

# Beamer Template

Nic Duquette  
USC Price

# Today's Topics

1. Basic Slides Stuff

2. Code

3. Tables and Figures

# Basic Slides Stuff

## What is this

- ▶  $\text{\LaTeX}$  is a system for compiling documents from markup
- ▶ That means the same markup can be compiled more than one way
- ▶ This template creates two versions of the same slide deck
- ▶ One to be projected during a classroom lecture
- ▶ And one to give to students as a handout

## Colors

- ▶ The classroom version of the slides use a light-on-dark theme, which is kind to eyes
- ▶ The handouts use dark-on-white, to avoid wasting ink
- ▶ A set of colors names beginning with "x" is defined differently in the two modes, to have appropriate contrast with light or dark backgrounds
- ▶ Colors: `xred`, `xorange`, `xyellow`, `xgreen`, `xblue`, `xpurple`, `xbrown`, `xmono`. “`xbg`” will use the background color.

## Overlays

- ▶ Use overlays to reveal stuff in the classroom
- ▶ Bit
- ▶ By by bit
- ▶ Only the final, complete slide will be in the handout version

## Solution frames

- ▶ Sometimes we don't want students to have the answer to in-class exercises
- ▶ ...yet we also want them to have the handout to follow along
- ▶ Can you guess what the solutionframe environment does?

## References

- ▶ Regular citation commands like \cite will work normally
  - ▶ Duquette (2016) is a paper you can read if you want to
- ▶ Sometimes we want to cite a claim, but don't want a heap of words on the slide
- ▶ Command \citep{...} will deemphasize a parenthetical citation
  - ▶ Scholars are known to make claims about things (Duquette 2016)
- ▶ The references slide at the end of the deck automatically puts the reference list in small text and breaks up across frames as needed.

## Math

- ▶ The general preamble defines a variety of common operators that are tricky to code manually, such as \expect for the expectation operator. For example,
- ▶  $\mathbb{E}[\varepsilon_i \varepsilon_j | X] = \sigma_i^2$  if  $i = j$ ,  $\sigma_{ij}$  if  $i \neq j$ , and

$$\mathbb{V}[\varepsilon_i | X_i] = \mathbb{E}[\varepsilon' \varepsilon | X_i] = \boldsymbol{\Omega} = \begin{bmatrix} \sigma_1^2 & \sigma_{1,2} & \cdots & \sigma_{1,m} \\ \sigma_{1,2} & \sigma_2^2 & \cdots & \sigma_{2,m} \\ \vdots & \ddots & \ddots & \vdots \\ \sigma_{1,m} & \sigma_{2,m} & \cdots & \sigma_m^2 \end{bmatrix}$$

## More buttons

- ▶ One of the best Beamer tricks is to use a \beamergotobutton to hyperlink to a hidden slide with supplementary content,  Like this button
- ▶ I also use buttons to include hyperlinks to files and resources outside of the slides, using \href and custom commands
  - ▶  File on Google Drive
  - ▶  Code on Github
  - ▶  Link to a generic web site

# Slide aesthetics

- ▶ Beamer clutter, like navigation controls and headers, are deliberately omitted
- ▶ Many of Paul Goldsmith-Pinkham's ideas are used here too
  - ▶ Wide-itemize environment spaces out bullet points
  - ▶ 16:9 aspect ratio
  - ▶ \sectionframe adds colorful frames defining sections
- ▶ \transitionframe also adds the colorful frame, but does not add a section to the table of contents



The Goldsmith-Pinkham Beamer Template

# Code

## Insert and color bits of code

- ▶ To use code highlighting with the same color scheme as the rest of the presentation, copy the style files included with the template to your Pygments styles directory
- ▶ See directions in the README for more detail
- ▶ Code is only highlighted if the slides are compiled using the options in the shell script
- ▶ Otherwise, the slides default to “draft” mode (monospace font, no color).

