- Relate for both the simulation and resampling several points
- Relate the simulation sampling to P(D| restricted models).
- For true p-values you are simulating or sampling under restricted model but fitting under full

Monte Carlo Simulations for Inference and Power analyses

October 9th 2014

Readings.

- There are three PDF's on the ANGEL site in the simulation/monte carlo folder (resources sub-folder).
- These provide some very deep insight into thinking about power.
- For deeper insight into Monte Carlo methods, link to book in the same folder.

Goals

- Develop a monte carlo approach for making inferences, such as P-values and confidence intervals.
- Learn how to write R scripts to perform these actions.
- Begin discussing the concept of power.

Groups

A -
Andrew
Colleen
Alejandro
Eleanor

B-
Emily
Sahar
Carina
Zachary
Thomas

C-
Maria Natalia
Klara
Patric
Jie

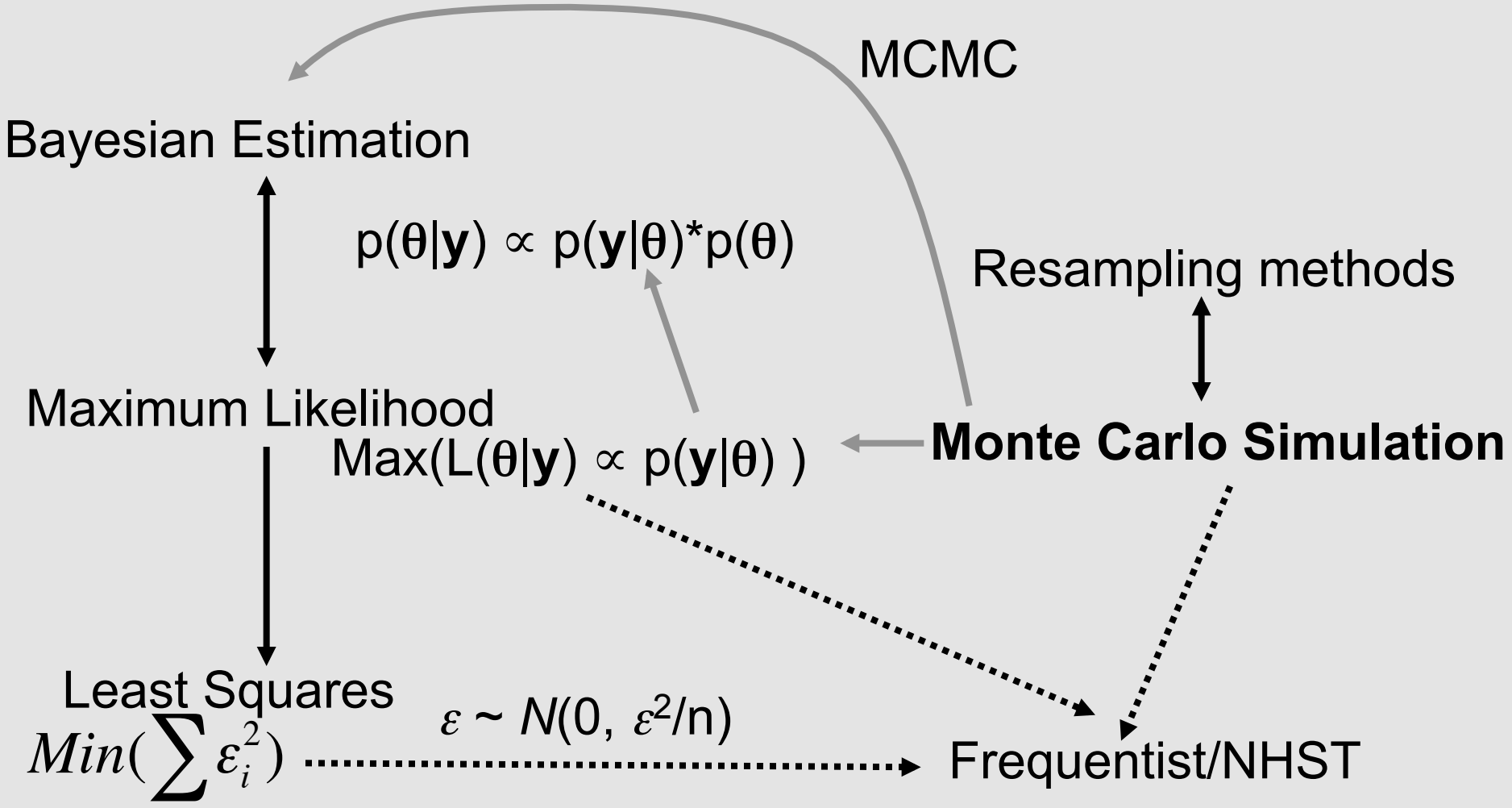
D-
Prateek
Sam
Danielle
Alita
David

E-
Kileigh
Carina
Jay
Kevin

Group Roles

- Role 1- Makes sure everyone participates/speaks up (point if you have to).
- Role 2- Synthesis and scribe
- Role 3- News Anchor
- Role 4 - critic

Relationship between Estimation/inference methods



What is monte Carlo simulation?

