
Stata 12/13 Tutorial 4

TOPIC: Conditional and Marginal Effects of Continuous Explanatory Variables in Linear Regression Models

DATA: `auto1.dta` (a *Stata*-format dataset first created in *Stata 12/13 Tutorial 1*)

TASKS: *Stata 12/13 Tutorial 4* deals with computing the **conditional and marginal effects of individual continuous explanatory variables** on the dependent variable in linear regression models. It also demonstrates how to create a line graph of the marginal effect of a continuous explanatory variable and how to save the line graph in several different file formats for future use.

- The **conditional effect** of an explanatory variable on the dependent variable is the *ceteris paribus* relationship between the values of that explanatory variable and the conditional mean of the dependent variable for fixed values of all other explanatory variables in the regression function.
- The **marginal effect** of an explanatory variable on the dependent variable is the *ceteris paribus* (or partial) effect of a unit increase in that explanatory variable on the conditional mean of the dependent variable.
- The **Stata commands** that constitute the primary subject of this tutorial are:

| | |
|--------------------|---|
| regress | Used to perform OLS estimation of multiple linear regression models. |
| scalar | Used to save as scalars elements of the coefficient vector and the coefficient covariance matrix after OLS estimation. |
| lincom | Used after estimation to compute linear combinations of coefficient estimates and associated statistics. |
| margins | Used after OLS estimation to compute estimates of the marginal effects of continuous explanatory variables . |
| marginsplot | Used to draw line graphs of the marginal effects of continuous explanatory variables in linear regression models. |
| graph | Used to graph or plot the conditional and marginal effects of continuous explanatory variables in linear regression models |
| graph save | Saves the graph currently displayed in the Graph window to a disk file in the current <i>Stata</i> working directory. |

- graph export** Exports the graph currently displayed in the Graph window to a file in the current *Stata* working directory.
- graph use** Loads a graph on disk into memory and displays it in the Graph window.

NOTE: *Stata* commands are *case sensitive*. All *Stata* command names must be typed in the Command window in **lower case letters**.

HELP: *Stata* has an extensive on-line **Help** facility that provides fairly detailed information (including examples) on all *Stata* commands. Students should become familiar with the *Stata* on-line **Help** system. In the course of doing this tutorial, take the time to browse the **Help** information on some of the above *Stata* commands. To access the on-line **Help** for any *Stata* command:

- choose (click on) **Help** from the *Stata* main menu bar
- click on **Stata Command** in the **Help** drop down menu
- type the full name of the *Stata* command in the *Stata* command dialog box and click **OK**

□ **Preparing for Your *Stata* Session**

Before beginning your *Stata* session, use Windows Explorer to copy the *Stata*-format dataset **auto1.dta** to the *Stata working directory* on the C:-drive or D:-drive of the computer at which you are working. If you did not save the *Stata*-format dataset **auto1.dta** you created during *Stata 12/13 Tutorial 1* and bring it with you on a portable storage device such as a flash memory stick, you will have to repeat the relevant section of *Stata 12/13 Tutorial 1* before proceeding with this tutorial.

- **On the computers in Dunning 350**, the default *Stata* working directory is usually C:\data.

□ **Start Your *Stata* Session**

To start your *Stata* session, double-click on the ***Stata 12 or Stata 13 icon*** on the Windows desktop.

After you double-click the ***Stata 12 or Stata 13 icon***, you will see the familiar screen of five *Stata* windows.

□ **Record Your *Stata* Session -- log using**

To record your *Stata* session, including all the *Stata* commands you enter and the results (output) produced by these commands, make a text-format **.log** file named **452tutorial4.log**. To open (begin) the log file **452tutorial4.log**, enter in the Command window:

```
log using 452tutorial4.log
```

This command opens a text-format (ASCII) file called **452tutorial4.log** in the current *Stata* working directory.

Note: It is important to include the **.log** file extension when opening a log file; if you do not, your log file will be in smcl format, a format that only *Stata* can read.

□ Record Only Your *Stata* Commands -- cmdlog using

To record only the *Stata* commands you type during your *Stata* session, you can use the *Stata* **cmdlog using** command. To start (open) the command log file **452tutorial4.txt**, enter in the Command window:

```
cmdlog using 452tutorial4
```

This command opens a plain text-format (ASCII) file called **452tutorial4.txt** in the current *Stata* working directory. All commands you enter during your *Stata* session are recorded in this file. At the end of this tutorial, you will close your command log file in the same way you close your regular log file.

□ Loading a *Stata*-Format Dataset into *Stata* – use

To check that the *Stata*-format dataset ‘auto1.dta’ is in the current *Stata* working directory of the computer at which you are working, type in the Command window:

```
dir auto1.*
```

You should see in the *Stata* Results window the filename ‘auto1.dta’.

To load, or read, into memory the *Stata*-format dataset auto1.dta, type in the Command window:

```
use auto1
```

This command loads into memory the *Stata*-format dataset **auto1.dta**.

To summarize the contents of the current dataset, use the **describe** command. Recall from *Stata 12/13 Tutorial 1* that the **describe** command displays a summary of the contents of the current dataset in memory, which in this case is the *Stata*-format data file **auto1.dta**. Type in the Command window:

```
describe
```

To compute summary statistics for the variables in the current dataset, use the **summarize** command. Recall from *Stata 12/13 Tutorial 1* that the **summarize** command computes descriptive summary statistics for all *numeric* variables in the current dataset in memory. Type in the Command window:

```
summarize
```

□ **The *Conditional* Effects of Individual Explanatory Variables: Definition**

Nature: The ***conditional effect*** of an explanatory variable on the dependent variable is the *ceteris paribus* relationship between the values of that explanatory variable and the conditional mean of the dependent variable for fixed values of all other explanatory variables in the regression function.

Consider the following linear regression model for car prices (which is Model 2 of *Stata 12/13 Tutorial 2*):

$$\text{price}_i = \beta_0 + \beta_1 \text{wgt}_i + \beta_2 \text{mpg}_i + \beta_3 \text{wgt}_i^2 + \beta_4 \text{mpg}_i^2 + \beta_5 \text{wgt}_i \text{mpg}_i + u_i . \quad (1)$$

The population regression function for regression equation (1) is written in general as:

$$E(\text{price}_i | \text{wgt}_i, \text{mpg}_i) = \beta_0 + \beta_1 \text{wgt}_i + \beta_2 \text{mpg}_i + \beta_3 \text{wgt}_i^2 + \beta_4 \text{mpg}_i^2 + \beta_5 \text{wgt}_i \text{mpg}_i \dots (2)$$

There are two explanatory variables in model (1) – wgt_i and mpg_i . The conditional effect of each is obtained by setting the *other* explanatory variable equal to some specified (or selected) value in the population regression equation – in other words, by holding constant the other explanatory variable.

Conditional effect of wgt_i on price_i

The ***conditional effect of wgt_i on price_i*** is obtained by setting the other explanatory variable in regression function (2) – namely mpg_i – equal to some specified value such as $\text{mpg}_i = \text{mpg}_0$. The conditional effect of wgt_i on price_i for $\text{mpg}_i = \text{mpg}_0$ is thus:

$$\begin{aligned}
 E(\text{price}_i | \text{wgt}_i, \text{mpg}_0) &= \beta_0 + \beta_1 \text{wgt}_i + \beta_2 \text{mpg}_0 + \beta_3 \text{wgt}_i^2 + \beta_4 \text{mpg}_0^2 + \beta_5 \text{wgt}_i \text{mpg}_0 \\
 &= \beta_0 + \beta_2 \text{mpg}_0 + \beta_4 \text{mpg}_0^2 + \beta_1 \text{wgt}_i + \beta_5 \text{wgt}_i \text{mpg}_0 + \beta_3 \text{wgt}_i^2 \\
 &= (\beta_0 + \beta_2 \text{mpg}_0 + \beta_4 \text{mpg}_0^2) + (\beta_1 + \beta_5 \text{mpg}_0) \text{wgt}_i + \beta_3 \text{wgt}_i^2 \\
 &\quad \dots (3)
 \end{aligned}$$

As the above equation indicates, the conditional effect of wgt_i on price_i is a quadratic function of wgt_i once the value of mpg_i is fixed at the specified value mpg_0 .

Interpretation:

The conditional effect of wgt_i on price_i represents the *ceteris paribus*, or "other things equal," relationship between mean price and the explanatory variable wgt_i .

