

# 550.400: Mathematical Modeling and Consulting

Lecture Notes

**Instructor:**

Dr. N. H. Lee

JHU AMS 2012 FALL

Last Compiled on September 19, 2012

# Outline

[Lesson Plan](#)

[Class Info.](#)

[Textbook](#)

Writing about Numbers

Math. Modeling

[Example](#)

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

[Tutorials](#)

LATEX

Git

Vim

R

[Exercises](#)

[Project](#)

Work Statement

Glossary

## Today's Lesson Plan

1. Share Your Favorites: WMA Chapter 2
2. Share Your Favorites: IMM Chapter 1
3. Share Your Project Idea & Who would be your client/sponsor?
4. Is Tennis Fair?
5. Work Statement Example
6. How to sync two git local folders

# Resume HW

## Common Problems:

- Liked “quotes”
- 1 page only means 1 page only
- Use the template! mean use the template
- Typos; .edu  $\neq$  .eud
- No need for GPA
- No facebook link
- Use a darker color

# Syllabus

- Grade Policy
- Attendance
- *Tentative Schedule*
- Blackboard
- Misc.

# OH Location

Clark Hall 320B

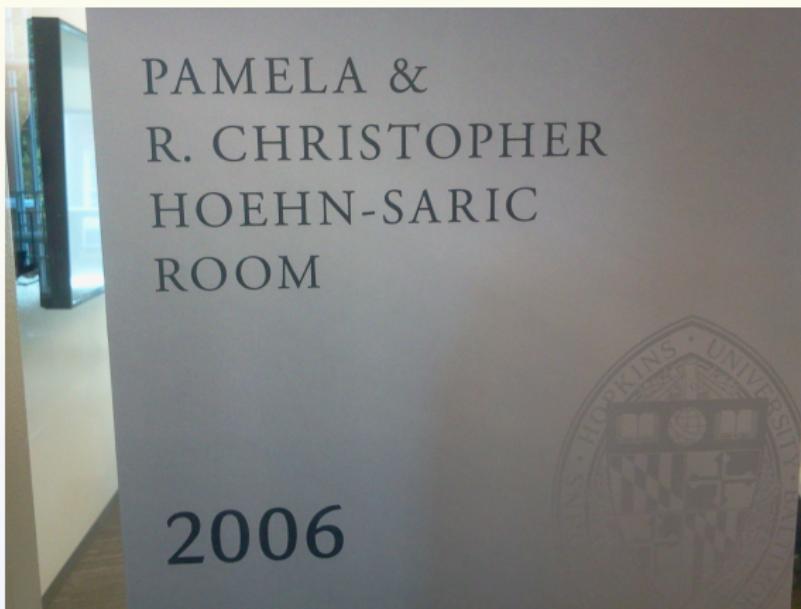


# Course Book Reserve

JHU Library Reserve Service

# Presentations in this class

For your presentation recording needs



# Unofficial Way to Access the Course Folder

<http://cis.jhu.edu/~nhlee/550400.html/>

# Outline

Lesson Plan

Class Info.

Textbook

Writing about Numbers

Math. Modeling

Example

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

Project

Work Statement

Glossary

# Seven Basic Principles

1. Set the context
2. Choose effective examples and analogies
3. Choose vocabulary to suit your readers
4. Decide whether to present #s in text, tables, or figures
5. Report and interpret #s in the text
6. Specify the direction *and* size of an association between variables
7. For many #s, summarize overall pattern

# Outline

Lesson Plan

Class Info.

**Textbook**

Writing about Numbers

Math. Modeling

Example

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

Project

Work Statement

Glossary

# Models and Reality: “Disclaimer”

*Here we are concerned exclusively with mathematical models, that is, models that mimic reality by using the language of mathematics. Whenever we use “model” without a modifier, we mean “mathematical model.”*

# Models and Reality

*What makes Mathematical models useful? If we “speak in mathematics,”,*

- *We must formulate our ideas precisely and so are less likely to let implicit assumptions slip by,*
- *We have a concise “language” which encourages manipulation,*
- *We have a large number of potential theorems available,*
- *We have high speed computers available for carrying out calculations.*

# Properties of Models

*A mathematical model is an abstract, simplified, mathematical construct related to a part of reality and created for a particular purpose.*

*Since a dozen different people are likely to come up with a dozen different definitions, don't take this one too seriously;  
rather, think of it as a crude starting point around which to build your own understanding of mathematical modeling.*

# Properties of Models

*As far as a model is concerned, the world can be divided into three parts:*

1. *Things whose effects are neglected,*
2. *Things that affect the model but whose behavior the model is not designed to study,*
3. *Things the model is designed to study the behavior of.*

## Building a Model: “Disclaimer”

*Model building involves imagination and skill. Giving rules for doing it is like listing rules for being an artist; at best this provides a framework around which to build skills and develop imagination.*

*It may be impossible to teach imagination. I won't try, but I hope this book provides an opportunity for your skills and imagination to grow. With these warnings, I present an outline of the modeling process.*

# Building a Model

*With these warnings, I present an outline of the modeling process.*

1. *Formulate a problem*
2. *Outline the model*
3. *Is it Useful?*
3. *Test the model*

## Building a Model

*Some models may require no data. If a model makes the same prediction regardless of the data, we are not getting something for nothing because this prediction is based on the assumptions of the model.*

*To some extent, the distinction between data and assumptions is artificial. In an extreme case, a model may be so specialized that its data are all built into the assumptions.*

# Building a Model

*The manager of a large commercial printing company asks your advice on how many salespeople to employ.*

*Qualitatively, more salespeople will increase sales overhead, while fewer salespeople may mean losing potential customers.*

*Thus there should be some optimum number.*

## IMM Problem: “Disclaimer”

*Some of the problems in this book lead you step by step through the development of a model and thus resemble the mathematics problems you have seen in other courses;*

*however, many problems are closer to real life: They are vaguely stated, have multiple answers (models), or are open ended.*

*I strongly recommend working in small groups on the problems to bring out various ideas and evaluate them critically.*

## Models and Reality

*The ultimate test of a model is how well it performs when it is applied to the problem it was designed to handle.*

*A model is used, it may lead to incorrect predictions. The model is often modified, frequently discarded, and sometimes used anyway because it is better than nothing. This is the way science develops.*

# Outline

Lesson Plan

Class Info.

Textbook

Writing about Numbers

Math. Modeling

Example

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

Project

Work Statement

Glossary

# What is Mathematical Modeling?

## Trillion Dollar Bet

### NOVA Online | Trillion Dollar Bet

[www.pbs.org/wgbh/nova/stockmarket/](http://www.pbs.org/wgbh/nova/stockmarket/)

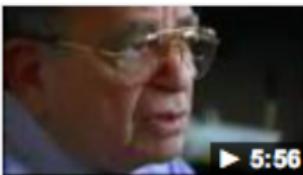
Welcome to the companion Web site to "**Trillion Dollar Bet**," originally broadcast on February 8, 2000. The film tells the fascinating story of the invention of the ...

