College of
EDUCATION

ESTIMATION METHODS

EDF 9780: Multivariate Educational Research (Spring 2026 Semester)

Lecture #4

Lecture Parts: Building Blocks

The building blocks: The basics of mathematical statistics:

- Random variables: definitions and types
- Univariate distributions
  - General terminology
  - Univariate normal (aka, Gaussian)
  - Other popular (continuous) univariate distributions
- Types of distributions: marginal, conditional, and joint
- Expected values: means, variances, and the algebra of expectations
- Linear combinations of random variables

The finished product: How the GLM fits within statistics

- The GLM with the normal distribution
- The statistical assumptions of the GLM
- How to assess these assumptions

Least Squares and Maximum Likelihood

The basics of maximum likelihood estimation

- The engine that drives most modern statistical methods

Additional information from maximum likelihood estimator (MLEs)

- Likelihood ratio tests
- Wald tests
- Information criteria

MLEs for GLMs

- An introduction to the NLME (non-linear mixed effects) and LME (linear mixed effects) packages in R
- We'll also use the lavaan package in R (ML for Path Analysis)

Bayesian Estimation

An introduction to Bayesian statistics:

- What it is
- What it does
- Why people use it

An introduction to Markov Chain Monte Carlo (MCMC estimation)

- How it works
- Features to look for when using MCMC
- Why people use it

RANDOM VARIABLES AND STATISTICAL DISTRIBUTIONS

Random Variables

Random: situations in which the certainty of the outcome is unknown and is at least in part due to chance

+

Variable: a value that may change given the scope of a given problem or set of operations

=

Random Variable: a variable whose outcome depends on chance

(possible values might represent the possible outcomes of a yet-to-be-performed experiment)

Today we will denote a random variable with a lower-cased:

$$x$$

Types of Random Variables

Random variables have different types:

1. Continuous

   Examples of continuous random variables:

   x represents the height of a person, drawn at random

   Y_p (the outcome/DV in a GLM)

2. Discrete (also called categorical, generally)

   Example of discrete random variables:

   x represents the gender of a person, drawn at random

3. Mixture of Continuous and Discrete:

   Example of mixture random variables:

   x represents $\begin{cases} \text{response time (if between 0 and 45 seconds)} \\ 0 \end{cases}$

Key Features of Random Variables

Random variables each are described by a **probability density/mass function (PDF)** $f(x)$ that indicates relative frequency of occurrence

- A PDF is a mathematical function that gives a rough picture of the distribution from which a random variable is drawn

The type of random variable dictates the name and nature of these functions:

- Continuous random variables:
  - $f(x)$ is called a probability density function
  - Area under curve must equal 1 (found by calculus – integration)
  - Height of curve (the function value $f(x)$):
    - Can be any positive number
    - Reflects relative likelihood of an observation occurring

- Discrete random variables:
  - $f(x)$ is called a probability mass function
  - Sum across all values must equal 1
  - The function value $f(x)$ is a probability (so must range from 0 to 1)

Other Key Terms

The **sample space** is the set of all values that a random variable x can (possibly) take:

- The sample space for a random variable x from a normal distribution ($x \sim N(\mu_x, \sigma_x^2)$) is $(-\infty, \infty)$ (all real numbers)
- The sample space for a random variable x representing the outcome of a coin flip is $\{H, T\}$
- The sample space for a random variable x representing the outcome of a roll of a die is $\{1, 2, 3, 4, 5, 6\}$

When using generalized models, the trick is to pick a distribution with a sample space that matches the range of values **obtainable** by data

Uses of Distributions in Data Analysis

Statistical models make distributional assumptions on various parameters and/or parts of data

These assumptions govern:

- How models are estimated
- How inferences are made
- How missing data may be imputed

If data do not follow an assumed distribution, inferences may be inaccurate

- Depending on the size of the mismatch can lead to substantial inaccuracies

Therefore, it can be helpful to check distributional assumptions prior to (or while) running statistical analyses

CONTINUOUS UNIVARIATE DISTRIBUTIONS

Continuous Univariate Distributions

To demonstrate how continuous distributions work and look, we will discuss three:

- Uniform distribution
- Normal distribution
- Chi-square distribution

Each are described a set of **parameters**, which we will later see are what give us our inferences when we analyze data

What we then do is put constraints on those parameters based on hypothesized effects in data

Uniform Distribution

The uniform distribution is shown to help set up how continuous distributions work

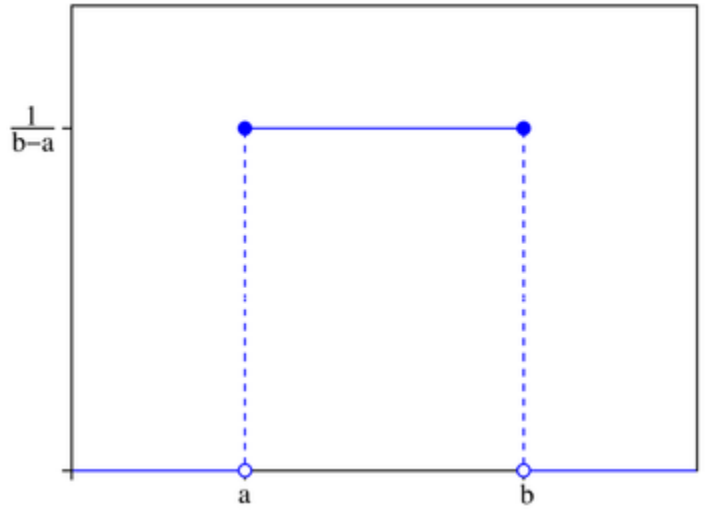
For a continuous random variable x that ranges from (a, b), the uniform probability density function is:

$$f(x) = \frac{1}{b-a}$$

The uniform distribution has two parameters:

- a – the lower limit
- b – the upper limit

$$x \sim U(a, b)$$



More on the Uniform Distribution

To demonstrate how PDFs work, we will try a few values:

x	a	b	$f(x)$
.5	0	1	$\frac{1}{1-0} = 1$
.75	0	1	$\frac{1}{1-0} = 1$
15	0	20	$\frac{1}{20-0} = .05$
15	10	20	$\frac{1}{20-10} = .1$

The uniform PDF has the feature that all values of x are **equally likely** across the sample space of the distribution

- Therefore, you do not see x in the PDF $f(x)$

The mean of the uniform distribution is $\frac{1}{2}(a+b)$

The variance of the uniform distribution is $\frac{1}{12}(b-a)^2$

Univariate Normal Distribution

For a continuous random variable x (ranging from $-\infty$ to ∞) the univariate normal distribution function is:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left(-\frac{(x-\mu_x)^2}{2\sigma_x^2}\right)$$

The shape of the distribution is governed by two parameters:

- The mean μ_x
- The variance σ_x^2
- These parameters are called **sufficient statistics** (they contain all the information about the distribution)

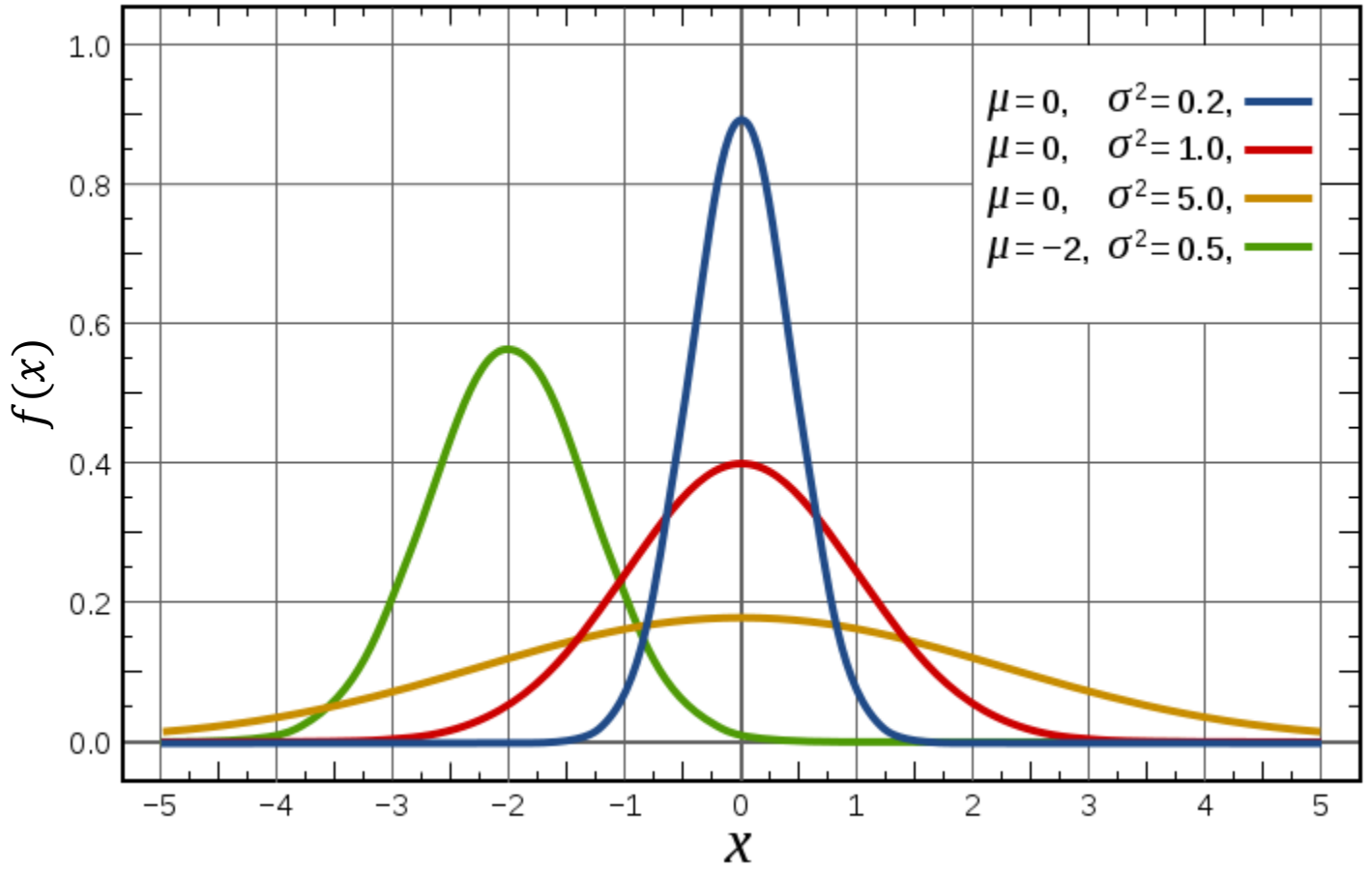
The skewness (lean) and kurtosis (peakedness) are fixed

Standard notation for normal distributions is $x \sim N(\mu_x, \sigma_x^2)$

- Read as: "x follows a normal distribution with a mean μ_x and a variance σ_x^2"

Linear combinations of random variables following normal distributions result in a random variable that is normally distributed

Univariate Normal Distribution



$f(x)$ gives the height of the curve (relative frequency) for any value of x, μ_x, and σ_x^2

More of the Univariate Normal Distribution

To demonstrate how the normal distribution works, we will try a few values:

x	μ_x	σ_x^2	$f(x)$
.5	0	1	0.352
.75	0	1	0.301
.5	0	5	0.079
.75	-2	1	0.009
-2	-2	1	0.399

The values from $f(x)$ were obtained by using Excel

- The "=normdist()" function
- Most statistics packages have a normal distribution function
  - In R you can use the dnorm() function

The mean of the normal distribution is μ_x

The variance of the normal distribution is σ_x^2

Chi-Square Distribution

Another frequently used univariate distribution is the Chi-Square distribution

- Sampling distribution of the variance follows a chi-square distribution
- Likelihood ratios follow a chi-square distribution

For a continuous random variable x (ranging from 0 to ∞), the chi-square distribution is given by:

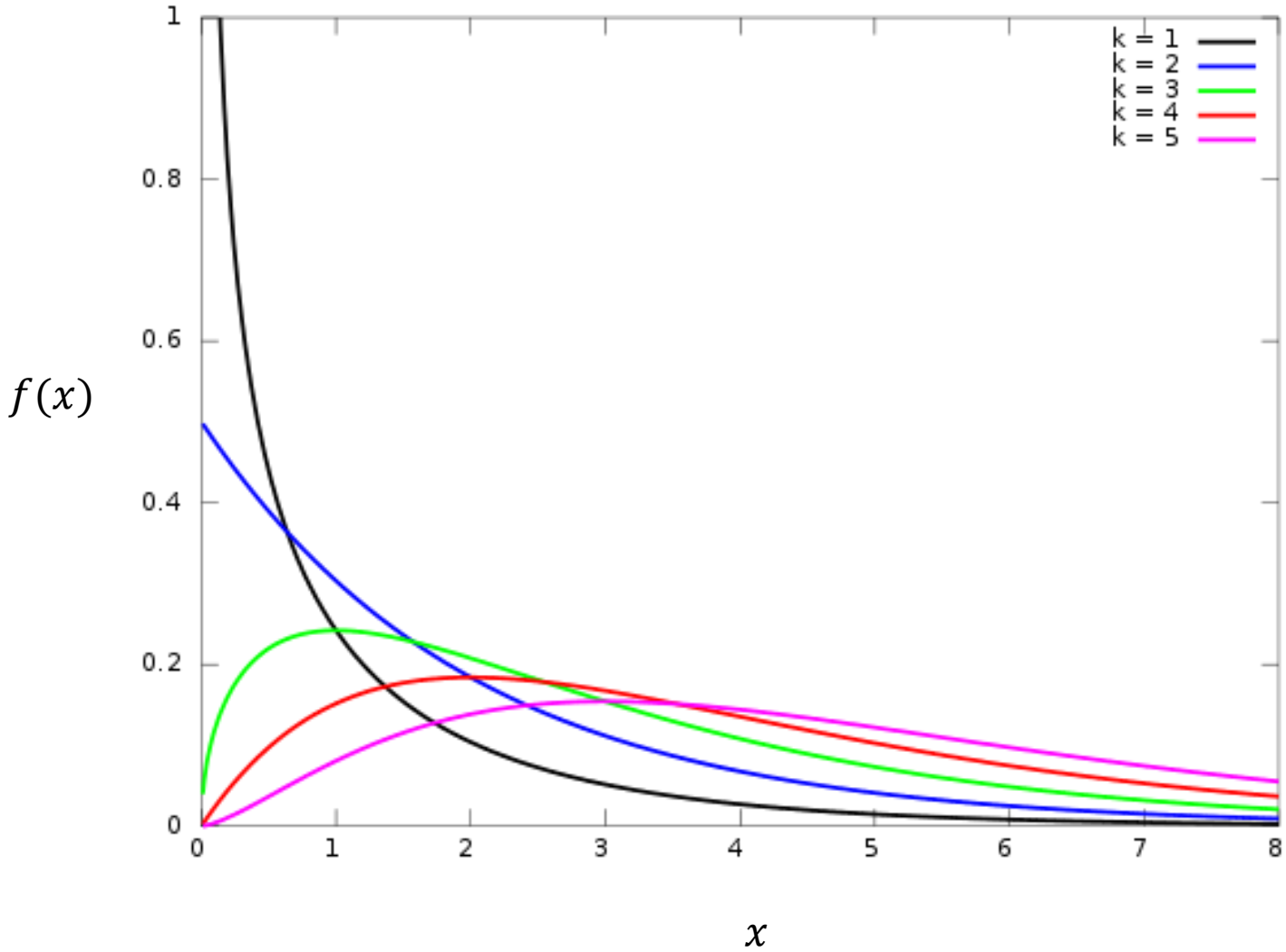
$$f(x) = \frac{1}{2^{\frac{\nu}{2}}\Gamma\left(\frac{\nu}{2}\right)} x^{\frac{\nu}{2}-1} \exp\left(-\frac{x}{2}\right)$$

