

# 通过 CSV 文件生成 L<sup>A</sup>T<sub>E</sub>X 表格的几种方法

耿楠

西北农林科技大学信息工程学院, 陕西·杨凌, 712100

2019 年 4 月 19 日

## 摘要

CSV 文件称**逗号分隔值** (Comma-Separated Values, CSV, 有时也称为**字符分隔值**, 因为分隔字符也可以不是逗号) 文件, 以纯文本形式存储表格数据 (数字和文本)。纯文本意味着该文件是一个字符序列, 不含必须像二进制数字那样被解读的数据。CSV 文件由任意数目的记录组成, 记录间以某种换行符分隔; 每条记录由字段组成, 字段间的分隔符是其它字符或字符串, 最常见的是逗号或制表符。通常, 所有记录都有完全相同的字段序列。CSV 文件都是纯文本文件, 可以使用记事本、Excel 等软件进行生成或编辑, 是一种比较方便的数据管理方式。在 L<sup>A</sup>T<sub>E</sub>X 中可以采用 csvsimple、pgfplotstable、datatool、csvtools 等宏包直接使用 CSV 文件的数据生成 L<sup>A</sup>T<sub>E</sub>X 表格。

## 一、准备 CSV 文件数据

CSV 文件数据可以使用记事本、Excel 等软件生成, 也可以在导言区用 `filecontents*` 环境生成。

```
1 \begin{filecontents*}{db1.csv}
2   姓名, 性别, 年龄
3   张三, 男, 18
4   李四, 男, 45
5   马五, 女, 16
6 \end{filecontents*}
```

该代码会在当前工作目录下生成 “db1.csv” 数据文件。

## 二、使用 csvsimple 宏包生成 L<sup>A</sup>T<sub>E</sub>X 表格

csvsimple 是一个用于处理 CSV 文件数据的宏包, 它采用了 pgfkeys 的 key-value 语法, 是一个基于已有数据生成表格的轻量级工具包。在导言区使用 “`\usepackage {csvsimple}`” 便可以使用该宏包提供的功能, 详情请查阅其说明<http://www.ctan.org/pkg/csvsimple>或 <https://github.com/T-F-S/csvsimple>。

### 1、简单方式生成表格

使用 csvsimple 宏包的最简单方式是直接 `\csvautotabular` 命令生成表格, 其代码如下, 生成的表格如表1所示。

```

1 \begin{table}[htb]
2   \centering
3   \caption{使用\cs{csvautotabular}命令生成表格\label{tab01}}
4   \csvautotabular{db1.csv}
5 \end{table}

```

表 1: 使用\csvautotabular 命令生成表格

姓名	性别	年龄
张三	男	18
李四	男	45
马五	女	16

## 2、读入数据生成

为了能够更为灵活地控制生成的表格，可以使用\csvreader 命令读入数据，并对表格属性进行必要地设置，其代码如下，生成的表格如表2所示。

```

1 \begin{table}[htb]
2   \centering
3   \caption{使用\cs{csvreader}命令生成表格\label{tab02}}
4   \csvreader[tabular=|c|c|c|c|,% 列格式
5     table head=\hline & 姓名 & 性别 & 年龄\\ \hline,% 表头
6     late after line=\\ \hline % 表格线
7   ]%
8     {db1.csv}% 数据文件
9     {姓名 =\name, 性别 =\gender, 年龄 =\age}% 字段命名
10    {\thecsvrow & \name & \gender & \age}% 读入数据生成表格
11 \end{table}

```

表 2: 使用\csvreader 命令生成表格

	姓名	性别	年龄
1	张三	男	18
2	李四	男	45
3	马五	女	16

## 3、读入数据生成三线表

结合 booktabs 宏包，可以非常方便的用\csvautobooktabular 直接生成三线表格，其代码如下，生成的表格如表3所示。

```

1 \begin{table}[htb]
2   \centering
3   \caption{使用\cs{csvautobooktabular}命令生成三线表\label{tab03}}
4   \csvautobooktabular{db1.csv}% 数据文件
5 \end{table}