[The Formula that Shook The ...](#) - [Transcripts](#) - [A Trader's Lexicon](#) - [Resources](#)

### Videos for trillion dollar bet - Report videos



[Trillion Dollar Bet 1 - YouTube](#)  
[youtube.com](http://youtube.com)  
Jan 8, 2009



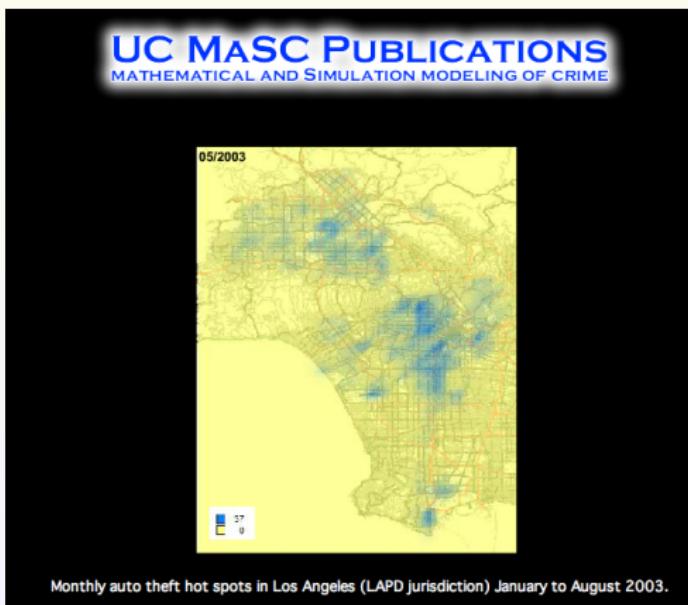
[The Trillion Dollar Bet - YouTube](#)  
[youtube.com](http://youtube.com)  
Sep 15, 2007



[The Midas Formula Stockmarket ...](#)  
[youtube.com](http://youtube.com)  
Aug 23, 2011

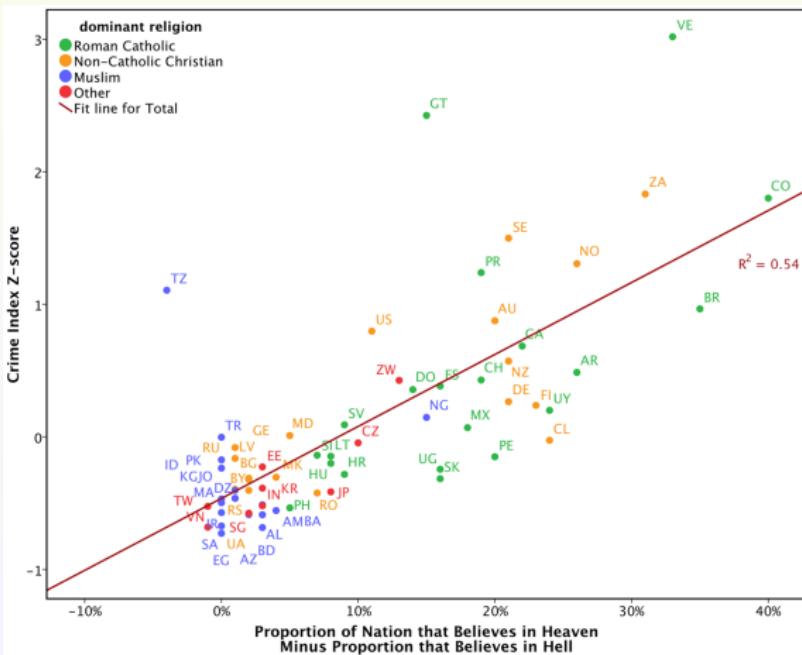
# What is Mathematical Modeling?

## LAPD Fighting Crime with Math



# What is Mathematical Modeling?

Crime rates and religious beliefs



## More Project Ideas

<http://www.stat.berkeley.edu/>

<http://www.math.msu.edu/>

<http://www.mathgoespop.com/>

<http://www.math.hmc.edu/clinic/>

# Outline

Lesson Plan

Class Info.

Textbook

Writing about Numbers

Math. Modeling

Example

Random Bits

**Insurance Redlining**

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

Project

Work Statement

Glossary

# Insurance Redlining

## Insurance Redlining

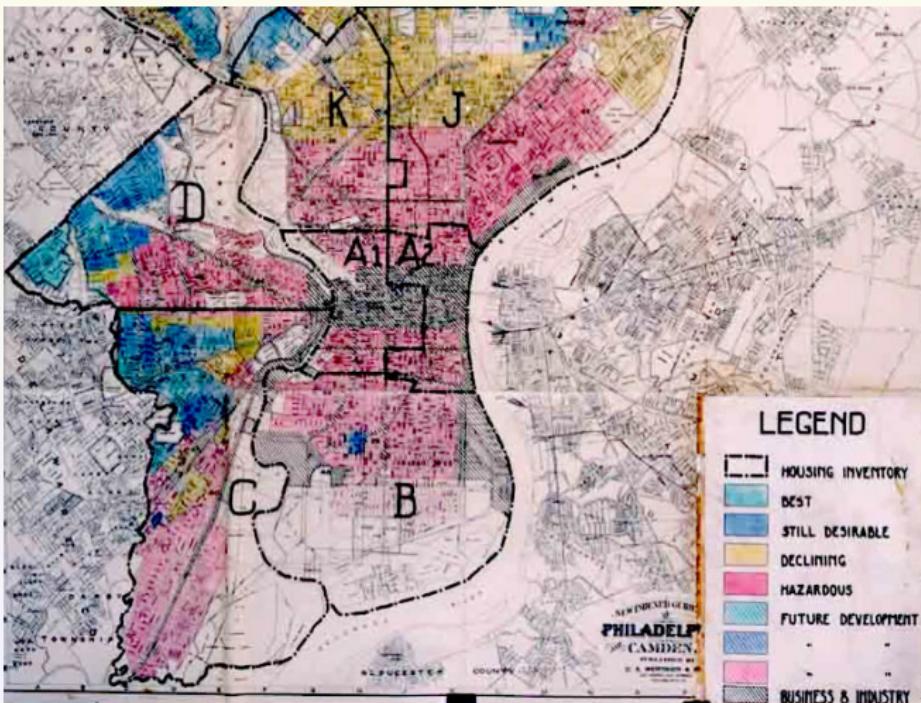
*Insurance redlining* refers to the practice of refusing to issue insurance to certain types of people or within some geographic area.

## FAIR

The *FAIR* plan was offered by the city of Chicago as a default policy to homeowner who had been rejected by the voluntary market.

# Insurance Redlining

## Insurance Redlining



# Insurance Redlining

## Sponsor

The U.S. Commission on Civil Rights examined charges by several Chicago community organizations that insurance companies were redlining their neighborhoods.

## Data

The number of FAIR plan policies written and renewed in Chicago by zip code for the number of months of December 1977 through May 1978.

# Insurance Redlining

Variables to consider:

`race` Racial composition in percentage of minority,

`fire` Fire per 100 housing units,

`theft` Theft per 1000 population,

`age` percent of housing unit built before 1939,

`involact` New FAIR plan policies and renewal per 100 housing units,

`income` Median family income in thousands of dollars,

`side` North or South side of Chicago.