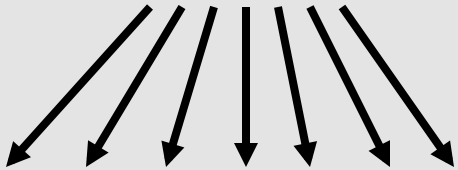
Why might we want to do simulations for statistics?

What sorts of applications might it have?

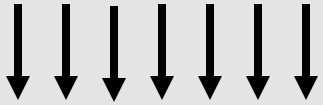
Sampling distributions

Repeated sampling

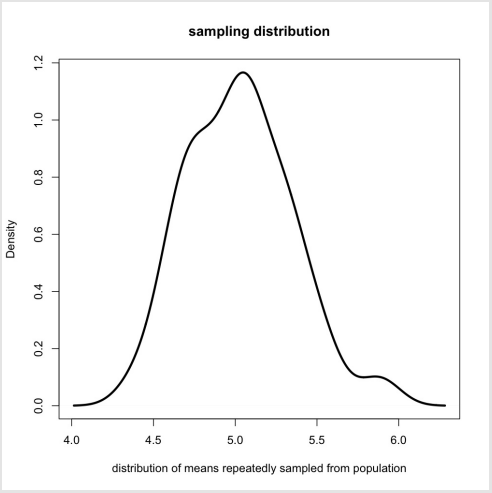
Population parameters
(True, fixed)



Sample statistics



Sampling distribution



So what do we need to know and what to assume?

1)What are we trying to simulate?

2)What do we know (what information do we have)?

3)What do we need to assume?

So what do we need to know and what to assume?

1)What are we trying to find?

- Simulated values of our response (simulated y values).

So what do we need to know and what to assume?

What do we know:

- We have observed values for our predictors.
- Observed values for our response.
- Arguably we know the structure of the model itself.

So what do we need to know and what to assume?

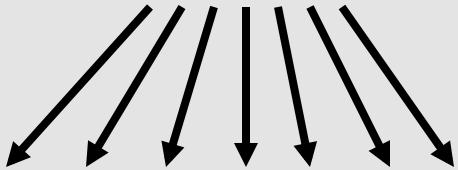
What do we need to assume for the simulation?

- parameter values (β_0, β_1,…, σ^2)
- form of the model,
- Distribution(s) of unexplained (“error”) variance.

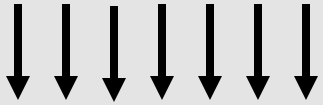
Sampling distributions

Repeated sampling

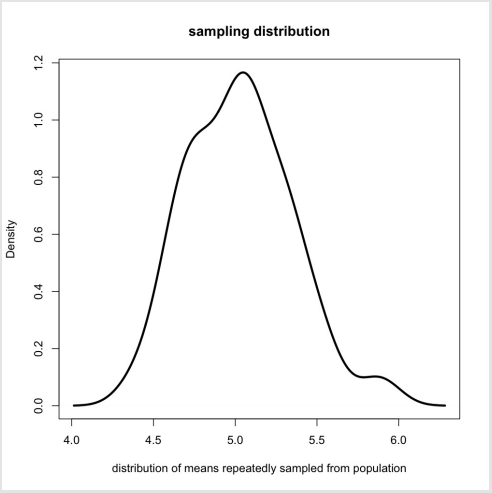
Population parameters
(True, fixed)



Sample statistics



Sampling distribution



Going in reverse

- Essentially we are working through the same sampling idea as on the previous slide, just in reverse.

Using Monte Carlo simulation to generate values for a linear regression.

- Start with an imaginary sample (n=100 observations, with a set of x values that are known (or simulated):

$$y_1 \sim \beta_0 + \beta_1 x_1 + \varepsilon_1$$

$$Y \sim N(\beta_0 + \beta_1 X, \sigma^2) = N(X\beta, \sigma^2)$$

Using Monte Carlo simulation to generate values for a linear regression.

$$y_1 \sim \beta_0 + \beta_1 x_1 + \varepsilon_1$$

$$Y \sim N(\beta_0 + \beta_1 X, \sigma^2) = N(X\beta, \sigma^2)$$

- How might you use simulations to generate data from such a model?
- What do you need? What do you know?

Using Monte Carlo simulation to generate values for a linear regression.

$$y_1 \sim \beta_0 + \beta_1 x_1 + \varepsilon_1$$
$$Y \sim N(\beta_0 + \beta_1 X, \sigma^2) = N(X\beta, \sigma^2)$$

- How might you use simulations to generate data from such a model?
- What do you need? What do you know?

- Make a step-by-step list of what you need to do to simulate the data.

Using Monte Carlo simulation to generate values from a simple linear model. STEPS:

- Start with an imaginary simple (n=100 observations) example: $Y \sim N(\beta_0 + \beta_1 X, \sigma^2)$
- Generate parameter values (intercept and slope).
- Add uncertainty (based on some distribution)
- Simulate values for Y

Using Monte Carlo simulation to generate values from a simple linear model. STEPS:

- Start with an imaginary simple (n=100 observations) example: $Y \sim N(\beta_0 + \beta_1 X, \sigma^2)$

Using Monte Carlo simulation to generate values from a simple linear model.

