

**texdoc 2.0**  
**An update on creating LaTeX documents from within Stata**

**Example 1**

Ben Jann  
University of Bern, ben.jann@soz.unibe.ch

2016 German Stata Users Group Meeting  
GESIS, Cologne, June 10, 2016

**Contents**

<b>1</b>	<b>The texdoc source file</b>	<b>2</b>
<b>2</b>	<b>The resulting <math>\LaTeX</math> source file</b>	<b>6</b>
<b>3</b>	<b>The resulting PDF</b>	<b>9</b>

# 1 The texdoc source file

— *the-auto-data.texdoc* —

```
texdoc init the-auto-data, replace logdir(log) ///  
  gropts(optargs(width=0.8\textwidth))  
set linesize 100  
  
/**/  
\documentclass[12pt]{article}  
\usepackage{fullpage}  
\usepackage{hyperref,graphicx,booktabs,dcolumn}  
\usepackage{stata}  
  
\title{The Auto Data}  
\author{Ben Jann}  
\date{\today}  
  
\begin{document}  
  
\maketitle  
  
\begin{abstract}  
  I really like the auto data because it is so awesome. You can do all kinds  
  of stuff with the auto data, like tabulating a variable or computing  
  descriptive statistics. You can even use the auto data to estimate  
  regression models. I am really amazed by the richness of this dataset.  
  There is information on many different makes and models and you can learn,  
  for example, about the gear ratio of a Dodge Diplomat (a stunning 2.47). In  
  this article I will illustrate the auto data and I will show you what you  
  can do with it. I am convinced that you will love this dataset as much as I  
  do after having read this paper.  
\end{abstract}  
  
\tableofcontents  
  
\section{Introduction}  
  
What we want to do in the introductory section is to open the data and have a  
look at what is inside of it. Since the auto data is shipped with Stata, we can  
use the \stcmd{sysuse} command to open it (see \dref{sysuse}). Furthermore, the  
\stcmd{describe} command will list the variables and display some other  
information (see \dref{describe}). So let's start:  
  
***/  
  
texdoc stlog  
sysuse auto  
texdoc stlog cnp  
describe  
texdoc stlog close
```

```
/**
```

Wow! 74 observations! And what a wealth of variables! Make, price, miles per gallon, and many more. I am very motivated to learn more about this amazing data set.

```
\section{Descriptives}
```

Let's now look at some descriptive statistics. Maybe also let's do a graph.

```
*/
```

```
texdoc stlog
summarize
pspline price weight
texdoc stlog close
texdoc graph, label(fig1) ///
    caption(What a crazy relation between price and weight)
```

```
/**
```

In figure~\ref{fig1} we see that for some unknown reason expensive cars seem to be heavier.

Actually, I really only want to print a graph without printing the code that produced the code. Hm, how can we do that? Maybe the `\stcmd{nolog}` option will do.

```
*/
```

```
texdoc stlog, nolog
pspline price mpg
texdoc stlog close
texdoc graph, label(fig2) ///
    caption(Another crazy relation)
```

```
/**
```

In figure~\ref{fig2} we see that price is also related to miles per gallon. How interesting!

```
\section{Varieties of logs}
```

The default is to include a log of the commands and their output:

```
*/
```

```
texdoc stlog
display "2 + 2 = " 2 + 2
display "sqrt(2) = " sqrt(2)
```

```
texdoc stlog close
```

```
/**
```

```
\noindent
```

Use the `\stcmd{cmdlog}` option to only include a copy of the commands without output:

```
*/
```

```
texdoc stlog, cmdlog
display "2 + 2 = " 2 + 2
display "sqrt(2) = " sqrt(2)
texdoc stlog close
```

```
/**
```

```
\noindent
```

Conversely, use `\stcmd{cmdstrip}` to print only the output, but not the commands:

```
*/
```

```
texdoc stlog, cmdstrip
display "2 + 2 = " 2 + 2
display "sqrt(2) = " sqrt(2)
texdoc stlog close
```

```
/**
```

```
\noindent
```

Furthermore, use `\stcmd{texdoc stlog oom}` to omit output from selected commands:

```
*/
```

```
texdoc stlog
sysuse auto
texdoc stlog oom ///
list make
display make[1]
texdoc stlog close
```

```
/**
```

```
\section{Regression tables}
```

Finally we get to regressions! To include a table in the document, it is a good idea to enclose the corresponding commands in a `\stcmd{texdoc stlog}` environment and specify the `\stcmd{nolog}` option to suppress the output. This allows you to use the `\stcmd{nodo}` option later on to skip computing the table.

In table~\ref{table1} we see that an additional pound of car costs around 3.5

dollars once we control for milage and origin.