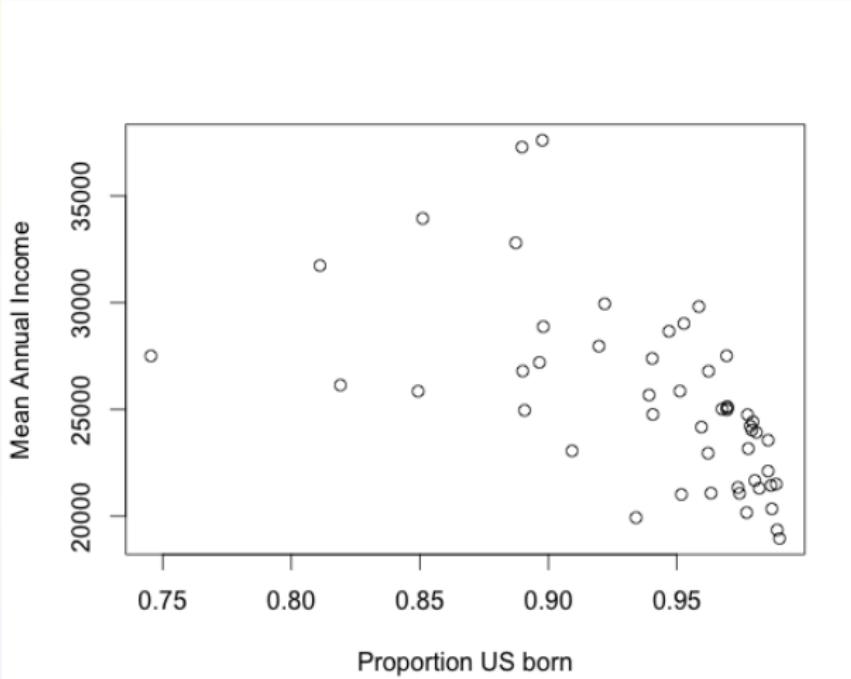
# Insurance Redlining: Ecological Fallacy

## Ecological Fallacy

*When data are collected at the group level, we may observe a correlation between two variables. The **ecological fallacy** is concluding that the same correlation holds at the individual level.*

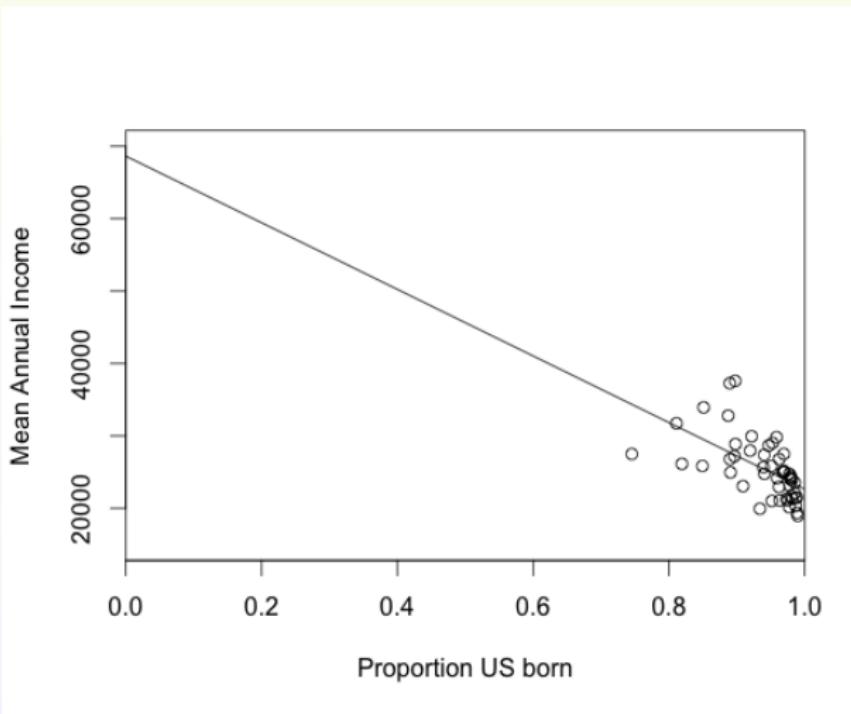
# Insurance Redlining: Ecological Fallacy

1998 annual per capita income and proportion U.S. born for 50 states plus D.C.



# Insurance Redlining: Ecological Fallacy

1998 annual per capita income and proportion U.S. born for 50 states plus D.C.



## Insurance Redlining

*For the ecological fallacy example, the assumption would be that the incomes of the native born do not depend on the proportion of native born within the state (and similarly for naturalized citizens).*

*For the insurance redlining example, we only have aggregate data. We must inform the sponsor that unless more detailed data becomes available, the results for the aggregated data may not hold true at the individual level.*

## Work Statement: Introduction

The work statement should contain a short description of your sponsor.

For the insurance redlining example, *U.S. Commission on Civil Rights* would be the sponsor.

*Boilerplating* from the sponsor's webpage is often acceptable.

<http://www.usccr.gov>

## Work Statement: Problem Statement

*Can the insurance companies claim that the discrepancy is due to greater risks in some zip codes?*

*The insurance companies could claim that they were denying insurance in neighborhoods where they had sustained large fire-related losses and any discriminatory effect was a by-product of legitimate business practice.*

# Outline

Lesson Plan

Class Info.

Textbook

Writing about Numbers

Math. Modeling

Example

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

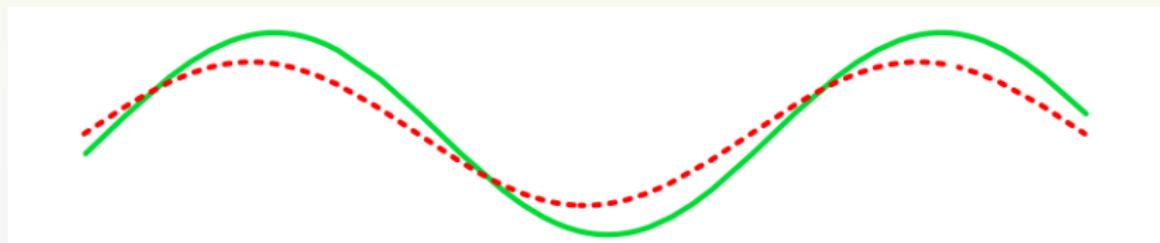
Project

Work Statement

Glossary

# Sherlock Holmes and the Bicycle Tracks: Problem Statement

Which one is the front wheel?



# Sherlock Holmes and the Bicycle Tracks

*"This track, as you perceive, was made by a rider who was going from the direction of the school."*

*"Or Toward it?"*

*"No, no, my dear Watson. The more deeply sunk impression is, of course, the hind wheel, upon which the weight rests. You perceive several places where it has passed across and obliterated the more shallow mark of the front one. It was undoubtedly heading away from the school."*

– *The Adventure of the Priory School* by Arthur Conan Doyle

# Sherlock Holmes and the Bicycle Tracks

$$f_x(t) = r_x(t) + \frac{L}{\sqrt{1 + (r'_y(t)/r'_x(t))^2}}$$

$$f_y(t) = r_y(t) + \frac{Lr'_y(t)/r'_x(t)}{\sqrt{1 + (r'_y(t)/r'_x(t))^2}}$$

# Outline

Lesson Plan

Class Info.

Textbook

Writing about Numbers

Math. Modeling

Example

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

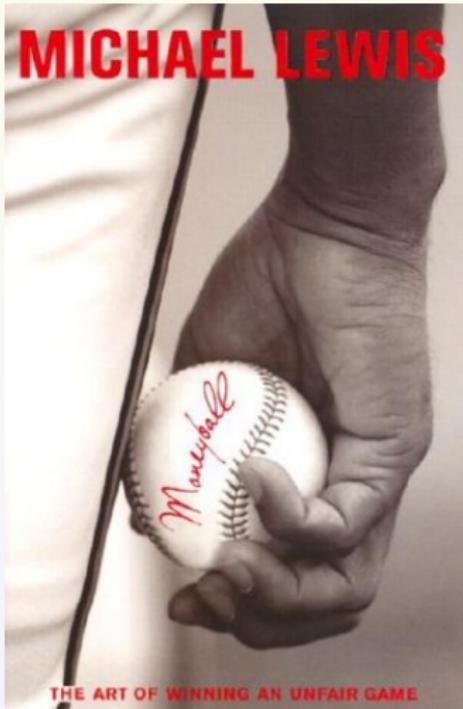
Project

Work Statement

Glossary

# Is a Sport Game Fair?: Problem Statement

How can we decide if a game with two competitors is fair?