$\Gamma(\cdot)$ is called the gamma function

The chi-square distribution is governed by one parameter: ν (the degrees of freedom)

- The mean is equal to ν; the variance is equal to 2ν

(Univariate) Chi-Square Distribution



MARGINAL, JOINT, AND CONDITIONAL DISTRIBUTIONS

Moving from One to Multiple Random Variables

When more than one random variable is present, there are several different types of statistical distributions:

We will first consider two discrete random variables:

- x is the outcome of the flip of a penny (H_p, T_p)
  $f(x = H_p) = .5$; $f(x = T_p) = .5$
- z is the outcome of the flip of a dime (H_d, T_d)
  $f(z = H_d) = .5$; $f(z = T_d) = .5$

We will consider the following distributions:

- Marginal distribution
  - The distribution of one variable only (either $f(x)$ ***or*** $f(z)$)
- Joint distribution
  - $f(x, z)$: the distribution of both variables (both x ***and*** z)
- Conditional distribution
  - The distribution of one variable, conditional on values of the other:
    - $f(x|z)$: the distribution of x given z
    - $f(z|x)$: the distribution of z given x

Marginal Distributions

Marginal distributions are what we have worked with exclusively up to this point: they represent the distribution of one variable by itself

- Continuous univariate distributions:
  - Uniform
  - Normal
  - Chi-square

- Categorical distributions in our example:
  - The flip of a penny $f(x)$
  - The flip of a dime $f(z)$

Joint Distributions

Joint distributions describe the distribution of more than one variable, simultaneously

- Representations of multiple variables collected

Commonly, the joint distribution function is denoted with all random variables separated by commas

- In our example, $f(x, z)$ is the joint distribution of the outcome of flipping both a penny and a dime
  - As both are discrete, the joint distribution has four possible values:
    $f(x = H_p, z = H_d), f(x = H_p, z = T_d), f(x = T_p, z = H_d), f(x = T_p, z = T_d)$

Joint distributions are **multivariate distributions**

We will use joint distributions to introduce two topics

- Joint distributions of independent variables
- Joint likelihoods – used in maximum likelihood estimation

Joint Distributions of Independent Random Variables

Random variables are said to be independent if the occurrence of one event makes it neither more nor less probable of another event

- For joint distributions, this means: $f(x,z) = f(x)f(z)$

In our example, flipping a penny and flipping a dime are independent – so we can complete the following table of their joint distribution:

Joint (Penny, Dime)

Dime

Penny	$z = H_d$	$z = T_d$	
$x = H_p$	$f(x = H_p, z = H_d)$	$f(x = H_p, z = T_d)$	$f(x = H_p)$
$x = T_p$	$f(x = T_p, z = H_d)$	$f(x = T_p, z = T_d)$	$f(x = T_d)$
	$f(z = H_d)$	$f(z = T_d)$	

Marginal (Penny)

Marginal (Dime)

Joint Distributions of Independent Random Variables

Because the coin flips are independent, this becomes:

Joint (Penny, Dime)

Penny \ Dime	$z = H_d$	$z = T_d$	
$x = H_p$	$f(x = H_p)f(z = H_d)$	$f(x = H_p)f(z = T_d)$	$f(x = H_p)$
$x = T_p$	$f(x = T_p)f(z = H_d)$	$f(x = T_p)f(z = T_d)$	$f(x = T_p)$
	$f(z = H_d)$	$f(z = T_d)$	

Marginal (Penny)

Marginal (Dime)

Then, with numbers:

Joint (Penny, Dime)

Penny \ Dime	$z = H_d$	$z = T_d$	
$x = H_p$	.25	.25	.5
$x = T_p$	.25	.25	.5
	.5	.5	

Marginal (Penny)

Marginal (Dime)

Marginalizing Across a Joint Distribution

If you had a joint distribution, $f(x, z)$, but wanted the marginal distribution of either variable ($f(x)$ or $f(z)$) you would have to **marginalize** across one dimension of the joint distribution

For categorical random variables, **marginalize = sum across**

$$f(x) = \sum_{z} f(x, z)$$

For example, $f(x = H_p) = f(x = H_p, z = H_p) + f(x = H_p, z = T_p) = .5$

For continuous random variables, **marginalize = integrate across**

No integration needed from you – just a conceptual understanding

Here, the integral = an eraser!

$$f(x) = \int_{z} f(x, z)dz$$

Conditional Distributions

For two random variables x and z, a conditional distribution is written as: $f(z|x)$

- The distribution of z given x

The conditional distribution is also equal to the joint distribution divided by the marginal distribution of the conditioning random variable

$$f(z|x) = \frac{f(z,x)}{f(x)}$$

Conditional distributions are found everywhere in statistics

- The general linear model uses the conditional distribution of the dependent variable (where the independent variables are the conditioning variables)

Conditional Distributions

For discrete random variables, the conditional distribution can be shown in a contingency table:

Joint (Penny, Dime)

Penny \ Dime	$z = H_d$	$z = T_d$	
$x = H_p$	.25	.25	.5
$x = T_p$	.25	.25	.5
	.5	.5	

Marginal (Penny)

Marginal (Dime)

Conditional: $f(z|x = H_p)$:

$$f(z = H_d|x = H_p) = \frac{f(z = H_d, x = H_p)}{f(x = H_p)} = \frac{.25}{.5} = .5$$

$$f(z = T_d|x = H_p) = \frac{f(z = T_d, x = H_p)}{f(x = H_p)} = \frac{.25}{.5} = .5$$

We will show a continuous conditional distribution with the GLM in a few slides

EXPECTED VALUES AND THE ALGEBRA OF EXPECTATIONS

Expected Values

Expected values are statistics taken the sample space of a random variable: they are essentially weighted averages

The weights used in computing this average correspond to the probabilities (for a discrete random variable) or to the densities
(for a continuous random variable).

Notation: the expected value is represented by: $E(x)$

- *The actual statistic that is being weighted by the PDF is put into the parentheses where x is now*

Expected values allow us to understand what a statistical model implies about data, for instance:

- How a GLM specifies the (conditional) mean and variance of a DV

Expected Value Calculation

For discrete random variables, the expected value is found by:

$$E(x) = \sum_x xP(X = x)$$

For example, the expected value of a roll of a die is:

$$E(x) = (1)\frac{1}{6} + (2)\frac{1}{6} + (3)\frac{1}{6} + (4)\frac{1}{6} + (5)\frac{1}{6} + (6)\frac{1}{6} = 3.5$$

For continuous random variables, the expected value is found by:

$$E(x) = \int_x xf(x)dx$$

We won't be calculating theoretical expected values with calculus…we use them only to see how models imply things about our data

Variance and Covariance… As Expected Values

A distribution's theoretical variance can also be written as an expected value:

$$V(x) = E\big(x - E(x)\big)^2 = E(x - \mu_x)^2$$

This formula will help us understand predictions made GLMs and how that corresponds to statistical parameters we interpret

For a roll of a die, the theoretical variance is:

$$V(x) = E(x - 3.5)^2 = \frac{1}{6}(1 - 3.5)^2 + \frac{1}{6}(2 - 3.5)^2 + \frac{1}{6}(3 - 3.5)^2 + \frac{1}{6}(4 - 3.5)^2 + \frac{1}{6}(5 - 3.5)^2 + \frac{1}{6}(6 - 3.5)^2 = 2.92$$

Likewise, the SD is then $\sqrt{2.92} = 1.71$

Likewise, for a pair of random variables x and z, the covariance can be found from their joint distributions:

$$Cov(x, z) = E(xz) - E(x)E(z) = E(xz) - \mu_x\mu_z$$

LINEAR COMBINATIONS OF RANDOM VARIABLES

Linear Combinations of Random Variables

A **linear combination** is an expression constructed from a set of terms by multiplying each term by a constant and then adding the results

$$x = a_1 v_1 + a_2 v_2 + \cdots + a_n v_n$$

The linear regression equation is a linear combination

More generally, linear combinations of random variables have specific implications for the mean, variance, and possibly covariance of the new random variable

As such, there are predicable ways in which the means, variances, and covariances change

These terms are called the algebra of expectations

To guide us through this process, we will use the descriptive statistics from the height/weight/gender example

Descriptive Statistics for Height/Weight Data

Variable	Mean	SD	Variance
Height	67.9	7.44	55.358
Weight	183.4	56.383	3,179.095
Female	0.5	0.513	0.263

Diagonal: Variance

Above Diagonal: Covariance

Correlation /Covariance	Height	Weight	Female
Height	55.358	334.832	-2.263
Weight	.798	3,179.095	-27.632
Female	-.593	-.955	.263

Below Diagonal: Correlation

Algebra of Expectations

Here are some properties of expected values (true for any type of random variable): x and z are random variables, c and d constants

Sums of Constants:

$$E(x+c) = E(x) + c$$
$$V(x+c) = V(x)$$
$$Cov(x+c, z) = Cov(x, z)$$

Products of Constants:

$$E(cx) = cE(x)$$
$$V(cx) = c^2V(x)$$
$$Cov(cx, dz) = cdCov(x, z)$$

Sums of Random Variables:

$$E(cx+dz) = cE(x) + dE(z)$$
$$V(cx+dz) = c^2V(x) + d^2V(z) + 2cd\big(Cov(x, z)\big)$$

Examples for Algebra of Expectations

Image you wanted to convert weight from pounds to kilograms (where 1 pound = 0.453 kg)

$$Weight_{kg} = .453 Weight_{lb}$$

The mean (expected value) of weight in kg:

$$E(Weight_{kg}) = E(.453 Weight_{lb}) = .453 E(Weight_{lb}) = .453 \overline{Weight}_{lb} = .453 * 183.4 = 83.08\text{kg}$$

The variance of weight in kg:

$$V(Weight_{kg}) = V(.453 Weight_{lb}) = .453^2 V(Weight_{lb}) = .453^2 * 3{,}179.095 = 652.38 kg^2$$

The covariance of weight in kg with height in inches:

$$Cov(Weight_{kg}, Height) = Cov(.453 Weight_{lb}, Height) = .453 Cov(Weight_{lb}, Height) = .453 * 334.832 = 151.68\text{kg} * \text{inches}$$

Don’t Take My Word For It…

R syntax for transforming weight, marginal descriptive statistics, and covariances:

```
> describe(data01)
         vars  n   mean    sd median trimmed   mad    min    max  range skew kurtosis    se
id          1 20  10.50  5.92  10.50   10.50  7.41   1.00  20.00  19.00 0.00    -1.38  1.32
sex*        2 20   1.50  0.51   1.50    1.50  0.74   1.00   2.00   1.00 0.00    -2.10  0.11
heightIN    3 20  67.90  7.44  68.00   67.88  7.41  54.00  82.00  28.00 0.05    -0.81  1.66
weightLB    4 20 183.40 56.38 175.00  182.12 72.65 109.00 269.00 160.00 0.11    -1.80 12.61
weightKG    5 20  83.08 25.54  79.28   82.50 32.91  49.38 121.86  72.48 0.11    -1.80  5.71
> #show covariance matrix
> cov(data01[c("heightIN", "weightKG", "weightLB")])
           heightIN  weightKG  weightLB
heightIN  55.35789  151.6787  334.8316
weightKG 151.67871  652.3789 1440.1299
weightLB 334.83158 1440.1299 3179.0947
> #show correlation matrix
> cor(data01[c("heightIN", "weightKG", "weightLB")])
           heightIN  weightKG  weightLB
heightIN 1.0000000 0.7981507 0.7981507
weightKG 0.7981507 1.0000000 1.0000000
weightLB 0.7981507 1.0000000 1.0000000
```

Where We Use This…The ghlt() Function from multcomp

The ghlt() function in the multcomp package computes the expected value and standard error (square root of variance) for a new random variable

- The new random variable is a linear combination of the original model parameters (the fixed effects)
- The original model parameters are considered "random" here as their sampling distribution is used (assuming normal errors and a large N)

$$Estimate = 1 * \beta_{experience4} + 1 * \beta_{G2*experience4}$$

Where:

- $\beta_{experience4}$ has mean $\widehat{\beta_{experience4}}$ and variance $se\left(\widehat{\beta_{experience4}}\right)^2$
- $\beta_{G2*experience4}$has mean $\widehat{\beta_{G2*experience4}}$ and variance $se\left(\widehat{\beta_{G2*experience4}}\right)^2$
- There exists a covariance between $\widehat{\beta_{experience4}}$ and $\widehat{\beta_{G2*experience4}}$
  - We'll call this $Cov\left(\widehat{\beta_{experience4}}, \widehat{\beta_{G2*experience4}}\right)$

More ghlt() Fun

So…if the estimates are:

Parameter	Estimate	Standard Error	t Value	Pr > |t|
Intercept	75.49934727	0.38707620	195.05	<.0001
Dgroup2	-10.07267266	0.54896179	-18.35	<.0001
Dgroup3	4.17623925	0.54961852	7.60	<.0001
Dgroup4	-6.04195829	0.54912685	-11.00	<.0001
experience4	-0.38518388	0.29569936	-1.30	0.1943
enthusiasm	-5.00727782	0.18730609	-26.73	<.0001
Dgroup2*experience4	-0.63103823	0.39198136	-1.61	0.1091
Dgroup3*experience4	-0.10925920	0.41111045	-0.27	0.7907
Dgroup4*experience4	0.16959725	0.41917025	0.40	0.6862

And $Cov(\widehat{\beta_{experience4}}, \widehat{\beta_{G2*experience4}}) = -.08756$

…What is:

$$E(Estimate) = E(1 * \beta_{experience4} + 1 * \beta_{G2*experience4}) = 1 * E(\beta_{experience4}) + 1 * E(\beta_{G2*experience4})$$
$$= -.385 - .631 = -1.016$$

$$V(Estimate) = V(1 * \beta_{experience4} + 1 * \beta_{G2*experience4})$$
$$= 1^2 V(\beta_{experience4}) + 1^2 V(\beta_{G2*experience4}) + 2 * 1 * 1 Cov(\beta_{experience4}, \beta_{G2*experience4}) =$$
$$.296^2 + .391^2 + 2 * .08756 = .0653$$

$$se(Estimate) = \sqrt{V(Estimate)} = .257$$

THE GENERAL LINEAR MODEL WITH WHAT WE HAVE LEARNED TODAY

The General Linear Model, Revisited

The general linear model for predicting Y from X and Z:

$$Y_p = \beta_0 + \beta_1 X_p + \beta_2 Z_p + \beta_3 X_p Z_p + e_p$$

In terms of random variables, under the GLM:

- e_p is considered random: $e_p \sim N(0, \sigma_e^2)$

- Y_p is dependent on the linear combination of X_p, Z_p, and e_p

The GLM provides a model for the **conditional distribution** of the dependent variable, where the conditioning variables are the independent variables: $f(Y_p | X_p, Z_p)$

  - There are no assumptions made about X_p and Z_p - they are constants
  - The regression slopes $\beta_0, \beta_1, \beta_2, \beta_3$ are constants that are said to be fixed at their values (hence, called fixed effects)

Combining the GLM with Expectations

Using the algebra of expectations predicting Y from X and Z:

The expected value (mean) of $f(Y_p|X_p, Z_p)$:

$$\hat{Y}_p = E(Y_p) = E(\beta_0 + \beta_1 X_p + \beta_2 Z_p + \beta_3 X_p Z_p + e_p)$$

Constants

Random Variable with $E(e_p) = 0$

$$= \beta_0 + \beta_1 X_p + \beta_2 Z_p + \beta_3 X_p Z_p + E(e_p) = \beta_0 + \beta_1 X_p + \beta_2 Z_p + \beta_3 X_p Z_p$$

The variance of $f(Y_p|X_p, Z_p)$:

$$V(Y_p) = V(\beta_0 + \beta_1 X_p + \beta_2 Z_p + \beta_3 X_p Z_p + e_p) = V(e_p) = \sigma_e^2$$

Distribution of $f(Y_p|X_p, Z_p)$

We just found the mean (expected value) and variance implied by the GLM for the conditional distribution of Y_p given X_p and Z_p

The next question: what is the distribution of $f(Y_p|X_p, Z_p)$?