## Stata code

- ▶ Individual fragments of Stata code can be inserted using `\stata{code text}`
  - ▶ `regress Y X, vce(cluster id)`
- ▶ Or, an entire frame of Stata code can be created from an external file using `\InsertStataFrame{frametitle}{filepath.do}`

# A Stata Frame

```
di "hello"

forval i=1(3)7 {

    // Print the value of i
    display "`i'"

}
```

## Python code

- ▶ Similarly, individual fragments of Python code can be inserted using  
  \python{*code text*}
  - ▶ `print("Hello World")`
- ▶ Or, an entire frame of Python code can be created from an external file using  
  \InsertPythonFrame{*frametitle*}{*filepath.py*}

# A Python Frame

```
# Use the "statsmodels" package of econometric commands with "sm" shortcut
import statsmodels as sm

# Define a model regressing y on X using OLS from the sm module
ols_model = sm.OLS(y, X)

# Estimate the model using the .fit attribute
ols_results = ols_model.fit()

# Report the results to the terminal
print(ols_results.summary())
```

## Tweak the code slides

- ▶ The `\Insert...Frame` commands will take an optional argument to adjust the text size, e.g. `\InsertPythonFrame[\scriptsize]{Title}{Filepath}`
- ▶ You can fit up to 30-ish lines of code on a slide, max, using `\tiny`
- ▶ Aim for 20 or fewer lines for readability

## Doing something else

- ▶ Adding other languages is straightforward
  1. Define a new minted environment in both preambles for your favorite language  
 Supported languages
  2. Copy the new command definitions for Stata or Python and adapt them as needed
- ▶ What about other tweaks to code blocks? You may get frustrated.
- ▶ Beamer and verbatim text (like code) often don't get along; expect bugs.

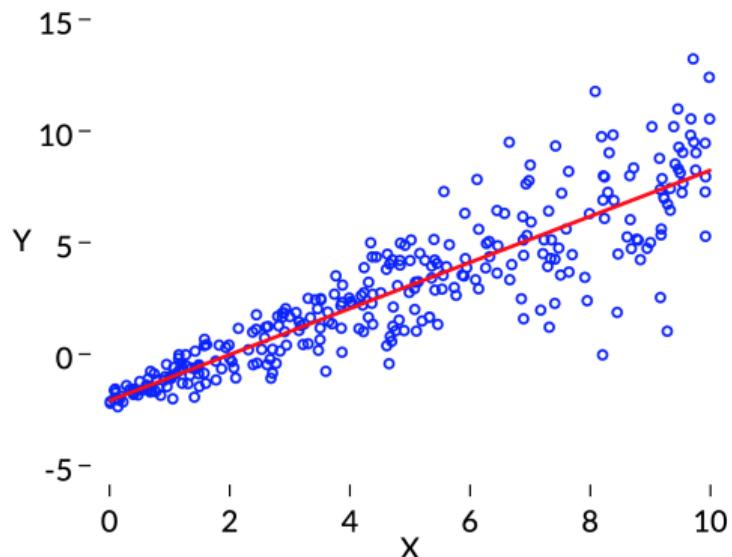
# Tables and Figures

# L<sup>A</sup>T<sub>E</sub>Xtables

	(1)	(2)	(3)
Income	0.298*** (0.026)	0.638*** (0.050)	0.488*** (0.042)
Education	0.092*** (0.015)	0.077*** (0.016)	0.082*** (0.015)

- ▶ Insert .tex tables from statistical software using \input{}
- ▶ \tablepath defines the relative path to the table directory
- ▶ The begin{column} environment makes it easy to put comments alongside your float

# Images can change with the slide style



- ▶ The macro \figpath points to images in the `figs_dark` directory in classroom mode
- ▶ But it points to `figs_light` in handout mode
- ▶ Automate this process by using the color profiles in this template to create two versions of your charts

