Beamer Class Demonstration

Gary King and Ista Zahn

IQSS

September 18, 2018

Outline

Beamer Features

Some of Gary's Examples

Other Features

Structural Features

More Features

Blocks

Appendix

What's this course about?

- Specific statistical methods for many research problems - How to learn (or create) new methods - Inference: Using facts you know to learn about facts you don't know

What's this course about?

- Specific statistical methods for many research problems - How to learn (or create) new methods - Inference: Using facts you know to learn about facts you don't know
- How to write a publishable scholarly paper

What's this course about?

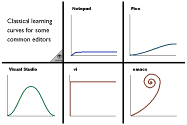
- Specific statistical methods for many research problems - How to learn (or create) new methods - Inference: Using facts you know to learn about facts you don't know
- How to write a publishable scholarly paper
- All the practical tools of research — theory, applications, simulation, programming, word processing, plumbing, whatever is useful

What's this course about?

- Specific statistical methods for many research problems - How to learn (or create) new methods - Inference: Using facts you know to learn about facts you don't know
- How to write a publishable scholarly paper
- All the practical tools of research — theory, applications, simulation, programming, word processing, plumbing, whatever is useful
- $\rightsquigarrow$ Outline and class materials:

What's this course about?

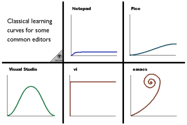
- Specific statistical methods for many research problems - How to learn (or create) new methods - Inference: Using facts you know to learn about facts you don't know
- How to write a publishable scholarly paper
- All the practical tools of research — theory, applications, simulation, programming, word processing, plumbing, whatever is useful
- ⇝ Outline and class materials:

  j.mp/G2001 

  -

What's this course about?

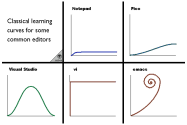
- Specific statistical methods for many research problems - How to learn (or create) new methods - Inference: Using facts you know to learn about facts you don't know
- How to write a publishable scholarly paper
- All the practical tools of research — theory, applications, simulation, programming, word processing, plumbing, whatever is useful
- ⇝ Outline and class materials:

  j.mp/G2001 

  -
  - The syllabus gives topics, not a weekly plan.

What's this course about?

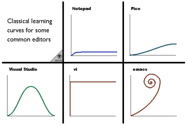
- Specific statistical methods for many research problems - How to learn (or create) new methods - Inference: Using facts you know to learn about facts you don't know
- How to write a publishable scholarly paper
- All the practical tools of research — theory, applications, simulation, programming, word processing, plumbing, whatever is useful
- ⇝ Outline and class materials:

  j.mp/G2001 

  -
  - The syllabus gives topics, not a weekly plan.
  - We will go as fast as possible subject to everyone following along

What’s this course about?

- Specific statistical methods for many research problems - How to learn (or create) new methods - Inference: Using facts you know to learn about facts you don’t know
- How to write a publishable scholarly paper
- All the practical tools of research — theory, applications, simulation, programming, word processing, plumbing, whatever is useful
- ⇝ Outline and class materials:

  j.mp/G2001 

  -
  - The syllabus gives topics, not a weekly plan.
  - We will go as fast as possible subject to everyone following along
  - We cover different amounts of material each week

How much math will you scare us with?

- All math requires two parts: proof and concepts & intuition
- Different classes emphasize:
  - Baby Stats: dumbed down proofs, vague intuition
  - Math Stats: rigorous mathematical proofs
  - Practical Stats: deep concepts and intuition, proofs when needed
    - Goal: how to do empirical research, in depth
    - Use rigorous statistical theory — when needed
    - Insure we understand the intuition — always
    - Always traverse from theoretical foundations to practical applications
    - Includes "how to" computation
    - $\rightsquigarrow$ Fewer proofs, more concepts, better practical knowledge
- Do you have the background for this class?

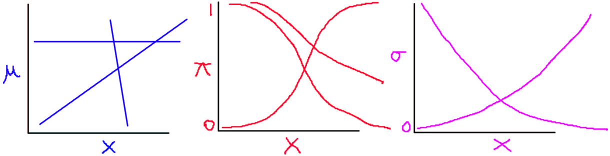
How much math will you scare us with?

- All math requires two parts: proof and concepts & intuition
- Different classes emphasize:
  - Baby Stats: dumbed down proofs, vague intuition
  - Math Stats: rigorous mathematical proofs
  - Practical Stats: deep concepts and intuition, proofs when needed
    - Goal: how to do empirical research, in depth
    - Use rigorous statistical theory — when needed
    - Insure we understand the intuition — always
    - Always traverse from theoretical foundations to practical applications
    - Includes "how to" computation
    - $\rightsquigarrow$ Fewer proofs, more concepts, better practical knowledge
- Do you have the background for this class?

A Test: What's this?

$$b = (X'X)^{-1}X'y$$

Systematic Components: Examples



- $E(Y_i) \equiv \mu_i = X_i\beta = \beta_0 + \beta_1 X_{1i} + \cdots + \beta_k X_{ki}$
- $\Pr(Y_i = 1) \equiv \pi_i = \frac{1}{1+e^{-x_i\beta}}$
- $V(Y_i) \equiv \sigma_i^2 = e^{x_i\beta}$
- Interpretation:
  - Each is a class of functional forms
  - Set β and it picks out one member of the class
  - β in each is an "effect parameter" vector, with different meaning

