*D
764     mat list cD
765
766     mat D_c = D/c
767     mat list D_c
768
769
770     *
771     *-4.2.2 矩阵的转置: 行列互换
772
773     matrix A = (-1, 2 \ 3, 4 )
774     matrix B = ( 4, 1, 2, 5 )
775     mat      C = (4,1 \ 2, 5)
776
777     mat list A
778     mat At = A'
779     mat list At
780
781     mat list B
782     mat Bt = B'
783     mat list Bt
784
785     * 公式:  $(A*C)' = C'*A' \neq A'*C'$ 
786     mat ACt = (A*C)'
787     mat AtCt = A'*C'
788     mat CtAt = C'*A' // 转置运算优先于乘法运算
789     mat list ACt
790     mat list CtAt
791     mat list AtCt
792
793
794     *
795     *-4.2.3 矩阵的逆矩阵
796
797     *-4.2.3.1 矩阵的行列式: 描述矩阵特征的一个统计量
798
799     mat A = (-1, 2 \ 3, 4)
800     mat list A
801     scalar detA = det(A)
802     dis detA
803     dis -1*4 - 3*2
804
805     *= 性质:
806     * (1) 若A不可逆, 则  $|A|=0$ , 反之亦然
807     * (2)  $|A'| = |A|$ 
808     * (3)  $|A*B| = |A| * |B|$ 
809     * (4)  $|5*A| = 5^n * |A|$ 
810     *
811     * (5)  $\begin{vmatrix} A & 0 \\ 0 & B \end{vmatrix} = |A| * |B|$ 
812
813
814
```

```

815
816     *-4.2.3.2 矩阵求逆
817
818     dis issym(A)           // 判断一个矩阵是否为对称矩阵
819     mat invA = inv(A)
820     mat IA   = A*invA
821     mat list A
822     mat list invA
823     mat list IA
824
825
826     *
827     *-4.2.4 矩阵的向量化
828
829     *- 向量化矩阵 类似于变量操作中 stack 命令
830     mat A = (-1, 2 \ 3, 4)
831     mat vA = vec(A)
832     mat list A
833     mat list vA
834
835     *- 向量化方阵的对角元素
836     mat E = e + 0.9*I(5)
837     mat dA = vecdiag(A)
838     mat dE = vecdiag(E)
839     mat list A
840     mat list dA
841     mat list E
842     mat list dE
843     *-例:
844     sysuse auto, clear
845     reg price wei len foreign
846     mat b = e(b)
847     mat V = e(V)
848     mat list V
849     mat se2 = vecdiag(e(V))
850     mat list se2
851     mat se2 = diag(se2)           // 向量的对角化
852     mat list se2
853     mat se = cholesky(se2)       // 裘氏分解
854     mat list se
855     mat t = diag(b)*inv(se)
856     mat list t
857     reg price wei len foreign // 验证一下
858
859
860     *- 矩阵向量化的性质
861
862     * 1.  $\text{vec}(ABC) = (C' \# A) \text{vec}(B)$ 
863     *
864     * 2.  $\text{vec}(ab') = b \# a$ 
865     *
866     * 3.  $\text{vec}(a' \# B) = a \# \text{vec}(B)$ 
867     *
868     * 4.  $\text{vec}(a \# B) = (I_k \# a \# I_n) \text{vec}(B) = \text{vec}(B \# a')$  (B是  $n \times k$  矩阵)
869     *
870     * 5.  $\text{tr}(AB) = \text{vec}'(A') \text{vec}(B) = \text{vec}'(B') \text{vec}(A)$ 
871     *
872     * 6.  $\text{tr}(ABCD) = \text{vec}'(A)(B \# D') \text{vec}(C') = \text{vec}'(A')(D' \# B) \text{vec}(C)$ 
873     *
874     * 7.  $a'BCDF = (c' \# a' \# D)[\text{vec}(B) \# F]$ 
875     *
876     * 8.  $Abc'D = (b' \# I_n) \text{vec}(A) \text{vec}'(D')(c' \# I_m) = (b' \# I_n \# c')[\text{vec}(A) \# D]$ 
877     * 其中, A是 $n \times k$ 矩阵, D是 $m \times j$ 矩阵
878     *  $I_n$ 表示 $n \times n$ 单位阵
879
880     * 练习: 请采用stata命令验证上述性质
881
882
883
884     *
885     *-4.2.5 矩阵的对角值(trace)
886
887     *-定义: 方阵的对角元素之和
888

```

```

889  *--性质:
890  * (1)  $\text{tr}(AB) = \text{tr}(BA)$  // 要求: A,B可乘
891  * (2)  $\text{tr}(cA) = c*\text{tr}(A)$  // c 是单值
892
893  *--示例:
894  matrix Atr = trace(A)
895  scalar Etr = trace(e)
896  mat list A
897  mat list Atr
898  mat list e
899  dis Etr
900
901
902
903  *
904  *--4.2.6 交乘矩阵的定义
905
906  * [P] matrix accum -- Form cross-product matrices
907
908  help matrix accum
909
910  *--4.2.6.1 简单交乘矩阵 -matrix accum-, -matrix vecaccum-
911
912  *--应用背景
913  *
914  * OLS估计:  $b = \text{inv}(X'X)*X'y$ 
915  *
916  * x 是一个 N*K 维矩阵,
917  * 当N较大时(如N=20000), 将超过stata矩阵的上限(11000)
918  * 但  $X'X$  则是一个较小的矩阵, 维度为: K*K
919  *
920  *-- matrix accum 的定义
921  *
922  * matrix accum (A) =  $A'A$  其中,  $A = (x1,x2,x3,.....)$ 
923  *
924  *-- matrix vecaccum 的定义
925  *
926  * matrix vecaccum(A) =  $x1'*X$  其中,  $X = (x2,x3,.....)$ 
927  *
928  *-- 几个重要选项:
929  * (1) noconstant 不在 x 矩阵中自动附加常数项;
930  * (2) deviation 采用离差的形式
931
932  *--eg1- 线性模型的 OLS 估计
933
934  *--目的: 求取  $b = \text{inv}(X'X)*X'y$ 
935  * 其中,  $y = \text{price}$ ,
936  *  $X = (\text{weight}, \text{mpg}, \text{Cons})$ 
937
938  * 方法1: 结合使用 matrix accum 和 matrix vecaccum
939  sysuse auto, clear
940  mat accum XX = weight mpg
941  mat vecaccum yX = price weight mpg
942  mat Xy = yX' // 这里要注意
943  mat b = inv(XX)*Xy
944  mat list b
945  reg price weight mpg, noheader // 检验上述结果
946
947  * 方法2: 仅使用 matrix accum 命令
948  * 思路: 若  $A = (y, X)$ , 则
949  *
950  * 
$$\text{mat accum } (A) = S = (y, X)'(y, X) = \begin{bmatrix} y'y & y'X \\ X'y & X'X \end{bmatrix}$$

951  *
952  *
953  *
954  * 其中, x 的最后一列会被自动加入常数项
955  * 可见,  $X'X$  和  $X'y$  矩阵都可以从 S 矩阵中抽取
956  matrix accum S = price weight mpg // y=price, X=[weight mpg 1]
957  mat list S
958  matrix XX = S[2..., 2...]
959  mat list XX
960  matrix Xy = S[2..., 1]
961  mat b = inv(XX)*Xy
962  mat list b

```

```

963         reg price weight mpg,nohead           // 检验上述结果
964
965     *-eg2- 获取变量的相关系数矩阵
966         sysuse auto, clear
967         corr price weight mpg length
968         ret list
969     *-自行生成矩阵
970         matrix accum R = price weight mpg length, noconstant deviation
971         matrix R = corr(R)
972         mat list R, format(%6.4f)
973
974
975     *-4.2.6.2 加权交乘矩阵    -mat glsaccum-
976
977     * 用于生成 GLS 估计中的相关矩阵
978     *
979     *-mat glsaccum 的定义
980     *
981     *      mat glsaccum(X) = S = X'BX
982     *
983     * 其中, B 为权重矩阵, 定义如下:
984     *
985     *      [ w_1    0    ...    0 ]
986     *      [ 0    w_2    ...    0 ]
987     *      [ .      .      .      . ]
988     *      [ .      .      .      . ]
989     *      [ 0      0    ...    w_k ]
990     *
991     * w_k(k=1,2,...,K) 表示第 k 组观察值的权重矩阵, 是一个方阵
992     *
993     * 若 X 也根据组别定义, 则可表示为:
994     *
995     *      [ x_1 ]
996     *      [ x_2 ]
997     *      [ .   ]
998     *      [ .   ]
999     *      [ x_k ]
1000     *
1001     * 由此可以更为细致的了解到 glsaccum 的定义方式:
1002     *
1003     *      X'BX = X1'W1X1 + X2'W2X2 + ... + X_k'*W_k*X_k
1004
1005     *- 应用举例: White(1980) 异方差稳健性标准误的计算
1006     *
1007     *      Var(b) = inv(X'X)*(X'WX)*inv(X'X)    // White(1980)稳健性方差-协方差矩阵
1008     *
1009     * 其中,
1010     *
1011     *      [ e1^2    0    ...    0 ]
1012     *      [ 0    e2^2    ...    0 ]
1013     *      [ .      .      .      . ]
1014     *      [ .      .      .      . ]
1015     *      [ 0      0    ...    eN^2 ]    NXN 矩阵
1016     *
1017     * ei 表示第 i 个观察值对应的残差
1018     *
1019     * 问题的关键: 求得 (X'WX) 矩阵即可, 可采用 -mat glsaccum- 命令
1020
1021     *-1 获得OLS估计值
1022         sysuse auto, clear
1023         mat accum XX = wei len mpg
1024         mat vecaccum Xy = price wei len mpg
1025         mat Xy = Xy'
1026         mat b = inv(XX)*Xy           // 系数的 OLS 估计值
1027         mat list b
1028
1029     *-2 求取残差之平方向量: e2
1030         mkmat price, mat(y)
1031         gen cons = 1
1032         mkmat wei len mpg cons, mat(X) // 注意附加常数项
1033         mat e = y - X*b               // 残差向量
1034         mat colnames e = c1
1035         mgen e2=c1^2, in(e) out(e2)   // 权重: 残差的平方项
1036

```