Linear combinations of random variables that are normally distributed result in variables that are normally distributed

Because $e_p \sim N(0, \sigma_e^2)$ is the only random term in the GLM, the resulting conditional distribution of Y_p is normally distributed:

$$Y_p|X_p, Z_p \sim N(\beta_0 + \beta_1 X_p + \beta_2 Z_p + \beta_3 X_p Z_p, \sigma_e^2)$$

Model for the means: from fixed effects; literally gives mean of $f(Y_p|X_p, Z_p)$

Model for the variances: from random effects; gives variance of $f(Y_p|X_p, Z_p)$

Examining What This Means in the Context of Data

If you recall from the regression analysis of the height/weight data, the final model we decided to interpret: Model 5

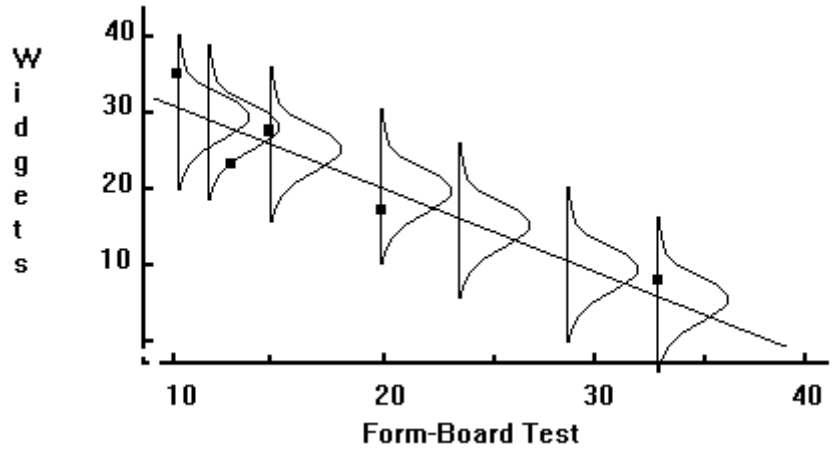
$$W_p = \beta_0 + \beta_1\left(H_p - \bar{H}\right) + \beta_2 F_p + \beta_3\left(H_p - \bar{H}\right)F_p + e_p$$

where $e_p \sim N(0, \sigma_e^2)$

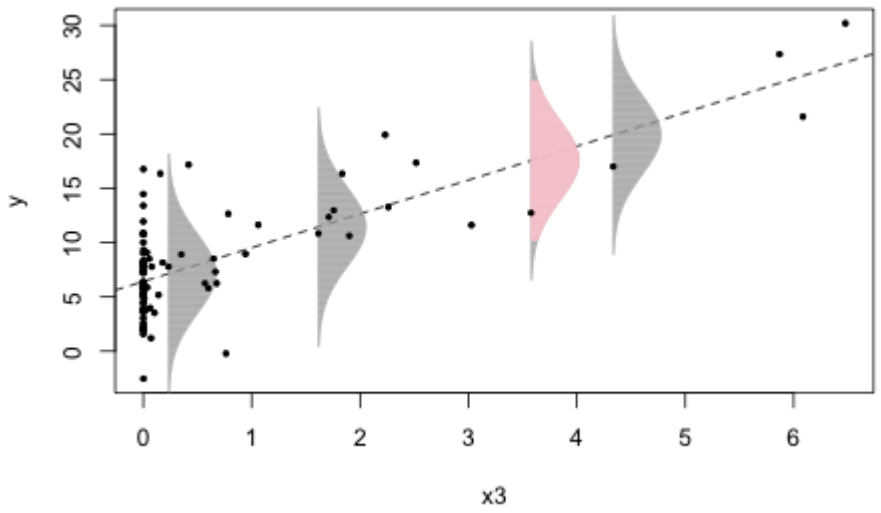
```
Call:
lm(formula = weightLB ~ heightIN_MC + female + female * heightIN_MC,
    data = data01)

Residuals:
    Min      1Q  Median      3Q     Max
-3.8312 -1.7797  0.4958  1.3575  3.3585

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        222.1842     0.8381  265.11  < 2e-16 ***
heightIN_MC          3.1897     0.1114   28.65 3.55e-15 ***
female             -82.2719     1.2111  -67.93  < 2e-16 ***
heightIN_MC:female  -1.0939     0.1678   -6.52 7.07e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.175 on 16 degrees of freedom
Multiple R-squared:  0.9987,    Adjusted R-squared:  0.9985
F-statistic:  4250 on 3 and 16 DF,  p-value: < 2.2e-16
```

Picturing the GLM with Distributions

The distributional assumptions of the GLM are the reason why we do not need to worry if our dependent variable is normally distributed



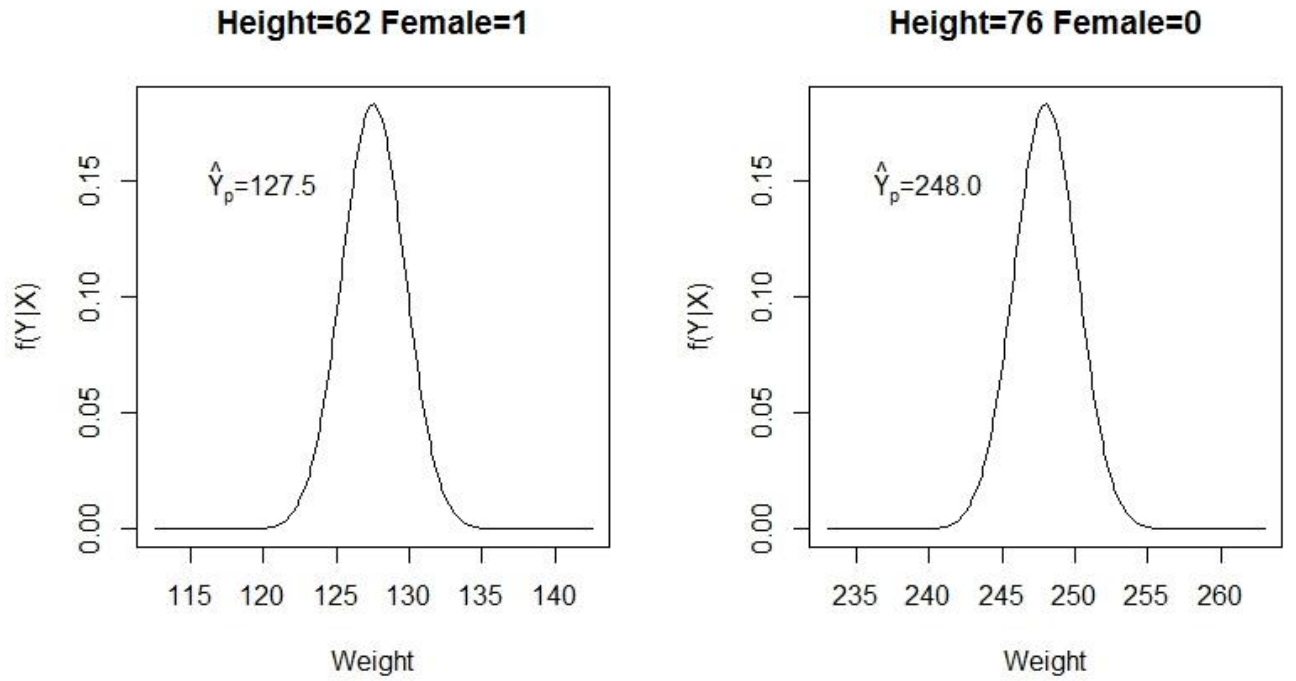
Our dependent variable should be **conditionally** normal



We can check this assumption by checking our assumption about the residuals, $e_p \sim N(0, \sigma_e^2)$

More Pictures of the GLM

Treating our estimated values of the slopes $(\beta_0, \beta_1, \beta_2, \beta_3)$ and the residual variance (σ_e^2) as the true values* we can now see what the theoretical* distribution of $f\left(Weight_p \middle| Height_p, Female_p\right)$ looks like for a given set of predictors



*Note: these distributions change when sample estimates are used (think standard error of the prediction)

Behind the Pictures…

To emphasize the point that PDFs provide the height of the line, here is the normal PDF (with numbers) that produced those plots:

$$f\left(W_p \middle| H_p, F_p\right) = \frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left(-\frac{\left(W_p - \widehat{W}_p\right)^2}{2\sigma_e^2}\right)$$

Model for the Means

$$= \frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left(-\frac{\left(W_p - \left(\beta_0 + \beta_1\left(H_p - \bar{H}\right) + \beta_2 F_p + \beta_3\left(H_p - \bar{H}\right)F_p\right)\right)^2}{2\sigma_e^2}\right)$$

Model for the Variance

$$= \frac{1}{\sqrt{2\pi(4.73)}} \exp\left(-\frac{\left(W_p - \left(222.18 + 3.19\left(H_p - \bar{H}\right) - 82.27F_p - 1.09\left(H_p - \bar{H}\right)F_p\right)\right)^2}{2(4.73)}\right)$$

The plots were created using the following value for the predictors:

$\bar{H} = 67.9$

Left plot: $H_p = 62; F_p = 1$

Right plot: $H_p = 76; F_p = 0$

ASSESSING UNIVARIATE NORMALITY IN R

Assessing Univariate Normality in SAS

The assumption of normally distributed residuals permeates GLM

- Good news: of all the distributional assumptions, this seems to be the least damaging to violate. GLMs are robust to violations of normality.

Methods exist to examine residuals from an analysis and thereby determine the adequacy of a model

- Graphical methods: Quantile-Quantile plots
- Hypothesis tests

Both approaches have problems

- Graphical methods do not determine how much deviation is by chance
- Hypothesis tests become overly sensitive to small deviations when sample size is large (have great power)

To emphasize how distributions work, we will briefly discuss both

Assessing Distributional Assumptions Graphically

A useful tool to evaluate the plausibility of a distributional assumption is that of the Quantile versus Quantile Plot
(more commonly called a Q-Q plot)

A Q-Q plot is formed by comparing the observed quantiles of a variable with that of a known statistical distribution

- A quantile is the particular ordering of a given observation
- In our data, a person with a height of 71 is the 39^{th} tallest person (out of 50)
- This would correspond to the person being at the $\frac{39-.5}{50} = .77$ or .77 percentile of the distribution (taller than 77% of the distribution)
- The Q-Q plot then converts the percentile to a quantile using the sample mean and variance
  - A quantile is the value of an observation at the 77^{th} percentile

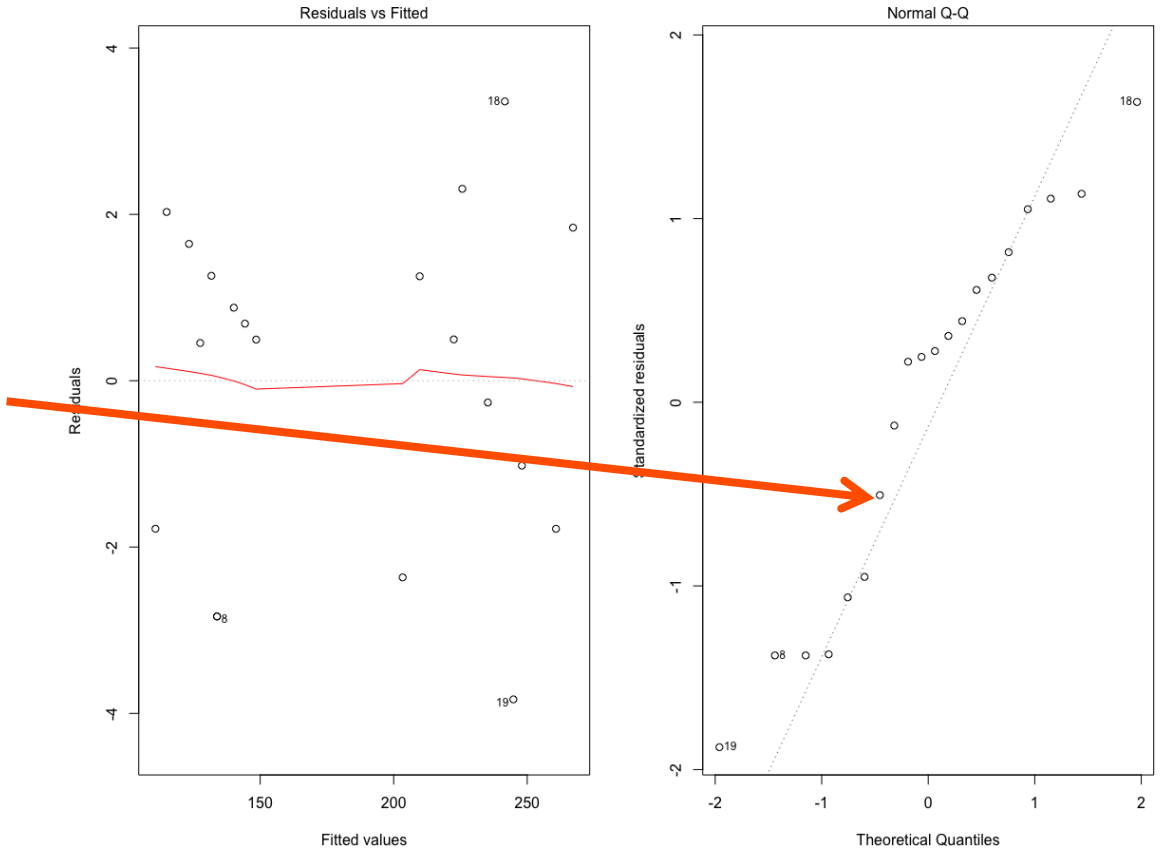
If the data deviate from a straight line, the data are not likely to follow from that theoretical distribution