In other words, it isolates the relationship between wgt_i and price_i from the relationships of the other explanatory variable(s) to price_i .

But it controls for, or adjusts for, the effect(s) on price_i of the other explanatory variable(s) – in this case the explanatory variable mpg_i . It is therefore often referred to as the ***regression-adjusted relationship of wgt_i to price_i*** .

Estimation:

The conditional effect of wgt_i on price_i for $\text{mpg}_i = \text{mpg}_0$ is estimated by replacing the population regression coefficients β_j ($j = 0, 1, \dots, 5$) in expression (3) for

$E(\text{price}_i | \text{wgt}_i, \text{mpg}_0)$ with their OLS estimates $\hat{\beta}_j$ ($j = 0, 1, \dots, 5$):

$$\begin{aligned}
 \hat{E}(\text{price}_i | \text{wgt}_i, \text{mpg}_0) &= (\hat{\beta}_0 + \hat{\beta}_2 \text{mpg}_0 + \hat{\beta}_4 \text{mpg}_0^2) + (\hat{\beta}_1 + \hat{\beta}_5 \text{mpg}_0) \text{wgt}_i + \hat{\beta}_3 \text{wgt}_i^2 \\
 &\quad \dots (4)
 \end{aligned}$$

Conditional effect of mpg_i on price_i

The ***conditional effect of mpg_i on price_i*** is obtained by setting the other explanatory variable in regression function (2) – namely wgt_i – equal to some specified value such as $\text{wgt}_i = \text{wgt}_0$. The conditional effect of mpg_i on price_i for $\text{wgt}_i = \text{wgt}_0$ is thus:

$$\begin{aligned}
E(\text{price}_i | \text{wgt}_0, \text{mpg}_i) &= \beta_0 + \beta_1 \text{wgt}_0 + \beta_2 \text{mpg}_i + \beta_3 \text{wgt}_0^2 + \beta_4 \text{mpg}_i^2 + \beta_5 \text{wgt}_0 \text{mpg}_i \\
&= \beta_0 + \beta_1 \text{wgt}_0 + \beta_3 \text{wgt}_0^2 + \beta_2 \text{mpg}_i + \beta_5 \text{wgt}_0 \text{mpg}_i + \beta_4 \text{mpg}_i^2 \\
&= (\beta_0 + \beta_1 \text{wgt}_0 + \beta_3 \text{wgt}_0^2) + (\beta_2 + \beta_5 \text{wgt}_0) \text{mpg}_i + \beta_4 \text{mpg}_i^2 \\
&\dots (5)
\end{aligned}$$

The above equation indicates that the conditional effect of mpg_i on price_i is a quadratic function of mpg_i once the value of wgt_i is fixed at the specified value wgt_0 .

Estimation:

The conditional effect of mpg_i on price_i for $\text{wgt}_i = \text{wgt}_0$ is estimated by replacing the population regression coefficients β_j ($j = 0, 1, \dots, 5$) in expression (5) for $E(\text{price}_i | \text{wgt}_0, \text{mpg}_i)$ with their OLS estimates $\hat{\beta}_j$ ($j = 0, 1, \dots, 5$):

$$\begin{aligned}
\hat{E}(\text{price}_i | \text{wgt}_0, \text{mpg}_i) &= (\hat{\beta}_0 + \hat{\beta}_1 \text{wgt}_0 + \hat{\beta}_3 \text{wgt}_0^2) + (\hat{\beta}_2 + \hat{\beta}_5 \text{wgt}_0) \text{mpg}_i + \hat{\beta}_4 \text{mpg}_i^2 \\
&\dots (6)
\end{aligned}$$

□ *The Marginal Effects of Individual Explanatory Variables: Definition*

The marginal effect of an individual explanatory variable is a function that gives the change in the mean value of the dependent variable in response to a one-unit increase in the value of the explanatory variable, holding constant the effects of all other explanatory variables in the model.

Consider again Model 2 for car prices, which is given by the population regression equation (1):

$$\text{price}_i = \beta_0 + \beta_1 \text{wgt}_i + \beta_2 \text{mpg}_i + \beta_3 \text{wgt}_i^2 + \beta_4 \text{mpg}_i^2 + \beta_5 \text{wgt}_i \text{mpg}_i + u_i. \quad (1)$$

Re-write the population regression function (or PRF) for Model 2 as:

$$\begin{aligned}
E(\text{price}_i | \text{wgt}_i, \text{mpg}_i) &= \beta_0 + \beta_1 \text{wgt}_i + \beta_2 \text{mpg}_i + \beta_3 \text{wgt}_i^2 + \beta_4 \text{mpg}_i^2 + \beta_5 \text{wgt}_i \text{mpg}_i \\
&\dots (2)
\end{aligned}$$

Marginal effect of wgt_i on $price_i$

- The **marginal effect of wgt_i on $price_i$** is obtained by partially differentiating regression equation (1), or equivalently the population regression function (2), with respect to wgt_i :

$$\frac{\partial price_i}{\partial wgt_i} = \frac{\partial E(price_i | wgt_i, mpg_i)}{\partial wgt_i} = \beta_1 + 2\beta_3 wgt_i + \beta_5 mpg_i \quad (7)$$

- Estimates of the marginal effect of wgt_i on $price_i$** are obtained by substituting in (7) the OLS coefficient estimates $\hat{\beta}_j$ for the population regression coefficients β_j :

$$\text{estimate of } \frac{\partial price_i}{\partial wgt_i} = \frac{\partial \hat{E}(price_i | wgt_i, mpg_i)}{\partial wgt_i} = \hat{\beta}_1 + 2\hat{\beta}_3 wgt_i + \hat{\beta}_5 mpg_i \quad (8)$$

The estimated marginal effect of wgt_i on $price_i$ can be evaluated at any chosen values of wgt_i and mpg_i .

Marginal effect of mpg_i on $price_i$

- The **marginal effect of mpg_i on $price_i$** is obtained by partially differentiating regression equation (1), or equivalently the population regression function (2), with respect to mpg_i :

$$\frac{\partial price_i}{\partial mpg_i} = \frac{\partial E(price_i | wgt_i, mpg_i)}{\partial mpg_i} = \beta_2 + 2\beta_4 mpg_i + \beta_5 wgt_i \quad (9)$$

- Estimates of the marginal effect of mpg_i on $price_i$** are obtained by substituting in (9) the OLS coefficient estimates $\hat{\beta}_j$ for the population regression coefficients β_j :

$$\text{estimate of } \frac{\partial price_i}{\partial mpg_i} = \frac{\partial \hat{E}(price_i | wgt_i, mpg_i)}{\partial mpg_i} = \hat{\beta}_2 + 2\hat{\beta}_4 mpg_i + \hat{\beta}_5 wgt_i \quad (10)$$

The estimated marginal effect of mpg_i on price_i can be evaluated at any chosen values of wgt_i and mpg_i .

□ Computing "Typical" Values of Variables as Scalars

Computing both conditional and marginal effects requires the selection of certain fixed values of the explanatory variables *wgt* and *mpg*. In the foregoing expressions for the conditional and marginal effects of *wgt* and *mpg*, these selected fixed values are denoted as wgt_0 and mpg_0 . But what specific values might one select for wgt_0 and mpg_0 ?

Common choices for the values of wgt_0 and mpg_0 are sample means and sample percentiles such as the median (50-th percentile), the first quartile (25-th percentile), and the third quartile (75-th percentile). In this section, we use the **summarize** and **scalar** commands to create these values as scalars.