```

1037      *-3 求取 (X'WX) 矩阵
1038      gen id = _n          // 最简单的情况：每个观察值归属于一个组别
1039      sort id
1040      mat e2 = diag(e2)    // 将残差向量变换为对角方阵
1041      mat glsaccum XWX = wei len mpg, group(id) glsmat(e2) row(id)
1042      mat list XWX
1043
1044      *-4 求取稳健性标准误
1045      mat var_b = inv(XX)*XWX*inv(XX)          // 计算 White(1980) 估计式
1046      mat se_rob = cholesky(diag(vecdiag(var_b))) // 对角元素开根号，求得[s.e.]
1047      mat se_rob = se_rob/sqrt(70/74)          // 调整自由度
1048      mat list b
1049      mat list se_rob
1050      reg price wei len mpg, robust nohead      // 验证一下
1051
1052      *-5 计算 t 值
1053      mat t_rob = diag(b)*inv(se_rob)          // t-value = b/se
1054      mat list t_rob
1055
1056
1057      *-4.2.6.3 用户自行设定的权重
1058
1059      *- mat (vec)accum 与 mat glsaccum 的关系
1060
1061      * 上述三个命令所返回的矩阵具有如下一般形式：
1062      *
1063      *      X1'*B*X2
1064      *
1065      * (1) mat accum:      X1=X2, B=I  ==>  X'X
1066      * (2) mat glsaccum:  X1=X2      ==>  X'BX
1067      * (3) mat vecaccum:  B=I, X1 是一个列向量, x2是一个矩阵
1068      *                      ==>  y'X
1069
1070      *- 自行指定权重
1071
1072      *-基本思想：
1073      *      X1'*B*X2 可采用一般化形式表示为
1074      *      X1'W1*B*W1*X2, 其中 W1 = W^{1/2}
1075      *
1076      *      若用户不自行设定权重，则 W = I
1077      *      若用户自行设定权重，如 pweights(v), 则 W = diag(v)
1078      *      此处，v 是一个变量
1079      *
1080      *-用途：若设定 B=I, X1=X2, 由于 W1*W1' = W, 则上式可表示为：
1081      *      X'*W*X
1082      *      这与 mat glsaccum 命令返回的矩阵形式相似，
1083      *      区别在于我们可以通过变量 v 来设定权重矩阵，
1084      *      而不必采用矩阵的形式来设定
1085
1086      *-eg：一个难题的解决：用-mat accum-替代 -mat glsaccum-
1087
1088      *-参见：
1089      *      view browse http://statalist.org/archive/2002-10/msg00144.html
1090
1091      *-问题：
1092      *      在上述 mat glsaccum 命令中：
1093      *      mat glsaccum XWX = wei len mpg, group(id) glsmat(e2) row(id)
1094      *      我们必须设定 glsmat() 选项，以便指定权重矩阵，
1095      *      其中，e2 是一个 NXN 矩阵，
1096      *      然而，当 N=11000，或更大的数值时，我们是无法够造出 e2 矩阵的
1097      *
1098      *-解决方法：
1099      *      使用如下替代命令：
1100      *      mat accum H = wei len mpg [pw=e2], noc
1101      *      此处，e2 是一个变量，所以可以避免上述问题
1102      *
1103      *-示例检验：
1104      *      sysuse auto, clear
1105      *      reg price wei len mpg
1106      *      predict e, res
1107      *      gen e2 = e^2          // 权重序列
1108
1109      *-方法1: mat glsaccum 命令
1110      *      mkmat wei len mpg, mat(X)
1111      *      mkmat e2, mat(B)

```



```

1111      mat B = diag(B)
1112      mat S = X'*B*X
1113      mat list S
1114      *-方法2: mat accum 命令, 附加 [pw] 副指令
1115      mat accum H = wei len mpg [pw=e2], noc
1116      mat list H
1117
1118
1119      *-4.2.6.3 特殊加权交乘矩阵      -mat opaccum-
1120
1121      * 同样用于生成 GLS 估计中的相关矩阵
1122      * mat opaccum 可以视为 mat glsaccum 的特例
1123      *
1124      * mat glsaccum 的定义方式:
1125      *
1126      *   A = X'BX = X1'W1X1 + X2'W2X2 + ... + X_k'*W_k*X_k
1127      *
1128      * 这里的权重矩阵 wi 具有一般化的定义方式 (想想异方差和序列相关情形)
1129      * 在很多情况下, wi 具有比较特殊的形式, 如某个变量的外积(outer product):
1130      *
1131      *       Wi = e_i*e_i'
1132      *
1133      * 其中, e_i 是一个 n_ix1 矩阵, n_i 是第 i 个公司的样本数
1134
1135      *       N   [
1136      *       SUM [ (X_i)'e_i(e_i)'X_i ]
1137      *       i=1 [
1138
1139      *-eg:
1140      use maccumxmpl.dta, clear
1141      xtides
1142      mat opaccum A = x1 x2, opvar(e) group(id)
1143      mat list A
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155      *
1156      * ----- 计量分析与STATA应用 -----
1157      *
1158
1159      *       主讲人: 连玉君 博士
1160
1161      *       单   位: 中山大学岭南学院金融系
1162      *       电   邮: arlionn@163.com
1163      *       主   页: http://blog.cnfol.com/arlion
1164
1165      *
1166      *       ::第一部分::
1167      *       Stata 操作
1168      *       =====
1169      *       第四讲 矩阵操作
1170      *       =====
1171      *       -4.3- 矩阵的解析
1172
1173      *-----
1174      *-4.3 矩阵的解析
1175      *-----
1176
1177      *       ==本节目录==
1178
1179      *       4.3.1 线性相关、线性独立和正交向量
1180      *       4.3.2 矩阵的秩
1181      *       4.3.3 特征根和特征向量
1182      *       4.3.4 正定矩阵和负定矩阵
1183      *       4.3.5 裘氏分解
1184
1185      *
1186      *       本节命令

```

```

1185 *-----
1186 * rank(), mat symeigen, mat eigenvalues, cholesky()
1187 *-----
1188
1189
1190 *
1191 *-4.3.1 线性相关、线性独立和正交向量
1192
1193 * 线性相关和独立
1194
1195 * 矩阵 A = [A1, A2, ..., An]
1196
1197 * 对于  $c_1 A_1 + c_2 A_2 + \dots + c_n A_n = 0$  ( $c_i$ 为常数)
1198
1199 * 若存在一组系数  $c_1, c_2, \dots, c_n$  使得上式成立, 则称  $A_1, A_2, \dots, A_n$  线性相关;
1200
1201 * 反之, 称其线性独立。
1202
1203 * 正交向量
1204
1205 * 若  $A_i' A_j = 0, (i \neq j)$ , 则称向量  $A_i$  与  $A_j$  正交
1206
1207
1208
1209 *
1210 *-4.3.2 矩阵的秩(rank)
1211
1212 *  $\text{rank}(A) = \min(\text{行向量中线性独立的个数}, \text{列向量中线性独立的个数})$ 
1213
1214 * 含义: 彼此线性相关的两个变量并不能提供更多的信息,
1215 * 如, 薪水、基本工资、奖金, 给定任意两个变可计算出第三个
1216
1217 mata
1218     A = (1,2,3 \ 3,2,1)'
1219     A
1220     rank(A)
1221     B = (1,2,3 \ 3,2,1 \ 4,4,4)'
1222     B
1223     rank(B)
1224 end
1225
1226 *- 由于 matrix 环境下没有直接计算 rank() 的函数,
1227 * 这里使用了 mata 语句
1228
1229
1230
1231 *
1232 *-4.3.3 特征根和特征向量
1233
1234 *=定义:
1235 *
1236 * 给定方阵 A, 若能找到行向量 h 和一个单值 e, 使得
1237 *  $A \cdot h = e \cdot h$ 
1238 * 成立, 则称 h 为 A 的特征向量, 而 e 为 A 的特征根。
1239
1240 *=含义:
1241 *
1242 * 相当于把矩阵的一个方向分解出来, 而 A 可能包含 n 个方向
1243 * 即, 特征根:  $\text{Lamda} = (e_1, e_2, \dots, e_n)$ ; 特征向量:  $H = (h_1, h_2, \dots, h_n)$ 
1244
1245 *=性质:
1246 *
1247 * (1)  $\text{rank}(A)$  = 非零特征值的个数 (如果有一个特征值为0, 则矩阵非满秩)
1248 *
1249 * (2)  $\det(A)$  = 特征值的乘积 =  $e_1 \cdot e_2 \cdot \dots \cdot e_n$ 
1250 *
1251 * (3)  $\text{trace}(A)$  = 特征值的和 =  $e_1 + e_2 + \dots + e_n$ 
1252 *
1253 * (4)  $\text{inv}(A)$  的特征值为  $1/e_1, 1/e_2, \dots, 1/e_n$ 
1254
1255 *=Stata操作:
1256 *
1257 *-语法格式:
1258 *

```