# Is a Tennis Match Fair?

One simple answer is:

*If the roles of the competitors are reversed, their probability of winning does not change.*

Isn't that always true? No. For example, going first may give a player an advantage or disadvantage.

# Is a Tennis Match Fair?: Problem Statement

How can we decide if a game with two competitors is fair?



# Is Tennis Fair?

Review of our work done last time.

- What is tennis (for our mathematical modeling purpose)?
- What was assumed to be ignored?
- How was the server's advantage modeled?
- What did we do to simplify our model?
- Is there a “generalized” rule that models the tennis set winning determination?
- Who could be your sponsor/client? (Try USTA)

# Programmings in this class

- **LATEX :**
  - moderncv
  - beamer
  - report
  - pgf/TikZ
- Git
  - git gui
- R:
  - lm
  - ggplot2
  - tikzDevice
  - R CMD build

# Where to get some help for $\text{\LaTeX}$

<http://en.wikibooks.org/wiki/LaTeX/>

# Tutorial: L<sup>A</sup>T<sub>E</sub>X

L<sup>A</sup>T<sub>E</sub>X is a computer language for writing a scholarly paper:

Table: HTML vs L<sup>A</sup>T<sub>E</sub>X

	HTML	L <sup>A</sup> T <sub>E</sub> X
Code		
1	<html>	\begin{document}
2	. . .	. . .
3	</html>	\end{document}
Compiler	Firefox and etc.	pdflatex and etc.
Output	Web-page	PDF file

# Tutorial: $\text{\LaTeX}$

TeXworks is:

- an editing tool that is separate from  $\text{\LaTeX}$ ,
- available in Linux, OSX and Windows,
- available in:

<http://code.google.com/p/texworks/>

# Tutorial: $\text{\LaTeX}$

- Demo on preparing a resume using  $\text{\LaTeX}$  `moderncv` package:
  - Install  $\text{\LaTeX}$  (MikTeX in Windows and MacTeX in OSX),
  - Download `moderncv` package files from the course folder,
  - Change file names to reflect you,
  - Edit the TeX file,
  - Compile using your favorite  $\text{\LaTeX}$  editor,
  - Look at the resulting PDF file.

# Tutorial: L<sup>A</sup>T<sub>E</sub>X

Typing mathematics in L<sup>A</sup>T<sub>E</sub>X:

- 1 Hello  $\int_0^1 \sin(x) dx$  World
  - 2 \vskip0.5in
  - 3 Hello \$\$\int\_0^1 \sin(x) dx\$\$ World
- 

Hello  $\int_0^1 \sin(x) dx$  World

Hello

$$\int_0^1 \sin(x) dx$$

World

# Cautions: L<sup>A</sup>T<sub>E</sub>X

There are numerous quirky L<sup>A</sup>T<sub>E</sub>X rules:

- opening quotation is not the same as the closing quotation,
- period yields *two* blank spaces,
- for %, need to type \%,
- for \, need to type \textbackslash,
- for /, need to type /,
- for {, need to type \{,
- for \$, need to type \\$,
- ~ yields a single blank space,
- and etc.

# Tutorial: L<sup>A</sup>T<sub>E</sub>X

How to make slides with notes:

```
1 \usepackage{handoutWithNotes}
2 \pgfpagesuselayout{4 on 1 with notes}[a4paper, border shrink=5mm]
```

---

This will

- put four scaled-back slides in one page with ruled-note space on the right
- make the “hyper links” invalid

# The place to get some Git helps

<http://git-scm.com/doc/>

# Tutorial: Git

```
1 sudo apt-get install git
```

---

An alternative: git gui

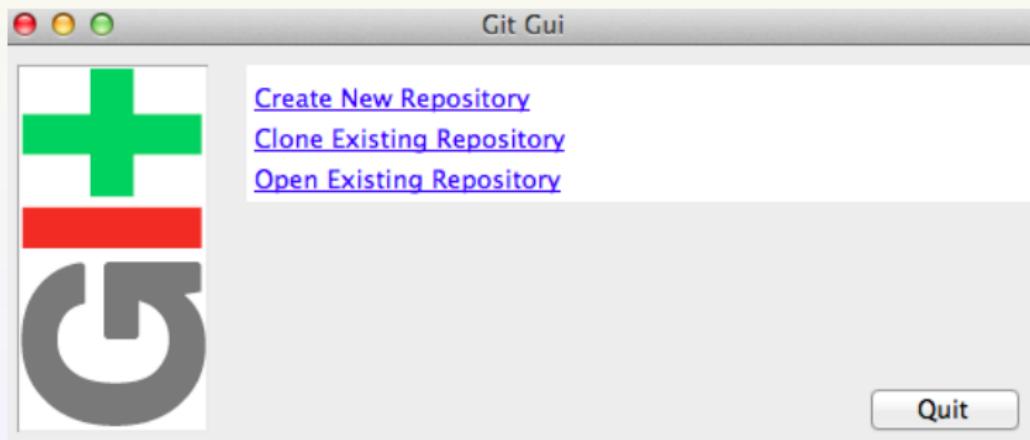
The screenshot shows the official Git website's "GUI Clients" section. At the top, there is a large red diamond icon containing a white stylized 'g' and the word "git" in a bold, lowercase sans-serif font. Below this, the slogan "distributed-is-the-new-centralized" is written in a smaller, gray, monospace-style font. On the left side, there is a sidebar with links: "About", "Documentation", "Downloads" (highlighted in red), "GUI Clients" (also highlighted in red), and "Logos". On the right side, the main content area has a title "GUI Clients" in large, bold, dark gray letters. Below the title, there is a paragraph of text: "Git comes with built-in GUI tools for committing changes, and there are many third-party tools for users looking for platforms like GitHub or Bitbucket." At the bottom of the main content area, there is a button labeled "Show GUIs for all OSes" and a link "7 Mac GUIs available".

# Tutorial: Git

```
1 cd ~/  
2 git clone http://cis.jhu.edu/~nhlee/550400.git
```

---

An alternative: git gui

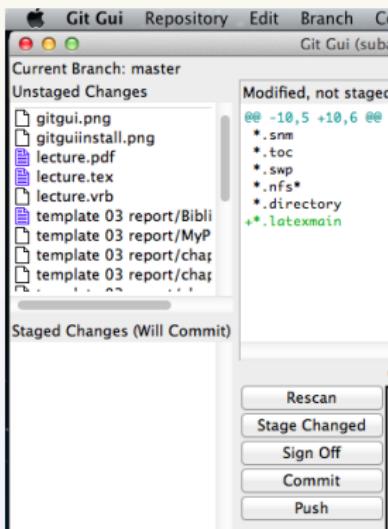


# Tutorial: Git

```
1 cd ~/550400  
2 git reset --hard HEAD  
3 git pull origin master
```

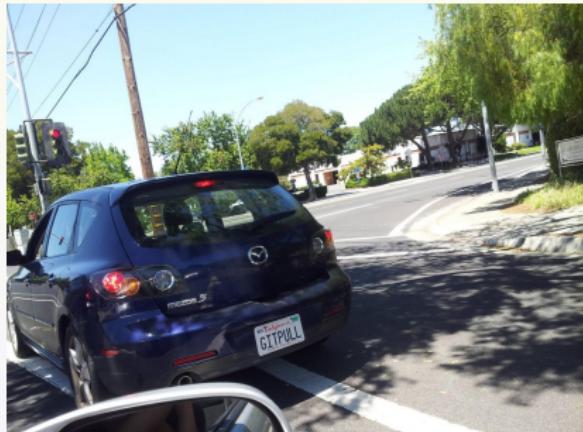
---

An alternative: git gui



# Tutorial: Git

After years of using git, you might find this funny:



1 git pull origin master

# Tutorial: Git

After years of using git, you might find this funny:



1 git push origin master

---

# Tutorial: Git

For \$19.99, you can also have  
your own:



```
1 cd ~/
2 mkdir hub.git
3 mkdir computerA.git
4 mkdir computerB.git
5
6 git init --bare hub.git
7
8 cd hub.git
9 cd hooks
10 cp post-update.sample post-update
```

---

# Tutorial: Git

```
1 cd computerA.git          1 cd computerB.git
2 git init                   2 git init
3 git remote add origin ~/hub.git
4 echo 'Hello A' >> commonfile.txt
5 git add commonfile.txt
6 git commit -am 'from A'
7 git pull origin master
8 git push origin master      1 cd computerB.git
                                2 git init
                                3 git remote add origin ~/hub.git
                                4 echo ' World B' >> commonfile.txt
                                5 git add commonfile.txt
                                6 git commit -am 'from B'
                                7 git pull origin master
                                8 git push origin master
```

---

# Tutorial: Git

```
1 cd ~/550400
2
3 git gui
4 git reset --hard HEAD
5
6 git branch personal
7 git branch
8 git checkout personal
9
10 edit some file
11 git status
12 git add .
13 git commit -am 'personal edit'
14
15 git checkout master
16 git branch -D personal
```

---

- checks if there has been any change to the folder
- build and update the master git branch
- create and update a personal git branch

# Tutorial: Git

## .gitignore?

- N.B. the course folder already has one
- Use it to let *git* know the files to *ignore* while version controlling
- one particular usage: create `.gitignore` at the root of your git folder
- files already been list under the git watch list will not be ignored even after creation of `.gitignore`

# Vim

## Vim

*Vim is a highly customizable text editor*

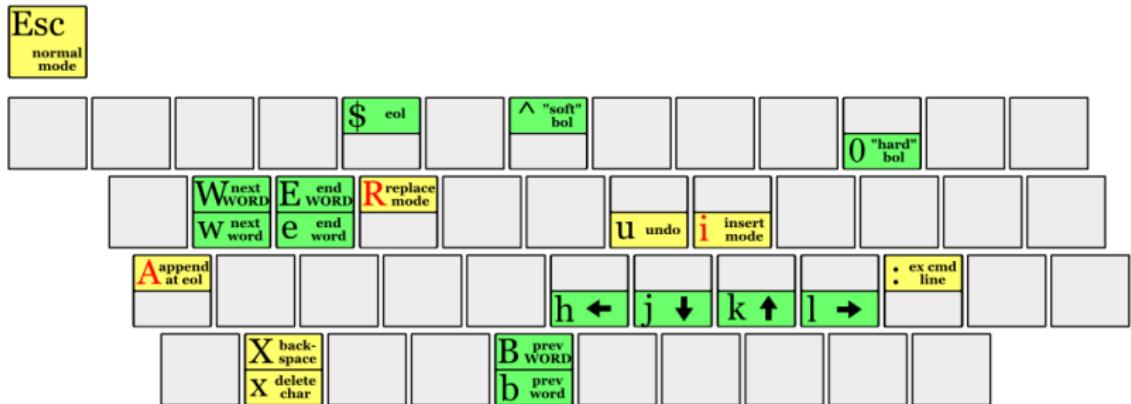
1. L<sup>A</sup>T<sub>E</sub>X, R, C/C++, Java, Python, Git and etc.
2. Regular expression, syntax coloring, autocompletion
3. Try Firefox + Wasavi/Vimperator/Vimium
4. <ESC>-mode
  - :-mode, aka., the last line mode
  - i-mode, aka., the insert mode

# Tutorial: Vi

version 1.1  
April 1st, 06

## vi/vim lesson 1 - basic editing

**motion** moves the cursor, or defines the range for an operator  
**command** direct action command, if red, it enters insert mode



### Basics:

**h j k l** are vi/vim cursor keys – use them as they are much closer than regular cursor keys!

Use **i** to enter insert mode, cursor turns from a block into a vertical line, and you can type in text. Use **Esc** to return to normal mode.

Use **x** to delete the current character, or **X** to delete the one to the left

Use **A** to go insert text at the end of the line (wherever you are in the line!)

(Note: insert mode is actually very similar to a regular editor, you can use cursor/navigation keys, backspace, delete...)

### Extras:

**u** to undo the last action – traditional vi has a single level, while vim supports unlimited undo (CTRL - **R** to redo)

**0** jumps directly to the beginning of the line, **\$** to the end, and **^** to the first non-blank

Use **w b e** to move along 'words'. A 'word' is a sequence of all alphanumeric or punctuation signs: **quux(foo, bar, baz)**

Use **W B E** to move along WORDs. A 'WORD' is a sequence of any non-blank characters: **quux(foo, bar, baz)**

Use **R** to enter insert mode with an overstrike cursor, which types over existing characters.

**:** **w** and press enter to save, **:q** and enter to quit.

# Tutorial: Vi

version 1.1  
April 1st, 06

## vi/vim lesson 2 - operators & repetition

  learned in previous lessons  
  motion moves the cursor, or defines the range for an operator  
  command direct text entry command.  
  operator requires a motion afterwards, operates between cursor & destination



### Basics:

**W**, followed by another key, moves the cursor to the next instance of that character on the current line, **F** does the same backwards.

**1** and **T** do the same, but they stop right before the character.

**d** (delete), followed, by any motion deletes the text between the cursor and that motion's destination (**d w**, **d f**, **d -**...).

**C** (change) does the same, but leaves you in insert mode.

Some motions, such as **h** and **k**, are linewise – deletion includes the full start/end lines.

**.** repeats the last editing action: text input, delete or change, etc... motion is recalculated at the new place.

### Extras:

Prepend a count to any command/motion to repeat it that number of times:

**1 2 W** to delete up to the second word.

**2 1 T** to delete up to but not including the second comma.

**2 i** repeats the text after you press (Esc) to finish the input session.

Repeat operator (**c c** or **d d**) to operate on the current line.

Only in vim, **V** enters visual mode. Move around with motions, the text will be highlighted. Press an operator to operate on that selection.

**V** enters visual-lines mode – like **V**, but selecting whole lines.

CTRL - **V** selects rectangular blocks.

# Tutorial: Vi

version 1.1  
April 1st, 06

## vi/vim lesson 3 - yank & paste

<b>lesson</b>	learned in previous lesson
<b>motion</b>	moves the cursor, or defines the range for an operator
<b>command</b>	direct action command, if red, it enters insert mode
<b>operator</b>	requires a motion afterwards, operates between cursor & destination
<b>extra</b>	special functions, requires extra input



### Basics

Use **y** followed by any motion to 'yank' (copy).

Use **p** to paste after (if charwise, to the right, if linewise, below).

Use **P** to paste before.

**y** copies the current line.

**y** also works in visual mode.

Text deleted with **d**, **c**, **x**... is also copied!

### Extras

**"** and an **a** - **z** character before any yank/delete/paste command chooses a register.

An **A** - **Z** register before yank/delete means "append-copy".