Q-Q Plots of GLM Residuals

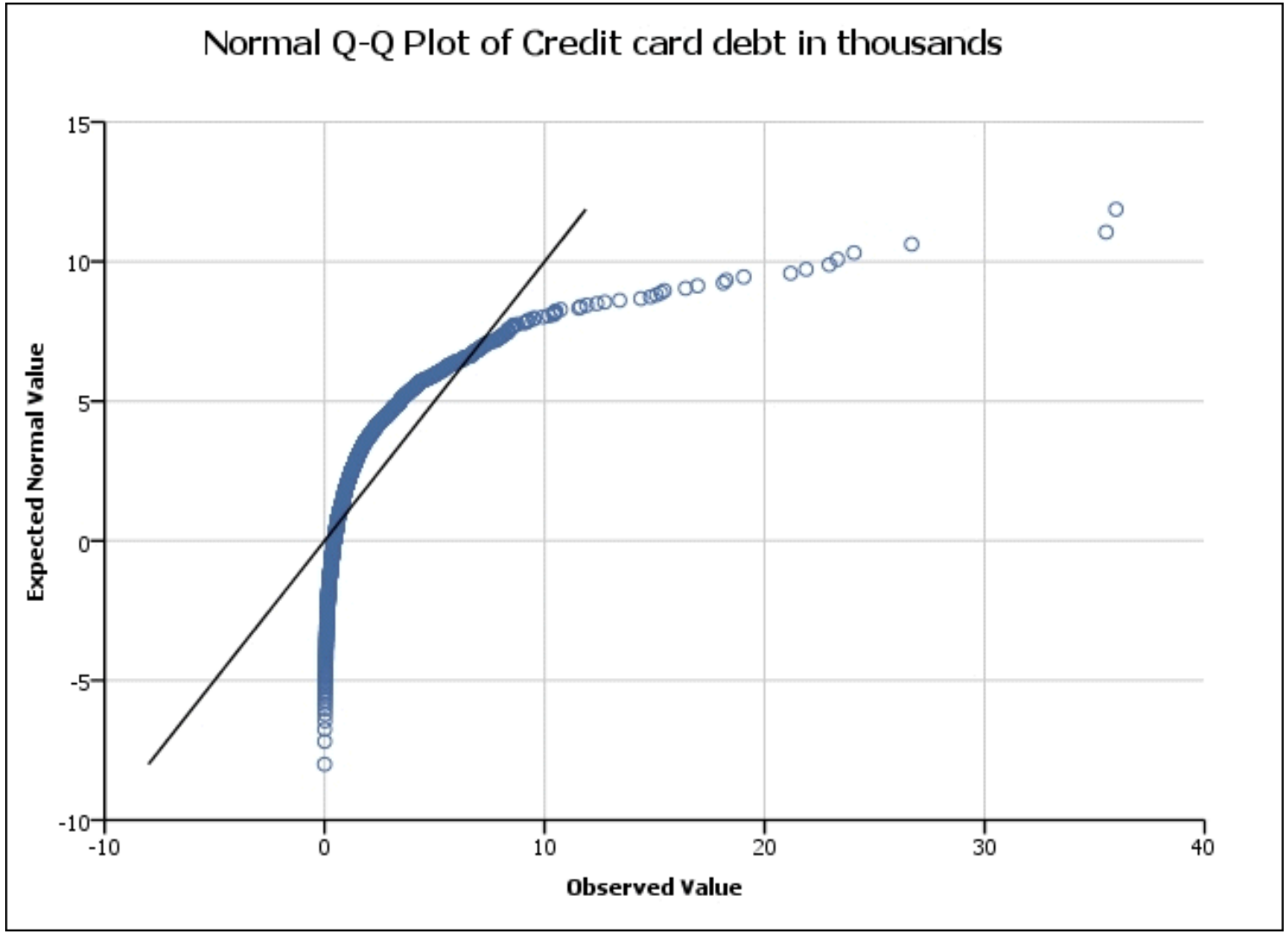
R has a built-in generic function named plot() that will plot residuals from your lm() model:

```
# plotting residuals of model 05:
plot(model05)
```

If residuals are normally distributed, they will fall on the line



Example Q-Q Plot of Non-Normal Data



Hypothesis Tests for Normality

```
> shapiro.test(model05$residuals);

        Shapiro-Wilk normality test

data:  model05$residuals
W = 0.95055, p-value = 0.3756
```

If a given test is **significant**, then it is saying that your data **do not** come from a normal distribution

In practice, test will give diverging information quite frequently:
the best way to evaluate normality is to consider both plots and tests (approximate = good)

Distributions Wrap Up

This was an introduction to mathematical statistics to understand the implications statistical models make about data

Although many of these topics do not seem directly relevant, they help provide insights that untrained analysts may not easily attain

- They also help you to understand when and when not to use a model!

ESTIMATION METHODS

Today's Example Data #1

Imagine an employer is looking to hire employees for a job where IQ is important

We will only use 5 observations so as to show the math behind the estimation calculations

The employer collects two variables:

IQ scores

Job performance

Descriptive Statistics:

Variable	Mean	SD
IQ	114.4	2.30
Performance	12.8	2.28

Covariance Matrix		
IQ	5.3	5.1
Performance	5.1	5.2

Observation	IQ	Performance
1	112	10
2	113	12
3	115	14
4	118	16
5	114	12

How Estimation Works (More or Less)

Most estimation routines do one of three things:

1. **Minimize Something:** Typically found with names that have “least” in the title. Forms of least squares include “Generalized”, “Ordinary”, “Weighted”, “Diagonally Weighted”, “WLSMV”, and “Iteratively Reweighted.” Typically the estimator of last resort…

2. **Maximize Something:** Typically found with names that have “maximum” in the title. Forms include “Maximum likelihood”, “ML”, “Residual Maximum Likelihood” (REML), “Robust ML”. Typically the gold standard of estimators

3. **Use Simulation to Sample from Something:** more recent advances in simulation use resampling techniques. Names include “Bayesian Markov Chain Monte Carlo”, “Gibbs Sampling”, “Metropolis Hastings”, “Metropolis Algorithm”, and “Monte Carlo”. Used for complex models where ML is not available or for methods where prior values are needed.

AN INTRODUCTION TO MAXIMUM LIKELIHOOD ESTIMATION

Properties of Maximum Likelihood Estimators

Provided several assumptions (“regularity conditions”) are met, maximum likelihood estimators have good statistical properties:

1. Asymptotic Consistency: as the sample size increases, the estimator converges in probability to its true value

2. Asymptotic Normality: as the sample size increases, the distribution of the estimator is normal (with variance given by “information” matrix)

3. Efficiency: No other estimator will have a smaller standard error

Because they have such nice and well understood properties, MLEs are commonly used in statistical estimation

Maximum Likelihood: Estimates Based on Statistical Distributions

Maximum likelihood estimates come from statistical distributions – assumed distributions of data

- We will begin today with the univariate normal distribution but quickly move to other distributions

For a single random variable x, the univariate normal distribution is

$$f(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left(-\frac{(x-\mu_x)^2}{2\sigma_x^2}\right)$$

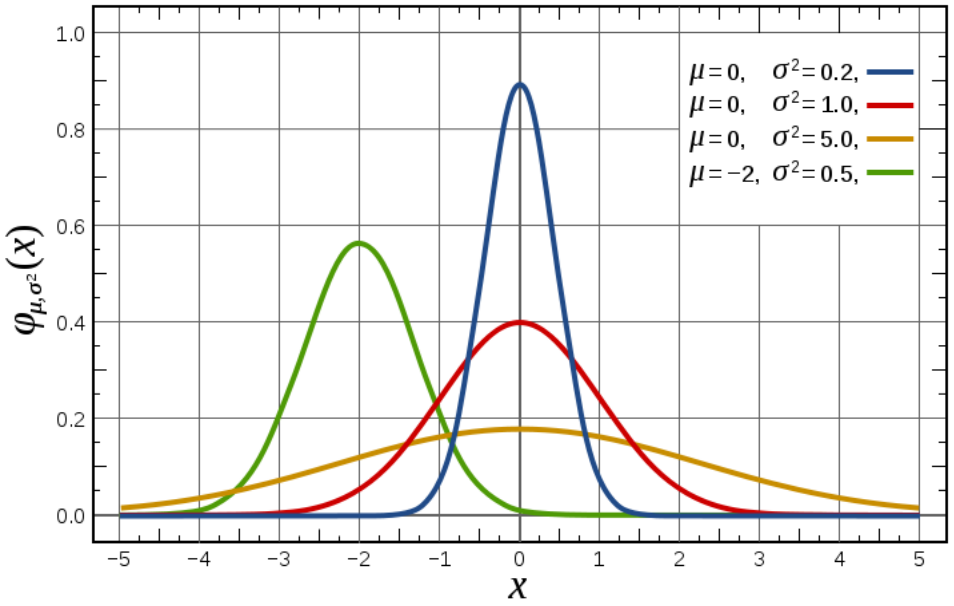
- Provides the height of the curve for a value of x, μ_x, and σ_x^2

Before we pretended we knew μ_x and σ_x^2

- Today we will only know x (and maybe σ_x^2)

Univariate Normal Distribution

$f(x)$



For any value of x , μ_x, and σ_x^2, $f(x)$ gives the height of the curve (relative frequency)

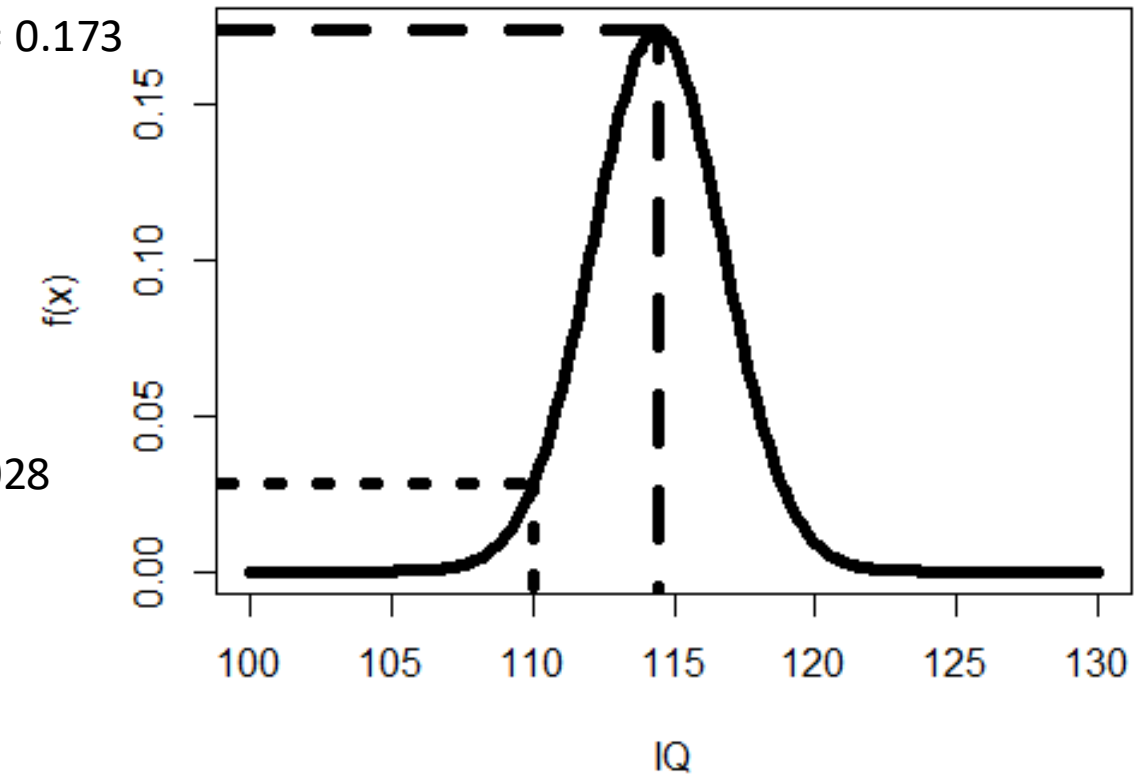
Example Distribution Values

Let's examine the distribution values for the IQ variable

- We assume that we **know** $\mu_x = 114.4$ and $\sigma_x^2 = 5.29$ $(\sigma_x = 2.30)$
  - In reality, we do not know what these values happen to be

For $x = 114.4$, $f(114.4) = 0.173$

For $x = 110$, $f(110) = 0.028$



Constructing a Likelihood Function

Maximum likelihood estimation begins by building a **likelihood function**

- A likelihood function provides a value of a likelihood (think height of a curve) for a set of statistical parameters

Likelihood functions start with probability density functions (PDFs)

- Density functions are provided for each observation individually (marginal)

The likelihood function for the entire sample is the function that gets used in the estimation process

- The sample likelihood can be thought of as a joint distribution of all the observations, simultaneously
- In univariate statistics, observations are considered independent, so the joint likelihood for the sample is constructed through a product

To demonstrate, let's consider the likelihood function for one observation

A One-Observation Likelihood Function

Let's assume the following:

- We have observed the first value of IQ ($x = 112$)
- That IQ comes from a normal distribution
- That the variance of x is known to be 5.29 ($\sigma_x^2 = 5.29$)
  - This is to simplify the likelihood function so that we only don't know one value
  - More on this later…empirical under-identification

For this one observation, the likelihood function takes its assumed distribution and uses its PDF:

$$f(x, \mu_x, \sigma_x^2) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left(-\frac{(x-\mu_x)^2}{2\sigma_x^2}\right)$$

The PDF above now is expressed in terms of the three unknowns that go into it:
x, μ_x, σ_x^2

A One-Observation Likelihood Function

Because we know two of these terms ($x = 112; \sigma_x^2 = 5.29$), we can create the likelihood function for the mean:

$$L(\mu_x | x = 112, \sigma_x^2 = 5.29) = \frac{1}{\sqrt{2\pi * 5.29}} \exp\left(-\frac{(112 - \mu_x)^2}{2 * 5.29}\right)$$

For every value of μ_x *could be*, the likelihood function now returns a number that is called **the likelihood**

- The actual value of the likelihood is not relevant (yet)

The value of μ_x with the highest likelihood is called the **maximum likelihood estimate (MLE)**