```

1259 * 非对称方阵: mat eigenvalues 特征根实部 特征根虚部 = 矩阵名
1260 * 对称方阵: mat symeigen 特征向量名 特征根名 = 矩阵名
1261
1262 *-eg1: 非对称矩阵
1263 matrix A = (23,12,-9 \ 2,4,-6 \ 5,1,3)
1264 dis det(A)
1265 mat eigenvalues H Lamda = A
1266 mat list H // 特征根实部
1267 mat list Lamda // 特征根虚部
1268
1269 *-eg2: 非满秩对称矩阵
1270 mat A = (1,2,3,4,5 \ 2,3,4,5,6 \ 3,4,5,6,7 \ 4,5,6,7,8 \ 5,6,7,8,9)
1271 mat list A
1272 dis det(A)
1273
1274 mata // 矩阵 A 的 rank
1275 A = (1,2,3,4,5 \ 2,3,4,5,6 \ 3,4,5,6,7 \ 4,5,6,7,8 \ 5,6,7,8,9)
1276 rank(A)
1277 end
1278
1279 mat symeigen H Lamda = A
1280 mat list H,format(%6.2f) // 特征向量
1281 mat Lamda = diag(Lamda)
1282 mat list Lamda
1283 mat list Lamda,format(%5.4f) // 特征根
1284
1285
1286 *-eg3: 满秩对称矩阵
1287 mat A = (12,35,-13 \ 35,108,0.3 \ -13,0.3,42)
1288 mat list A
1289 mat symeigen H L = A
1290 mat list L // 特征根
1291 mat list H // 特征向量
1292
1293
1294 *-验证上述性质:
1295
1296 *-秩(rank) 3
1297 mata
1298 A = (12,35,-13 \ 35,108,0.3 \ -13,0.3,42)
1299 rank(A)
1300 end
1301
1302 *-横列式(determine)
1303 dis det(A)
1304 dis L[1,1] * L[1,2] * L[1,3]
1305
1306 *-对角和(trace)
1307 dis trace(A)
1308 dis L[1,1] + L[1,2] + L[1,3]
1309
1310 *-逆矩阵的特征根: 练习一下吧
1311
1312
1313 *
1314 *-4.3.4 正定矩阵、负定矩阵
1315
1316 *-定义:
1317 * 给定 n*n 正方形矩阵 A 和`任意' n*1 向量 x, 矩阵的二次型定义为:
1318 *  $x'Ax$  (一个单值)
1319 * A 正定: 若  $x'Ax > 0$ 
1320 * A 负定: 若  $x'Ax < 0$ 
1321 * A 半正定: 若  $x'Ax \geq 0$ 
1322 * A 半负定: 若  $x'Ax \leq 0$ 
1323
1324 sysuse auto, clear
1325 reg price wei len fore
1326 mat V = e(V) // 正定
1327 mat NV = -V // 负定
1328 mat list V
1329
1330 mat x = matuniform(4,1) // 验证 V
1331 mat xVx = x'*V*x
1332 mat list x

```

```
1333     mat list xVx
1334
1335     mat x = matuniform(4,1)    // 验证 NV
1336     mat xNVx = x'*NV*x
1337     mat list x
1338     mat list xNVx
1339
1340
1341
1342 *-----
1343 *-4.3.5 裘氏分解
1344
1345     *-裘氏分解(cholesky factorization)
1346     * 相当于矩阵开根号
1347     * ! 只有正定对称矩阵才可进行此分解
1348
1349     mat A = (23,12,-9 \ 2,4,-6 \ 5,1,3) // 非对称
1350     mat chA = cholesky(A)
1351     mat A = (1,2,3,4,5 \ 2,3,4,5,6 \ 3,4,5,6,7 \ 4,5,6,7,8 \ 5,6,7,8,9)
1352     mat chA = cholesky(A)          // 非正定
1353
1354     mat A = J(4,4,1) + 3*I(4)    // 正定且对称
1355     mat B = cholesky(A)          // A=B*B'
1356     mat BT = B'
1357     mat list A
1358     mat list B          // B
1359     mat list BT         // B'
1360     mat AA = B*B'
1361     mat list AA
1362
1363     *- 应用实例: OLS 估计值的标准误
1364     sysuse auto, clear
1365     reg price wei len
1366     mat list e(b)
1367     mat list e(V)
1368     *- Q: 如何利用方差-协方差矩阵 e(V) 计算出各个系数的标准误?
1369     *- A: s.e. 其实就是 e(V) 矩阵中对角线元素的开方
1370     mat V = e(V)
1371     mat list V
1372     mat se2 = vecdiag(V)
1373     mat list se2
1374     mat se2 = diag(se2)
1375     mat list se2
1376     mat se = cholesky(se2)
1377     mat list se
1378     reg, nohead          // 检验一下
1379     *- 一条命令即可搞定:
1380     mat ss = cholesky(diag(vecdiag(e(V))))
1381     mat list ss
1382
1383     *- 练习: 如何根据 e(b) 矩阵和 se 矩阵求取 t 值?
1384     *- 提示: t[j] = b[j] / se[j]
1385     mat b = e(b)
1386     mat t = diag(b)*inv(se)
1387     mat list t
1388
1389
1390
1391
1392 *-----
1393 *-4.4 有关矩阵的进一步说明
1394 *-----
1395
1396     *          ==本节目录==
1397
1398     *          4.4.1 矩阵函数
1399     *          4.4.2 返回系统中的矩阵
1400     *          4.4.3 定义约束矩阵
1401     *          4.4.4 矩阵与暂元的相关操作
1402     *          4.4.5 矩阵对内存的需求
1403
1404
1405 *-----
1406 *-4.4.1 矩阵函数
```

```

1407
1408     help matrix functions
1409
1410
1411 *
1412 *-4.4.2 返回系统中的矩阵      -matrix get-
1413
1414     help matrix get
1415
1416     sysuse auto, clear
1417     regress price weight mpg
1418     matrix list e(b)
1419     matrix list e(V)
1420
1421     matrix b = get(_b)      // 估计系数向量
1422     matrix V = get(VCE)     // 方差-协方差矩阵
1423     matrix list b
1424     matrix list V
1425
1426     test weight = 1, notest
1427     test mpg = 40, accum
1428     matrix rxtr = get(Rr)   // 约束条件矩阵
1429     matrix list rxtr
1430
1431
1432 *
1433 *-4.4.3 定义约束矩阵(用于假设检验)
1434
1435     *-Wald 检验中, 约束条件通常表示为
1436     *
1437     *   R*b = r
1438     *
1439     * 如, 对于模型  $y = [x_1 \ x_2 \ x_3 \ x_4] * (b_1 \ b_2 \ b_3 \ b_4)'$ 
1440     *    $x_1 - x_3 = 2.8$ 
1441     *    $x_2 - x_3 = 0$ 
1442     * 这两个约束条件可表示如下:
1443     *
1444     *   [ 1  0  -1  0  0 ] [b1]   | 2.8 |
1445     *   [ 0  1  -1  0  0 ] [b2]   |  0  |
1446     *                       [b3]
1447     *                       [b4]
1448     *
1449     * -mat_put_rr- 命令用于定义矩阵  $z = [R \ b]$ 
1450
1451     sysuse auto, clear
1452     regress price wei len mpg foreign
1453     mat z = (1,0,-1,0,0,2.8 \ 0,1,-1,0,0,0)
1454     mat_put_rr z
1455     test
1456
1457     *-等价于
1458     test wei - mpg = 2.8
1459     test len = mpg, accum
1460
1461
1462
1463 *
1464 *-4.4.4 矩阵与暂元的相关操作
1465
1466     help matmacfunc
1467
1468     sysuse auto, clear
1469     mkmat price wei len turn, mat(A)
1470     local rnames: rowfullnames A
1471     local cnames: colfullnames A
1472     dis "`rnames'"
1473     dis "`cnames'"
1474
1475     sureg (price foreign weight length)   ///
1476          (mpg foreign weight turn)       ///
1477          (displ foreign weight)
1478     mat b = get(_b)
1479     local rn: rnames b
1480     local cn: colnames b

```

```

1481     dis "`rn'"
1482     dis "`cn'"
1483
1484     *-应用：参见第二讲 A2_data 第 2.7.4 小节：样本的堆砌 (Line:1986)
1485
1486
1487
1488     *-4.4.5 矩阵对内存的需求
1489
1490
1491         表4-1 不同版本下参数的设定
1492
1493         +-----+-----+-----+-----+-----+-----+
1494         | Parameter | -- Intercooled Stata -- | ----- Stata/SE ----- |
1495         |           | Default   min   max   | Default   min   max   |
1496         +-----+-----+-----+-----+-----+-----+
1497         | maxvar   |      2,047 2,047 2,047 |      5,000 2,047 32,766 |
1498         | matsize  |        200   10   800 |        400   10 11,000 |
1499         | memory   |         1M 500K   ... |         10M 500K   ... |
1500         +-----+-----+-----+-----+-----+-----+
1501
1502         表4-2 矩阵大小对内存的需求
1503
1504         +-----+-----+
1505         | matsize | memory use |
1506         +-----+-----+
1507         |      400 |      1.254M |
1508         |      800 |      4.950M |
1509         |     1,600 |     19.666M |
1510         |     3,200 |     78.394M |
1511         |     6,400 |    313.037M |
1512         |    11,000 |    924.080M |
1513         +-----+-----+
1514
1515         *-设定矩阵的默认尺寸
1516
1517         set matsize 200
1518         mat a = J(300,1,0) // 错误
1519         set matsize 400
1520         mat a = J(300,1,0) // 正确
1521
1522         * ----- over -----
1523
1524
1525

```