**"\*** or **"+** select the system clipboard.

**o** enters insert mode in a new empty line below the current one.

**O** does the same above the current line.

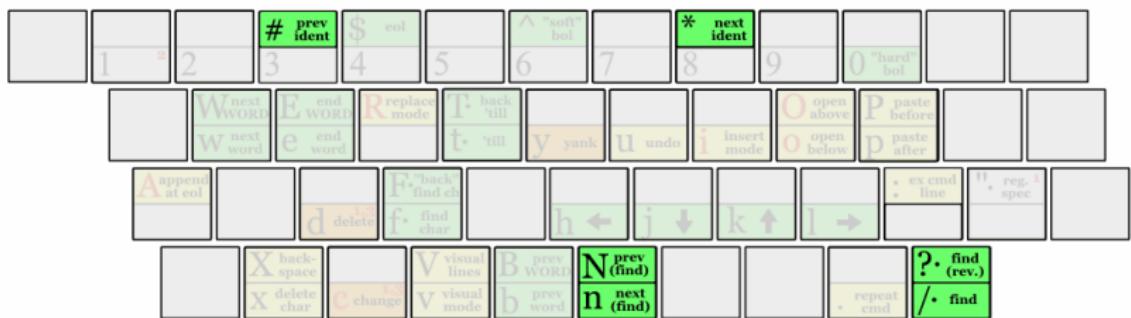
# Tutorial: Vi

version 1.1  
April 1st, 06

## vi/vim lesson 4 - searching

	learned in previous lessons
	moves the cursor, or defines the range for an operator
	direct action command, if <code>red</code> , it enters insert mode
	operator if <code>red</code> , it enters insert mode requires a motion afterwards, operates between cursor & destination
	extra special functions, requires extra input

**Esc**  
normal mode



### Basics:

is the basic search motion – type the text you are searching for after the slash, and then press return. Being a motion, you can use this after an operator, or in visual mode.

does the same, backwards.

repeats the last search in the same direction, repeats it in the reverse direction

Be careful, because the search target is interpreted as a regular expression: `a*b` means zero or more 'a's followed by a 'b', `^abc` means 'abc' at the beginning of a line, `[0-9]` looks for the next digit, etc...

### Extras:

The following very useful motions work only in vim:

searches forward for the next instance of the identifier under the cursor.

does the same backwards.

# Tutorial: Vi

version 1.1  
April 1st, 06

## vi/vim lesson 5 - marks & macros

<b>lesson</b>	learned in previous lesson
<b>motion</b>	moves the cursor, or defines the range for an operator
<b>command</b>	direct action command, if red, it enters insert mode
<b>operator</b>	requires a motion afterwards, operates between cursor & motion
<b>extra</b>	special functions, requires extra input



### Marks:

- Use **m** followed by an **a** - **z** character to set a mark.
- Use **g** followed by a character to go to that mark.
- Use **g** and a character to go to the first non-blank in that line.
- A**-**Z** marks are global, **a** - **z** per-buffer.
- ~** refers to the position of the last modification.

### Macros:

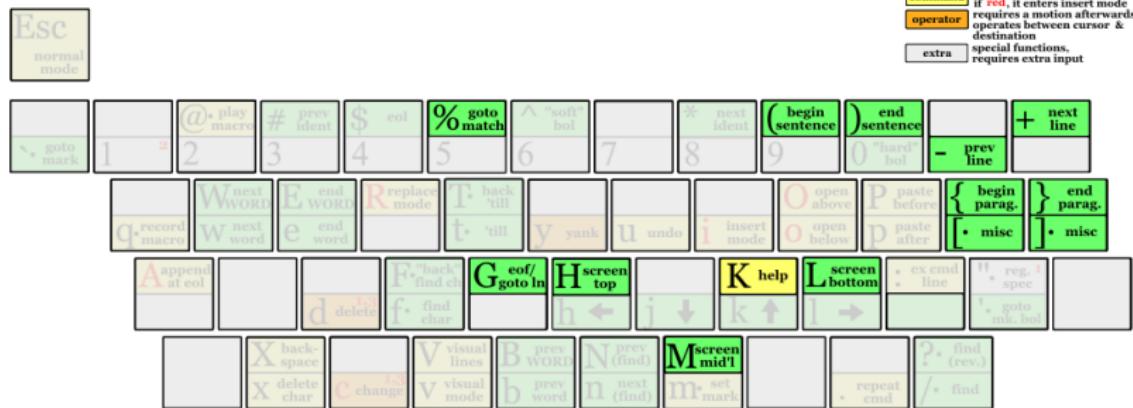
- Use **q** followed by an **a** - **z** character to start recording.
- Use **q** afterwards to stop recording.
- @** followed by a character replays that macro.
- @ @** to repeat the last macro played.

# Tutorial: Vi

version 1.1  
April 1st, 06

## vi/vim lesson 6 – various motions

	learned in previous lessons
<b>motion</b>	moves the cursor, or defines the range for an operator
<b>command</b>	direct action command.
<b>operator</b>	if red, it enters insert mode requires a motion afterwards, operates between cursor & destination
<b>extra</b>	special functions, requires extra input



**%** jumps between matching pairs of ‘‘()’’, ‘‘[]’’, ‘‘{}’’, etc...

**H M L** jump directly to the top/middle/bottom of the screen.

**G** jumps to the end of the file, or to the line # typed before it.

**-** / **+** jump to the previous/next line.

**K**, not technically a motion, jumps to the help for the word under the cursor: vim help, man page under unix, etc...

**C** and **D** jump to the beginning/end of the current sentence.

**C** and **Y** jump to the previous/next empty line.

**I I** jumps to the previous ‘‘I’’ in column 0.

**I I** jumps to the next ‘‘I’’ in column 0.

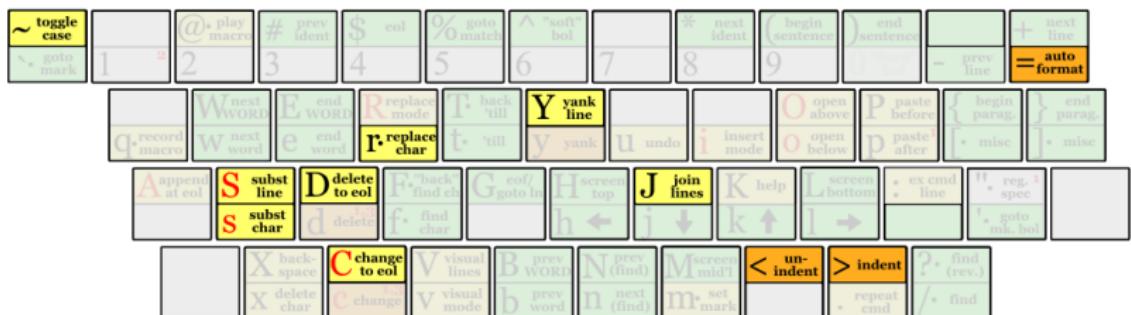
# Tutorial: Vi

version 1.1  
April 1st, 06

## vi/vim lesson 7 - various commands

	learned in previous lessons
	move to the cursor, or defines the range for an action
	direct action command.
	if red, it enters insert mode
	operator requires a motion afterwards, operates between cursor & destination
	extra special functions, requires extra input

**Esc**  
normal mode



### Basics:

**J** joins the current line with the next one, or all the lines in the current visual selection.

**r** followed by any character replaces the current character with that one.

**C** is shorthand for **c** **\$**, changes to end of line.

**D** is shorthand for **d** **\$**, deletes to end of line.

**Y** is shorthand for **y** **y**, yanks the whole line.

**S** deletes the character under the cursor and enters insert mode.

**s** clears the current line and enters insert mode.

### Extras:

**>** and a motion to indent one or more lines.

**<** and a motion to unindent.

**=** and a motion to reformat a range of text.

All of them work in visual mode, or can be repeated (**>** **>**, etc...) to operate on the current line.