- To create as scalars the sample mean and the 25-th, 50-th and 75-th sample percentiles of the variable *wgt_i*, enter the following commands:

```
summarize wgt, detail
return list
scalar wgtbar = r(mean)
scalar wgt50 = r(p50)
scalar wgt25 = r(p25)
scalar wgt75 = r(p75)
scalar list wgtbar wgt25 wgt50 wgt75
```

- To create as scalars the sample mean and the 25-th, 50-th and 75-th sample percentiles of the variable *mpg_i*, enter the following commands:

```
summarize mpg, detail
return list
scalar mpgbar = r(mean)
scalar mpg50 = r(p50)
scalar mpg25 = r(p25)
scalar mpg75 = r(p75)
scalar list mpgbar mpg25 mpg50 mpg75
```

□ Computing the *Conditional* Effects of Individual Explanatory Variables

- First, if your current dataset does not already include them, you will have to generate the regressors wgt_i^2 , mpg_i^2 and $wgt_i mpg_i$ that enter Model 2 but that are not included in the current dataset. In that case, enter the commands:

```
generate wgtsq = wgt^2
generate mpgsq = mpg^2
generate wgtmpg = wgt*mpg
summarize price wgt mpg wgtsq mpgsq wgtmpg
```

- Now estimate regression equation (1) by OLS. Enter the command

```
regress price wgt mpg wgtsq mpgsq wgtmpg
```

- Use the **scalar** command to save the OLS coefficient estimates, and the **scalar list** command to display the saved scalars. Enter the commands:

```
scalar b1 = _b[wgt]
scalar b2 = _b[mpg]
scalar b3 = _b[wgtsq]
scalar b4 = _b[mpgsq]
scalar b5 = _b[wgtmpg]
scalar b0 = _b[_cons]
scalar list b1 b2 b3 b4 b5 b0
```

Conditional Effect of wgt_i

The estimated conditional effect of wgt_i on $price_i$ for some given value mpg_0 of mpg_i is the following quadratic function of wgt_i :

$$\hat{E}(price_i | wgt_i, mpg_0) = (\hat{\beta}_0 + \hat{\beta}_2 mpg_0 + \hat{\beta}_4 mpg_0^2) + (\hat{\beta}_1 + \hat{\beta}_5 mpg_0) wgt_i + \hat{\beta}_3 wgt_i^2 \dots (4)$$

- Create as a variable the conditional effect of wgt_i on $price_i$ for $mpg_0 =$ the sample mean of mpg_i . Enter *on one line* the **generate** command:

```
generate cewgtbar = b0 + b2*mpgbar + b4*(mpgbar^2) + (b1 +
b5*mpgbar)*wgt + b3*wgtsq
```

- Create as a variable the conditional effect of wgt_i on $price_i$ for $mpg_0 =$ the 25-th sample percentile of mpg_i . Enter *on one line* the **generate** command:

```
generate cewgt25 = b0 + b2*mpg25 + b4*(mpg25^2) + (b1 +
b5*mpg25)*wgt + b3*wgtsq
```

- Create as a variable the conditional effect of wgt_i on $price_i$ for $mpg_0 =$ the 50-th sample percentile of mpg_i . Enter *on one line* the **generate** command:

```
generate cewgt50 = b0 + b2*mpg50 + b4*(mpg50^2) + (b1 +
b5*mpg50)*wgt + b3*wgtsq
```

- Create as a variable the conditional effect of wgt_i on $price_i$ for $mpg_0 =$ the 75-th sample percentile of mpg_i . Enter *on one line* the **generate** command:

```
generate cewgt75 = b0 + b2*mpg75 + b4*(mpg75^2) + (b1 +
b5*mpg75)*wgt + b3*wgtsq
```

- List and compute summary statistics for the values of the four conditional effects of wgt_i you have just created, after first sorting the sample observations in ascending order of the sample values of wgt_i . Enter the commands:

```
sort wgt
list wgt mpg cewgtbar cewgt25 cewgt50 cewgt75
summarize wgt mpg cewgtbar cewgt25 cewgt50 cewgt75
```

Conditional Effect of mpg_i

The estimated conditional effect of mpg_i on $price_i$ for some given value wgt_0 of wgt_i is the following quadratic function of mpg_i :

$$\hat{E}(price_i | wgt_0, mpg_i) = (\hat{\beta}_0 + \hat{\beta}_1 wgt_0 + \hat{\beta}_3 wgt_0^2) + (\hat{\beta}_2 + \hat{\beta}_5 wgt_0) mpg_i + \hat{\beta}_4 mpg_i^2 \dots (6)$$

- Create as a variable the conditional effect of mpg_i on $price_i$ for $wgt_0 =$ the sample mean of wgt_i . Enter *on one line* the **generate** command:

```
generate cempgbar = b0 + b1*wgtbar + b3*(wgtbar^2) + (b2 +
b5*wgtbar)*mpg + b4*mpgsq
```

- Create as a variable the conditional effect of mpg_i on price_i for $\text{wgt}_0 =$ the 25-th sample percentile of wgt_i . Enter *on one line* the **generate** command:

```
generate cempg25 = b0 + b1*wgt25 + b3*(wgt25^2) + (b2 +
b5*wgt25)*mpg + b4*mpgsq
```

- Create as a variable the conditional effect of mpg_i on price_i for $\text{wgt}_0 =$ the 50-th sample percentile of wgt_i . Enter *on one line* the **generate** command:

```
generate cempg50 = b0 + b1*wgt50 + b3*(wgt50^2) + (b2 +
b5*wgt50)*mpg + b4*mpgsq
```

- Create as a variable the conditional effect of mpg_i on price_i for $\text{wgt}_0 =$ the 75-th sample percentile of wgt_i . Enter *on one line* the **generate** command:

```
generate cempg75 = b0 + b1*wgt75 + b3*(wgt75^2) + (b2 +
b5*wgt75)*mpg + b4*mpgsq
```

- List and compute summary statistics for the four conditional effects of mpg_i you have just created, after first sorting the sample observations in ascending order of the sample values of mpg_i . Enter the commands:

```
sort mpg
list mpg wgt cempgbar cempg25 cempg50 cempg75
summarize mpg wgt cempgbar cempg25 cempg50 cempg75
```

□ Graphing the *Conditional Effects of Individual Explanatory Variables*

The estimated conditional effect of wgt_i on price_i for any given value mpg_0 of mpg_i is a quadratic function of wgt_i . In order to better understand the nature of these conditional effect functions, it is often instructive to draw line graphs of the estimated conditional effect of wgt_i on price_i ; these plot the estimated values of the conditional effect of wgt_i against the sample values of wgt_i sorted in ascending order.

Graphing the Conditional Effect of wgt_i

- First, to ensure that the sample observations are sorted in ascending order of the sample values of wgt_i , enter the following **sort** command:

```
sort wgt
```

- Use the **graph twoway line** command to draw a twoway line graph of the estimated conditional effect of wgt_i on $price_i$ as a function of wgt_i while holding the value of the other continuous variable mpg_i constant at its sample mean value. Enter the commands:

```
graph twoway line cewgtbar wgt
line cewgtbar wgt
```

Note that the command name **graph twoway line** can be abbreviated to simply **line**.

- Use the **ytitle()** and **xtitle()** options to add your own titles to the vertical y-axis and horizontal x-axis of the above line graph. Enter *on one line* the command:

```
line cewgtbar wgt, ytitle("Conditional Mean Car Price ($)")
xtitle("Car Weight (pounds)")
```

- Use the **title()** and **subtitle()** options to add titles to the above figure. Enter *on one line* the command:

```
line cewgtbar wgt, ytitle("Conditional Mean Car Price ($)")
xtitle("Car Weight (pounds)") title("Conditional Effect of
Weight on Car Price") subtitle("for Sample Mean Value of Fuel
Efficiency (mpg)")
```

- Use the **graph twoway line** command to draw in the same diagram line graphs of the estimated conditional effects of wgt_i on $price_i$ as functions of wgt_i for the 25-th, 50-th and 75-th sample percentile values of mpg_i . Enter *on one line* the command:

```
graph twoway line cewgt25 cewgt50 cewgt75 wgt,
ytitle("Conditional Mean Car Price ($)") xtitle("Car Weight
(pounds)") title("Conditional Effect of Weight on Car Price")
subtitle("for Selected Percentile Values of Fuel Efficiency
(mpg)")
```

- Modify the above diagram by using the **ylabel()** and **xlabel()** options to change the default labeling and ticking of values on the left vertical and bottom horizontal axes of the diagram. Enter *on one line* the command:

```

line cewgt25 cewgt50 cewgt75 wgt, ytitle("Conditional Mean Car
Price ($)") xtitle("Car Weight (pounds)") title("Conditional
Effect of Weight on Car Price") subtitle("for Selected
Percentile Values of Fuel Efficiency (mpg)")
ylabel(0(5000)20000) xlabel(2000(500)5000)

```

Note the effects of the added options on the appearance of the figure, in particular on the value labeling of the two axes. Note too that the command name has been abbreviated from **graph twoway line** to simply **line**.