```
1
2
3
4
5      *=====
6      *      计量分析与STATA应用
7      *=====
8
9      *      主讲人：连玉君 博士
10
11     *      单   位：中山大学岭南学院金融系
12     *      电   邮：arlionn@163.com
13     *      主   页：http://blog.cnfol.com/arlion
14
15     *      ::第一部分::
16     *      Stata 操作
17     *      =====
18     *      第五讲  STATA 编程初步
19     *      =====
20
21
22     *cd D:\stata11\ado\personal\Net_course_A\A5_prog
23     cd `c(sysdir_personal)'Net_course_A\A5_prog
24
25
26     *-----
27     *      本讲目录
28     *-----
29     * 5.1 stata程序简介
30     * 5.2 单值(scalar)
31     * 5.3 暂元
32     * 5.4 其它暂时性物件
33     * 5.5 控制语句
34     * 5.6 引用 Stata 命令的返回值
35
36
37
38
39
40     *=====
41     *      计量分析与STATA应用
42     *=====
43
44     *      主讲人：连玉君 博士
45
46     *      单   位：中山大学岭南学院金融系
47     *      电   邮：arlionn@163.com
48     *      主   页：http://blog.cnfol.com/arlion
49
50     *      ::第一部分::
51     *      Stata 操作
52     *      =====
53     *      第五讲  STATA 编程初步
54     *      =====
55     *      -5.1- stata程序简介
56
57
58     *-----
59     *-> 5.1 stata程序简介
60     *-----
61
62     *      ==本节目录==
63
64     *      5.1.1 Stata 程序的基本结构
65     *      5.1.2 程序的执行
66     *          5.1.2.1 第一种执行方式：ado 文档执行方式
67     *          5.1.2.2 第二种执行方式：run(Ctrl+R)
68     *      5.1.3 程序的管理
69     *      5.1.4 避免列印过多的结果
70     *      5.1.5 避免程序因错误而中断
71     *      5.1.6 避免数据在程序执行过后有所变动
72
73
74     *-----
```

```
75  *--5.1.1 Stata 程序的基本结构
76
77      program define myprog
78      version 8.0
79      dis "I Iove This Game!"
80      end
81
82      * 注：保存为 myprog.ado (文件的扩展名为 '.ado')
83
84
85  *-----
86  *--5.1.2 程序的执行
87
88      *--5.1.2.1 第一种执行方式：ado 文档执行方式
89
90      myprog
91
92      adopath + D:\stata10\ado\personal\Net_course_A\A5_prog
93
94      myprog
95
96
97      *--说明：
98      *      (1) 这种执行方式与stata官方命令完全相同；
99      *      (2) 对于需要经常执行的命令，采用这种方式很好；
100
101      *--建议：
102      *      (1) 把自己编写的程序统一存放于 .....\personal\_Myado下；
103      *      (2) 并在profile.do文件中定义如下
104      *      adopath + D:\stata11\ado\personal\_Myado
105      *      (3) 该文件夹下可进一步设定 a-z 等子文件夹，
106      *      存放相应字母开头的文件
107      *      (4) 对于临时的 ado 文档，可以采用 -adopath- 命令定义存放地址
108
109
110      *--5.1.2.2 第二种执行方式：run
111
112      *--Step1：在内存中定义程序；
113      *具体方法：选中，点击`Execute Quietly(run)`键（快捷键：Ctrl+R）
114
115      *--Step2：执行程序(方式同前)
116
117      *--示例：
118
119      program define mynike
120      version 8.0
121      dis in red "Just do it! "
122      end
123
124      mynike
125
126
127  *-----
128  *--5.1.3 程序的管理
129
130      doedit myprog.ado      // 修改程序
131      program dir             // 查找内存中的程序
132      program list myprog    // 列示程序代码
133      program list _all
134      program drop mynike    // 删除内存中调入的程序，但不影响硬盘中存储的文件
135      mynike                 // 错误信息，因为程序已不在内存中
136
137      program define mynike
138      version 8.0
139      dis in red "Just do it! hahaha! "
140      end
141
142      mynike
143
144      *--说明：
145      *      (1) 修改程序后，必须先将旧版本从内存中清除(program drop)，
146      *      然后再调入新定义的程序
147      *      (2) 更为合理的定义方法：
148
```



```
149         capture program drop mynike           // 新增语句
150         program define mynike
151             version 8.0
152             dis in red "Just do it! ha    ha! " // 请修改后执行
153         end
154
155         mynike
156
157
158 *-----
159 *-5.1.4 避免列印过多的结果           -quietly-
160
161     sysuse auto, clear
162     quietly sum price, meanonly // 静悄悄地做, 单行
163     scalar avg = r(mean)
164     dis avg
165
166     quietly{                               // 静悄悄地做, 整段
167         sum price if foreign == 0
168         scalar avg1 = r(mean)
169         sum price if foreign == 1
170         scalar avg2 = r(mean)
171         scalar diff = avg2 - avg1
172     }
173     dis diff
174
175
176 *-----
177 *-5.1.5 避免程序因错误而中断         -capture-
178
179     sysuse auto, clear
180     drop prcie           // 错误! 声张, 停止
181
182     capture drop prcie   // 错误! 不声张, 不停止
183     sum mpg
184     dis _rc              // -help _rc-
185
186     capture drop price   // 正确! 不声张
187     dis _rc              // 注意该值的变化
188
189     sysuse auto, clear
190     cap noisily drop prcie // 错误! 声张, 不停止
191     sum mpg
192
193 *- 示例:
194
195     cap program drop varies
196     program define varies
197         version 10.0
198         args varname           // 设定输入项
199         cap sum `varname'
200         if _rc ~= 0{
201             dis as error "错误: 未发现变量 `varname'"
202             exit _rc
203         }
204     end
205
206     varies pp
207     varies weight
208
209
210 *- 特别说明:
211 * capture 后的任何一个 argument 错误, 则所有 args 都不会被执行
212
213     sysuse auto, clear
214     order price weight length
215
216     cap drop price weigth length // Q:哪些变量会被 drop ?
217
218 *-正确方法:
219     cap drop price
220     cap drop weigth
221     cap drop length
222
```

```

223      *-或使用 -dropvars-
224      sysuse auto, clear
225      order price weight length
226      dropvars price weight length
227
228
229      *-----
230      *-5.1.6 避免数据在程序执行过后有所变动      -preserve-
231
232      sysuse auto, clear
233      preserve                      // 备份当前状态 s1
234      keep price weight foreign
235      drop if price > 10000
236      sum
237      save auto_new.dta, replace
238      restore                      // 恢复到状态 s1
239      sum
240
241      use auto_new.dta, clear
242
243      *-说明:
244      * (1) 多数情况下, 我们改动资料都是为了得到特定的结果;
245      * (2) 在 preserve 和 restore 之间对资料进行的任何修改都无法保留;
246      * (3) preserve 和 restore 不可“嵌套”使用
247      *      stata11 提供了“嵌套”功能, 参见
248
249      help snapshot
250
251
252      *-实例: 上市公司资本结构影响因素分析
253
254      *-数据
255      use xtcs2, clear
256      tsset code year
257      xtodes                      // unbalance panel data
258
259      * 基本模型: 非平行面板分析结果
260      xtreg tl size-tobin, fe
261      est store fe_unb
262
263      * 稳健性检验: 平行面板回归结果
264      preserve
265      xtbalance, range(1998 2004) // 处理为平行面板
266      xtreg tl size-tobin, fe
267      est store fe_bal
268      restore
269
270      * 随机效应模型(针对非平行面板)
271      xtreg tl size-tobin, re
272      est store re_unb
273
274      * 结果汇总
275      esttab fe_unb fe_bal re_unb, nogap stat(N N_g)
276
277
278
279
280
281
282
283
284      *=====
285      *      计量分析与STATA应用
286      *=====
287
288      *      主讲人: 连玉君 博士
289
290      *      单 位: 中山大学岭南学院金融系
291      *      电 邮: arlionn@163.com
292      *      主 页: http://blog.cnfol.com/arlion
293
294      *      ::第一部分::
295      *      Stata 操作
296      *      =====

```

---