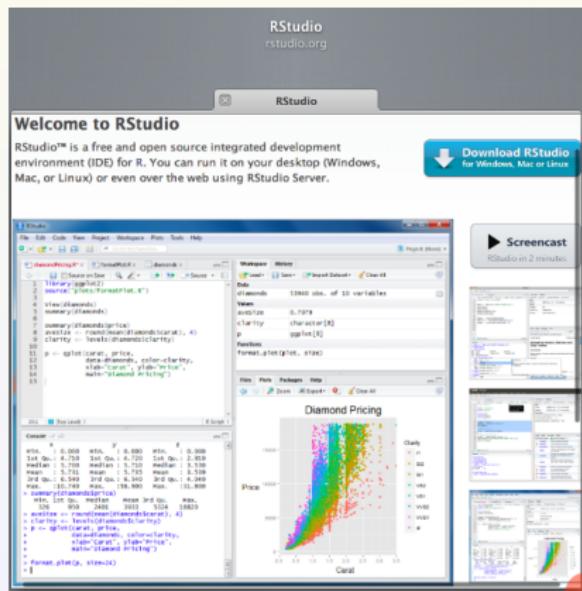
**-** toggles the case of the character under the cursor.

Now go grab the full cheat sheet and learn the rest.

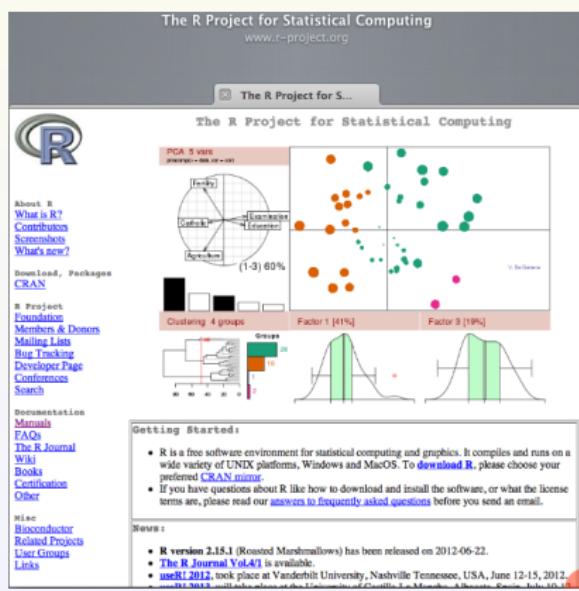
Start with **I** **a** **,** and **;**. Piece of cake!

# Demo: R + LATEX

R Studio



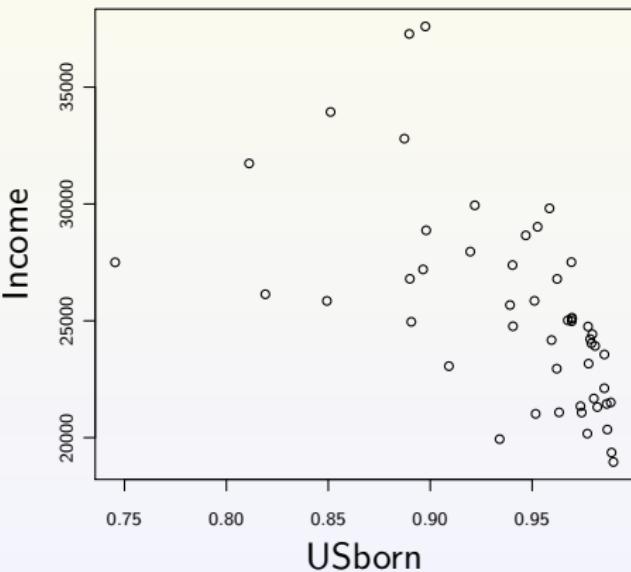
R



# Demo: R + $\text{\LaTeX}$

```
1 install.packages(faraway)
2 install.packages(tikzDevice)
3 require(faraway)
4 require(tikzDevice)
5 data(eco)
6 tikz('embeddedfig1.tex',
7       standAlone=F,
8       width=5,height=5)
9 plot(income ~ usborn,
10      data=eco,
11      xlab='Proportion US born',
12      ylab='Mean Annual Income',
13      )
14 dev.off()
```

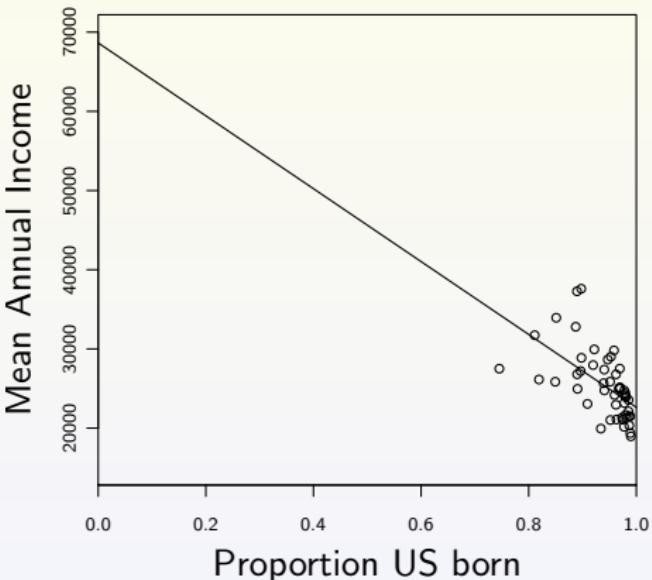
---



# Demo: R + L<sup>A</sup>T<sub>E</sub>X

```
1 tikz('embeddedfig2.tex',
2       standAlone=F,
3       width=5,height=5)
4 plot(income ~ usborn,
5       data = eco,
6       xlab='Proportion US born',
7       ylab='Mean Annual Income',
8       xlim=c(0,1),
9       ylim=c(15000,70000),
10      xaxs='i')
11 g<-lm(income~usborn,eco)
12 abline(coef(g))
13 dev.off()
```

---



# Outline I

Lesson Plan

Class Info.

Textbook

Writing about Numbers

Math. Modeling

Example

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

Project

# Outline II

Work Statement

Glossary

## WMA Problem 2.5a & 2.6a

*The Williams family's income of \$25,000 falls below 185% of the Federal Poverty Threshold for a family of four, qualifying them for food stamps.*

**Problem 2.5a** Identify terms that need to be defined or restated for a nontechnical audience

**Problem 2.6a** Rewrite the sentences in the previous questions for an audience with a fifth-grade education. Convey the main point, not the calculation or the jargon.

**FYI** Off-the-chart

## WMA Problem 2.8a

Rewrite each of these sentences to specify the direction and magnitude of the association:

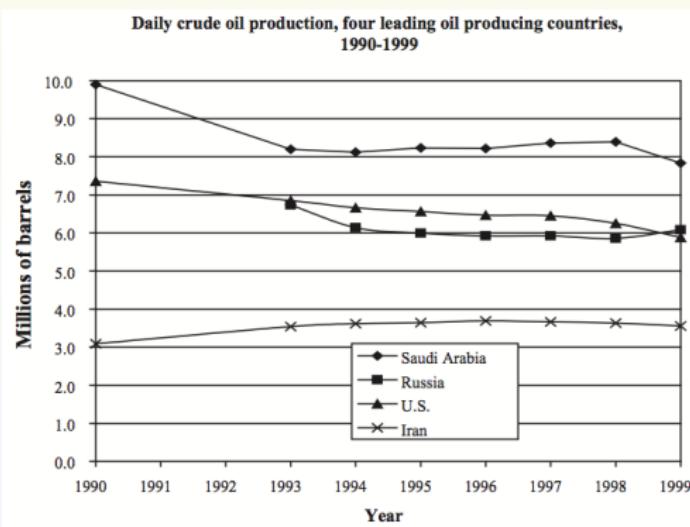
*In the United States, race is correlated with income.*

Table: Median income by race and Hispanic origin, United States, 1999

Race/Hispanic origin	Median Income
White	\$42,504
Black	\$27,910
Asian/Pacific Islander	\$51,205
Hispanic (can be of any race)	\$30,735

## WMA Problem 2.9

Use the GEE approach to describe the patterns in the figure below, including an introductory sentence about the purpose of the chart before summarizing the patterns.