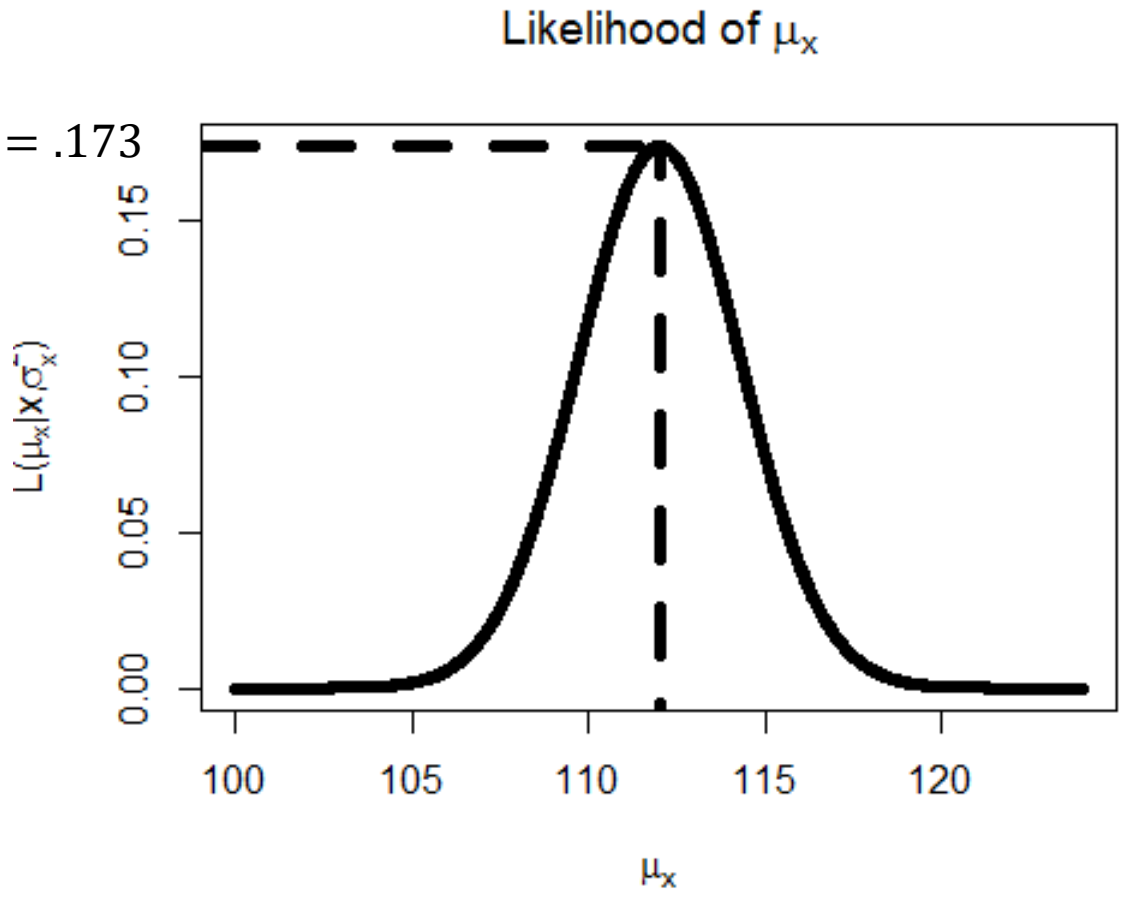
- For this one observation, what do you think the MLE would be?
- This is asking: what is the most likely mean that produced these data?

The MLE is…

The value of μ_x that maximizes $L(\mu_x|x, \sigma_x^2)$ is $\hat{\mu}_x = 112$

- The value of the likelihood function at that point is $L(112|x, \sigma_x^2) = .173$

For $\hat{\mu}_x = 112$, $L(112|x, \sigma_x^2) = .173$



From One Observation…To The Sample

The likelihood function shown previously was for one observation, but we will be working with a sample

- Assuming the sample observations are independent and identically distributed, we can form the joint distribution of the sample
- For normal distributions, this means the observations have the same mean and variance

$$L(\mu_x, \sigma_x^2 | x_1, \dots, x_N) = L(\mu_x, \sigma_x^2 | x_1) \times L(\mu_x, \sigma_x^2 | x_2) \times \cdots \times L(\mu_x, \sigma_x^2 | x_N)$$

$$= \prod_{p=1}^{N} f(x_p) = \prod_{p=1}^{N} \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left(-\frac{(x_p - \mu_x)^2}{2\sigma_x^2} \right) =$$

$$(2\pi\sigma_x^2)^{-\frac{N}{2}} \exp\left(-\sum_{p=1}^{N} \frac{(x_p - \mu_x)^2}{2\sigma_x^2} \right)$$

Multiplication comes from independence assumption:
Here, $L(\mu_x, \sigma_x^2 | x_p)$ is the univariate normal PDF for x_p, μ_x, and σ_x^2

The Sample Likelihood Function

From the previous slide:

$$L(x_1, \ldots, x_N | \mu_x, \sigma_x^2) = L = (2\pi\sigma_x^2)^{-\frac{N}{2}} \exp\left(-\sum_{p=1}^{N} \frac{(x_p - \mu_x)^2}{2\sigma_x^2} \right)$$

For this function, there is one mean (μ_x), one variance (σ_x^2), and all of the data $(x_1, \ldots, x_N)$

If we observe the data but **do not know** the mean and/or variance, then we call this the sample likelihood function

Rather than provide the height of the curve of any value of x, it provides the ***likelihood*** for any possible values of μ_x ***and*** σ_x^2

- ***Goal of Maximum Likelihood is to find* values of μ_x *and* σ_x^2 *that maximize this function***

Likelihood Function for All Five Observations

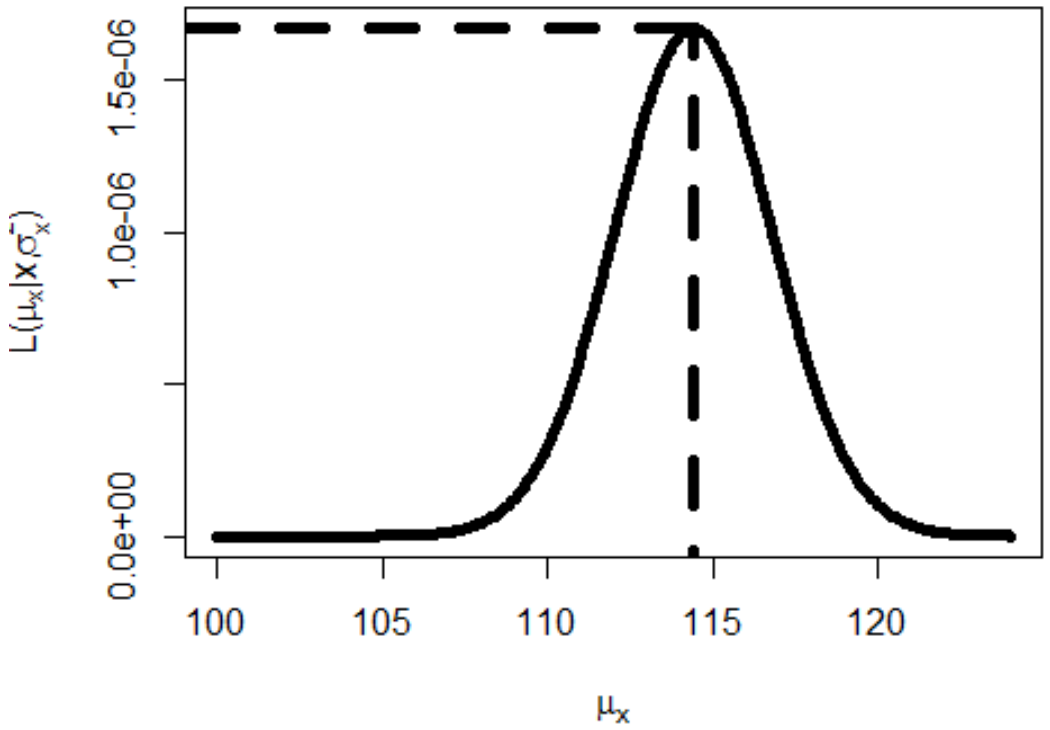
Imagine we know that $\sigma_x^2 = 5.29$ but we do not know μ_x

The likelihood function will give us the likelihood of a range of values of μ_x:

The value of μ_x where L is the maximum is the MLE for μ_x:

- $\hat{\mu}_x = 114.4$
- $L = 1.67e-06$

Note: likelihood value abbreviated as L



The Log-Likelihood Function

The likelihood function is more commonly re-expressed as the log-likelihood: $\log L = \ln(L)$

The natural log of L

$$\log L = \log L(\mu_x, \sigma_x^2 | x_1, \dots, x_N) = \log\left(L(\mu_x, \sigma_x^2 | x_1) \times L(\mu_x, \sigma_x^2 | x_2) \times \cdots \times L(\mu_x, \sigma_x^2 | x_N)\right)$$

$$= \sum_{p=1}^{N} \log L\left(\mu_x, \sigma_x^2 | x_p\right) = \log\left[(2\pi\sigma_x^2)^{-\frac{N}{2}} \exp\left(-\sum_{p=1}^{N} \frac{(x_p - \mu_x)^2}{2\sigma_x^2}\right)\right] =$$

$$-\frac{N}{2}\log(2\pi) - \frac{N}{2}\log(\sigma_x^2) - \sum_{p=1}^{N} \frac{(x_p - \mu_x)^2}{2\sigma_x^2}$$

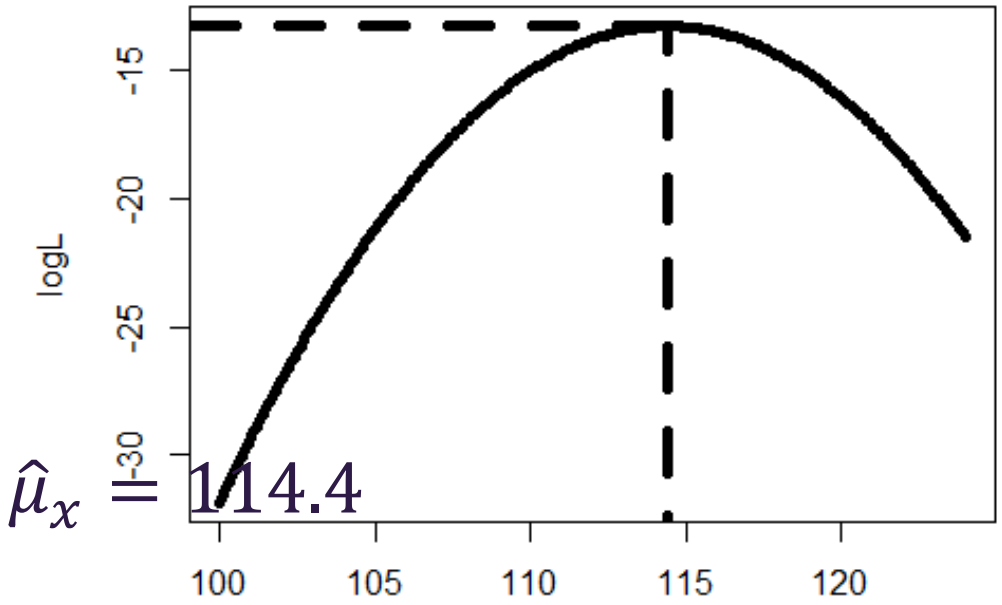
The log-likelihood and the likelihood have a maximum at the same location of μ_x and σ_x^2

Log-Likelihood Function In Use

Imagine we know that $\sigma_x^2 = 5.29$ but we do not know μ_x

The log-likelihood function will give us the likelihood of a range of possible values of μ_x

The value of μ_x where $\log L$ is the maximum is the MLE for μ_x:



$$\log L = \log 1.67e - 06 = -13.3$$

But…What About the Variance?

Up to this point, we have assumed the sample variance was known

- Not likely to happen in practice

We can jointly estimate the mean and the variance using the same log likelihood (or likelihood) function

- The variance is now a parameter in the model
- The likelihood function now will be with respect to two dimensions
  - Each unknown parameter is a dimension

$$\log L = \log L(\mu_x, \sigma_x^2 | x_1, \ldots, x_N) = -\frac{N}{2}\log(2\pi) - \frac{N}{2}\log(\sigma_x^2) - \sum_{p=1}^{N} \frac{(x_p - \mu_x)^2}{2\sigma_x^2}$$

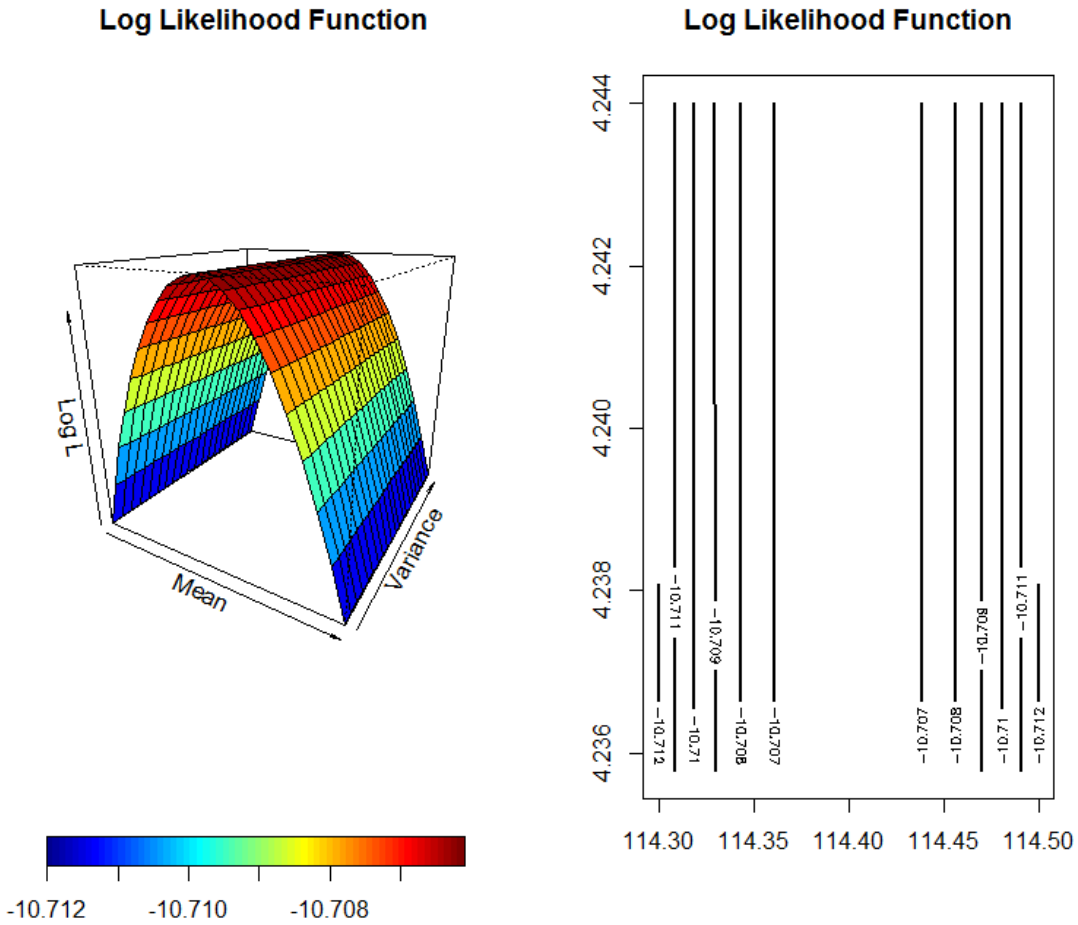
The Log Likelihood Function for Two Parameters

The point where log L is the maximum is the MLE for μ_x and σ_x^2

$$\log L = -10.7$$

Wait…$\sigma_x^2 = 4.24$?