```

297      *      第五讲 STATA 编程初步
298      *      =====
299      *      -5.2- 单值(scalar)
300      *      -5.3- 暂 元
301      *      -5.4- 其它暂时性物件
302
303
304      *-----
305      *-> 5.2 单值(scalar)
306      *-----
307
308      *      ==本节目录==
309
310      *      5.2.1 存放数值
311      *      5.2.1 存放数值
312      *      5.2.2 存放字符串
313      *      5.2.3 执行命令后的单值结果
314      *      5.2.4 单值的管理
315
316      scalar a = 3
317      scalar b = ln(a) + (3^4.2)/exp(2)
318      dis a
319      dis b
320
321
322      *-----
323      *-5.2.2 存放字符串
324
325      scalar c = .a
326      dis c
327      scalar s1 = "hello, Arlion"
328      scalar s2 = substr(s1,1,5) // 单值的引用很简单
329      dis s1
330      dis s2
331
332      * display 命令还是一个简单的计算器
333      dis ln(3) + (3^4.2)/exp(2)
334      dis %6.2f ln(3) + (3^4.2)/exp(2)
335
336      * 标示出变量的特定观察值
337      sysuse auto,clear
338      dis price[3]
339      list price in 1/3
340      sort price
341      gen pmax = price[_N]
342      list pmax in 1/20
343      sum price
344
345
346      *-----
347      *-5.2.3 执行命令后的单值结果
348
349      sum price
350      return list
351      dis r(N)
352      scalar range = r(max) - r(min)
353      dis range
354      gen qq = r(sd)
355      list qq in 1/10
356
357      *-示例: 求取 mean(price) 的标准误
358
359      *- 公式: s.e.(mean) = s.d.(price)/sqrt(N)
360
361      sysuse auto, clear
362      sum price
363      scalar se_price = r(sd)/sqrt(r(N))
364      dis "sd(price) = " r(sd)
365      dis "se(price) = " se_price
366
367      *-用途: t 检验
368      scalar t_value = r(mean) / se_price
369      dis t_value
370

```

---

```
371      *-stata t 检验命令(验证)
372      ttest price=0
373
374
375      *-----
376      *-5.2.4 单值的管理
377
378      scalar dir
379      scalar list
380      scalar drop a
381      scalar list
382      scalar drop _all
383      scalar list
384
385
386
387
388      *-----
389      *-> 5.3 暂元
390      *-----
391
392      * ==本节目录==
393
394      * 5.3.1 暂元的定义和引用
395      *      5.3.1.1 暂元的基本功能
396      *      5.3.1.2 数学运算符的处理
397      *      5.3.1.3 复合双引号: `"' "'
398      *      5.3.1.4 暂元中的暂元
399      *      5.3.1.5 暂元引用机制的简化
400      * 5.3.2 全局暂元
401      * 5.3.3 暂元的管理
402
403
404      *-----
405      *-5.3.1 暂元的定义和引用
406
407      *-5.3.1.1 暂元的基本功能
408
409      *-存放数字
410      local a = 5
411      dis `a'
412
413      local b = `a' + 7
414      dis `b'
415
416      *-存放文字
417      local name1 "Arlion: "
418      dis "`name1'"
419
420      local name2 中山大学 岭南学院
421      dis "`name2'"
422
423      local name3 `name1'`name2'
424      dis "`name3'"
425
426      *-存放变量名称
427      sysuse auto, clear
428      local varlist price weight rep78 length
429      sum `varlist'
430      des `varlist'
431
432      dis `varlist' // 列印各变量的第一个观察值
433      dis price weight rep78 length
434      list price weight rep78 length in 1/1
435      dis "`varlist'" // 列印变量名称
436
437
438      *-5.3.1.2 数学运算符的处理
439
440      local a "2+2"
441      dis `a'
442      dis "`a'"
443
444      local b = 2+2 // 与上面的定义有何差异?
```

```
445     dis `b'
446     dis "`b'"
447
448
449     *-5.3.1.3 复合双引号: `"'  "'
450
451     *-适用于文字中包含 ` ' 和 " 的情形
452     local tt John's "car"
453     dis "`tt'"           // 错误方式
454     dis "John's " car ""
455     local tt John's "car" // 正确方式
456     dis "`" `tt' "'
457
458
459     *-5.3.1.4 暂元中的暂元
460
461     local a1 = 2
462     local a2 "var"
463     local a3 = 2*`a1'
464     local a4 `a`a1''
465     local `a2`a1' = 2*`a3'
466     local `a`a3'' "`a`a1'2'" // 从第一个完整的 ` ' 开始分析
467
468     dis `a1'
469     dis "`a2'"
470     dis `a3'           // 4
471     dis "`a4'"         // 暂元 a2 中的内容
472     dis ``a2`a1''      // 8
473     dis "`a`a3''"      // ?
474     dis "`a`a3'''"     // 8
475
476
477     *-5.3.1.5 暂元引用机制的简化
478
479     *-数学运算式的简化
480     sysuse auto, clear
481     local i = 19
482     local j = int(sqrt(`i'))
483     dis `j'
484     dis price[`j']
485     *-等价于:
486     local i = 19
487     dis price[`=int(sqrt(`i'))'] // price`=j'
488     *-验证
489     list price in 1/5
490
491     *-逻辑运算的简化
492     gen price1 = price if foreign==1
493     gen price0 = price if foreign==0
494     local i = 0
495     sum price`=(`i`>0)''
496
497     *-暂元内数值的递增和递减
498     local i = `i' + 1
499     local i++           // 执行运算`之后' 加 1
500     local ++i           // 执行运算`之前' 加 1
501
502     local j = `j' - 1
503     local j--
504     local --j
505
506     * e.g.
507     local i = 1
508     dis `i++'
509     dis `i'
510     local i = 1
511     dis `++i'
512     dis `i'
513
514
515     *-----
516     *-5.3.2 全局暂元
517
518     *- 定义和引用方式
```

```

519     global aa "This is my first program!"
520     dis "$aa"
521
522     global x1 = 5
523     global x2 = 2^$x1
524     dis $x2
525
526     *- 示例:
527     sysuse nlsw88, clear
528     global option ", vce(bootstrap, reps(50))" // 公共选项
529     global reg "regress" // 估计方法
530
531     local x1 "hours ttl_exp"
532     $reg wage `x1' $option
533     est store m1
534
535     local x2 "hours ttl_exp married union"
536     $reg wage `x1' $option // `x1' 中的内容个已经失效
537     $reg wage `x2' $option
538     est store m2
539
540     esttab m1 m2, nogap
541
542
543     *-----
544     *-5.3.3 暂元的管理
545
546     macro list
547     macro dir
548     macro drop x2
549     macro dir x2
550     macro dir aa
551
552
553
554     *-----
555     *-> 5.4 其它暂时性物件
556     *-----
557
558     * ==本节目录==
559
560     * 5.4.1 暂时性变量
561     * 5.4.2 暂时性矩阵和暂时性单值
562     * 5.4.3 暂时性文件
563
564
565     *-----
566     *-5.4.1 暂时性变量 -tempvar-
567
568     sysuse nlsw88, clear
569     tempvar x1 x2
570     gen `x1' = hours^2
571     gen `x2' = ln(wage)
572     sum `x1' `x2'
573
574     * 暂时性变量的名称可与永久性变量同名，因为二者的引用方法有别
575
576     *-----一个实例-----
577     cap program drop mysum
578     program define mysum
579     version 8.0
580     args var // 输入项 help args
581
582     tempvar x1 x2 // 定义两个暂时性变量
583     gen `x1' = sqrt(`var')
584     gen `x2' = ln(`var')
585
586     dis in y "The summary of `var' is: "
587     sum `var'
588     dis _n in y "The summary of sqrt(`var') is:"
589     sum `x1'
590     dis _n in y "The summary of ln(`var') is:"
591     sum `x2'
592

```

```
593     end
594     *-----
595
596     sysuse nlsw88, clear
597     mysum wage
598
599
600 *-----
601 *-5.4.2 暂时性矩阵和暂时性单值          -tempname-
602
603     local j = 7
604     tempname mymat          // 定义暂时性矩阵
605     mat `mymat' = I(`j')    // 引用方式
606     mat list `mymat'
607
608 *-----一个实例-----
609 * 求取一个矩阵各行的和
610     mat a = J(4,4,1) + I(4)
611     mat a[3,2] = 9
612     tempname one rowsum
613     local c = colsof(a)      // 返回矩阵 a 的列数
614     mat `one' = J(`c',1,1)   // 定义一个 cx1 的列向量
615     mat `rowsum' = a * `one' // 求和
616     mat list a
617     mat list `one'
618     mat list `rowsum'
619 *-----
620
621 *-练习：如何求取各行的算术平均值和加权平均值？
622
623
624 *-----封装为程序-----
625     *! 求取给定矩阵的各行之和和各列之和
626     cap program drop mat_sum
627     program define mat_sum
628     version 10
629         args matname
630         tempname one rowsum colsum
631         local c = colsof(`matname')
632         mat `one' = J(`c',1,1) // (1)
633         mat `rowsum' = `matname' * `one'
634         dis in g _n "矩阵 `matname' 的" in w " 各行加总为： "
635         mat list `rowsum', noheader nonames
636
637         local c = rowsof(`matname')
638         mat `one' = J(1,`c',1) // (2) 重复利用 one 这个空盒子，比较"环保"
639         mat `colsum' = `one' * `matname'
640         dis in g _n "矩阵 `matname' 的" in w " 各列加总为： "
641         mat list `colsum', noheader nonames
642     end
643     exit
644 *-----
645 *- 语句可精简：mat `one' = J(`=colsof(`matname')',1,1)
646
647     mat list a
648     mat_sum a
649
650     mat b = (-1.3, 2.6 \ 3.89, 0.42 \ 50.1, -0.634)
651     mat list b
652     mat_sum b
653
654
655 * 关于暂时性单值的两点说明：
656 * (1) 可以将其视为 1*1 暂时性矩阵
657 * (2) 尽量避免暂时性单值的使用，而用暂元替代之
658
659
660
661 *-----
662 *-5.4.3 暂时性文件          -tempfile-
663
664 *-定义： tempfile file1
665 *-调用： use "`file1'"
666
```