## IMM Problem 1.1

Suppose people enter the elevators in a skyscraper at random during the morning rush. The result will be several elevators stopping on each floor to discharge one or two passengers each.

- Discuss schemes for improving the situation.
- How could improvement be measured?
- How could you model the situation to decide what scheme to adopt?

## IMM Problem 1.6

Unless you have been extremely lucky, you have had a large class in a poorly designed lecture hall.

- (a) What are some criteria to be considered in designing a large lecture hall?

## IMM Problem 1.6

Unless you have been extremely lucky, you have had a large class in a poorly designed lecture hall.

(b) One criterion is legibility of material written on the boards.

- Construct a model of legibility as a function of
  - *the distance* your seat is from the board
  - *the angle* at which you look at the board
- What will the curves of constant legibility look like on a floor plan?
- How can you test this prediction? Try it.
- Does this suggest shaping the back of the hall differently than is usually done? How?

## IMM Problem 1.6

Unless you have been extremely lucky, you have had a large class in a poorly designed lecture hall.

- (c) Can mathematical modeling help with any other criteria besides the one mentioned in (b)? Try to pick a criterion from among these possibilities and develop a model for it.

# Collect All: L<sup>A</sup>T<sub>E</sub>X + Git

## The Blind Men and the Elephant

*In-class Group Exercise (Scavenger hunt):*

- Start up a git folder,
- Create and edit the `.gitignore` file,
- Download the template for a beamer file,
- Look up the poem from the book,
- One slide per stanza,
- Use `verse` environment,
- Compile after each stanza,
- Commit after creating each stanza,
- Repeat until done.

# Outline

Lesson Plan

Class Info.

Textbook

Writing about Numbers

Math. Modeling

Example

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

Project

Work Statement

Glossary

# Mission Impossible?: an analogy

Mission Impossible Season 2 Episode (00:00 – 06:25)



# Project in Industry: Frequently Recurring Elements

A stylized timeline:

1. Work Statement,
2. Midterm Presentation,
3. Progress Report,
4. Final Presentation,
5. Final Report.



# Outline

Lesson Plan

Class Info.

Textbook

Writing about Numbers

Math. Modeling

Example

Random Bits

Insurance Redlining

Sherlock Holmes and the Bicycle Tracks

Fair Play

Tutorials

LATEX

Git

Vim

R

Exercises

Project

Work Statement

Glossary

# What is Work Statement

This is the written proposal and definition of the project and constitutes the team's "contract" with the sponsor. It should be approximately 2-5 pages long. It sets forth the nature of the project, the specific objectives of the project, the results expected, and the "deliverables" for the project. The scope of the project must be within the timetable for the program and that the deliverables are reasonable and appropriate; given the nature of research, it should not include promises that the team cannot be certain to achieve. It is ultimately given to the sponsor for review and signature.

# Template 1

1. Abstract
2. Background
3. Problem description
4. Approach (“time permitting” clause for some work)
5. Schedule (dates for completing milestones and tasks and for deliverables)
6. Milestones (major checkpoints your team will use to stay on track)
7. Deliverables (specific work products you will deliver to the sponsor)

# Templates 2

1. Introduction
2. Problem background
3. Mathematical background
4. Computing background
5. Possible solutions and project objectives
6. Deliverables (“time permitting” clause for some work)
7. Timeline

## Template 3

1. Project background
2. Goals (major direction you see the work aimed at, not necessarily what you bid to do)
3. Proposed mathematical approach
4. Objectives (specific aims of your project, and schedule of results you expect to achieve)
5. Optional objectives
6. Deliverables
7. Milestones
8. Work flowchart
9. Schedule

# Template 4

1. Abstract
2. Problem background
3. Problem description
4. Approach
5. Deliverables
6. Timetable
7. Team members

# Work Statement

In the initial segment (“Abstract”, “Introduction”, “Background”)

- Brief description of the company
- Major product lines(s)
- A brief (abstract) description of the project

# Work Statement

## Throughout

- Spell out terminology – avoid undefined jargon or acronym
- When options must be resolved, give dates by which they must be resolved
- Give modest objectives, not boastful ones

# Work Statement

List of deliverables should include

- Site visits (to be arranged)
- Midterm oral presentation
- Midterm report
- Final presentation
- Final report
- Software (if appropriate)
  - Specify sponsor-approved OS, platform
  - Documentations

# Work Statement

Work Statement Examples

See *Protein Pathways* Project Work Statement

# Glossary I

## GOAL

The overall, long range, end result that your research is aimed at, what you are trying to achieve ultimately. Stating a goal does not mean you believe you will get there this time around. It is the grand view towards which you strive. The goal of AIDS research is to find a cure for AIDS.

# Glossary

## OBJECTIVES

The specific things you will try to achieve in your project, the immediate targets of your research. Your objectives spell out how you have parsed the problem of heading towards the goal into smaller pieces that you will work on. The objectives set practical limits on your work. They point to where the project can reasonably expect to wind up. It should be clear that the objectives fit into and work towards the long-range goal.

# Glossary

## TASKS

These are the specific things you will do in order to achieve your objectives. The tasks drive your determination of what skills and other resources (such as data, software, hardware, written materials, work environment) will be needed for your project. If among the resources needed are ones that must be supplied by the sponsor, then you will need to specify these items in your Work Statement.

# Glossary

## DELIVERABLES

The things you promise to deliver to the sponsor. For a project, these include a mid-term and final report, a mid-term presentation and a final presentation on Projects Day. They may also include site visits to the sponsor (usually one near the beginning of the project to get acquainted with the sponsor, and one after Projects Day to present the work at the sponsor's location), software, perhaps hardware in some cases, written results of literature searches, white papers (i.e., written background information on such things as plans, methods or concepts prepared for internal use), etc. These additional items are to be decided by you in consultation with your sponsors mentor.

## Glossary

### MILESTONES

A list of specific accomplishments that you may use to mark progress and maintain pace and coordination within your project. They are used to help your team stay on track and to determine the success of a chosen line of attack on your problem. Milestones may or may not be included in your Work Statement, but you should definitely think these through for your own use as you plan your project and Work Statement. They are check-points for you (and for your sponsor, if they are included in the Work Statement), not necessarily deliverables. You may want to specify major milestones in your Work Statements to indicate what you would do if your research leads to the conclusion that some objective cannot be accomplished. For example, "if by such a date we have found it impossible to achieve X, then we will begin Y." Research is exploration of the unknown, so you may encounter an intractable obstacle and need to work around it. You can't know everything ahead of time. Give some thought to this and try to allow for milestones by which you can judge where you are and what you need to do to proceed effectively in the event you don't meet a milestone.

# Glossary

## SCHEDULE

This specifies when you will finish major parts of your research and provides a timetable for completion of deliverables. Internally, you should maintain as fine-grained a schedule as you need to keep your team coordinated and on track, but in your Work Statement it is best to make the schedule and list of deliverables as modest as the sponsor will allow.