- It was 5.29 on a previous slide
- Why?
  - Think $\frac{1}{N}$…



Maximizing the Log Likelihood Function

The process of finding the values of μ_x *and* σ_x^2 that maximize the likelihood function is complicated

- What was shown was a grid search: trial-and-error process

For relatively simple functions, we can use calculus to find the maximum of a function mathematically

- Problem: not all functions can give closed-form solutions (i.e., one solvable equation) for location of the maximum
- Solution: use efficient methods of searching for parameter (i.e., Newton-Raphson)

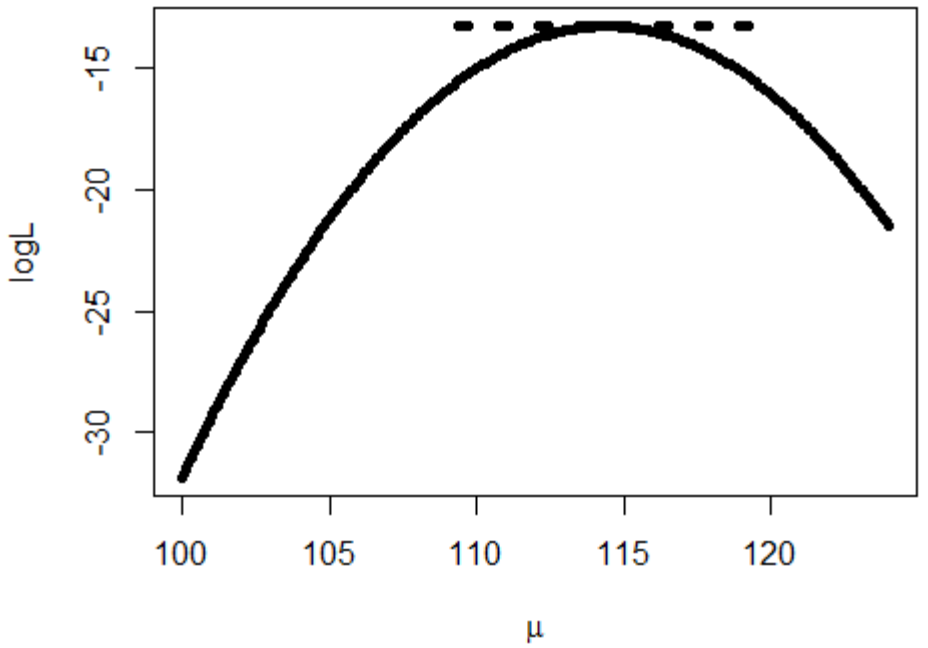
Using Calculus: The First Derivative

The calculus method to find the maximum of a function makes use of the first derivative

- Slope of line that is tangent to a point on the curve

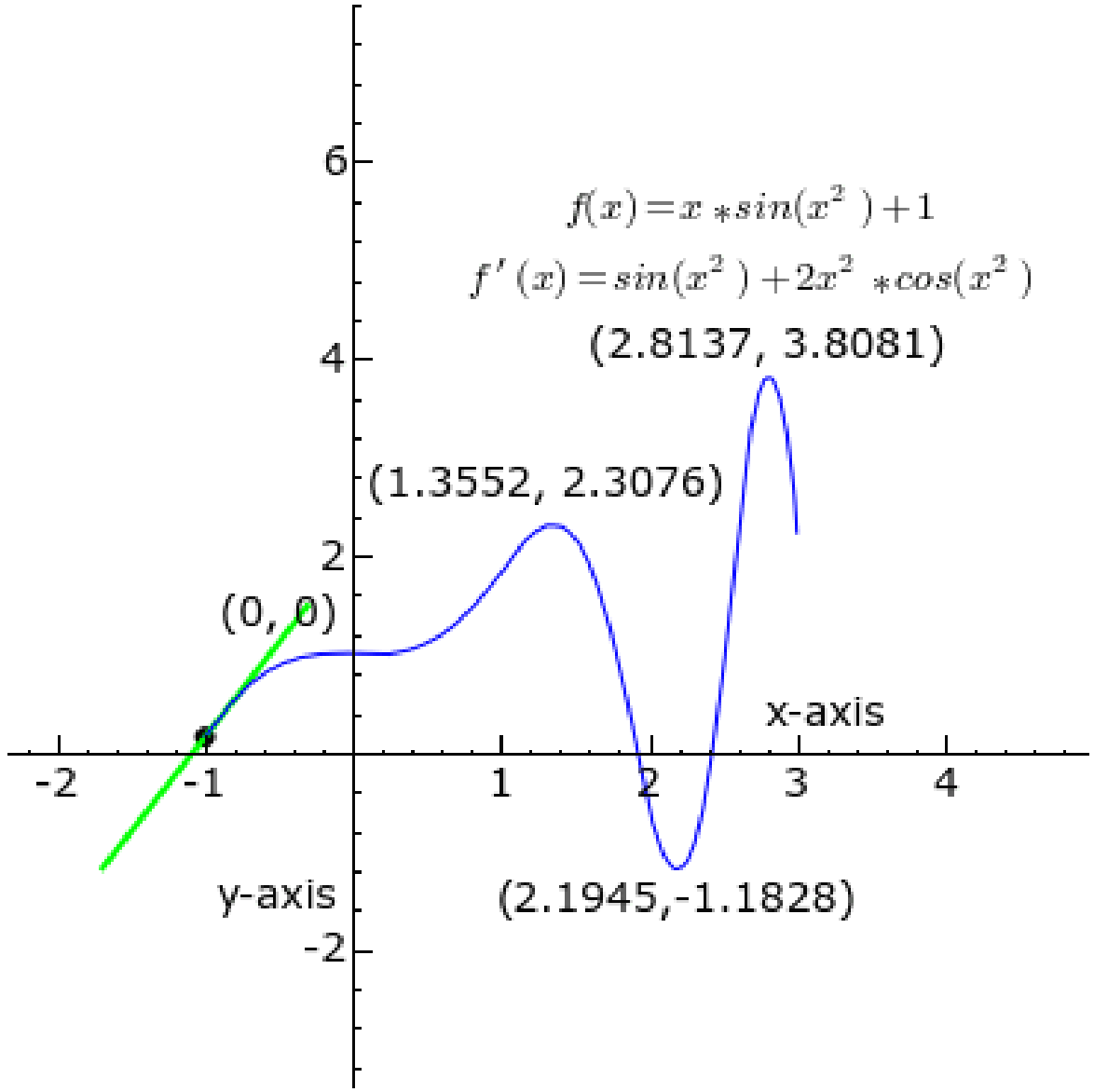
When the first derivative is zero (slope is flat), the maximum of the function is found

- Could also be at a minimum – but our functions will be inverted "U" shaped (called convex)



First Derivative = Tangent Line

From:
Wikipedia



The First Derivative for the Sample Mean

Using calculus, we can find the first derivative for the mean from our normal distribution example (the slope of the tangent line for any value of μ_x):

$$\frac{\partial \log L}{\partial \mu_x} = \frac{1}{\sigma_x^2}\left(-N\mu_x + \sum_{p=1}^{N} x_p\right)$$

To find where the maximum is, we set this equal to zero and solve for μ_x (giving us an ML estimate $\hat{\mu}_x$):

$$\frac{1}{\sigma_x^2}\left(-N\mu_x + \sum_{p=1}^{N} x_p\right) = 0 \rightarrow \hat{\mu}_x = \frac{1}{N}\sum_{p=1}^{N} x_p$$

The First Derivative for the Sample Variance

Using calculus, we can find the first derivative for the variance (slope of the tangent line for any value of σ_x^2):

$$\frac{\partial \log L}{\partial \sigma_x^2} = -\frac{\mathrm{N}}{2\sigma_x^2} + \sum_{p=1}^{N} \frac{\left(x_p - \mu_x\right)^2}{2\sigma_x^4}$$

To find where the maximum is, we set this equal to zero and solve for σ_x^2 (giving us an ML estimate $\hat{\sigma}_x^2$):

$$-\frac{\mathrm{N}}{2\sigma_x^2} + \sum_{p=1}^{N} \frac{\left(x_p - \mu_x\right)^2}{2\sigma_x^4} = 0 \rightarrow \hat{\sigma}_x^2 = \frac{1}{\mathrm{N}} \sum_{p=1}^{N} \left(x_p - \mu_x\right)^2$$

- Where the $\frac{1}{N}$ version of the variance/standard deviation comes from

Standard Errors: Using the Second Derivative

Although the estimated values of the sample mean and variance are needed, we also need the standard errors

For MLEs, the standard errors come from the **information matrix**, which is found from the square root of -1 times the inverse matrix of second derivatives (only one value for one parameter)

- Second derivative gives curvature of log-likelihood function

Variance of the sample mean:

$$\frac{\partial^2 \log L}{\partial \mu_x^2} = \frac{-N}{\sigma_x^2} \rightarrow Var(\hat{\mu}_x) = \frac{\sigma_x^2}{N}$$

ML ESTIMATION OF GLMS: THE NLME/LME4 PACKAGES IN R

Maximum Likelihood Estimation for GLMs in R: NLME and LME4

Maximum likelihood estimation of GLMs can be performed in the NLME and LME4 packages in R

- Also: SAS PROC MIXED; XTMIXED in Stata

These packages will grow in value to you as time goes on: most multivariate analyses can be run with these programs:

- Multilevel models
- Repeated measures
- Some factor analysis models

The **MIXED** part of Non-Linear/Linear Mixed Effects refers to the type of model it can estimate: **General Linear Mixed Models**

- Mixed models *extend* the GLM to be able to model dependency between observations (either within a person or within a group, or both)

Likelihood Functions in NLME and LME4

Both packages use a common (but very general) log-likelihood function based on the GLM: the conditional distribution of Y given **X**

$$f(Y_p|X_p, Z_p) \sim N(\beta_0 + \beta_1 X_p + \beta_2 Z_p + \beta_3 X_p Z_p, \sigma_e^2)$$

- Y is normally distributed conditional on the values of the predictors

The log likelihood for Y is then

$$\log L = \log L(\sigma_e^2 | x_1, \ldots, x_N) = -\frac{N}{2}\log(2\pi) - \frac{N}{2}\log(\sigma_e^2) - \sum_{p=1}^{N} \frac{\left(Y_p - \hat{Y}_p\right)^2}{2\sigma_e^2}$$

Furthermore, there is a **closed form** (a set of equations) for the fixed effects (and thus $\hat{Y}_p$) for any possible value of σ_e^2

- The programs seek to find σ_e^2 at the maximum of the log likelihood function – and after that finds everything else from equations
- Begins with a naïve guess…then uses Newton-Raphson to find maximum

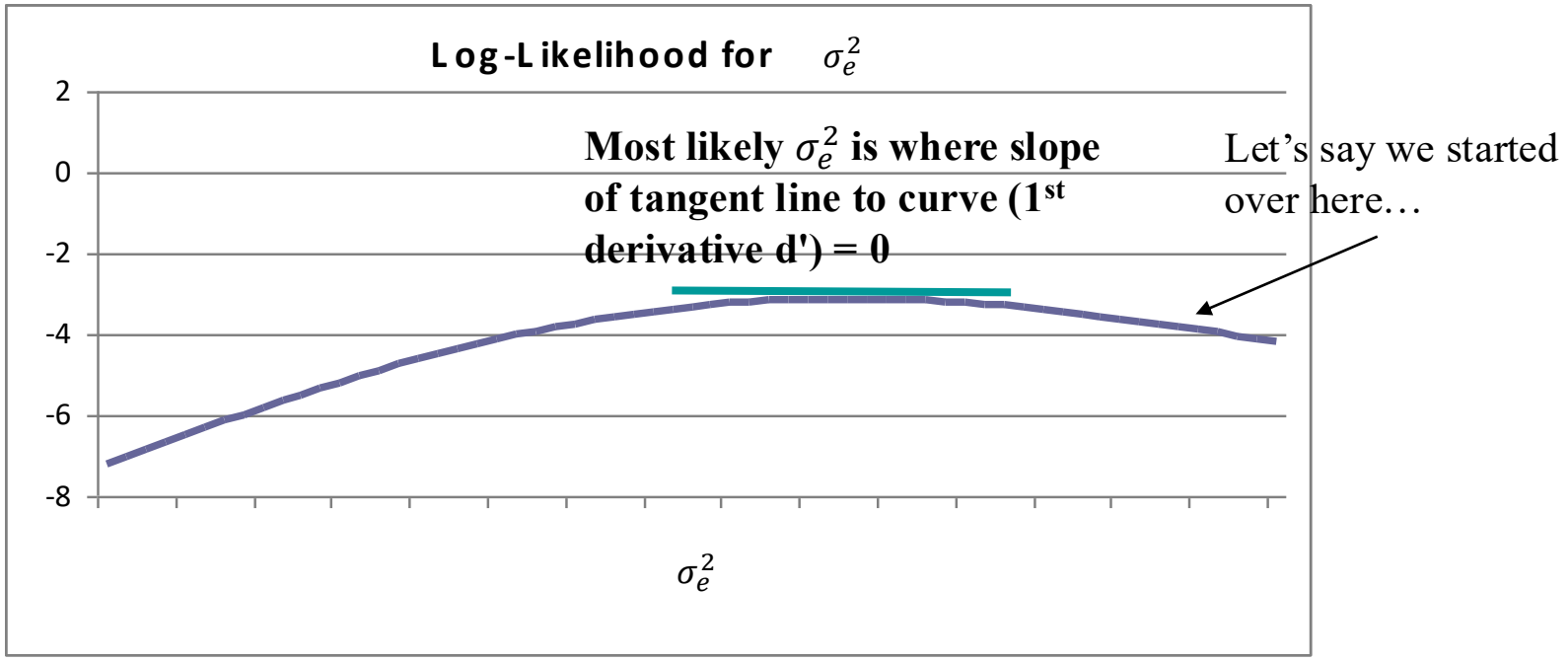
σ_e^2 Estimation via Newton Raphson

We could calculate the likelihood over wide range of σ_e^2 for each person and plot those log likelihood values to see where the peak is…

- But we have lives to lead, so we can solve it mathematically instead by finding where the slope of the likelihood function (the 1st derivative, d') = 0 (its peak)

Step 1: Start with a guess of σ_e^2, **calculate 1st derivative d'** of the log likelihood with respect to σ_e^2 at that point