```

表 3: 使用\csvautobooktabular 命令生成三线表

姓名	性别	年龄
张三	男	18
李四	男	45
马五	女	16

#### 4、生成跨页长表格

可以通过\csvloop 命令引入\longtable 环境，以生成跨页表格。其示例代码如下，生成的表格如表4所示。

```

1 \csvloop{
2   file=db4.csv,
3   respect all,
4   separator=comma,
5   no head,
6   before reading={
7     \begin{longtable}[c]{c*{6}{r}}
8     \caption[实验数据]{实验数据，这个题注十分的长，注意这在索引中的处理
9       \toprule
10    },
11    command={\csviffirstrow
12      {
13        \headrow
14        \midrule
15        \endfirsthead
16      }
17      {\csvcoli & \csvcolii & \csvcoliii & \csvcoliv & \csvcolv &
18        \csvcolvi & \csvcolvii}
19    },
20    late after line=\\,
21    late after first line=\ctntabcmd,
22    late after last line=\\bottomrule,
23    after reading=\end{longtable}
24 }

```

表 4: 实验数据, 这个题注十分的长, 注意这在索引中的处理方式, 还有 \caption 后面的双反斜杠

测试程序	正常运行 时间 (s)	同步 时间 (s)	检查点 时间 (s)	卷回恢复 时间 (s)	进程迁移 时间 (s)	检查点 文件 (KB)
CG.A.2	23.05	0.002	0.116	0.035	0.589	32491
CG.A.4	15.06	0.003	0.067	0.021	0.351	18211
CG.A.8	13.38	0.004	0.072	0.023	0.210	9890
CG.B.2	867.45	0.002	0.864	0.232	3.256	228562
CG.B.4	501.61	0.003	0.438	0.136	2.075	123862
CG.B.8	384.65	0.004	0.457	0.108	1.235	63777
MG.A.2	112.27	0.002	0.846	0.237	3.930	236473
MG.A.4	59.84	0.003	0.442	0.128	2.070	123875
MG.A.8	31.38	0.003	0.476	0.114	1.041	60627
MG.B.2	526.28	0.002	0.821	0.238	4.176	236635
MG.B.4	280.11	0.003	0.432	0.130	1.706	123793
MG.B.8	148.29	0.003	0.442	0.116	0.893	60600
LU.A.2	2116.54	0.002	0.110	0.030	0.532	28754
LU.A.4	1102.50	0.002	0.069	0.017	0.255	14915
LU.A.8	574.47	0.003	0.067	0.016	0.192	8655
LU.B.2	9712.87	0.002	0.357	0.104	1.734	101975
LU.B.4	4757.80	0.003	0.190	0.056	0.808	53522
LU.B.8	2444.05	0.004	0.222	0.057	0.548	30134
CG.B.2	867.45	0.002	0.864	0.232	3.256	228562
CG.B.4	501.61	0.003	0.438	0.136	2.075	123862
CG.B.8	384.65	0.004	0.457	0.108	1.235	63777
MG.A.2	112.27	0.002	0.846	0.237	3.930	236473
MG.A.4	59.84	0.003	0.442	0.128	2.070	123875
MG.A.8	31.38	0.003	0.476	0.114	1.041	60627
MG.B.2	526.28	0.002	0.821	0.238	4.176	236635
MG.B.4	280.11	0.003	0.432	0.130	1.706	123793
MG.B.8	148.29	0.003	0.442	0.116	0.893	60600
LU.A.2	2116.54	0.002	0.110	0.030	0.532	28754
LU.A.4	1102.50	0.002	0.069	0.017	0.255	14915
LU.A.8	574.47	0.003	0.067	0.016	0.192	8655
LU.B.2	9712.87	0.002	0.357	0.104	1.734	101975
LU.B.4	4757.80	0.003	0.190	0.056	0.808	53522
LU.B.8	2444.05	0.004	0.222	0.057	0.548	30134