- Finally, use the **legend ()** option to change the default legend of the above diagram. Enter *on one line* the command:

```

line cewgt25 cewgt50 cewgt75 wgt, ytitle("Conditional Mean Car
Price ($)") xtitle("Car Weight (pounds)") title("Conditional
Effect of Weight on Car Price") subtitle("for Selected
Percentile Values of Fuel Efficiency (mpg)")
ylabel(0(5000)20000) xlabel(2000(500)5000) legend(label(1 "mpg
= 25-th percentile") label(2 "mpg = 50-th percentile") label(3
"mpg = 75-th percentile"))

```

Graphing the Conditional Effect of mpg_i

- First, sort the sample observations in ascending order of the sample values of mpg_i. Enter the following **sort** command:

```
sort mpg
```

- Use the **graph twoway line** command to draw a line graph of the estimated conditional effect of mpg_i on price_i as a function of mpg_i, conditional on the sample mean value of wgt_i. Enter *on one line* the command:

```
graph twoway line cempgbar mpg, ytitle("Conditional Mean Car
Price ($)") xtitle("Fuel Efficiency (mpg = miles per gallon)")

```

- Use the **graph twoway line** command to draw in the same diagram line graphs of the estimated conditional effect of mpg_i on price_i as a function of mpg_i for the 25-th, 50-th and 75-th sample percentile values of wgt_i. Use appropriate **title()**, **subtitle()**, **ylabel()**, and **legend()** options to make your figure more readable. Enter *on one line* the command:

```

line cempg25 cempg50 cempg75 mpg, ytitle("Conditional Mean Car
Price ($)") xtitle("Fuel Efficiency (mpg = miles per gallon)")
title("Conditional Effect of Fuel Efficiency on Car Price")
subtitle("for Selected Percentile Values of Car Weight (wgt)")
ylabel(0(5000)20000) legend(label(1 "wgt = 25-th percentile")
label(2 "wgt = 50-th percentile") label(3 "wgt = 75-th
percentile"))

```

□ Computing the *Marginal* Effects of Individual Explanatory Variables

The **lincom** command is a convenient tool not only for evaluating marginal effects for selected values of the explanatory variables, but also for performing two-tail tests of the null hypothesis that the marginal effect is zero for the specified values of the explanatory variables.

- First, re-estimate the model by OLS by entering either one of the following commands:

```

regress
regress price wgt mpg wgtsq mpgsq wgtmpg

```

Marginal Effects of wgt_i

The estimated marginal effect of wgt_i on $price_i$ for some given values wgt_0 of wgt_i and mpg_0 of mpg_i is calculated by the following linear function of wgt_i and mpg_i :

$$\text{estimate of } \frac{\partial price_i}{\partial wgt_i} = \frac{\partial \hat{E}(price_i | wgt_i, mpg_i)}{\partial wgt_i} = \hat{\beta}_1 + 2\hat{\beta}_3 wgt_0 + \hat{\beta}_5 mpg_0$$

- Use the following **lincom** command to evaluate the estimated marginal effect of wgt_i on $price_i$ for the sample mean values of wgt_i and mpg_i :

```

lincom _b[wgt] + 2*_b[wgtsq]*wgtbar + _b[wgtmpg]*mpgbar

```

Note that the **lincom** command also computes (1) the estimated standard error of the estimated marginal effect of wgt_i on $price_i$, and (2) the t-ratio for a two-tail test of the null hypothesis that the marginal effect of wgt_i on $price_i$ equals zero for the average car in the sample. Would you retain or reject the null hypothesis at the 5 percent significance level, at the 1 percent significance level?

- Use the following **lincom** command to evaluate the estimated marginal effect of wgt_i on $price_i$ for the 25-th sample percentiles of wgt_i and mpg_i . Enter the commands:

```
lincom _b[wgt] + 2*_b[wgtsq]*wgt25 + _b[wgtmpg]*mpg25
return list
```

Would you retain or reject the null hypothesis that the marginal effect of wgt_i on $price_i$ equals zero for a car whose weight and fuel efficiency equal the 25-th sample percentile values?

- Use the following **lincom** command to evaluate the estimated marginal effect of wgt_i on $price_i$ for the 50-th sample percentiles of wgt_i and mpg_i :

```
lincom _b[wgt] + 2*_b[wgtsq]*wgt50 + _b[wgtmpg]*mpg50
```

Would you retain or reject the null hypothesis that the marginal effect of wgt_i on $price_i$ equals zero for the median car in the sample?

- Use the following **lincom** command to evaluate the estimated marginal effect of wgt_i on $price_i$ for a car in the 25-th sample percentile of wgt_i and the 75-th sample percentile in mpg_i :

```
lincom _b[wgt] + 2*_b[wgtsq]*wgt25 + _b[wgtmpg]*mpg75
```

Would you retain or reject the null hypothesis that the marginal effect of wgt_i on $price_i$ equals zero for a car in the 25-th sample percentile of wgt_i and the 75-th sample percentile in mpg_i ?

- Finally, you can evaluate the marginal effect of wgt_i on $price_i$ for all 74 cars in the sample. Enter the following commands:

```
generate mewgt = b1 + 2*b3*wgt + b5*mpg
summarize mewgt
```

- Use the **graph twoway scatter** command to plot the estimated marginal effect of wgt_i on $price_i$ against wgt_i for all cars in the sample, after first sorting the sample observations in ascending order of the observed values of wgt_i . Enter the commands:

```

sort wgt
list wgt mpg mewgt
graph twoway scatter mewgt wgt

```

- Add appropriate `ylabel()`, `xlabel()`, `title()`, `subtitle()`, `ytitle()` and `xtitle()` options to the above **graph twoway scatter** command to make the figure more readable. Enter *on one line* the command:

```

graph twoway scatter mewgt wgt, ylabel(-10(5)10)
xlabel(2000(500)5000) title("Marginal Effect of Weight on Car
Price") subtitle("As Function of WGT, All Cars in Sample")
ytitle("Marginal effect of wgt on price ($ per pound)")
xtitle("Car weight in pounds (wgt)")

```

- Use the **graph twoway scatter** command to plot the estimated marginal effect of wgt_i on $price_i$ against mpg_i for all cars in the sample, after first sorting the sample observations in ascending order of the observed values of mpg_i . Enter the commands:

```

sort mpg
list mpg wgt mewgt
graph twoway scatter mewgt mpg

```

- Add appropriate `ylabel()`, `xlabel()`, `ytitle()`, `xtitle()`, `title()`, and `subtitle()` options to the above **graph twoway scatter** command to make the figure more readable. Enter *on one line* the command:

```

graph twoway scatter mewgt mpg, ylabel(-10(5)10)
xlabel(10(5)40) ytitle("Marginal effect of wgt on price ($ per
pound)") xtitle("Miles per gallon (mpg)") title("Estimated
Marginal Effect of Wgt on Price") subtitle("As Function of
MPG, All Cars in Sample")

```

Marginal Effects of mpg_i

The estimated marginal effect of mpg_i on $price_i$ for some given values wgt_0 of wgt_i and mpg_0 of mpg_i is calculated by the following linear function of wgt_i and mpg_i :

$$\text{estimate of } \frac{\partial price_i}{\partial mpg_i} = \frac{\partial \hat{E}(price_i | wgt_i, mpg_i)}{\partial mpg_i} = \hat{\beta}_2 + 2\hat{\beta}_4 mpg_0 + \hat{\beta}_5 wgt_0$$

- Use the following **lincom** command to evaluate the estimated marginal effect of mpg_i on price_i for the sample mean values of wt_i and mpg_i :

```
lincom _b[mpg] + 2*_b[mpgsq]*mpgbar + _b[wgtmpg]*wtbar
```

Would you retain or reject the null hypothesis that the marginal effect of mpg_i on price_i equals zero for the "average" car in the sample?

- Use the following **lincom** command to evaluate the estimated marginal effect of mpg_i on price_i for the 25-th sample percentiles of wt_i and mpg_i :

```
lincom _b[mpg] + 2*_b[mpgsq]*mpg25 + _b[wgtmpg]*wt25
```

Would you retain or reject the null hypothesis that the marginal effect of mpg_i on price_i equals zero for a car whose weight and fuel efficiency equal the 25-th sample percentile values?

- Use the following **lincom** command to evaluate the estimated marginal effect of mpg_i on price_i for the 50-th sample percentiles of wt_i and mpg_i :

```
lincom _b[mpg] + 2*_b[mpgsq]*mpg50 + _b[wgtmpg]*wt50
```

Would you retain or reject the null hypothesis that the marginal effect of mpg_i on price_i equals zero for the median car in the sample?