\*\*\*/

```
texdoc stlog, nolog
sysuse auto
regress price weight
estimates store m1
regress price weight mpg
estimates store m2
regress price weight mpg foreign
estimates store m3
estttab m1 m2 m3 using log/table1.tex, replace se label ///
    nomtitles booktabs align(D{.}{.}{-1}) ///
    title(Some regression table\label{table1})
texdoc stlog close
texdoc write \input{log/table1.tex}
```

/\*\*\*

```
\end{document}
```

\*\*\*/

— *end of file* —

## 2 The resulting L<sup>A</sup>T<sub>E</sub>X source file

Applying

```
. texdoc do the-auto-data.texdoc
```

generates to the following L<sup>A</sup>T<sub>E</sub>X file.

— *the-auto-data.tex* —

```
\documentclass[12pt]{article}
\usepackage{fullpage}
\usepackage{hyperref,graphicx,booktabs,dcolumn}
\usepackage{stata}

\title{The Auto Data}
\author{Ben Jann}
\date{\today}

\begin{document}

\maketitle

\begin{abstract}
  I really like the auto data because it is so awesome. You can do all kinds
  of stuff with the auto data, like tabulating a variable or computing
  descriptive statistics. You can even use the auto data to estimate
  regression models. I am really amazed by the richness of this dataset.
  There is information on many different makes and models and you can learn,
  for example, about the gear ratio of a Dodge Diplomat (a stunning 2.47). In
  this article I will illustrate the auto data and I will show you what you
  can do with it. I am convinced that you will love this dataset as much as I
  do after having read this paper.
\end{abstract}

\tableofcontents

\section{Introduction}

What we want to do in the introductory section is to open the data and have a
look at what is inside of it. Since the auto data is shipped with Stata, we can
use the \stcmd{sysuse} command to open it (see \dref{sysuse}). Furthermore, the
\stcmd{describe} command will list the variables and display some other
information (see \dref{describe}). So let's start:

\begin{stlog}
\input{log/1.log.tex}
\end{stlog}

Wow! 74 observations! And what a wealth of variables! Make, price, miles per
gallon, and many more. I am very motivated to learn more about this amazing
data set.
```

```
\section{Descriptives}
```

Let's now look at some descriptive statistics. Maybe also let's do a graph.

```
\begin{stlog}
\input{log/2.log.tex}
\end{stlog}
```

```
\begin{figure}
  \centering
  \includegraphics[width=0.8\textwidth]{log/2.pdf}
  \caption{What a crazy relation between price and weight}
  \label{fig1}
\end{figure}
```

In figure~\ref{fig1} we see that for some unknown reason expensive cars seem to be heavier.

Actually, I really only want to print a graph without printing the code that produced the code. Hm, how can we do that? Maybe the `\stcmd{nolog}` option will do.

```
\begin{figure}
  \centering
  \includegraphics[width=0.8\textwidth]{log/3.pdf}
  \caption{Another crazy relation}
  \label{fig2}
\end{figure}
```

In figure~\ref{fig2} we see that price is also related to miles per gallon. How interesting!

```
\section{Varieties of logs}
```

The default is to include a log of the commands and their output:

```
\begin{stlog}
\input{log/4.log.tex}
\end{stlog}
```

```
\noindent
```

Use the `\stcmd{cmdlog}` option to only include a copy of the commands without output:

```
\begin{stlog}
\input{log/5.log.tex}
\end{stlog}
```

```
\noindent
```

Conversely, use `\stcmd{cmdstrip}` to print only the output, but not the commands:

```
\begin{stlog}
\input{log/6.log.tex}
\end{stlog}
```

```
\noindent
```

Furthermore, use `\stcmd{texdoc stlog oom}` to omit output from selected commands:

```
\begin{stlog}
\input{log/7.log.tex}
\end{stlog}
```

```
\section{Regression tables}
```

Finally we get to regressions! To include a table in the document, it is a good idea to enclose the corresponding commands in a `\stcmd{texdoc stlog}` environment and specify the `\stcmd{nolog}` option to suppress the output. This allows you to use the `\stcmd{nodo}` option later on to skip computing the table.

In `table~\ref{table1}` we see that an additional pound of car costs around 3.5 dollars once we control for milage and origin.

```
\input{log/table1.tex}
```

```
\end{document}
```

— *end of file* —



### 3 The resulting PDF

The following pages display the resulting PDF after compiling the  $\text{\LaTeX}$  source file.

# The Auto Data

Ben Jann

June 11, 2016

## Abstract

I really like the auto data because it is so awesome. You can do all kinds of stuff with the auto data, like tabulating a variable or computing descriptive statistics. You can even use the auto data to estimate regression models. I am really amazed by the richness of this dataset. There is information on many different makes and models and you can learn, for example, about the gear ratio of a Dodge Diplomat (a stunning 2.47). In this article I will illustrate the auto data and I will show you what you can do with it. I am convinced that you will love this dataset as much as I do after having read this paper.