续下页

续表 4 实验数据

测试程序	正常运行 时间 (s)	同步 时间 (s)	检查点 时间 (s)	卷回恢复 时间 (s)	进程迁移 时间 (s)	检查点 文件 (KB)
EP.A.2	123.81	0.002	0.010	0.003	0.074	1834
EP.A.4	61.92	0.003	0.011	0.004	0.073	1743
EP.A.8	31.06	0.004	0.017	0.005	0.073	1661
EP.B.2	495.49	0.001	0.009	0.003	0.196	2011
EP.B.4	247.69	0.002	0.012	0.004	0.122	1663
EP.B.8	126.74	0.003	0.017	0.005	0.083	1656

如果再结合 `siunitx`、`longtable`、`xcolor` 等宏包，则可以对生成的表格进行更为细致的控制，有关细节，请查阅 `csvsimple` 宏包使用手册。

### 三、使用 `pgfplotstable` 宏包生成 $\text{\LaTeX}$ 表格

`pgfplotstable` 是一个用于处理 CSV 文件数据的宏包，它为生成表格提供了丰富的设置命令。在导言区使用 “`\usepackage {pgfplotstable}`” 便可以使用该宏包提供的功能，详情请查阅其说明<https://ctan.org/pkg/pgfplotstable>。

#### 1、简单方式

使用 `pgfplotstable` 宏包的最简单方式是直接 `\pgfplotstabletypeset` 命令生成表格，利用该命令的可选参数，可以对最终生成表格的格式进行必要的设置，并且可以选择性的选取要输出的数据列或是对数据列的顺序进行调整。其示例代码如下，生成的表格如表5所示<sup>1</sup>。

```

1 \begin{table}[htb]
2   \centering
3   \caption{使用\cs{pgfplotstabletypeset}命令生成表格\label{tab04}}
4   \pgfplotstabletypeset[
5     col sep=comma,
6     string type,
7     columns/name/.style={column name= 姓名, column type={|l}},
8     columns/gender/.style={column name= 性别, column type={|l}},
9     columns/age/.style={column name= 年龄, column type={|c|}},
10    every head row/.style={before row=\hline,after row=\hline},
11    every last row/.style={after row=\hline},
12    ]{db2.csv}
13 \end{table}

```

<sup>1</sup>该例中使用了 `db2.csv` 数据，第 1 行是各列的名称，可以重新为各列命名。

表 5: 使用\pgfplotstabletypeset 命令生成表格

姓名	性别	年龄
张三	男	18
李四	男	45
马五	女	16

## 2、结合 booktabs 生成三线表

结合 booktabs 宏包, 通过\pgfplotstabletypeset 命令的的可选参数, 可以方便地实现三线表。其示例代码如下, 生成的表格如表6所示<sup>2</sup>。

```

1 \begin{table}[htb]
2   \centering
3   \caption{结合\pkg{booktabs}宏包生成三线表\label{tab05}}
4   \pgfplotstabletypeset[
5     col sep=comma,
6     string type,
7     columns/0/.style={column name= 姓名, column type={l}},
8     columns/1/.style={column name= 性别, column type={l}},
9     columns/2/.style={column name= 年龄, column type={c}},
10    every head row/.style={before row=\toprule,after row=\midrule},
11    every last row/.style={after row=\bottomrule},
12    ]{db2.csv}
13 \end{table}

```

表 6: 结合 booktabs 宏包生成三线表

name	gender	age
张三	男	18
李四	男	45
马五	女	16

## 3、结合 siunitx 控制数据显示精度

结合 siunitx 宏包, 通过\pgfplotstabletypeset 命令的的可选参数, 可以方便地实现对数据显示精度的控制。其示例代码如下, 生成的表格如表7所示<sup>3</sup>。