# Tell Stata the color profiles

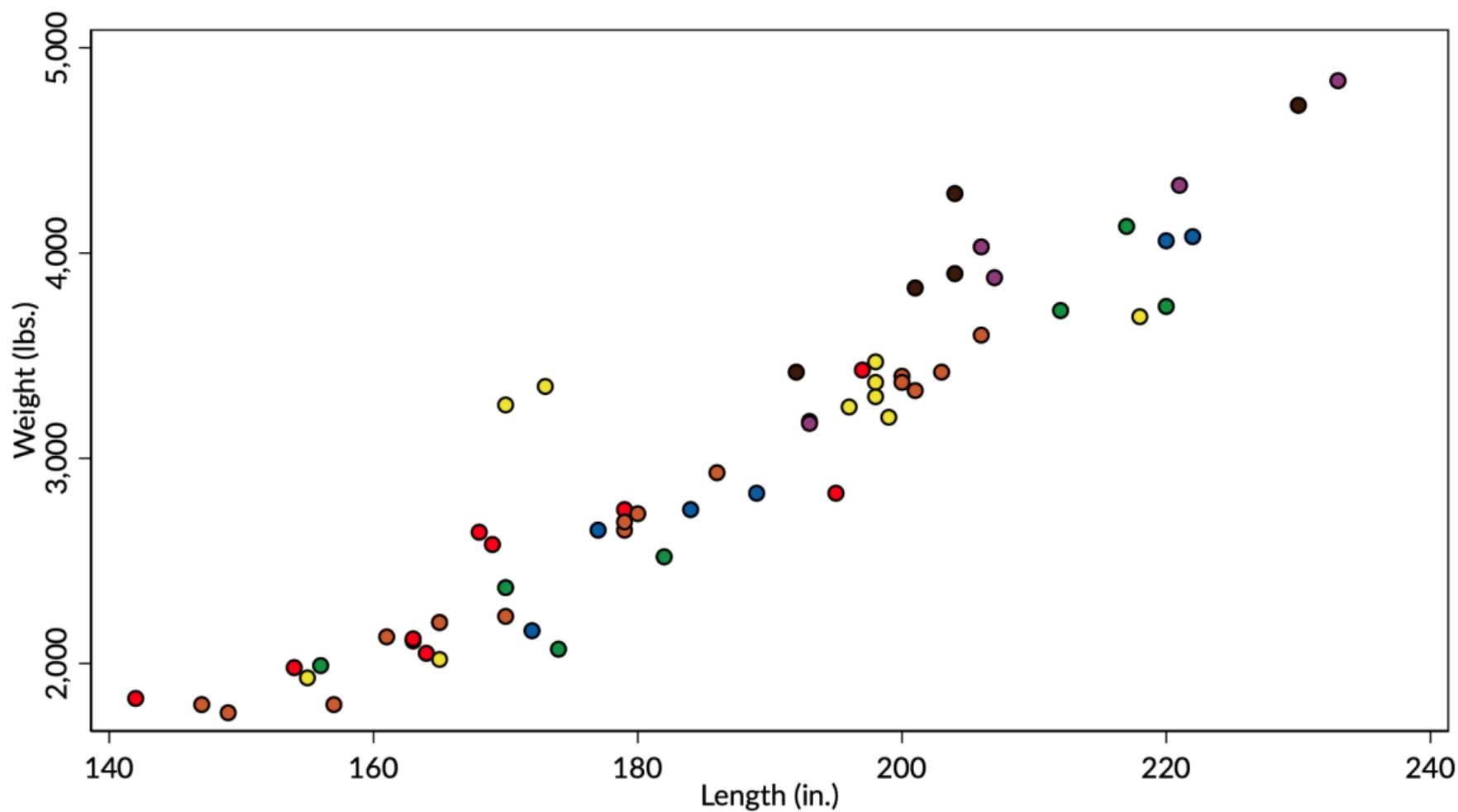
```
global ishandout=1
if ${ishandout}==1 {                                // Set globals this way if it's for a handout
    global xmono "black"                         // Colors
    global xred="255 30 30"
    global xgreen="0 158 85"
    global xblue ="0 114 178"
    global xpurple="163 82 143"
    global xyellow="240 228 66"
    global xorange="213 111 62"
    global xbrown="76 35 10"
    global xbg "white"

    global xscheme "s1color"                      // Twoway plot theme

    global figsave "../figs_light"                 // Where to save the figure
}
else {
    global xmono "white"                        // Classroom globals
    global xred="255 81 81"
    global xgreen="0 250 95"
    global xblue ="100 150 255"
    global xpurple ="232 179 252"
    global xyellow ="240 228 66"
    global xorange ="255 140 33"
    global xbrown ="144 67 19"
    global xbg "black"
    global xscheme "s1rcolor"
    global figsave "../figs_dark"
}
```