- Are we there (d' = 0) yet? Positive d' = too low, negative d' = too high



σ_e^2 Estimation via Newton Raphson

Step 2: **Calculate the 2nd derivative** (slope of slope, d'') at that point

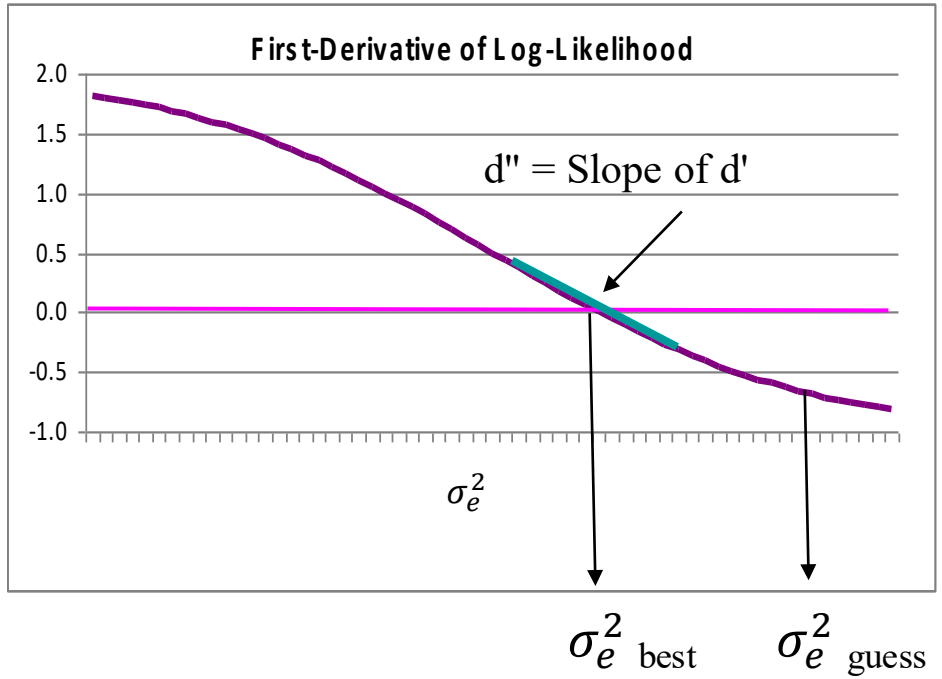
- Tells us **how far off we are**, and is used to figure out how much to adjust by
- d'' will always be negative as approach top, but d' can be positive or negative

Calculate new guess of σ_e^2 : σ_e^2 new = σ_e^2 old – (d'/d'')

- If (d'/d'') < 0 → σ_e^2 increases
  If (d'/d'') > 0 → σ_e^2 decreases
  If (d'/d'') = 0 then you are done

2nd derivative d'' also tells you how *good* of a peak you have

- Need to know where your best σ_e^2 is (at d'=0), as well as how precise it is (from d'')
- If the function is flat, d'' will be smallish
- **Want large d'' because 1/SQRT(d'') = σ_e^2's SE**



Trying it Out: Using NLME with Our Example Data

For now, we will know NLME to be largely like LM

- Even the glht function from MULTCOMP works the same

The first model will be the empty model where IQ is the DV

- Linking NLME's gls function to our previous set of slides
- After that, we will replicate a previous analysis : predicting Performance from IQ
- What we are estimating is $\sigma_x^2 = \sigma_e^2$ (the variance of IQ, used in the likelihood function) and $\beta_0^{IQ} = \mu_x$ (the mean IQ, found from equations)

The NLME function we will use is called gls

The only difference from the lm function is the inclusion of the option method="ML"

```
> model01 = gls(iq~1,data=data01,method="ML")
> summary(model01)
Generalized least squares fit by maximum likelihood
  Model: iq ~ 1
  Data: data01
      AIC      BIC    logLik
  25.4122 24.63108 -10.7061

Coefficients:
             Value Std.Error  t-value p-value
(Intercept) 114.4  1.029563 111.1151       0
```

The Basics of PROC MIXED Output

Here are some of the names of the object returned by the gls function:

```
> names(model01)
 [1] "modelStruct"  "dims"         "contrasts"    "coefficients" "varBeta"      "sigma"        "apVar"        "logLik"       "numIter"      "groups"       "call"
[12] "method"       "fitted"       "residuals"    "parAssign"    "na.action"
```

Dimensions: see Subjects and Max Obs Per Subject

```
> model01$dims
$N
[1] 5

$p
[1] 1

$REML
[1] 0
```

Note: if no results – no convergence – bad news

- If you do not have the MLE all the good things about the MLE don't apply to your results

Further Unpacking Output

The estimated σ_e is shown in the summary() function

```
Residual standard error: 2.059126
Degrees of freedom: 5 total; 4 residual
```

- Note: R found the same estimate of σ_e^2 as we did – just reported as the un-squared version
- Also: the SE of σ_e^2 is the SD of a variance – not displayed in this package but does happen in others

The Information Criteria section shows statistics that can be used for model comparisons

```
     AIC       BIC    logLik
25.4122  24.63108 -10.7061
```

Finally…the Fixed Effects

The coefficients (also referred to as fixed effects) are where the estimated regression slopes are listed – here $\beta_0^{IQ} = \mu_x$

```
Coefficients:
             Value Std.Error  t-value p-value
(Intercept) 114.4  1.029563 111.1151       0
```

- This also is the value we estimated in our example from before

Not listed: traditional ANOVA table with Sums of Squares, Mean Squares, and F statistics

- The Mean Square Error is no longer the estimate of σ_e^2: this comes directly from the model estimation algorithm itself
- The traditional R^2 change test also changes under ML estimation

```
> anova(model01)
Denom. DF: 4
            numDF  F-value p-value
(Intercept)     1 9877.253  <.0001
```

USEFUL PROPERTIES OF MAXIMUM LIKELIHOOD ESTIMATES

Useful Properties of MLEs

Next, we demonstrate three useful properties of MLEs (not just for GLMs)

- Likelihood ratio (aka Deviance) tests
- Wald tests
- Information criteria

To do so, we will consider our example where we wish to predict job performance from IQ (but will now center IQ at its mean of 114.4)

We will estimate two models, both used to demonstrate how ML estimation differs slightly from LS estimation for GLMs

- Empty model predicting just performance: $Y_p = \beta_0 + e_p$
- Model where mean centered IQ predicts performance:

$$Y_p = \beta_0 + \beta_1(IQ - 114.4) + e_p$$

R gls Syntax

Syntax for the empty model predicting performance:

```
#emtpy model predicting performance
model02 = gls(perf~1,data=data01,method="ML")
summary(model02)
```

Syntax for the conditional model where mean centered IQ predicts performance:

```
#centering IQ at mean of 114.4
data01$iq114 = data01$iq-114.4

#Regression with ML:
model03a = gls(perf~iq114,data=data01,method="ML")
summary(model03a)
```

Questions in comparing between the two models:

- How do we test the hypothesis that IQ predicts performance?
  - Likelihood ratio tests (can be multiple parameter/degree-of-freedom)
  - Wald tests (usually for one parameter)
- If IQ does significantly predict performance, what percentage of variance in performance does it account for?
  - Relative change in σ_e^2 from empty model to conditional model

Likelihood Ratio (Deviance) Tests

The likelihood value from MLEs can help to statistically test competing models assuming the models are nested

Likelihood ratio tests take the ratio of the likelihood for two models and use it as a test statistic

Using log-likelihoods, the ratio becomes a difference

- The test is sometimes called a **deviance test**

$$D = \Delta - 2\log L = -2 \times (\log L_{H0} - \log L_{HA})$$

- D is tested against a Chi-Square distribution with degrees of freedom equal to the difference in number of parameters

Deviance Test Example

Imagine we wanted to test the null hypothesis that IQ did not predict performance:

$$H_0: \beta_1 = 0$$
$$H_1: \beta_1 \neq 0$$

The difference between the empty model and the conditional model is one parameter

- Null model: one intercept β_0 and one residual variance σ_e^2 estimated = 2 parameters
- Alternative model: one intercept β_0, one slope β_1, and one residual variance σ_e^2 estimated = 3 parameters

Difference in parameters: 3-2 = 1 (will be degrees of freedom)

LRT/Deviance Test Procedure

Step #1: estimate null model (get -2*log likelihood)

```
> summary(model02)
Generalized least squares fit by maximum likelihood
  Model: perf ~ 1
  Data: data01
       AIC      BIC    logLik
  25.31696 24.53584 -10.65848
```

Step #2: estimate alternative model (get -2*log likelihood)

```
> summary(model03a)
Generalized least squares fit by maximum likelihood
  Model: perf ~ iq114
  Data: data01
       AIC      BIC    logLik
  12.92641 11.75472 -3.463204
```

Step #3: compute test statistic

$$D = -2 \times (\log L_{H0} - \log L_{HA}) = -2 \times (-10.658 - 3.463) = 14.4$$

Step #4: calculate p-value from Chi-Square Distribution with 1 DF

- I used the pchisq() function (with the upper tail)
- p-value = 0.000148

```
> lrt02v03
[1] 14.39055
> pchisq(lrt02v03,df=1,lower.tail=FALSE)
[1] 0.0001485457
```

Inference: the regression slope for IQ was significantly different from zero -- we prefer our alternative model to the null model

Interpretation: IQ significantly predicts performance

Likelihood Ratio Tests in R

R makes this process much easier by embedding likelihood ratio tests in the ANOVA() function for nested models:

```
> anova(model02,model03a)
         Model df      AIC      BIC      logLik   Test  L.Ratio p-value
model02      1  2 25.31696 24.53584 -10.658480
model03a     2  3 12.92641 11.75472  -3.463204 1 vs 2 14.39055   1e-04
```

Wald Tests (Usually 1 DF Tests in Software)

For each parameter θ, we can form the Wald statistic:

$$\omega = \frac{\hat{\theta}_{MLE} - \theta_0}{SE(\hat{\theta}_{MLE})}$$

(typically $\theta_0 = 0$)

As N gets large (goes to infinity), the Wald statistic converges to a standard normal distribution $\omega \sim N(0,1)$

- Gives us a hypothesis test of $H_0: \theta = 0$

If we divide each parameter by its standard error, we can compute the two-tailed p-value from the standard normal distribution (Z)

- Exception: bounded parameters can have issues (variances)

We can further add that variances are estimated, switching this standard normal distribution to a t distribution (R does this for us for some packages)

- Note: some don't like calling this a "true" Wald test

Wald Test Example

We could have used a Wald test to compare between the empty and conditional model, or:

$$H_0: \beta_1 = 0$$
$$H_1: \beta_1 \neq 0$$

R provides this for us in the from the summary() function:

```
> summary(model03a)
Generalized least squares fit by maximum likelihood
  Model: perf ~ iq114
  Data: data01
       AIC      BIC    logLik
  12.92641 11.75472 -3.463204

Coefficients:
                Value Std.Error  t-value p-value
(Intercept) 12.800000 0.2792623 45.83505  0.0000
iq114        0.962264 0.1356218  7.09521  0.0058
```

Note: these estimates are identical to the glht estimates from previously

Here, the slope estimate has a t-test statistic value of 7.095 (p = .0058), meaning we would reject our null hypothesis

Typically, Wald tests are used for one additional parameter

- Here, one slope

Model Comparison with R^2

To compute an R^2, we use the ML estimates of σ_e^2:

- Empty model: $\sigma_e^2 = 4.160\ (2.631)$
- Conditional model: $\sigma_e^2 = 0.234\ (0.148)$

The R^2 for variance in performance accounted for by IQ is:

$$R^2 = \frac{4.160 - 0.234}{4.160} = .944$$

- Hall of fame worthy

```
> (model02$sigma^2-model03a$sigma^2)/model02$sigma^2
[1] 0.9062651
>
```

Information Criteria

Information criteria are statistics that help determine the relative fit of a model for non-nested models

- Comparison is fit-versus-parsimony

R reports a set of criteria (from conditional model)

```
> anova(model02,model03a)
         Model df      AIC      BIC     logLik   Test  L.Ratio p-value
model02      1  2 25.31696 24.53584 -10.658480
model03a     2  3 12.92641 11.75472  -3.463204 1 vs 2 14.39055   1e-04
```

- Each uses -2*log-likelihood as a base
  - Choice of statistic is **very** arbitrary and depends on field

Best model is one with *smallest* value

Note: don't use information criteria for nested models

- LRT/Deviance tests are more powerful

How ML and LS Estimation of GLMs Differ

You may have recognized that the ML and the LS estimates of the fixed effects were identical

- And for these models, they will be

Where they differ is in their estimate of the residual variance σ_e^2:

- From Least Squares (MSE): $\sigma_e^2 = 0.390$ (no SE)
- From ML (model parameter): $\sigma_e^2 = 0.234$ (no SE in R)

The ML version uses a **biased estimate** of σ_e^2 (it is too small)

Because σ_e^2 plays a role in all SEs, the Wald tests differed from LS and ML

Troubled by this? Don't be: a fix will come in a few weeks…

- HINT: use method="REML" rather than method="ML" in gls()

Maximum Likelihood Wrap Up

This lecture was our first pass at maximum likelihood estimation

The topics discussed today apply to all statistical models,
not just GLMs

Maximum likelihood estimation of GLMs helps when the basic assumptions are obviously violated

- Independence of observations
- Homogeneous σ_e^2
- Conditional normality of Y (normality of error terms)

INTRODUCTION TO BAYESIAN STATISTICS AND MARKOV CHAIN MONTE CARLO ESTIMATION

Today's Class

An introduction to Bayesian statistics:

- What it is
- What it does
- Why people use it

An introduction to Markov Chain Monte Carlo (MCMC estimation)

- How it works
- Features to look for when using MCMC
- Why people use it

AN INTRODUCTION TO BAYESIAN STATISTICS

Bayesian Statistics: The Basics

Bayesian statistical analysis refers to the use of models where some or all of the parameters are treated as
random components

- Each parameter comes from some type of distribution

The likelihood function of the data is then augmented with an additional term that represents the likelihood of the **prior distribution** for each parameter

- Think of this as saying each parameter has a certain likelihood – the height of the prior distribution

The final estimates are then considered summaries of the **posterior distribution** of the parameter, conditional
on the data

- In practice, we use these estimates to make inferences, just as we have when using the non-Bayesian approaches to estimation (e.g., maximum likelihood/least squares)

Bayesian Statistics: Why It Is Used

Bayesian methods get used because the ***relative*** accessibility of one method of estimation (MCMC – to be discussed shortly)

There are four main reasons why people use MCMC:

1. Missing data
   - Multiple imputation: MCMC is used to estimate model parameters then "impute" data
   - More complicated models for certain types of missing data

2. Lack of software capable of handling large sized analyses
   - Have a zero-inflated negative binomial with 21 multivariate outcomes per 18 time points?

3. New models/generalizations of models not available in software
   - Have a new model?
   - Need a certain option not in your current software?

4. Membership in the cult of Bayesians
   - They believe philosophical differences exist between numbers from Bayesian analysis and other types of estimators

Bayesian Statistics: Perceptions and Issues

The use of Bayesian statistics has been controversial

- The use of certain prior distributions can produce results that are biased or reflect subjective judgment rather than objective science

Most MCMC estimation methods are **computationally intensive**

- Until recently, very few methods available for those who aren't into programming in Fortran, C, or C++

Understanding of what Bayesian methods are and how they work is limited outside the field of mathematical statistics

- Especially the case in the educational and social sciences

Over the past 20 years, Bayesian methods have become widespread – making new models estimable and becoming standard in some social science fields (quantitative psychology and educational measurement)

HOW BAYESIAN METHODS WORK

How Bayesian Statistics Work

The term Bayesian refers to Thomas Bayes (1701-1761)

- Formulated Bayes' Theorem

Bayesian methods rely on Bayes' Theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$P(A)$ is the **prior distribution (pdf) of A → WHY THINGS ARE BAYESIAN**
$P(B)$ is the **marginal distribution (pdf) of B**
$P(B|A)$ is the **conditional distribution (pdf) of B, given A**
$P(A|B)$ is the **posterior distribution (pdf) of A, given B**

Bayes' Theorem Example…

- Imagine a patient takes a test for a rare disease (present 1% of the population) that has a 95% accuracy rate…what is the probability the patient actually has the disease?