<sup>2</sup>该例中使用了 db2.csv 数据, 第 1 行是各列的名称, 该例中使用列号选择各列, 并为各列重新命名。

<sup>3</sup>该例中使用了 db3.csv 数据, 第 1 行是各列的名称, 注意数据中小数点不可省略。

```

1 \begin{table}[htb]
2   \centering
3   \caption{结合\pkg{siunitx}宏包控制数据显示精度\label{tab06}}
4   \pgfplotstabletypeset[
5     multicolumn names,
6     col sep=comma,
7     display columns/0/.style={
8       column name=$Value 1$,
9       column type={S[table-format = 3.0 ,round-precision=0]},string type},
10    display columns/1/.style={
11      column name=$Value 2$,
12      column type={S[table-format = 3.2 ,round-precision=2]},
13      string type},
14    every head row/.style={
15      before row={\toprule},
16      after row={
17        \si{\ampere} & \si{\volt}\\
18        \midrule}
19      },
20    every last row/.style={after row=\bottomrule}, %
21    ]{db3.csv} %
22 \end{table}

```

表 7: 结合 siunitx 宏包控制数据显示精度

<i>Value1</i>	<i>Value2</i>
A	V
1	11.43
2	2342.23
3	55.00

#### 4、使用 longtable 宏包生成跨页表格

可以通过`\begin table`和`\end table`参数引入`\longtable`环境，以生成跨页表格。其示例代码如下，生成的表格如表8所示。

```

1 \pgfplotstabletypeset[
2   multicolumn names, % 允许合并列名
3   col sep=comma, % 逗号数据分割
4   begin table=\begin{longtable}, % 使用 longtable 环境
5   end table=\end{longtable},
6   display columns/0/.style={
7     column name={\multirow{2}{*}{测试程序}}, % 列名称
8     column type={c},string type},
9   display columns/1/.style={
10    column name= 正常运行, % 列名称
11    column type={r},int detect},
12   display columns/2/.style={
13    column name= 同步, % 列名称
14    column type={r},int detect},
15   display columns/3/.style={
16    column name= 检测点, % 列名称
17    column type={r},int detect},
18   display columns/4/.style={
19    column name= 卷回恢复, % 列名称
20    column type={r},int detect},
21   display columns/5/.style={
22    column name= 进程迁移, % 列名称
23    column type={r},int detect},
24   display columns/6/.style={
25    column name= 检查点, % 列名称
26    column type={r},int detect},
27   every head row/.append style={
28     before row={\caption[实验数据]{实验数据, 这个题注十分的长, 注意这在索引中的处理方式, 还有 \cs{caption} 后面的双反斜杠\label{tabpgfltab}}\toprule},
29     after row={%
30       & 时间 (s) & 时间 (s) & 时间 (s) & 时间 (s) & 时间 (s) & 文件 (KB)\% 数据单
31       位, 用 & 分割
32       \midrule
33       \endfirsthead
34       \ctncap\%
35       \toprule
36       \headrow
37       \midrule
38       \endhead
39       \hline
40       \multicolumn{7}{r}{续下页} \\\
41       \endfoot
42       \endlastfoot
43     },
44     every last row/.style={after row=\bottomrule}, %
45 ]{db4.csv} % filename/path to file