## Use them in a chart

```
// An example graph
webuse auto, clear
twoway (scatter weight length if price<4000, mcolor(${xred}) mlcolor(${xmono})) ///
(scatter weight length if price<4500 & price>=4000, mcolor(${xorange}) mlcolor(${xmono})) ///
(scatter weight length if price<5000 & price>=4500, mcolor(${xyellow}) mlcolor(${xmono})) ///
(scatter weight length if price<7000 & price>=6000, mcolor(${xgreen}) mlcolor(${xmono})) ///
(scatter weight length if price<10000 & price>=7000, mcolor(${xblue}) mlcolor(${xmono})) ///
(scatter weight length if price<12000 & price>=10000, mcolor(${xpurple}) mlcolor(${xmono})) ///
(scatter weight length if price>=12000, mcolor(${xbrown}) mlcolor(${xmono})) ///
, scheme(${xscheme}) legend(off)
graph display, xsize(16) ysize(9)
graph export "${figsave}/auto.png", as(png) replace
```



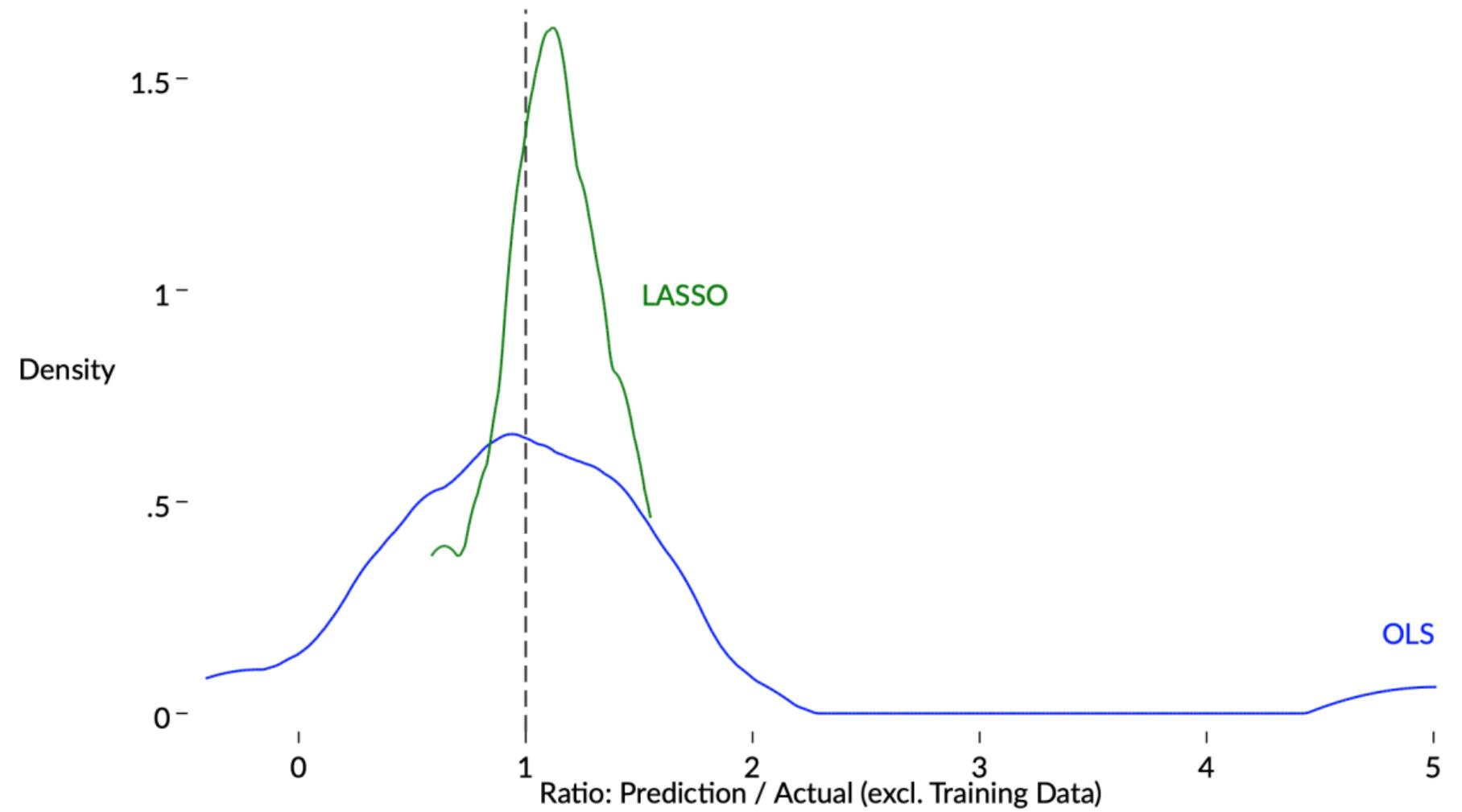
## Images can also be the same



- ▶ Sometimes you want to include an image that is the same for both versions of the slides