## Contents

1	Introduction	1
2	Descriptives	2
3	Varieties of logs	4
4	Regression tables	4

## 1 Introduction

What we want to do in the introductory section is to open the data and have a look at what is inside of it. Since the auto data is shipped with Stata, we can use the `sysuse` command to open it (see [D] `sysuse`). Furthermore, the `describe` command will list the variables and display some other information (see [D] `describe`). So let's start:

```
. sysuse auto
(1978 Automobile Data)
```

```
. describe
Contains data from /Applications/Stata14/ado/base/a/auto.dta
  obs:          74          1978 Automobile Data
  vars:         12          13 Apr 2014 17:45
  size:        3,182          (_dta has notes)
```

---

variable name	storage type	display format	value label	variable label
make	str18	%-18s		Make and Model
price	int	%8.0gc		Price
mpg	int	%8.0g		Mileage (mpg)
rep78	int	%8.0g		Repair Record 1978
headroom	float	%6.1f		Headroom (in.)
trunk	int	%8.0g		Trunk space (cu. ft.)
weight	int	%8.0gc		Weight (lbs.)
length	int	%8.0g		Length (in.)
turn	int	%8.0g		Turn Circle (ft.)
displacement	int	%8.0g		Displacement (cu. in.)
gear_ratio	float	%6.2f		Gear Ratio
foreign	byte	%8.0g	origin	Car type

---

```
Sorted by: foreign
```

Wow! 74 observations! And what a wealth of variables! Make, price, miles per gallon, and many more. I am very motivated to learn more about this amazing data set.

## 2 Descriptives

Let's now look at some descriptive statistics. Maybe also let's do a graph.

```
. summarize
```

Variable	Obs	Mean	Std. Dev.	Min	Max
make	0				
price	74	6165.257	2949.496	3291	15906
mpg	74	21.2973	5.785503	12	41
rep78	69	3.405797	.9899323	1	5
headroom	74	2.993243	.8459948	1.5	5
trunk	74	13.75676	4.277404	5	23
weight	74	3019.459	777.1936	1760	4840
length	74	187.9324	22.26634	142	233
turn	74	39.64865	4.399354	31	51
displacement	74	197.2973	91.83722	79	425
gear_ratio	74	3.014865	.4562871	2.19	3.89
foreign	74	.2972973	.4601885	0	1

```
. pspline price weight
(pilot goodness-of-fit chi2(16) = 32.38; p = 0.0089)
(using penalized model ...)
```

In figure 1 we see that for some unknown reason expensive cars seem to be heavier.

Actually, I really only want to print a graph without printing the code that produced the code. Hm, how can we do that? Maybe the `nolog` option will do.

In figure 2 we see that price is also related to miles per gallon. How interesting!