```



表 8: 实验数据, 这个题注十分的长, 注意这在索引中的处理方式, 还有 \caption 后面的双反斜杠

测试程序	正常运行 时间 (s)	同步 时间 (s)	检测点 时间 (s)	卷回恢复 时间 (s)	进程迁移 时间 (s)	检查点 文件 (KB)
CG.A.2	$2.31 \cdot 10^1$	$2 \cdot 10^{-3}$	$1.16 \cdot 10^{-1}$	$3.5 \cdot 10^{-2}$	$5.89 \cdot 10^{-1}$	32,491
CG.A.4	$1.51 \cdot 10^1$	$3 \cdot 10^{-3}$	$6.7 \cdot 10^{-2}$	$2.1 \cdot 10^{-2}$	$3.51 \cdot 10^{-1}$	18,211
CG.A.8	$1.34 \cdot 10^1$	$4 \cdot 10^{-3}$	$7.2 \cdot 10^{-2}$	$2.3 \cdot 10^{-2}$	$2.1 \cdot 10^{-1}$	9,890
CG.B.2	$8.67 \cdot 10^2$	$2 \cdot 10^{-3}$	$8.64 \cdot 10^{-1}$	$2.32 \cdot 10^{-1}$	$3.26 \cdot 10^0$	228,562
CG.B.4	$5.02 \cdot 10^2$	$3 \cdot 10^{-3}$	$4.38 \cdot 10^{-1}$	$1.36 \cdot 10^{-1}$	$2.08 \cdot 10^0$	123,862
CG.B.8	$3.85 \cdot 10^2$	$4 \cdot 10^{-3}$	$4.57 \cdot 10^{-1}$	$1.08 \cdot 10^{-1}$	$1.24 \cdot 10^0$	63,777
MG.A.2	$1.12 \cdot 10^2$	$2 \cdot 10^{-3}$	$8.46 \cdot 10^{-1}$	$2.37 \cdot 10^{-1}$	$3.93 \cdot 10^0$	236,473
MG.A.4	$5.98 \cdot 10^1$	$3 \cdot 10^{-3}$	$4.42 \cdot 10^{-1}$	$1.28 \cdot 10^{-1}$	$2.07 \cdot 10^0$	123,875
MG.A.8	$3.14 \cdot 10^1$	$3 \cdot 10^{-3}$	$4.76 \cdot 10^{-1}$	$1.14 \cdot 10^{-1}$	$1.04 \cdot 10^0$	60,627
MG.B.2	$5.26 \cdot 10^2$	$2 \cdot 10^{-3}$	$8.21 \cdot 10^{-1}$	$2.38 \cdot 10^{-1}$	$4.18 \cdot 10^0$	236,635
MG.B.4	$2.8 \cdot 10^2$	$3 \cdot 10^{-3}$	$4.32 \cdot 10^{-1}$	$1.3 \cdot 10^{-1}$	$1.71 \cdot 10^0$	123,793
MG.B.8	$1.48 \cdot 10^2$	$3 \cdot 10^{-3}$	$4.42 \cdot 10^{-1}$	$1.16 \cdot 10^{-1}$	$8.93 \cdot 10^{-1}$	60,600
LU.A.2	$2.12 \cdot 10^3$	$2 \cdot 10^{-3}$	$1.1 \cdot 10^{-1}$	$3 \cdot 10^{-2}$	$5.32 \cdot 10^{-1}$	28,754
LU.A.4	$1.1 \cdot 10^3$	$2 \cdot 10^{-3}$	$6.9 \cdot 10^{-2}$	$1.7 \cdot 10^{-2}$	$2.55 \cdot 10^{-1}$	14,915
LU.A.8	$5.74 \cdot 10^2$	$3 \cdot 10^{-3}$	$6.7 \cdot 10^{-2}$	$1.6 \cdot 10^{-2}$	$1.92 \cdot 10^{-1}$	8,655
LU.B.2	$9.71 \cdot 10^3$	$2 \cdot 10^{-3}$	$3.57 \cdot 10^{-1}$	$1.04 \cdot 10^{-1}$	$1.73 \cdot 10^0$	101,975
LU.B.4	$4.76 \cdot 10^3$	$3 \cdot 10^{-3}$	$1.9 \cdot 10^{-1}$	$5.6 \cdot 10^{-2}$	$8.08 \cdot 10^{-1}$	53,522
LU.B.8	$2.44 \cdot 10^3$	$4 \cdot 10^{-3}$	$2.22 \cdot 10^{-1}$	$5.7 \cdot 10^{-2}$	$5.48 \cdot 10^{-1}$	30,134
CG.B.2	$8.67 \cdot 10^2$	$2 \cdot 10^{-3}$	$8.64 \cdot 10^{-1}$	$2.32 \cdot 10^{-1}$	$3.26 \cdot 10^0$	228,562
CG.B.4	$5.02 \cdot 10^2$	$3 \cdot 10^{-3}$	$4.38 \cdot 10^{-1}$	$1.36 \cdot 10^{-1}$	$2.08 \cdot 10^0$	123,862
CG.B.8	$3.85 \cdot 10^2$	$4 \cdot 10^{-3}$	$4.57 \cdot 10^{-1}$	$1.08 \cdot 10^{-1}$	$1.24 \cdot 10^0$	63,777
MG.A.2	$1.12 \cdot 10^2$	$2 \cdot 10^{-3}$	$8.46 \cdot 10^{-1}$	$2.37 \cdot 10^{-1}$	$3.93 \cdot 10^0$	236,473
MG.A.4	$5.98 \cdot 10^1$	$3 \cdot 10^{-3}$	$4.42 \cdot 10^{-1}$	$1.28 \cdot 10^{-1}$	$2.07 \cdot 10^0$	123,875
MG.A.8	$3.14 \cdot 10^1$	$3 \cdot 10^{-3}$	$4.76 \cdot 10^{-1}$	$1.14 \cdot 10^{-1}$	$1.04 \cdot 10^0$	60,627
MG.B.2	$5.26 \cdot 10^2$	$2 \cdot 10^{-3}$	$8.21 \cdot 10^{-1}$	$2.38 \cdot 10^{-1}$	$4.18 \cdot 10^0$	236,635
MG.B.4	$2.8 \cdot 10^2$	$3 \cdot 10^{-3}$	$4.32 \cdot 10^{-1}$	$1.3 \cdot 10^{-1}$	$1.71 \cdot 10^0$	123,793
MG.B.8	$1.48 \cdot 10^2$	$3 \cdot 10^{-3}$	$4.42 \cdot 10^{-1}$	$1.16 \cdot 10^{-1}$	$8.93 \cdot 10^{-1}$	60,600
LU.A.2	$2.12 \cdot 10^3$	$2 \cdot 10^{-3}$	$1.1 \cdot 10^{-1}$	$3 \cdot 10^{-2}$	$5.32 \cdot 10^{-1}$	28,754
LU.A.4	$1.1 \cdot 10^3$	$2 \cdot 10^{-3}$	$6.9 \cdot 10^{-2}$	$1.7 \cdot 10^{-2}$	$2.55 \cdot 10^{-1}$	14,915
LU.A.8	$5.74 \cdot 10^2$	$3 \cdot 10^{-3}$	$6.7 \cdot 10^{-2}$	$1.6 \cdot 10^{-2}$	$1.92 \cdot 10^{-1}$	8,655
LU.B.2	$9.71 \cdot 10^3$	$2 \cdot 10^{-3}$	$3.57 \cdot 10^{-1}$	$1.04 \cdot 10^{-1}$	$1.73 \cdot 10^0$	101,975
LU.B.4	$4.76 \cdot 10^3$	$3 \cdot 10^{-3}$	$1.9 \cdot 10^{-1}$	$5.6 \cdot 10^{-2}$	$8.08 \cdot 10^{-1}$	53,522
LU.B.8	$2.44 \cdot 10^3$	$4 \cdot 10^{-3}$	$2.22 \cdot 10^{-1}$	$5.7 \cdot 10^{-2}$	$5.48 \cdot 10^{-1}$	30,134