- Use the following **lincom** command to evaluate the estimated marginal effect of mpg_i on price_i for a car in the 25-th sample percentile of wt_i and the 75-th sample percentile of mpg_i :

```
lincom _b[mpg] + 2*_b[mpgsq]*mpg75 + _b[wgtmpg]*wt25
```

Would you retain or reject the null hypothesis that the marginal effect of mpg_i on price_i equals zero for a car in the 25-th sample percentile of wt_i and the 75-th sample percentile of mpg_i ?

- Finally, you can evaluate the marginal effect of mpg_i on price_i for all 74 cars in the sample. Enter the following commands:

```
generate mempg = b2 + 2*b4*mpg + b5*wt
summarize mempg
```

- Use the **graph twoway scatter** command to plot the estimated marginal effect of mpg_i on price_i against mpg_i for all cars in the sample, after first sorting the sample observations in ascending order of the observed values of mpg_i . Enter the commands:

```
sort mpg
list mpg wgt mempg
graph twoway scatter mempg mpg
```

- Add appropriate **ytitle()**, **xtitle()**, **title()**, and **subtitle()** options to the above **graph twoway scatter** command to make the figure more readable. Enter *on one line* the command:

```
graph twoway scatter mempg mpg, title("Marginal Effect of MPG
on Car Price") subtitle("As Function of MPG, All Cars in
Sample") ytitle("Marginal effect of mpg on price ($ per mpg)")
xtitle("Miles per gallon (mpg)")
```

- Use the **graph twoway scatter** command to plot the estimated marginal effect of mpg_i on price_i against wgt_i for all cars in the sample, after first sorting the sample observations in ascending order of the observed values of wgt_i . Enter the commands:

```
sort wgt
list wgt mpg mempg
graph twoway scatter mempg wgt
```

- Add appropriate **ytitle()**, **xtitle()**, **title()**, and **subtitle()** options to the above **graph twoway scatter** command to make the figure more readable. Enter *on one line* the command:

```
graph twoway scatter mempg wgt, title("Marginal Effect of MPG
on Car Price") subtitle("As Function of WGT, All Cars in
Sample") ytitle("Marginal effect of mpg on price ($ per mpg)")
xtitle("Car weight (wgt)")
```

□ Computing *Marginal* Effects of Continuous Explanatory Variables – margins

In the two preceding sections, you have computed the marginal effect of wgt_i and the marginal effect of mpg_i for selected values of the explanatory variables using the **lincom** command. In this section, you are introduced to the *Stata* **margins** command

as a convenient means of computing estimates of the **marginal effects of *continuous explanatory variables*** following OLS estimation of a linear regression model using the **regress** command, and of performing two-tail tests of the null hypothesis that the marginal effect is zero for the specified values of the explanatory variables. For this purpose, the **margins** command is often much simpler and faster to use than the **lincom** command.

However, a requirement of the **margins** command is that the estimation command it follows is formulated in what *Stata* calls factor-variable notation, which identifies for *Stata* whether a particular explanatory variable is a *continuous* or a *categorical* variable.

1. If an explanatory variable is *continuous*, then its name must be entered in the variable list of the estimation command with the prefix **c .**, which identifies the variable as a *continuous* variable. For example, the variable wgt_i must appear in the variable list of the **regress** command as **c .wgt**. Similarly, the variable mpg_i must appear in the variable list of the regress command as **c .mpg**.
2. If an explanatory variable is *categorical*, then its name must be entered in the variable list of the estimation command with the prefix **i .**, which identifies the variable as a *categorical*, or *indicator*, variable that takes only non-negative integer values denoting the various levels of the variable. For example, if the binary categorical variable sex_i takes only the two values 0 for males and 1 for females, then it must appear in the variable list of the **regress** command as **i .sex**. Similarly, if the multi-level categorical variable $group_i$ takes the five values 1, 2, 3, 4 and 5, then it must appear in the variable list of the **regress** command as **i .group**. You will be introduced to the use of the **margins** command for categorical variables in a later tutorial.

Recall that Model 2 for car prices is given by the population regression equation (1):

$$price_i = \beta_0 + \beta_1 wgt_i + \beta_2 mpg_i + \beta_3 wgt_i^2 + \beta_4 mpg_i^2 + \beta_5 wgt_i mpg_i + u_i. \quad (1)$$

Re-write the population regression function (or PRF) for Model 2 as:

$$E(price_i | wgt_i, mpg_i) = \beta_0 + \beta_1 wgt_i + \beta_2 mpg_i + \beta_3 wgt_i^2 + \beta_4 mpg_i^2 + \beta_5 wgt_i mpg_i \dots (2)$$

- First, use the **regress** command to re-estimate by OLS regression equation (1) for price_i using the **c.** prefix to identify the continuous explanatory variables. Enter the command:

```
regress price c.wgt c.mpg c.wgt#c.wgt c.mpg#c.mpg c.wgt#c.mpg
```

Note the use of the factor-variable operator #, which specifies interactions or products of two variables. Thus, **c.wgt#c.wgt** is wgt_i^2 , **c.mpg#c.mpg** is mpg_i^2 , and **c.wgt#c.mpg** is the interaction term $\text{wgt}_i \text{mpg}_i$.

Computing Marginal Effects of wgt_i

The estimated marginal effect of wgt_i on price_i for some given values wgt_0 of wgt_i and mpg_0 of mpg_i is calculated by the following linear function of wgt_i and mpg_i :

$$\text{estimate of } \frac{\partial \text{price}_i}{\partial \text{wgt}_i} = \frac{\partial \hat{E}(\text{price}_i \mid \text{wgt}_i, \text{mpg}_i)}{\partial \text{wgt}_i} = \hat{\beta}_1 + 2\hat{\beta}_3 \text{wgt}_0 + \hat{\beta}_5 \text{mpg}_0$$

- Use any of the following **margins** commands to evaluate the estimated marginal effect of wgt_i on price_i at the **sample mean values of wgt_i and mpg_i** :

```
margins, dydx(c.wgt)
```

```
margins, dydx(wgt)
```

```
margins, dydx(c.wgt) at((mean) wgt mpg)
```

Note that all three of these **margins** commands produce identical results. That is because the default values of the explanatory variables at which the **margins** command computes estimates of marginal effects are the sample mean values.

In addition to an estimate of the conditional marginal effect of wgt_i on price_i , each of the above **margins** commands also computes (1) the estimated standard error of the estimated marginal effect of wgt_i on price_i , (2) a z-ratio (or large sample t-ratio) for performing a two-tail test of the null hypothesis that the marginal effect of wgt_i on price_i equals zero for the average car in the sample, (3) a two-tail p-value for the z-ratio, and (4) a two-sided 95% confidence interval for the marginal

effect of wgt_i on $price_i$. Would you retain or reject the null hypothesis at the 5 percent significance level, at the 1 percent significance level?

- Use the `at(...)` option on the **margins** command to estimate the marginal effect of wgt_i on $price_i$ for the **25-th sample percentiles of wgt_i and mpg_i** . Enter the two **margins** commands:

```
margins, dydx(c.wgt) at((p25) wgt (p25) mpg)
margins, dydx(c.wgt) at((p25) wgt mpg)
```

Note that these two **margins** commands produce identical results; the second just saves a few keystrokes. Would you retain or reject the null hypothesis that the marginal effect of wgt_i on $price_i$ equals zero for a car whose weight and fuel efficiency equal the **25-th sample percentile values**?

- Use any of the following **margins** commands with the `at(...)` option to estimate the marginal effect of wgt_i on $price_i$ for the **50-th sample percentiles of wgt_i and mpg_i** :

```
margins, dydx(c.wgt) at((p50) wgt (p50) mpg)
margins, dydx(c.wgt) at((median) wgt (median) mpg)
margins, dydx(c.wgt) at((median) wgt mpg)
```

Would you retain or reject the null hypothesis that the marginal effect of wgt_i on $price_i$ equals zero for the median car in the sample?

- Use the following **margins** command to evaluate the estimated marginal effect of wgt_i on $price_i$ of a car in the **25-th sample percentile of wgt_i and the 75-th sample percentile in mpg_i** :

```
margins, dydx(c.wgt) at((p25) wgt (p75) mpg)
```

Would you retain or reject the null hypothesis that the marginal effect of wgt_i on $price_i$ equals zero for a car in the 25-th sample percentile of wgt_i and the 75-th sample percentile in mpg_i ?