```

667 *----- 一个实例 -----
668 * 数据: 沪市的7家公司财务数据
669       use A5_tempfile1.dta, clear
670       list, sepby(id)
671 * 任务: 产生一个新的公司代码变量, 值为: 1,2,3
672 * 思路: 应用 _n 和 egen 命令
673       keep id
674       duplicates drop
675       gen id_new = _n
676       list, sep(0)
677
678 * 具体处理过程:
679       use A5_tempfile1.dta, clear
680       preserve // 备份数据, 以便抽取 id
681       keep id
682       duplicates drop
683       sort id
684       gen id_new = _n
685       list
686       tempfile id_data // 声明暂时性文件 id_data
687       sort id
688       save "`id_data'", replace // 存储数据至暂时性文件
689       restore // 抽取 id 完成, 恢复原数据
690       sort id year
691       merge id using "`id_data'" // 将新的 id 合并到原始数据中
692       sort id_new year
693
694       order id id_new year
695       list, sepby(id)
696       tsset id_new year // 使用新 id 标示 panel
697       xtset
698 *-----
699
700 *--一个简洁的处理方法:
701       use A5_tempfile1.dta, clear
702       xtset id year
703       egen id_new = group(id)
704       list, sepby(id)
705
706
707
708
709
710
711
712
713 *=====
714 *           计量分析与STATA应用
715 *=====
716
717 *           主讲人: 连玉君 博士
718
719 *           单 位: 中山大学岭南学院金融系
720 *           电 邮: arlionn@163.com
721 *           主 页: http://blog.cnfol.com/arlion
722
723 *           ::第一部分::
724 *           Stata 操作
725 *           =====
726 *           第五讲 STATA 编程初步
727 *           =====
728 *           -5.5- 循环语句
729
730       cd `c(sysdir_personal)'Net_course_A\A5_prog
731
732 *-----
733 *-> 5.5 循环语句
734 *-----
735
736 * ==本节目录==
737
738 * 5.5.1 条件循环: while 语句
739 * 5.5.2 forvalues 语句
740 * 5.5.3 foreach 语句

```



```

741
742
743
744 *           本节命令
745 *-----
746 * while, forvalues, foreach
747 *-----
748
749
750 *-----
751 *-5.5.1 循环语句
752
753     *-5.5.1.1 条件循环: while 语句
754
755     local j = 0
756     while `j' < 5 {
757         dis in y _s(10) `j'
758         local j = `j' + 1
759     }
760
761     *-或
762
763     local j = 0
764     while `j' < 5 {
765         dis in y _s(10) `j++'
766     }
767
768 *-----
769 *-示例: 采用数值法求取函数的极小值
770
771     twoway function y = 0.2*exp(x) - ln(x^2) + 3, ///
772         range(0 4) lw(*2)
773
774
775     local trace "set trace on"    // 解析具体过程
776 *-----
777     local delta = 0.05    // 步长
778     local x = 1    // x 的初始值
779     local j = 0    // 计数器: 记录迭代次数
780     local e = 1    // y1-y0
781     local e0 = 0.01 // 收敛判据
782     while `e' > `e0' {
783         `trace'
784         local y0 = 0.2*exp(`x') - ln(`x'^2) + 3
785         local x = `x' + `delta'
786         local y1 = (0.2*exp(`x') - ln(`x'^2)) + 3
787         local e = abs(`y1' - `y0')
788         dis in g "*" _c
789         local j = `j' + 1
790     }
791     dis "e = " `e'
792     dis "x = " `x'    // x 的解
793     dis "y = " `y1'   // y 的极小值
794     dis "j = " `j'    // 迭代次数
795
796     *-图示:
797     twoway function y = 0.2*exp(x) - ln(x^2) + 3, ///
798         range(0 4) lw(thick) xline(`x') yline(`y1') ///
799         text(`='y1'-0.5' `='x'+0.8' "(`x', `y1')")
800 *-----
801
802     *- 练习:
803     * (1) 设定 (delta=0.1, e0=0.01), -trace- 计算过程
804     * (2) 尝试将 (delta=0.001, e0=0.0001), 结果有何变化?
805     * (3) 若设定 (delta=0.02, e0=0.0001), 能否收敛?
806     * (4) 若设定 x=2 为初始值, 能否收敛?
807
808
809     *- 程序修改如下:
810 *-----
811     local h = 0.001    // 步长
812     local x = 1    // x 的初始值
813     local j = 0    // 计数器: 记录迭代次数
814     local e = 1    // y1-y0

```

```

815     local e0 = `h'/10 // 收敛判据 (修改为动态数值)
816     while abs(`e')>`e0'{
817         // 修改: abs(`e')
818         local y0 = 0.2*exp(`x') - ln(`x'^2) + 3
819         local x = `x' + `h'
820         local y1 = (0.2*exp(`x') - ln(`x'^2)) + 3
821         local e = `y1' - `y0' // 此前 e = abs(`y1'-`y0')
822         if (`e' > 0){
823             local h = -`h' // 新增: 反向搜索
824         }
825         dis in g "*" _c
826         local j = `j' + 1
827     }
828     dis "e = " `e'
829     dis "x = " `x' // x 的解
830     dis "y = " `y1' // y 的极小值
831     dis "j = " `j' // 迭代次数
832
833     *-图示:
834     local x: dis %4.3f `x' // 新增: 显示的更美观
835     local y: dis %4.3f `y1'
836     twoway function y = 0.2*exp(x) - ln(x^2) + 3, ///
837         range(0 4) lw(thick) xline(`x') yline(`y1') ///
838         text(`='y'-0.5' `='x'+0.4' "(`x', `y')")
839     *-----
840
841     *- 练习:
842     * (1) 尝试初始值 x=3, 是否能收敛?
843     * (2) 如何搜索更加有效? 程序如何编写?
844
845
846     *
847     *- 挑战: 请求取如下函数的全局极大值(0<x<80):
848
849     twoway function ///
850         y = 0.3*sin(0.5*x)+ 0.9*cos(0.2*x) + 0.5*ln(x), ///
851         range(0 80) lw(*2)
852     *
853     *
854     * 思路: dy/dx = 0
855     * dy/dx = 0.15*cos(0.5*x) -0.18*sin(0.2*x) + 0.5*(1/x)
856     *
857     * 尚未完成的解答:
858     doedit A5_while_max.do
859
860
861
862     *-5.5.1.2 forvalues 语句 // 数字的循环
863
864     forvalues i = 0(-1)-14{
865         dis in y _s(8) `i'
866     }
867
868     forvalues i = 0/4{
869         dis in y _s(10) `i'
870     }
871
872     forvalues i = 10(-2)1{
873         dis in y _s(8) `i'
874     }
875
876     mat mm = J(10,3,0)
877     forvalues i = 1/10{
878         forvalues j = 1/3{
879             mat mm[`i',`j'] = `i' + `j'
880         }
881     }
882     mat list mm
883
884
885     *-示例 1: 多个文件导入和合并
886     type d1.txt
887     type d2.txt
888     type d3.txt

```

```

889
890     *-导入
891     forvalues j = 1/3{
892         local varname id year invest market stock
893         insheet `varname' using d`j'.txt, clear
894         save s`j'.dta, replace
895     }
896
897     *-合并(纵向追加)
898     use s1.dta, clear
899     forvalues j = 2/3{
900         append using s`j'.dta
901     }
902     save alldata.dta, replace
903     browse
904
905
906     *-示例 2: Fama-French two-step regression
907     * viewsource xtfmb.ado
908     * help xtfmb
909
910     *- step1: 对面板分年度执行 OLS 回归, 记录之;
911     *- step2: 将各年度的估计值平均, 得到最终的 b, se, R2 等统计量
912
913     *- 简单处理方式
914     * model: reg mvalue invest kstock
915     use grunfeld.dta, clear
916     sort year company
917     tab year
918     mat R = J(20, 7, 0)
919     local i = 1
920     forvalues yr = 1935/1954{
921         qui reg mvalue invest kstock if (year == `yr')
922         mat b = e(b)
923         mat se = vecdiag(cholesky(diag(vecdiag(e(V)))))
924                 // 参见 A4_Matrix.do
925         mat R[`i++', 1] = (b, se, e(r2_a))
926     }
927     mat list R
928     mat list e(b) // 验证 1954 的结果
929
930
931
932     *- 一般化处理方式(可封装成程序)
933     *-----
934     use grunfeld.dta, clear
935     xtset company year
936     xtides
937     *-基本设定
938     qui xtset
939     egen tt = group(year) // 有什么好处?
940     tab tt
941     tsset company tt
942     local T = r(tmax) // 样本时间跨度
943     local y "mvalue" // 被解释变量
944     global xx "invest kstock" // 解释变量
945     *-设定存储结果的矩阵
946     local s = wordcount("$xx")
947     local c = (`s'+1)*2 + 1
948     mat R = J(`T', `c', 0)
949     *-第一步: 分年度回归
950     forvalues t = 1/`T'{
951         qui reg `y' $xx if (tt == `t')
952         mat b = e(b)
953         mat se2 = vecdiag(e(V))
954         math se = sqrt(se2) // Arlion 自编程序
955         *mat se = vecdiag(cholesky(diag(vecdiag(e(V)))))
956         mat R[`t', 1] = (b, se, `e(r2)')
957     }
958
959     *-第二步: 计算各年度平均值
960     mat one = J(1, `T', 1)/`T' // 每个元素都是 1/T
961     mat AR = one * R
962     mat list AR

```