续下页

续表 8 实验数据

测试程序	正常运行 时间 (s)	同步 时间 (s)	检查点 时间 (s)	卷回恢复 时间 (s)	进程迁移 时间 (s)	检查点 文件 (KB)
EP.A.2	$1.24 \cdot 10^2$	$2 \cdot 10^{-3}$	$1 \cdot 10^{-2}$	$3 \cdot 10^{-3}$	$7.4 \cdot 10^{-2}$	1,834
EP.A.4	$6.19 \cdot 10^1$	$3 \cdot 10^{-3}$	$1.1 \cdot 10^{-2}$	$4 \cdot 10^{-3}$	$7.3 \cdot 10^{-2}$	1,743
EP.A.8	$3.11 \cdot 10^1$	$4 \cdot 10^{-3}$	$1.7 \cdot 10^{-2}$	$5 \cdot 10^{-3}$	$7.3 \cdot 10^{-2}$	1,661
EP.B.2	$4.95 \cdot 10^2$	$1 \cdot 10^{-3}$	$9 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$1.96 \cdot 10^{-1}$	2,011
EP.B.4	$2.48 \cdot 10^2$	$2 \cdot 10^{-3}$	$1.2 \cdot 10^{-2}$	$4 \cdot 10^{-3}$	$1.22 \cdot 10^{-1}$	1,663
EP.B.8	$1.27 \cdot 10^2$	$3 \cdot 10^{-3}$	$1.7 \cdot 10^{-2}$	$5 \cdot 10^{-3}$	$8.3 \cdot 10^{-2}$	1,656