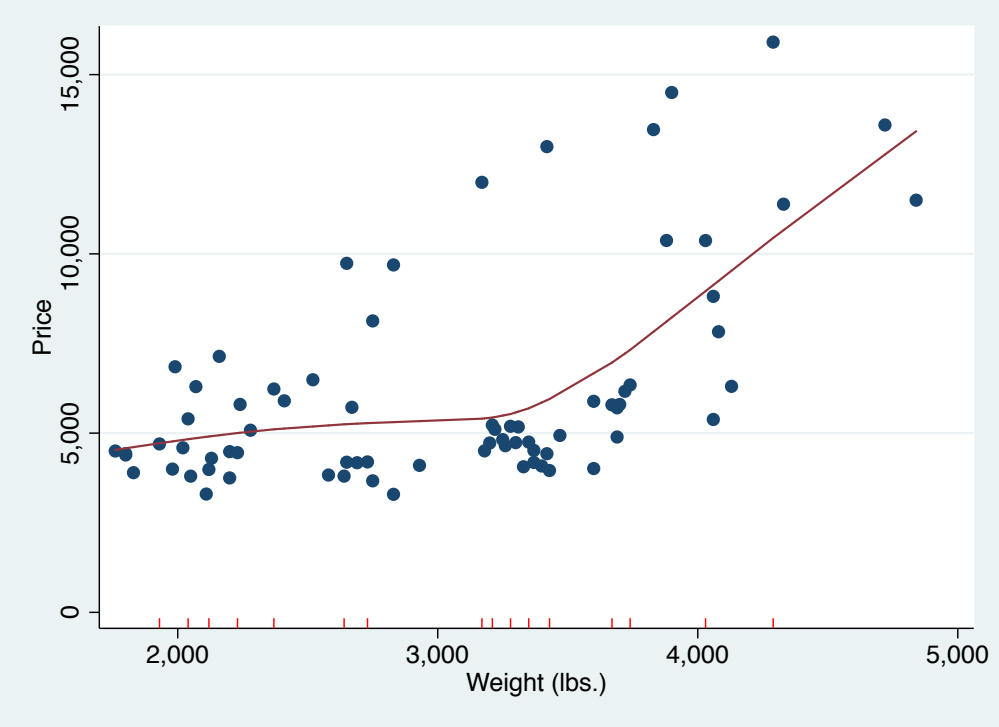


Figure 1: What a crazy relation between price and weight

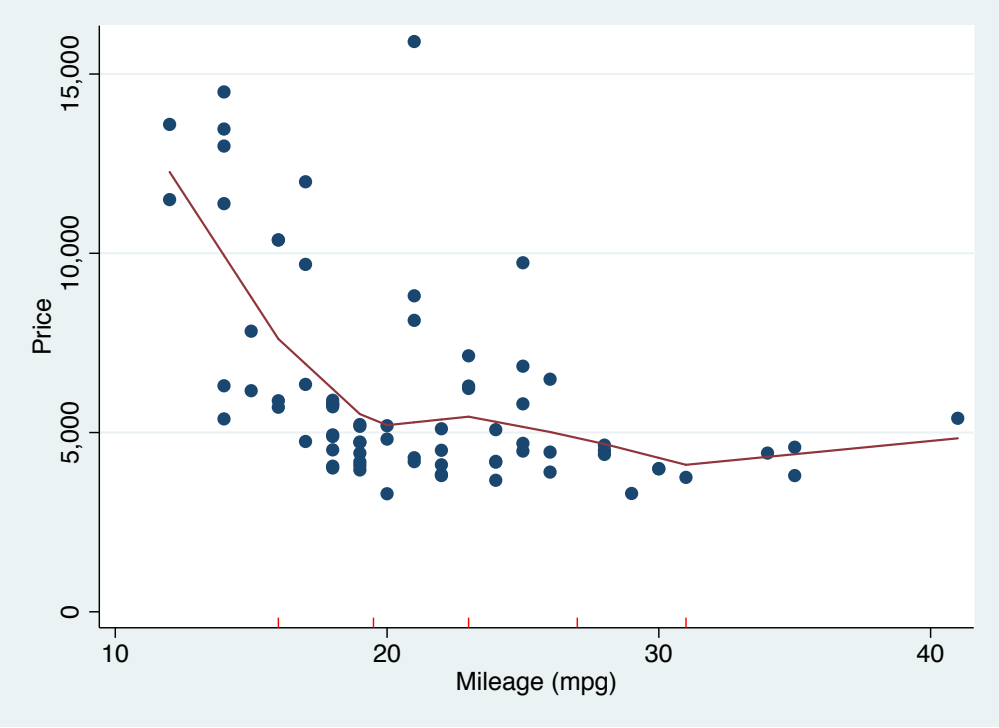


Figure 2: Another crazy relation

### 3 Varieties of logs

The default is to include a log of the commands and their output:

```
. display "2 + 2 = " 2 + 2
2 + 2 = 4
. display "sqrt(2) = " sqrt(2)
sqrt(2) = 1.4142136
```

Use the `cmdlog` option to only include a copy of the commands without output:

```
display "2 + 2 = " 2 + 2
display "sqrt(2) = " sqrt(2)
```

Conversely, use `cmdstrip` to print only the output, but not the commands:

```
2 + 2 = 4
sqrt(2) = 1.4142136
```

Furthermore, use `texdoc stlog oom` to omit output from selected commands:

```
. sysuse auto
(1978 Automobile Data)
. list make
  (output omitted)
. display make[1]
AMC Concord
```

### 4 Regression tables

Finally we get to regressions! To include a table in the document, it is a good idea to enclose the corresponding commands in a `texdoc stlog` environment and specify the `nolog` option to suppress the output. This allows you to use the `nodot` option later on to skip computing the table.

In table 1 we see that an additional pound of car costs around 3.5 dollars once we control for milage and origin.

Table 1: Some regression table

	(1)	(2)	(3)
Weight (lbs.)	2.044*** (0.377)	1.747** (0.641)	3.465*** (0.631)
Mileage (mpg)		-49.51 (86.16)	21.85 (74.22)
Car type			3673.1*** (684.0)
Constant	-6.707 (1174.4)	1946.1 (3597.0)	-5853.7 (3377.0)
Observations	74	74	74

Standard errors in parentheses

\*  $p < 0.05$ , \*\*  $p < 0.01$ , \*\*\*  $p < 0.001$