

通过 CSV 文件生成 L^AT_EX 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 15 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^AT_EX 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^AT_EX 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^AT_EX 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

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 I^AT_EX 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4所示¹。

```

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}

```

¹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 4: 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5所示²。

```

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 5: 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6所示³。

²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。

³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 6: 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

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7所示⁴。

```
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 7: \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8所示。

```
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 8: \DTLforeach 循环命令生成表格

姓名	年龄	性别
张三	18	男
马五	16	女

⁴ 此处使用\命令删除了第 3 行数据。

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^AT_EX 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。