另外,也可以使用 `multirow` 宏包、`\multicolumn` 命令实现表格的行列合并。如果再结合 `longtable`、`array`、`colortbl` 等宏包,则可以对生成的表格进行更为细致的控制,有关细节,请查阅 `pgfplotstable` 宏包使用手册。

## 四、使用 datatool 宏包

`datatool` 是一个用于处理 CSV 文件数据的宏包,使用它提供的 `\DTLloaddb` 结合 `\DTLdisplaydb` 命令或 `\DTLforeach` 循环,可以实现通过数据生成表格的操作。在导言区使用 “`\usepackage {datatool}`” 便可以使用该宏包提供的功能,详情请查阅其说明<https://ctan.org/pkg/datatool>。

### 1、简单方式

可以通过 `\DTLloaddb` 命令载入 CSV 文件数据,然后用 `\DTLdisplaydb` 命令生成表格。其示例代码如下,生成的表格如表9所示<sup>4</sup>。

```

1 \DTLloaddb[keys={col1,col2,col3}]{mydb}{db1.csv}% 可以在之前任何位置载入数据
2 \begin{table}[htb]
3   \centering
4   \caption{\cs{DTLdisplaydb}命令生成表格\label{tab07}}
5   \DTLdisplaydb{mydb} %
6 \end{table}

```

表 9: \DTLdisplaydb 命令生成表格

姓名	性别	年龄
张三	男	18
李四	男	45

### 2、使用 \DTLforeach 循环构建表格

可以通过 `\DTLloaddb` 命令载入 CSV 文件数据,然后用 `\DTLforeach` 循环命令生成表格,此时,可以按普通表格的编写方式进行数据处理。其示例代码如下,生成的表格如表10所示。

<sup>4</sup> 此处使用 `\DTLremove rowmydb3` 命令删除了第 3 行数据。

```

1 \DTLloaddb{table}{db2.csv}% 可以在之前任何位置载入数据
2 \begin{table}[htb]
3   \centering
4   \caption{\cs{DTLforeach}循环命令生成表格\label{tab08}}
5   \begin{tabular}{llc}
6     \toprule
7     姓名 & 年龄 & 性别 \tabularnewline
8     \midrule
9     \DTLforeach*{table}{\name=name, \gender=gender, \age=age}%
10      {\DTLiffirstrow}{\tabularnewline}%
11      \name & \age & \gender}\\ % 数据列位置可任意
12     \bottomrule
13   \end{tabular}
14 \end{table}

```

表 10: \DTLforeach 循环命令生成表格

姓名	年龄	性别
张三	18	男
马五	16	女

### 3、使用 longtable 宏包生成跨页表格

可以通过\DTLloaddb 命令载入 CSV 文件数据，然后用环境结合\DTLforeach 循环命令生成跨页表格。其示例代码如下，生成的表格如表11所示。

```

1 \DTLloadb{ltab}{db4.csv}% 可以在之前任何位置载入数据
2 \begin{longtable}[c]{c*{6}{r}}
3   \caption[实验数据]{实验数据，这个题注十分的长，注意这在索引中的处理方式，还有
   \cs{caption} 后面的双反斜杠}\label{tab09}\\
4   \toprule
5   \headrow
6   \midrule
7   \endfirsthead
8   \ctncap\tabularnewline
9   \toprule
10  \headrow
11  \midrule
12  \endhead
13  \hline
14  \multicolumn{7}{r}{续下页}
15  \endfoot

16  \endlastfoot
17  \DTLforeach*{ltab}{\cola=cola, \colb=colb, \colc=colc, \cold=cold,
18    \cole=cole, \colf=colf, \colg=colg}%
19    {\DTLiffirstrow{}{\tabularnewline}%
20    \cola & \colb & \colc & \cold & \cole & \colf & \colg}\\ % 数据列位置可任意
21  \bottomrule
22 \end{longtable}