- ▶ These are stored in the `figs_both` folder
- ▶ Use the filepath `\figboth/myimage.jpg` to load the image.

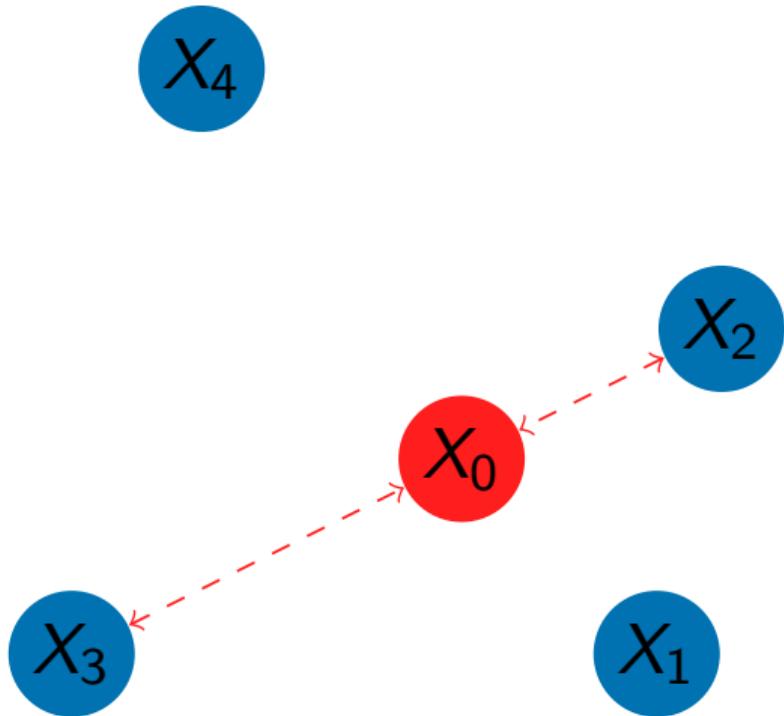
## Big images

- ▶ Sometimes, you don't want text next to your picture
- ▶ You want a big image you can see details of
- ▶ `\pictureframe{filepath}` will create a slide with one image filling as much space as it can
- ▶ Must be compiled twice for correct placement





## Diagrams



- ▶ If you use the theme color names, your tikz diagrams will change with the slide palette
- ▶ Overlays work! Add the visible on property to elements of the figure.

# References

# References I

**Duquette, Nicolas J.**, "Do Tax Incentives Affect Charitable Contributions? Evidence from Public Charities' Reported Revenues," *Journal of Public Economics*, May 2016, 137, 51–69.

Last update: April 26, 2021

# Hidden Slides

You found the hidden slide!