- Finally, you can evaluate the marginal effect of wgt_i on $price_i$ at several different values of wgt_i and a fixed value of mpg_i to examine how the conditional marginal effect of wgt_i varies with wgt_i while holding the values of the other explanatory

variables constant at specified fixed values. To illustrate, suppose we want to estimate the marginal effect of wgt_i on $price_i$ at the sample median value of mpg_i and the following 11 values of wgt_i : 2,000, 2,250, 2,500, 2,750, 3,000, 3,250, 3,500, 3,750, 4,000, 4,250, 4,500. These 11 values of wgt_i approximately span the range of sample values of wgt_i in the sample data. Enter the following **margins** command, noting in particular the formulation of the **at(...)** option:

```
margins, dydx(c.wgt) at(wgt = (2000(250)4500) (median) mpg)
```

The above **at(...)** option fixes the value of the continuous explanatory variable mpg_i at its sample median value and computes the conditional marginal effect of wgt_i on $price_i$ at 11 distinct values of wgt_i , beginning at 2,000 pounds and increasing in increments of 250 pounds to 4,500 pounds.

- Now we can use the **marginsplot** command to draw a line graph of the estimated marginal effect of wgt_i on $price_i$ against wgt_i for cars with the median value of mpg_i and the 11 distinct values of wgt_i specified in the preceding **margins** command. Enter the following **marginsplot** command:

```
marginsplot
```

Note that this **marginsplot** command displays the 95% confidence limits for the marginal effect estimates at each of the 11 data points.

- To suppress the display of the confidence intervals, add the **noci** option to the above **marginsplot** command; **noci** is short for “no confidence interval”. Enter the following **marginsplot** command with the **noci** option:

```
marginsplot, noci
```

- Add appropriate titles to the line graph produced by the foregoing **marginsplot** commands using the **title()**, **subtitle()** and **ytitle()** options. The **xtitle()**, **note()** and **caption()** options can also be used to further annotate the graph. Enter *on one line* the following **marginsplot** command with **title()**, **subtitle()** and **ytitle()** options:

```
marginsplot, title(Conditional Marginal Effect of WGT on Car Price) subtitle("at Sample Median of MPG = 20, with 95% Confidence Limits") ytitle(Change in car price (dollars))
```

You may want to save a graph to disk in a format such that you can use it, and perhaps modify it, either later in your current *Stata* session or during some future *Stata* session. The **graph save** command can be used to do this.

- To save the graph created by the preceding **marginsplot** command to a disk file in the current *Stata* working directory and give that disk file the name **graph1_tutorial4_f2013.gph**, enter the following **graph save** command:

```
graph save graph1_tutorial4_f2013.gph
```

Note that specifying the file extension **.gph** in the name given to the disk file is optional; all **graph save** commands will create disk files with the file extension **.gph**.

Computing Marginal Effects of mpg_i

The estimated marginal effect of mpg_i on price_i for some given values wgt₀ of wgt_i and mpg₀ of mpg_i is calculated by the following linear function of wgt_i and mpg_i:

$$\text{estimate of } \frac{\partial \text{price}_i}{\partial \text{mpg}_i} = \frac{\partial \hat{E}(\text{price}_i \mid \text{wgt}_i, \text{mpg}_i)}{\partial \text{mpg}_i} = \hat{\beta}_2 + 2\hat{\beta}_4 \text{mpg}_0 + \hat{\beta}_5 \text{wgt}_0$$

- Use any of the following **margins** commands to evaluate the estimated marginal effect of mpg_i on price_i at the **sample mean values of wgt_i and mpg_i**:

```
margins, dydx(c.mpg)
margins, dydx(mpg)
margins, dydx(c.mpg) at((mean) wgt (mean) mpg)
margins, dydx(c.mpg) at((mean) wgt mpg)
```

Note again that all four of these **margins** commands produce identical results because the default values of the explanatory variables at which the **margins** command computes estimates of marginal effects are the sample mean values.

Would you retain or reject the null hypothesis that the marginal effect of mpg_i on price_i equals zero for the average car in the sample at the 5 percent significance level, at the 1 percent significance level?

- Use the **at(...)** option on the **margins** command to estimate the marginal effect of mpg_i on $price_i$ for the **25-th sample percentiles of wgt_i and mpg_i** . Enter the two **margins** commands, which you will see produce identical results:

```
margins, dydx(c.mpg) at((p25) wgt (p25) mpg)
margins, dydx(c.mpg) at((p25) wgt mpg)
```

Would you retain or reject the null hypothesis that the marginal effect of mpg_i on $price_i$ equals zero for a car whose weight and fuel efficiency equal the 25-th sample percentile values?

- Use any of the following **margins** commands with the **at(...)** option to estimate the marginal effect of mpg_i on $price_i$ for the **50-th sample percentiles of wgt_i and mpg_i** :

```
margins, dydx(c.mpg) at((p50) wgt (p50) mpg)
margins, dydx(c.mpg) at((median) wgt (median) mpg)
margins, dydx(c.mpg) at((median) wgt mpg)
```

Would you retain or reject the null hypothesis that the marginal effect of mpg_i on $price_i$ equals zero for the median car in the sample?

- Use the following **margins** command to evaluate the estimated marginal effect of mpg_i on $price_i$ for a car in the **25-th sample percentile of wgt_i and the 75-th sample percentile in mpg_i** :

```
margins, dydx(c.mpg) at((p25) wgt (p75) mpg)
```

Would you retain or reject the null hypothesis that the marginal effect of mpg_i on $price_i$ equals zero for a car in the 25-th sample percentile of wgt_i and the 75-th sample percentile in mpg_i ?

- Finally, you can evaluate the marginal effect of mpg_i on $price_i$ at several different values of mpg_i and a fixed value of wgt_i to examine how the conditional marginal effect of mpg_i varies with mpg_i while holding the values of the other explanatory variables constant at specified fixed values. To illustrate, suppose we want to estimate the marginal effect of mpg_i on $price_i$ at the sample median value of wgt_i and the following 15 values of wgt_i : 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40. These 15 values of mpg_i approximately span the range of sample

values of mpg_i in the sample data. Enter the following **margins** command, noting in particular the formulation of the **at(...)** option:

```
margins, dydx(c.mpg) at(mpg = (12(2)40) (median) wgt)
```

The above **at(...)** option fixes the value of the continuous explanatory variable wgt_i at its sample median value and computes the conditional marginal effect of mpg_i on $price_i$ at 15 distinct values of mpg_i , beginning at 12 and increasing in increments of 2 to 40.

- We can now use the **marginsplot** command to draw a line graph of the estimated marginal effect of mpg_i on $price_i$ against mpg_i for cars with the median value of wgt_i and the 15 distinct values of mpg_i specified in the preceding **margins** command. Enter the following **marginsplot** command:

```
marginsplot
```

Note that this **marginsplot** command displays the 95% confidence limits for the marginal effect estimates at each of the 15 data points.

- To suppress the display of the confidence intervals, add the **noci** option to the above **marginsplot** command; **noci** is short for “no confidence interval”. Enter the following **marginsplot** command with the **noci** option:

```
marginsplot, noci
```

- Add appropriate titles to the line graph produced by the foregoing **marginsplot** commands using the **title()**, **subtitle()** and **ytitle()** options. The **xtitle()**, **note()** and **caption()** options can also be used to further annotate the graph. Enter *on one line* the following **marginsplot** command with **title()**, **subtitle()** and **ytitle()** options:

```
marginsplot, title(Conditional Marginal Effect of MPG on Car Price) subtitle("at median WGT (3,190 pounds), with 95% confidence limits") ytitle(Change in car price (US dollars))
```

- To save the graph created by the preceding **marginsplot** command to a disk file in the current *Stata* working directory and give that disk file the name **graph2_tutorial4_f2013.gph**, enter the following **graph save** command:

```
graph save graph2_tutorial4_f2013.gph
```

Note that specifying the file extension **.gph** in the name given to the disk file is optional; all **graph save** commands will create disk files with the file extension **.gph**.

□ Exporting a graph to disk – graph export

You may also want to export a graph created by *Stata* to a file on disk so that you can insert that graph into a document created by a word processor such as MS Word. The **graph export** command can be used to do this.