- (n=100) example:

$$y_1 \sim \beta_0 + \beta_1 x_1 + \varepsilon_1$$
$$Y \sim N(X\beta, \sigma^2)$$

- Now that we have the step-by-step approach, implement this in R, so that you generate 100 observations from this model, and then plot Y~X (scatterplot is fine).
- Make your x <-1:100, intercept =2, slope=4, sd=2

Generating simulated samples from linear model in R.

For each known value of X (x1, x2,…,x100) we are simulating one value from Y (y1, y2,…., y100)

Using Monte Carlo simulation to generate values for a linear regression with observed values of response (y) and predictors (x):

- How is this example different from when you need to simulate everything?
- What do you know?
- What are you trying simulate?
- What do you need to assume?

Using Monte Carlo simulation to generate values from a simple linear model.

- What would you do if you want to repeat this process many times?

- Can you think of any issues with this approach?

Using Monte Carlo simulation to generate values from a simple linear model.

- What would you do if you want to repeat this process many times?

  use a **for** loop

  or use **replicate()** in R.

Repeat this process 1000 times

- Generate a distribution for the simulated slopes and the simulated intercepts.

Using Monte Carlo simulation to generate values from a simple linear model.

- Can you think of any issues with this approach?

Using Monte Carlo simulation to generate values from a simple linear model.

- What issues are there to this approach?
  - The major difficulty with this approach is that we have not accounted for the co-variation among parameters. That is we assume that each parameter is completely independent of one another. This is rarely the case. We can deal with this by directly accounting for the co-variation.
  - See vcov()

  We are also making assumptions about RSE….

Using Monte Carlo simulation to construct confidence intervals.

- simple (n=100) example:

$$y_1 \sim \beta_0 + \beta_1 x_1 + \varepsilon_1$$
$$Y \sim N(X\beta, \sigma^2)$$

- How might you use simulations to construct confidence intervals BASED ON AN INITIAL FIT OF A MODEL TO DATA.
- Make a step-by-step list of what you need to simulate, and how you can use this to construct confidence intervals.

Using Monte Carlo simulation to construct confidence intervals. STEPS

Using Monte Carlo simulation to construct confidence intervals.

- How might you implement this in R?

Step-by-step approach to constructing CI's using monte carlo simulation

1. Estimate model parameters β's (i.e. using LS or MLE), and residual model variation σ (see below).
2. Simulate new sets of y values using the original parameters plus random deviates drawn from the appropriate distribution.*
3. Re-run the regression model with the simulated data.
4. Repeat this process (how many times? Does it make a difference)?
5. Construct confidence intervals using percentiles (quantiles) from simulated values or using ~ 2*SE.

Step-by-step approach to constructing CI's using monte carlo simulation

- This approach is approximate. What is missing?
- Can you think of other sources of variation we have not accounted for?
- How do we specify the width of the interval?

Using Monte Carlo simulation to approximate p-values.

- Start with an imaginary simple (n=100 observations) example:

$$y_1 \sim \beta_0 + \beta_1 x_1 + \varepsilon_1$$
$$Y \sim N(X\beta, \sigma^2)$$

- How might you use simulations to find a p-value? Start with reminding yourself of the definition of a p-value.
- Make a step-by-step list of what you need to simulate, and how you can use this to approximate a p-value.

Step-by-Step method to approximating p-values using Monte Carlo simulation

- Remember p-value = $P(D|H_O)$
- Estimate parameters as before for **full** model $y_1 \sim \beta_0 + \beta_1 x_1 + \varepsilon_1$
- Estimate parameters under null model.*

  I.e. ($y_1 \sim \beta_0 + \varepsilon_1$) (these are new estimates).
- Simulate values of y under the null model.
- Re-run the **full** regression model with simulated data.
- Compare observed parameters (or RSE) to simulated values.

Step-by-Step method to approximating p-values using Monte Carlo simulation

- How do we specify the precision of the p-value?

- Please implement this in R.

What are the kinds of assumptions we are making when using simulation for inference?

- Sample is representative of larger population.
- Assumptions about the distribution of the population
- Deterministic component of model is correct.
- Assumption of no unmodeled co-variation. How do we correct for this?
- Larger population. How large is the real population?
- Assuming the correct form for the null model (for p-value).
- Distribution of residual errors should follow known distribution.

Gelman and Hill utilize a better approach to account for variation in RSE.

- Page 143. (also see sim() in arm library)
- Step 1: Using classical regression to estimate model parameters. $\hat{\beta}, \mathrm{V}_{\beta}, \hat{\sigma}^2$
- Step 2: generate a number (N) simulations of β and σ. For Each simulation draw:
  - A) Simulate $\sigma = \hat{\sigma}\sqrt{(n-k)/X}$ where X is $\sim \chi^2(df = n-k)$
  - B) Given the random draw of sigma, simulate beta from a multivariate normal distribution with mean $\hat{\beta}$ and covariance matrix $\hat{\sigma}^2\mathrm{V}_{\beta}$.

n – k represents the number of df, i.e. n observations, k parameters.