```

963
964     *-第三步：呈现结果
965     qui tsset company year
966     global rowname ""
967     forvalues t = `r(tmin)'/ `r(tmax)'{
968         global rowname "$rowname `t'"
969     }
970     dis "$rowname"
971     mat rownames R = $rowname
972     global xx "$xx cons"
973     mat colnames R = $xx se se se R2
974     local coln : colnames R
975     mat colnames AR = `coln'
976     mat list R // 分年度估计结果
977     mat list AR // 各年度平均
978     *- 尚可进一步美化：
979     mat a = R[1, 1..3]
980     mat list a
981     mat RR = (AR[1, 1..3] \ AR[1, 4..6] \ AR[1, 7], ., .)
982     mat rownames RR = b se avg-R2
983     mat list RR
984     *-----
985     *- 测试结果：
986     *ssc install xtfrm, replace // 下载安装该命令
987     xtfrm mvalue invest kstock
988
989
990
991     *-5.5.1.3 foreach 语句 // 变量、暂元、文件等的循环
992
993     help foreach // 语法格式
994
995     *-a. 任意格式：foreach v in ...
996     type d1.txt
997     type d2.txt
998     type d3.txt
999     foreach file in d1 d2 d3{
1000         local varname id year invest market stock
1001         insheet `varname' using `file'.txt,clear
1002         save `file'.dta, replace
1003     }
1004
1005     * 追加样本
1006     use d1.dta, clear
1007     foreach file in d2.dta d3.dta{
1008         append using `file'
1009     }
1010     list
1011
1012
1013     *-b. 变量名循环：foreach v of varlist ...
1014
1015     *-示例 1：各变量的对数转换
1016     sysuse auto,clear
1017     global vars "price weight length"
1018     foreach v of varlist $vars{
1019         gen ln_`v' = ln(`v')
1020         label variable ln_`v' "ln(`v')"
1021     }
1022
1023     *-示例 2：各变量的缩尾处理(Winsorized)
1024     sysuse nls88, clear
1025     local vv "wage hours ttl_exp grade"
1026     foreach v of varlist `vv'{
1027         winsor `v' , gen(`v'_w) p(0.01)
1028     }
1029     d *_w
1030
1031
1032     *-c. 暂元循环：foreach cc of local ...
1033     sysuse auto,clear
1034     local vars price weight length
1035     foreach v of local vars{
1036         gen `v'_2 = `v'^2

```

```

1037     }
1038
1039     *-特别注意：这里的 vars 暂元在引用时无需加 ``
1040
1041
1042     *-d. 数字循环：foreach num of numlist ...
1043
1044     foreach num of numlist 1 4/8 13(2)21 103 {
1045         display in y _s(10) `num'
1046     }
1047
1048     foreach num of numlist 111 1111 11111 111111 1111111 11111111 {
1049         dis in g _s(10) %16.0f `num'^2
1050     }
1051
1052     *-这与 -forvalues- 语句有何差异?
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062     *=====
1063     *           计量分析与STATA应用
1064     *=====
1065
1066     *           主讲人：连玉君 博士
1067
1068     *           单   位：中山大学岭南学院金融系
1069     *           电   邮：arlionn@163.com
1070     *           主   页：http://blog.cnfol.com/arlion
1071
1072     *           ::第一部分::
1073     *           Stata 操作
1074     *           =====
1075     *           第五讲 STATA 编程初步
1076     *           =====
1077     *           -5.6- 条件语句
1078
1079     cd `c(sysdir_personal)'Net_course_A\A5_prog
1080
1081     *-----
1082     *-> 5.6 条件语句
1083     *-----
1084
1085     *   ==本节目录==
1086
1087     *   5.6.1 if 语句
1088     *   5.6.2 一些有用的条件函数
1089
1090
1091     *-----
1092     *-5.6.1 -if- 语句
1093
1094     *-基本要求
1095     * (1) 语法格式
1096     *     CASE I:
1097     *         if (条件){
1098     *             执行命令
1099     *         }
1100     *     CASE II:
1101     *         if (条件1){
1102     *             执行命令1
1103     *         }
1104     *         esle if (条件2){
1105     *             执行命令2
1106     *         }
1107     * (2) 左括弧 "{" 紧接着条件；右括弧 "}" 另起一行
1108     * (3) 条件判断可嵌套
1109
1110     *-示例 1

```

```

1111      clear
1112      scalar tt = 7^2 + 3*29 + ln(100)
1113      if tt>0{
1114          dis in g "The value is" in y " positive! "
1115      }
1116      dis tt
1117
1118
1119  *--示例 2
1120      scalar aa = 1    // 测试, 修改为 aa==1
1121      if aa ==1{
1122          dis "这小子真帅!"
1123      }
1124      else if aa==0{
1125          dis "这女孩真靓!"
1126      }
1127
1128
1129  *--示例 3
1130      sysuse nlsw88.dta, clear
1131      sort hours
1132      forvalues i = 1(1)20{
1133          if race[`i'] == 1{
1134              dis in y "`i'" in g " 号是" in y " 白人"
1135          }
1136          else if race[`i'] ==2{
1137              dis in y "`i'" in g " 号是" in y " 黑人"
1138          }
1139          else{
1140              dis in y "`i'" in g " 号是" in y " 其它人种"
1141          }
1142      }
1143
1144
1145  *--示例 4:
1146  *
1147  *-- 目的: Tukey power(n) function of variable (x)
1148  *
1149  *-- 变换规则:
1150  *
1151  *      {      x^n      if n > 0
1152  *      z = {      ln(x)   if n = 0
1153  *      {      -x^n      if n < 0
1154  *
1155  *-----mygen.ado-----
1156      cap program drop mygen
1157      program define mygen
1158      version 10
1159      syntax varname(numeric), Power(integer)
1160      if `power'>0 {
1161          generate `varlist' _p`power' = `varlist'^`power'
1162          label var `varlist' _p`power'    "`varlist'^`power'"
1163      }
1164      else if `power'==0 {
1165          generate ln_`varlist' = ln(`varlist')
1166          label var ln_`varlist'    "ln(`varlist')"
1167      }
1168      else {
1169          generate `varlist' _np`=-`power' = -`varlist'^(`power')
1170          label var `varlist' _np`=-`power'    "-`varlist'^(`power'"
1171      }
1172      end
1173  *-----
1174
1175  *--测试:
1176      sysuse auto, clear
1177      mygen price, power(-2)
1178      mygen price, p(0)
1179      mygen price, p(3)
1180      d *price*
1181
1182  *--如下命令是错误的
1183      mygen price , power(0.5)      // Power(integer)
1184      mygen price weight, power(0)  // varname

```

```

1185         mygen make, power(0)           // varname(numeric)
1186
1187
1188     *-示例 5: 寻找变量的最大值
1189     sysuse auto, clear
1190     local max = price[1]
1191     local N = _N
1192     forvalues i = 2/`N'{
1193         set trace on           // 具体过程解析
1194         if `max' < price[`i']{
1195             local max = price[`i']
1196         }
1197         else{                 // 这个语句不必要
1198             local max = `max'
1199         }
1200     }
1201     dis `max'
1202     sum price
1203
1204     *-解决方法 2: 使用 cond() 函数
1205     sysuse auto, clear
1206     gen max = price in 1/2           // Q: 为何 in 1/2 ?
1207     list price max in 1/10
1208     replace max = cond(price>max[_n-1], price, max[_n-1]) in 2/74
1209                                   // Q: 为何 in 2/74 ?
1210     order price max
1211     list price max
1212     local max = max[_N]
1213     dis `max'
1214
1215     *-练习: 如何对变量 price 的值进行排序? (不能使用sort或gsort命令)
1216
1217
1218
1219
1220     *-----
1221     *-5.6.2 一些有用的条件函数
1222
1223     *-参见 A2_data.do:    *-2.1.2.4 利用条件函数产生虚拟变量
1224
1225     * -cond()- 函数: 二元条件语句
1226     * 基本语法: cond(x, a, b)
1227     * 示例
1228     scalar aa = 1
1229     dis cond(aa==1, "这小子真帅!", "这女孩真靓!")
1230
1231     * -inrange()- 函数: 取值区间的判断
1232     * 基本语法: inrange(z,a,b)
1233     * 示例
1234     sysuse nls88, clear
1235     tab grade
1236     gen d_grade = inrange(grade, 12, 16)
1237     list grade d_grade in 1/40, sepby(d_grade)
1238
1239     * -inlist()- 函数: 枚举判断
1240     * 基本语法: inlist(z, a, b, ...)
1241     help inlist()
1242
1243     * -clip()- 函数: 分段区间判断
1244     * 基本语法: clip(x,a,b)
1245     help clip()
1246
1247     * -missing()- 函数:
1248     * 基本语法: missing(x1,x2,...,xn) or mi(x1,x2,...,xn)
1249     help mi()
1250
1251
1252
1253
1254
1255
1256
1257
1258

```