Basic Syntax

```
graph export newfilename.suffix [ , options ]
```

This command exports to a disk file in the current *Stata* working directory the graph currently displayed in the Graph window. The output format of this file is determined by the suffix of *newfilename.suffix*. An alternative way to specify the output format of the exported file is to use the **as(fileformat)** option, where *fileformat* specifies the desired format of the exported file.

The available *output formats* that can be specified with the **graph export** command are listed in the following table.

| <i>suffix</i> | implied option | output format |
|---------------|-----------------------|---------------------------------|
| .ps | as(ps) | PostScript |
| .eps | as(eps) | EPS (Encapsulated PostScript) |
| .wmf | as(wmf) | Windows Metafile |
| .emf | as(emf) | Windows Enhanced Metafile |
| .pict | as(pict) | Mackintosh PICT format |
| .pdf | as(pdf) | PDF |
| .png | as(png) | PNG (Portable Network Graphics) |
| .tif | as(tif) | TIFF |

Notes: **ps** and **eps** are available with all versions of *Stata*; **png** and **tif** are available for all versions of *Stata* except *Stata(console)* for Unix; **wmf** and **emf** are available only with *Stata for Windows*; and **pict** and **pdf** are available only with *Stata for Mackintosh*.

Source: StataCorp 2007. *Stata Statistical Software Release 10: Graphics*. College Station, TX: StataCorp LP, p. 124.

- To illustrate how to use the **graph export** command, first use the *Stata* **graph use** command to read into memory and display in the Graph window the first graph you saved to disk during this tutorial, specifically the graph named **graph1_tutorial4_f2013**. Enter the following command:

```
graph use graph1_tutorial4_f2013
```

- To export and save this line graph in EPS (Encapsulated PostScript) format to a file named **graph1_tutorial4_f2013.eps** in the current *Stata* working directory, enter the **graph export** command:

```
graph export graph1_tutorial4_f2013.eps
```

- To export and save the same line graph in Windows Metafile format to a file named **graph1_tutorial4_f2013.wmf** in the current *Stata* working directory, enter the **graph export** command:

```
graph export graph1_tutorial4_f2013.wmf
```

- To export and save the same line graph in Windows Enhanced Metafile format to a file named **graph1_tutorial4_f2013.emf** in the current *Stata* working directory, enter the **graph export** command:

```
graph export graph1_tutorial4_f2013.emf
```

- Finally, to export and save the same line graph in PNG (Portable Network Graphics) format to a file named **graph1_tutorial4_f2013.png** in the current *Stata* working directory, enter the **graph export** command:

```
graph export graph1_tutorial4_f2013.png
```

- To list the several graph files you have just saved to files in the current *Stata* working directory, enter the following **dir** command:

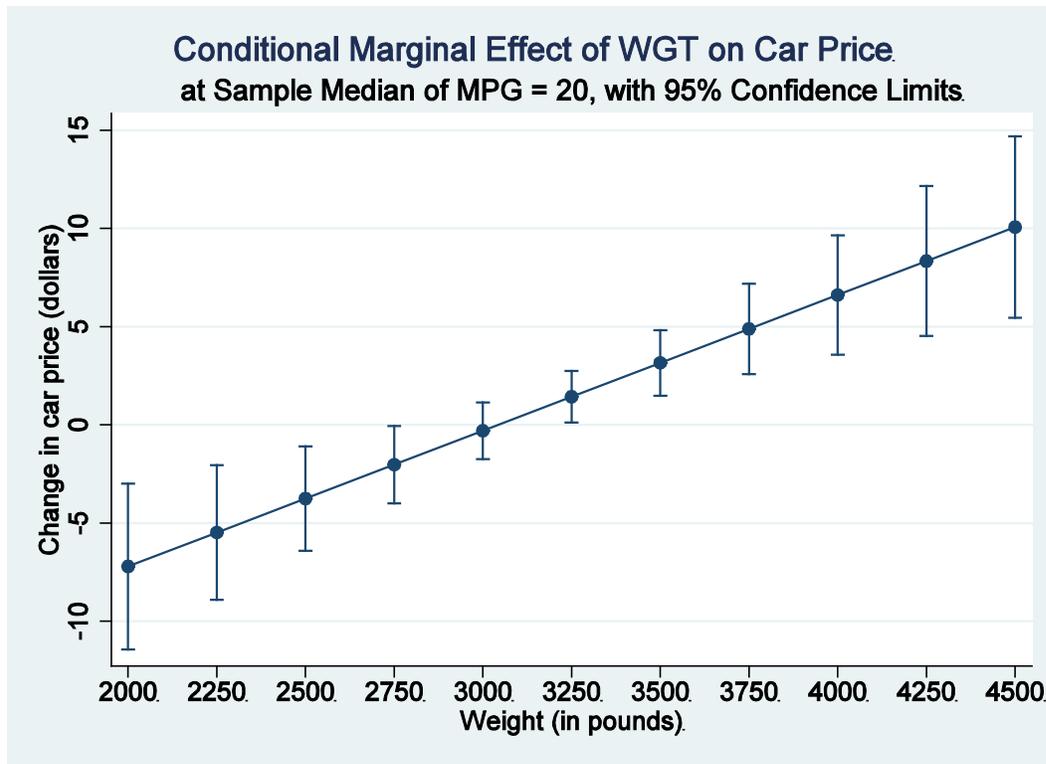
```
dir graph1_tutorial4_f2013.*
```

□ Inserting exported graph files into an MS Word document

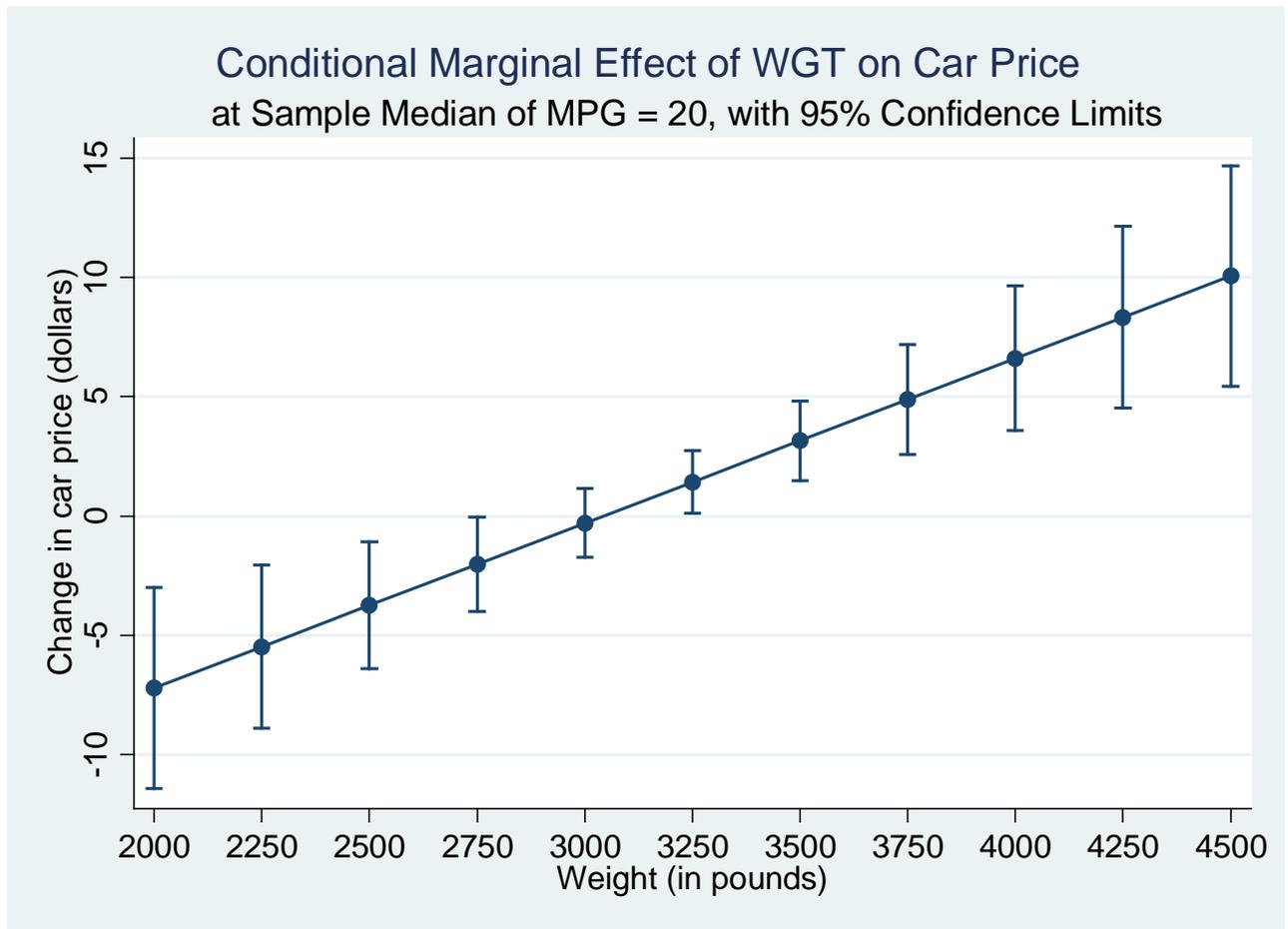
In this section, we illustrate how exported *Stata* graph files in various output formats appear when they are inserted into the current document, which is a Microsoft Word document. The exported files we insert are the following: the file `graph1_tutorial14_f2013.eps`, which is in EPS (Encapsulated PostScript) format; the file `graph1_tutorial14_f2013.wmf`, which is in Windows Metafile format; the file `graph1_tutorial14_f2013.emf`, which is in Windows Enhanced Metafile format; and the file `graph1_tutorial14_f2013.png`, which is in PNG (Portable Network Graphics) format.