```

表 11: 实验数据，这个题注十分的长，注意这在索引中的处理方式，还有 \caption 后面的双反斜杠

测试程序	正常运行 时间 (s)	同步 时间 (s)	检查点 时间 (s)	卷回恢复 时间 (s)	进程迁移 时间 (s)	检查点 文件 (KB)
CG.A.2	23.05	0.002	0.116	0.035	0.589	32491
CG.A.4	15.06	0.003	0.067	0.021	0.351	18211
CG.A.8	13.38	0.004	0.072	0.023	0.210	9890
CG.B.2	867.45	0.002	0.864	0.232	3.256	228562
CG.B.4	501.61	0.003	0.438	0.136	2.075	123862
CG.B.8	384.65	0.004	0.457	0.108	1.235	63777
MG.A.2	112.27	0.002	0.846	0.237	3.930	236473
MG.A.4	59.84	0.003	0.442	0.128	2.070	123875
MG.A.8	31.38	0.003	0.476	0.114	1.041	60627
MG.B.2	526.28	0.002	0.821	0.238	4.176	236635
MG.B.4	280.11	0.003	0.432	0.130	1.706	123793

续下页

续表 11 实验数据

测试程序	正常运行 时间 (s)	同步 时间 (s)	检查点 时间 (s)	卷回恢复 时间 (s)	进程迁移 时间 (s)	检查点 文件 (KB)
MG.B.8	148.29	0.003	0.442	0.116	0.893	60600
LU.A.2	2116.54	0.002	0.110	0.030	0.532	28754
LU.A.4	1102.50	0.002	0.069	0.017	0.255	14915
LU.A.8	574.47	0.003	0.067	0.016	0.192	8655
LU.B.2	9712.87	0.002	0.357	0.104	1.734	101975
LU.B.4	4757.80	0.003	0.190	0.056	0.808	53522
LU.B.8	2444.05	0.004	0.222	0.057	0.548	30134
CG.B.2	867.45	0.002	0.864	0.232	3.256	228562
CG.B.4	501.61	0.003	0.438	0.136	2.075	123862
CG.B.8	384.65	0.004	0.457	0.108	1.235	63777
MG.A.2	112.27	0.002	0.846	0.237	3.930	236473
MG.A.4	59.84	0.003	0.442	0.128	2.070	123875
MG.A.8	31.38	0.003	0.476	0.114	1.041	60627
MG.B.2	526.28	0.002	0.821	0.238	4.176	236635
MG.B.4	280.11	0.003	0.432	0.130	1.706	123793
MG.B.8	148.29	0.003	0.442	0.116	0.893	60600
LU.A.2	2116.54	0.002	0.110	0.030	0.532	28754
LU.A.4	1102.50	0.002	0.069	0.017	0.255	14915
LU.A.8	574.47	0.003	0.067	0.016	0.192	8655
LU.B.2	9712.87	0.002	0.357	0.104	1.734	101975
LU.B.4	4757.80	0.003	0.190	0.056	0.808	53522
LU.B.8	2444.05	0.004	0.222	0.057	0.548	30134
EP.A.2	123.81	0.002	0.010	0.003	0.074	1834
EP.A.4	61.92	0.003	0.011	0.004	0.073	1743
EP.A.8	31.06	0.004	0.017	0.005	0.073	1661
EP.B.2	495.49	0.001	0.009	0.003	0.196	2011
EP.B.4	247.69	0.002	0.012	0.004	0.122	1663
EP.B.8	126.74	0.003	0.017	0.005	0.083	1656

如果再结合 siunitx、longtable、booktabs、array、colortbl 等宏包，则可以对生成的表格进行更为细致的控制，有关细节，请查阅 pgfplotstable 宏包使用手册。

## 五、结论

L<sup>A</sup>T<sub>E</sub>X 排版技术经历了风风雨雨，已积累了大量的相关领域的宏包，借用这些宏包，可以大大减轻排版的工作量。发现这些宏包，善用这些宏包，就可以为我们的工作带来便利。