Negative Binomial Derivation

Recall:

one two three

Negative Binomial Derivation

Recall:

$$\Pr(A|B) = \frac{\Pr(AB)}{\Pr(B)} \implies \Pr(AB) = \Pr(A|B)\Pr(B)$$

one two three

Negative Binomial Derivation

Recall:

$$\Pr(A|B) = \frac{\Pr(AB)}{\Pr(B)} \implies \Pr(AB) = \Pr(A|B)\Pr(B)$$

one two three

$$\mathsf{NegBin}(y|\phi, \sigma^2) = \int_0^\infty \mathsf{Poisson}(y|\lambda) \times \mathsf{gamma}(\lambda|\phi, \sigma^2) d\lambda$$

Negative Binomial Derivation

Recall:

$$\Pr(A|B) = \frac{\Pr(AB)}{\Pr(B)} \implies \Pr(AB) = \Pr(A|B)\Pr(B)$$

one two three

$$\begin{aligned}\mathsf{NegBin}(y|\phi,\sigma^2) &= \int_0^\infty \mathsf{Poisson}(y|\lambda) \times \mathsf{gamma}(\lambda|\phi,\sigma^2) d\lambda \\ &= \int_0^\infty \P(y,\lambda|\phi,\sigma^2) d\lambda\end{aligned}$$

Negative Binomial Derivation

Recall:

$$\Pr(A|B) = \frac{\Pr(AB)}{\Pr(B)} \implies \Pr(AB) = \Pr(A|B)\Pr(B)$$

one two three

$$\begin{aligned}
\mathsf{NegBin}(y|\phi,\sigma^2) &= \int_0^\infty \mathsf{Poisson}(y|\lambda) \times \mathsf{gamma}(\lambda|\phi,\sigma^2) d\lambda \\
&= \int_0^\infty \P(y,\lambda|\phi,\sigma^2) d\lambda \\
&= \frac{\Gamma\left(\frac{\phi}{\sigma^2-1} + y_i\right)}{y_i!\Gamma\left(\frac{\phi}{\sigma^2-1}\right)} \left(\frac{\sigma^2-1}{\sigma^2}\right)^{y_i} \left(\sigma^2\right)^{\frac{-\phi}{\sigma^2-1}}
\end{aligned}$$

Outline

Beamer Features

Some of Gary's Examples

Other Features

Structural Features

More Features

Blocks

Appendix

Structural Features

Levels of Structure

- usual LaTeX \ section, \ subsection commands
- `frame` environments provide slides
- `block` environments divide slides into logical sections
- `columns` environments divide slides vertically (example later)
- overlays (‘a la prosper) change content of slides dynamically

Overlay Alerts

On the first overlay, this text is highlighted (or *alerted*).
On the second, this text is.

Structural Features

Levels of Structure

- usual LaTeX \ section, \ subsection commands
- `frame` environments provide slides
- `block` environments divide slides into logical sections
- `columns` environments divide slides vertically (example later)
- overlays (‘a la prosper) change content of slides dynamically

Overlay Alerts

On the first overlay, this text is highlighted (or *alerted*).
On the second, this text is.

Code blocks

```r
# Say hello in R
hello <- function(name) paste("hello", name)
```

Code blocks

```
# Say hello in R
hello <- function(name) paste("hello", name)

# Say hello in Python
def hello(name):
return("Hello" + " " + name)
```

Code blocks

```
# Say hello in R
hello <- function(name) paste("hello", name)

# Say hello in Python
def hello(name):
return("Hello" + " " + name)

-- Say hello in Haskell
hello name = "Hello" ++ " " ++ name
```

Code blocks

```
# Say hello in R
hello <- function(name) paste("hello", name)

# Say hello in Python
def hello(name):
return("Hello" + " " + name)

-- Say hello in Haskell
hello name = "Hello" ++ " " ++ name

/* Say hello in C */
#include <stdio.h>
int main()
{
  char name[256];
  fgets(name, sizeof(name), stdin);
  printf("Hello %s", name);
  return(0);
}
```

Alerts

- First level alert
- Second level alert
- Third level alert
- Fourth level alert
- Fifth level alert

Outline

Beamer Features

Some of Gary's Examples

Other Features

Structural Features

More Features

Blocks

Appendix

Other Features

Levels of Structure

- Clean, extensively customizable visual style
- Hyperlinks (http://github.com/izahn/iqss-beamer-theme
- No weird scaling prosper
  - slides are $96_{\text{mm}} \times {}_{128}$mm
  - text is 10-12pt on slide
  - slide itself magnified with Adobe Reader/xpdf/gv to fill screen
- pgf graphics framework easy to use
- include external JPEG/PNG/PDF figures
- output directly to pdf: no PostScript hurdles
- detailed User's Manual (with good presentation advice, too)

Theorems and Proofs

The proof uses *reductio ad absurdum.*

Theorem

There is no largest prime number.

Proof

- Suppose p were the largest prime number.

Theorems and Proofs

The proof uses *reductio ad absurdum.*

Theorem

There is no largest prime number.

Proof

- Suppose p were the largest prime number.
- Let q be the product of the first p numbers.

Theorems and Proofs

The proof uses *reductio ad absurdum.*

Theorem

There is no largest prime number.

Proof

- Suppose p were the largest prime number.
- Let q be the product of the first p numbers.
- Then $q + 1$ is not divisible by any of them.

Theorems and Proofs

The proof uses *reductio ad absurdum.*

Theorem

There is no largest prime number.

Proof

- Suppose p were the largest prime number.
- Let q be the product of the first p numbers.
- Then $q + 1$ is not divisible by any of them.
- But $q + 1$ is greater than 1, thus divisible by some prime number not in the first p numbers.

Blocks

Normal block

A set consists of elements.

Alert block

$2 = 2$.

Example block

The set $\{1, 2, 3, 5\}$ has four elements.

Outline

Beamer Features

Some of Gary's Examples

Other Features

Structural Features

More Features

Blocks

Appendix

Backup Slides

Details

Text omitted in main talk.

More details

Even more details