In MS Word 2010, each of these *Stata* graph files can be inserted into a document by selecting **Insert > Picture** and then choosing the filename of the graph you want from the list of all pictures.

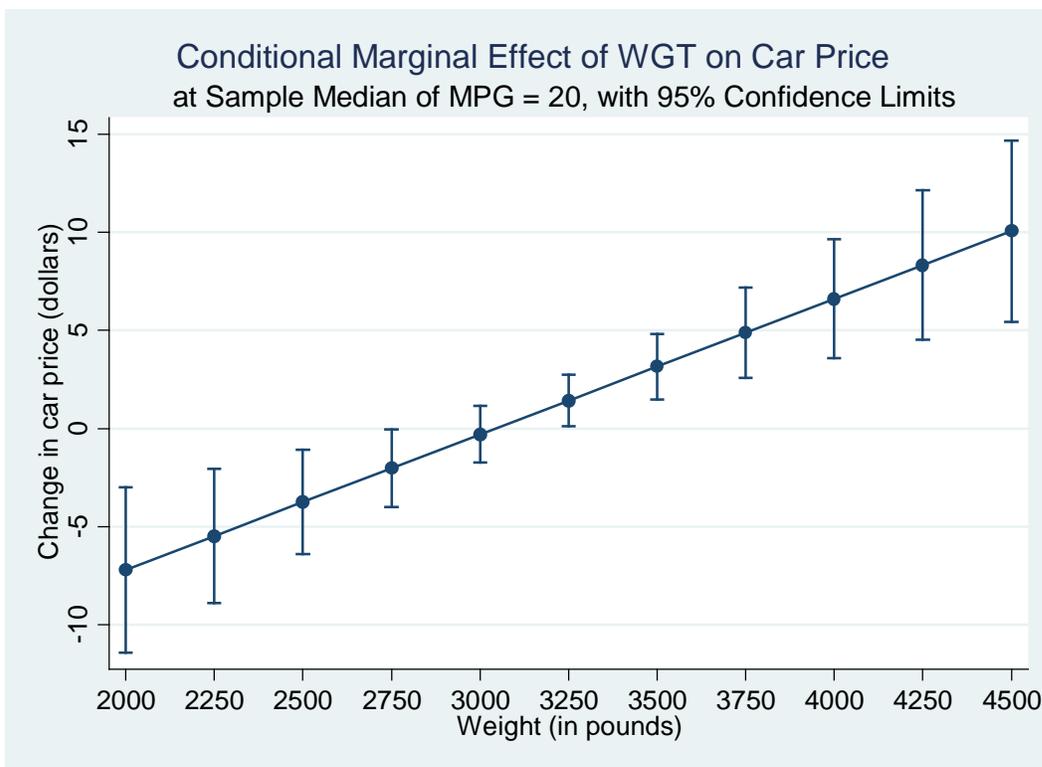
- ◆ The file `graph1_tutorial14_f2013.eps` in EPS (Encapsulated PostScript) format:



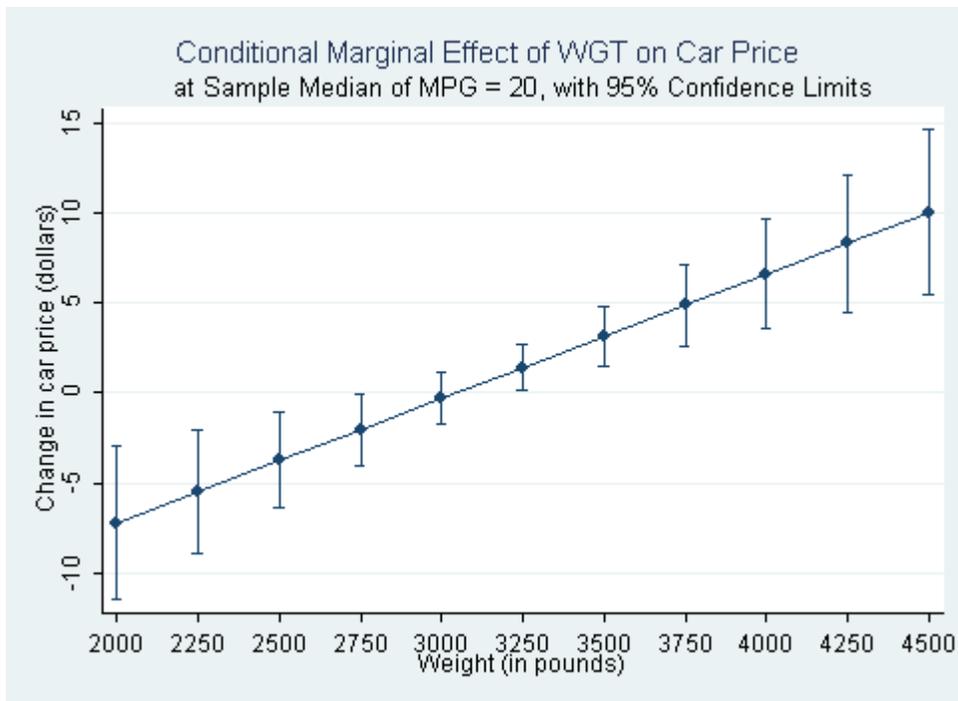
- ◆ The file `graph1_tutorial14_f2013.wmf` in Windows Metafile format:



- ◆ The file `graph1_tutorial4_f2013.emf` in Windows Enhanced Metafile format:



- ◆ The file `graph1_tutorial14_f2013.png` in PNG (Portable Network Graphics) format:



□ Preparing to End Your *Stata* Session

Before you end your *Stata* session, you should do three things.

- First, you will want to **save the current data set**. Enter either of the following **save** commands to save the current data set as *Stata*-format data set **auto4.dta**:

```
save auto4          or          save auto4, replace
```

- Second, **close the log file** you have been recording. Enter the command:

```
log close
```

- Third, **close the command log file** you have been recording. Enter the command:

```
cmdlog close
```

□ End Your *Stata* Session – exit

- **To end your *Stata* session**, use the **exit** command. Enter the command:

```
exit          or          exit, clear
```

□ Cleaning Up and Clearing Out

After returning to Windows, you should copy all the files you have used and created during your *Stata* session to your own portable electronic storage device such as a flash memory stick. These files will be found in the *Stata working directory*, which is usually **C:\data** on the computers in Dunning 350. There are three files you will want to be sure you have: the *Stata* log file **452tutorial4.log**; the *Stata* command log file **452tutorial4.txt**; and the saved *Stata*-format dataset **auto4.dta**. Finally, you will probably want to take with you the several *Stata* graph files you saved to disk during this tutorial. Use the Windows **copy** command to copy any files you want to keep to your own portable electronic storage device (e.g., a flash memory stick).

Finally, **as a courtesy to other users** of the computing classroom, please delete all the files you have used or created from the *Stata* working directory.