```

1259      *=====
1260      *      计量分析与STATA应用
1261      *=====
1262
1263      *      主讲人：连玉君 博士
1264
1265      *      单 位：中山大学岭南学院金融系
1266      *      电 邮：arlionn@163.com
1267      *      主 页：http://blog.cnfol.com/arlion
1268
1269      *      ::第一部分::
1270      *      Stata 操作
1271      *      =====
1272      *      第五讲 STATA 编程初步
1273      *      =====
1274      *      -5.7- 引用 Stata 命令的返回值
1275
1276
1277      *-----
1278      *-> 5.7  引用 Stata 命令的返回值
1279      *-----
1280
1281      *      ==本节目录==
1282
1283      *      5.7.1 留存在内存中的结果
1284      *      5.7.2 r-class
1285      *      5.7.3 e-class
1286      *      5.7.4 c-class
1287
1288
1289      *      本节命令
1290      *-----
1291      * return list, ereturn list, sreturn list, creturn list
1292      *-----
1293
1294
1295      *-----
1296      *-5.7.1 留存在内存中的结果
1297
1298      *- Stata 命令分为三种类型：
1299
1300      * (1) r-class 与模型估计无关的命令； 如， summary
1301      * (2) e-class 与模型估计有关的命令； 如， regress
1302      * (3) s-class 其它命令； 如， list
1303      * (4) c-class 存储系统参数
1304
1305      *- 显示留存值的方法：
1306      * r-class: return list
1307      * e-class: ereturn list
1308      * s-class: sreturn list
1309      * c-class: creturn list
1310
1311      *- 留存值分为四种类型：
1312      * 单值： 如， r(mean), r(max), r(N), e(r2), e(F)
1313      * 矩阵： 如， e(b), e(V)
1314      * 暂元： 如， e(cmd), e(depvar)
1315      * 函变量： 如， e(sample)
1316
1317
1318      *-----
1319      *-5.7.2 r-class
1320
1321      sysuse auto, clear
1322      sum price
1323      return list
1324      dis "汽车的平均价格是: " in g `r(mean)' // 两种方法均可
1325      dis "汽车的平均价格是: " in g r(mean)
1326      local ss = r(sum) // 引用留存值
1327      dis "所有汽车的价格总和是: " in g `ss'
1328
1329      *-----示 例-----
1330      * 计算一组变量的取值范围，并存于一个矩阵中
1331      sysuse auto, clear
1332      local vars "price weight gear_ratio"

```



```

1333     local n = wordcount("`vars'")
1334     mat aa = J(`n',4,0)
1335     local i = 1
1336     foreach v of varlist `vars'{
1337         qui sum `v'
1338         mat aa[`i++',1] = (r(mean),r(min),r(max),`=r(max)-r(min)')
1339     }
1340     mat colnames aa = mean min max range
1341     mat rownames aa = `vars'
1342     mat list aa
1343     *-----
1344
1345
1346     *-封装之，以便反复调用
1347
1348     *----- asum.ado -----begin---
1349     *! Author:   Roger Federer
1350     *! Date:     2010.10.10
1351     *! Version: 1.0.0
1352
1353     cap program drop asum
1354     program define asum, rclass // 程序类型为 r-class
1355     version 8.0
1356     syntax varlist             // 输入项
1357
1358     local n = wordcount("`varlist'")
1359     tempname aa                // 定义暂时性矩阵
1360     mat `aa' = J(`n',4,0)
1361     local i = 1
1362     foreach v of varlist `varlist'{
1363         qui sum `v'
1364         local range = r(max)-r(min)
1365         mat `aa'[`i++',1] = (r(mean), r(min), r(max), `range')
1366     }
1367     mat colnames `aa' = 平均值 最小值 最大值 取值范围
1368     mat rownames `aa' = `varlist'
1369
1370     * 列示结果
1371     dis _n in g _dup(20) "=" in y "我的统计结果" in g _dup(20) "="
1372     mat list `aa', noheader
1373
1374     * 返回值
1375     return add                // 加上该命令，看结果有何变化？
1376     return scalar range = `range'
1377     return matrix r = `aa'
1378
1379     end
1380     *----- asum.ado -----over---
1381
1382     sysuse auto, clear
1383     asum price weight length
1384     ret list
1385     mat list r(r)
1386
1387     *-存储之，以便永久使用
1388     doedit asum.ado
1389     adopath + D:\stata11\ado\personal\Net_course_A\A5_prog
1390     asum mpg turn
1391     which asum
1392
1393     *-该程序的缺陷：
1394     * (1) 结果的显示不够美观
1395     * (2) 程序没有选项，缺乏弹性
1396     * (3) 只能返回最后一个变量的计算结果
1397     ret list
1398
1399
1400
1401     *-----
1402     *-5.7.3 e-class
1403
1404     *-> 在高级视频中会非常详细的介绍
1405
1406     sysuse auto, clear

```

```

1407 regress price weight length foreign
1408 ereturn list
1409
1410 dis "The method is: " in g e(model)
1411 dis "最大似然值 = " in g e(ll)
1412 dis "R-sq = " r(r2) // ~~ 错误
1413 dis "R-sq = " e(r2) // ^-^ 正确
1414
1415 dis "系数向量为: "
1416 mat list e(b)
1417
1418 dis "系数的方差-协方差矩阵为: "
1419 mat list e(V), format(%6.2f)
1420
1421 *- e(sample) 的内容
1422 sysuse auto, clear
1423 count if rep78>4
1424 reg price weight length rep78 if rep78<=4
1425 sum price
1426 sum price if e(sample) == 1
1427 gen e_sample = e(sample)
1428 list rep78 e_sample in 1/20, sepby(e_sample)
1429
1430 *- 示例1: 控制缺漏值
1431 sysuse nlsw88, clear
1432 sum
1433 gen ln_wage = ln(wage)
1434 gen ln_hours = ln(hours)
1435 local vv "ln_wage married ln_hours ttl_exp"
1436 reg `vv'
1437 sum `vv' if e(sample)
1438 tabstat `vv' if e(sample), stat(mean sd min max N) ///
1439 format(%6.3f) c(s)
1440
1441 *- 示例2: 样本内预测
1442 sysuse auto, clear
1443 qui reg price weight length rep78 if rep78<=4
1444 predict y_hat_a // 回归拟合值
1445 predict res_a , res // 残差
1446 gen e_sample = e(sample)
1447 predict y_hat if e(sample)
1448 predict res if e(sample), res
1449 format y_hat* res* %6.2f
1450 list rep78 price y_hat* res* e_sample in 1/30, sepby(e_sample)
1451
1452 gsort -rep78
1453 list rep78 price y_hat* res* e_sample in 1/30, sepby(e_sample)
1454
1455
1456
1457 *-----
1458 *-5.7.4 c-class
1459
1460 *-> 提供了大量提供参数的返回值，编程时非常有用
1461
1462 creturn list
1463
1464 *- 常数值
1465 dis `c(pi)' // 圆周率
1466 dis "`c(alpha)'" // 英文字母
1467 dis "`c(seed)'" // 种子值
1468 dis `c(maxvar)' // 当前版本所允许的最大变量数
1469 dis `c(memory)'
1470 clear
1471 set memory 10m
1472 dis `c(memory)'
1473 sysuse nlsw88, clear
1474 dis `c(k)' // 变量的个数
1475 dis `c(N)' // 观察值个数
1476
1477 *- 文件路径
1478 dis "`c(sysdir_personal)'"
1479 dis "`c(sysdir_plus)'"
1480 cd `c(sysdir_personal)'Net_course_A

```

```

1481      cd
1482      cd A5_prog
1483      dis "`c(adopath)'"
1484      dis "`c(pwd)'"
1485      adopath
1486
1487      *- 时间
1488      dis "`c(current_date)'" // 当前日期
1489      dis "`c(current_time)'" // 当前时间
1490      *-应用
1491      doedit `c(sysdir_stata)'profile.do
1492      dir `c(sysdir_stata)'do\*.log
1493      shellout `c(sysdir_stata)'do\s1Apr2010084746.log
1494
1495      *- 应用实例 A:
1496      local date `c(current_date)'
1497      local time `c(current_time)'
1498      local vers `c(stata_version)'
1499      local mem `c(memory)'
1500      local flav = cond(`c(MP)', "MP", cond(`c(SE)', "SE", "IC"))
1501      local cwd `c(pwd)'
1502      display _newline "Run `date' at `time' on Stata/`flav' " ///
1503                      "version `vers', memory = `mem' bytes"
1504      display _newline "Current working directory: `cwd'"
1505
1506      *- 应用实例 B:
1507      sysuse auto, clear
1508      display _n "Datafile: `c(filename)' (N=`c(N)', k=`c(k))" ///
1509              " as of `c(filedate)'"
1510
1511
1512      *-一些有用的外部命令
1513
1514      * -adoedit- 在 do editor 中编辑 ado 文件
1515
1516
1517
1518      *=====后 记=====
1519
1520      * 至此，我们已经完成了 Stata 入门知识的学习
1521
1522      * 在第二部分中，我们将学习如何使用 Stata 分析和估计各种计量模型
1523
1524      *=====
1525
1526      exit
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536      *-----
1537      * 笑话: stata修炼宝典
1538      *-----
1539      *
1540      *-测试说明: 输入 t1-t2 的时间值，运行结果
1541
1542      *-----测试开始-----
1543
1544      local t1 = 16 // "蹂躏 STATA 的时间"
1545      local t2 = 0.5 // "打游戏的时间"
1546      local t3 = .2 // "陪女朋友的时间"
1547      local t4 = .2 // "打网球的时间"
1548
1549      if invlogit(`t1')>normal(`t2'+`t3'+`t4'-1.5){
1550          dis "恭喜: 你将成为 STATA 高手! "
1551      }
1552      else{
1553          dis "悲哀: STATA 将蹂躏你! "
1554      }

```

```
1555
1556      *-----测试结束-----
1557
1558
1559      * 结论：你不蹂躏STATA，STATA就会蹂躏你！
1560
1561      twoway (function y = invlogit(x), range(0 14)) ///
1562              (function y = normal(x-1.5), range(0 14))
1563
1564
1565
1566
1567
1568
```