Sidebar: My Pilgrimage to Bayes' Tomb

Bunhill Fields (London, England)

Bayes' Theorem Example

Imagine a patient takes a test for a rare disease (present 1% of the population) that has a 95% accuracy rate…what is the probability the patient actually has the disease?

- D = the case where the person actually has the disease
- ND = the case where the person does not have the disease
- + = the test for the disease is positive

The question is asking for: P(D|+)

From Bayes' Theorem:

$$P(D|+) = \frac{P(+|D)P(D)}{P(+)}$$

What we know:

$$P(D) = .01$$
$$P(+|D) = .95$$

Back to Distributions

We don't know $P(+)$ directly from the problem, but we can figure it out if we recall how distributions work:

- $P(+)$ is a marginal distribution
- $P(+|D)$ is a conditional distribution

We can get to the marginal by summing across the conditional:

$$P(+) = P(+|D)P(D) + P(+|ND)P(ND)$$
$$= .95 * .01 + .05 * .99 = .059$$

So, to figure out the answer, if a person tests positive for the disease, the **posterior probability** they actually have the disease is:

$$P(D|+) = \frac{P(+|D)P(D)}{P(+)} = \frac{.01 * .99}{.059} = .17$$

A (Perhaps) More Relevant Example

The old-fashioned Bayes' Theorem example I've found to be difficult to generalize to your actual data, so…

Imagine you administer an IQ test to a sample of 50 people

- y_p = person p's IQ test score

To put this into a linear-models context, the empty model for Y:

$$y_p = \beta_0 + e_p$$

Where $e_p \sim N(0, \sigma_e^2)$

From this empty model, we know that:

- β_0 is the mean of the Y (the mean IQ)
- σ_e^2 is the sample variance of Y
- The conditional distribution of Y is then: $f(y_p|\beta_0, \sigma_e^2) \sim N(\beta_0, \sigma_e^2)$

Non-Bayesian Analysis

Up to this point in the class, we have analyzed these data using ML and REML

For ML, we maximized the joint likelihood of the sample with respect to the two unknown parameters β_0 and σ_e^2

$$L(\beta_0, \sigma_e^2) = \prod_{p=1}^{N} f(y_p|\beta_0, \sigma_e^2) = \prod_{p=1}^{N} \frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left(-\frac{(y_p - \beta_0)^2}{2\sigma_e^2}\right)$$

Here, using gls(), I found:

$$\beta_0 = 102.769$$
$$\sigma_e^2 = 239.490$$

Also, I found:

$$LogL = -207.91$$

Setting up a Bayesian Approach

The (fully) Bayesian approach would treat each parameter as a random instance from some **prior distribution**

Let's say you know that this version of the IQ test is supposed to have a mean of 100 and a standard deviation of 15

- So β_0 should be 100 and σ_e^2 should be 225

Going a step further, let's say you have seen results for administrations of this test that led you to believe that the mean came from a normal distribution with a SD of 2.13

- This indicates the prior distribution for the **mean**…or

$$f(\beta_0) \sim N(100, 2.13^2)$$

Let's also say that you don't really have an idea as for the distribution of the variance, but you have seen it range from 200 to 400, so we can come up with a prior distribution for the **variance** of:

$$f(\sigma_e^2) \sim U(200, 400)$$

Here the prior is a uniform distribution meaning all values from 200 to 400 are equally likely

More on the Bayesian Approach

The Bayesian approach is now to seek to find the **posterior distribution** of the parameters given the data:

$$f\left(\beta_0, \sigma_e^2 | \boldsymbol{y}_p\right)$$

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

We can again use Bayes' Theorem (but for continuous parameters):

$$f\left(\beta_0, \sigma_e^2 | \boldsymbol{y}_p\right) = \frac{f\left(\boldsymbol{y}_p | \beta_0, \sigma_e^2\right) f(\beta_0, \sigma_e^2)}{f\left(\boldsymbol{y}_p\right)} = \frac{f\left(\boldsymbol{y}_p | \beta_0, \sigma_e^2\right) f(\beta_0) f(\sigma_e^2)}{f\left(\boldsymbol{y}_p\right)}$$

Because $f\left(\boldsymbol{y}_p\right)$ essentially is a constant (which involves integrating across β_0 and σ_e^2 to find its value), this term is often referred to as:

$$f\left(\beta_0, \sigma_e^2 | \boldsymbol{y}_p\right) \propto f\left(\boldsymbol{y}_p | \beta_0, \sigma_e^2\right) f(\beta_0) f(\sigma_e^2)$$

The symbol $\propto$ is read as "is proportional to" – meaning it is the same as when multiplied by a constant

- So it is the same for all values of β_0 and σ_e^2

Unpacking the Posterior Distribution

$f(\boldsymbol{y}_p|\beta_0, \sigma_e^2)$ is the **conditional distribution** of the data given the parameters – we know this already from our linear model (slide 12)

$$f(\boldsymbol{y}_p|\beta_0, \sigma_e^2) = \prod_{p=1}^{N} f(y_p|\beta_0, \sigma_e^2) = \prod_{p=1}^{N} \frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left(-\frac{(y_p - \beta_0)^2}{2\sigma_e^2}\right)$$

$f(\beta_0)$ is the **prior distribution** of β_0, which we decided would be $N(100, 2.13^2)$, giving the height of any β_0:

$$f(\beta_0) = \frac{1}{\sqrt{2\pi\sigma_{\beta_0}^2}} \exp\left(-\frac{(\beta_0 - \mu_{\beta_0})^2}{2\sigma_{\beta_0}^2}\right) = \frac{1}{\sqrt{2\pi * 2.13^2}} \exp\left(-\frac{(\beta_0 - 100)^2}{2 * 2.13^2}\right)$$

Unpacking the Posterior Distribution

$f(\sigma_e^2)$ is the **prior distribution** of σ_e^2, which we decided would be $\mathrm{U}(200,400)$, giving the height of any value of σ_e^2 as:

$$f(\sigma_e^2) = \frac{1}{b_{\sigma_e^2} - a_{\sigma_e^2}} = \frac{1}{400 - 200} = \frac{1}{200} = .005$$

Some useful terminology:

- The parameters of the model (for the data) get prior distributions
- The prior distributions each have parameters – these parameters are called **hyper-parameters**
- The hyper-parameters are not estimated in our example, but could be – giving us a case where we would call our priors **empirical priors**
  - **AKA random intercept variance**

Up Next: Estimation (first using non-MCMC)

Although MCMC is commonly thought of as the only method for Bayesian estimation, there are several other forms

The form analogous to ML (where the value of the parameters that maximize the likelihood or log-likelihood) is called **Maximum (or Modal) a Posteriori estimation (MAP)**

- The term modal comes from the maximum point coming at the peak (the mode) of the posterior distribution

In practice, this functions similar to ML, only instead of maximizing the joint likelihood of the data, we now have to worry about the prior:

$$f\left(\beta_0, \sigma_e^2 | \boldsymbol{y}_p\right) = \frac{f\left(\boldsymbol{y}_p | \beta_0, \sigma_e^2\right) f(\beta_0) f(\sigma_e^2)}{f\left(\boldsymbol{y}_p\right)} \propto f\left(\boldsymbol{y}_p | \beta_0, \sigma_e^2\right) f(\beta_0) f(\sigma_e^2)$$

Because it is often more easy to work with, the log of this is often used:

$$\log\left(f\left(\beta_0, \sigma_e^2 | \boldsymbol{y}_p\right)\right) \propto \log f\left(\boldsymbol{y}_p | \beta_0, \sigma_e^2\right) + \log f(\beta_0) + \log f(\sigma_e^2)$$

Grid Searching for the MAP Estimate of $\boldsymbol{\beta_0}$

To demonstrate, let's imagine we know $\sigma_e^2 = 239.490$

Later we won't know this…when we use MCMC

We will use Excel to search over a grid of possible values for β_0

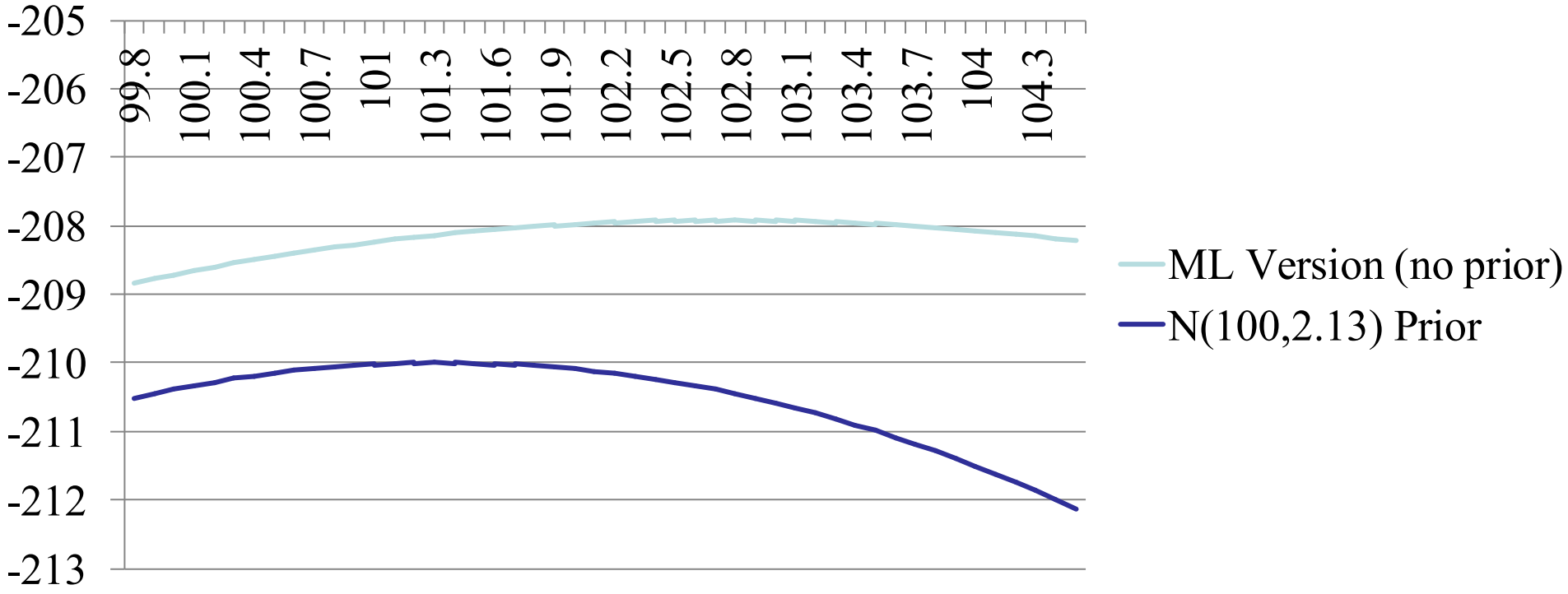
In each, we will use $\log f(\boldsymbol{y}_p|\beta_0) + \log f(\beta_0)$

As a comparison, we will also search over the ML log likelihood function $\log f(\boldsymbol{y}_p|\beta_0)$

ML v. Prior for β_0 of N(100, 2.13^2)

Maximum for ML: 102.8

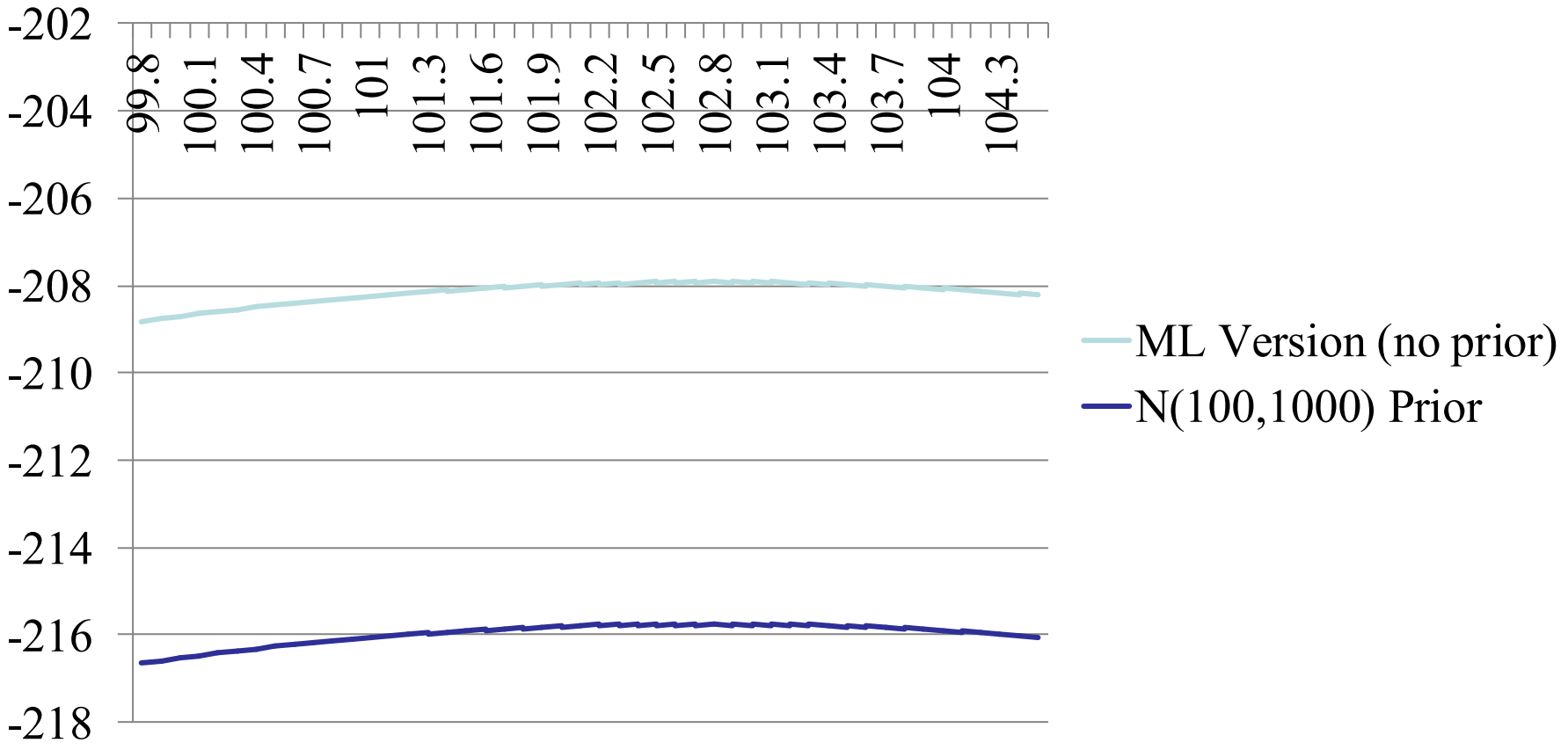
Maximum for Bayes: 101.4
(estimate is closer to mean of prior)



ML vs. Prior for β_0 of N(100, 1000^2)

Maximum for ML: 102.8

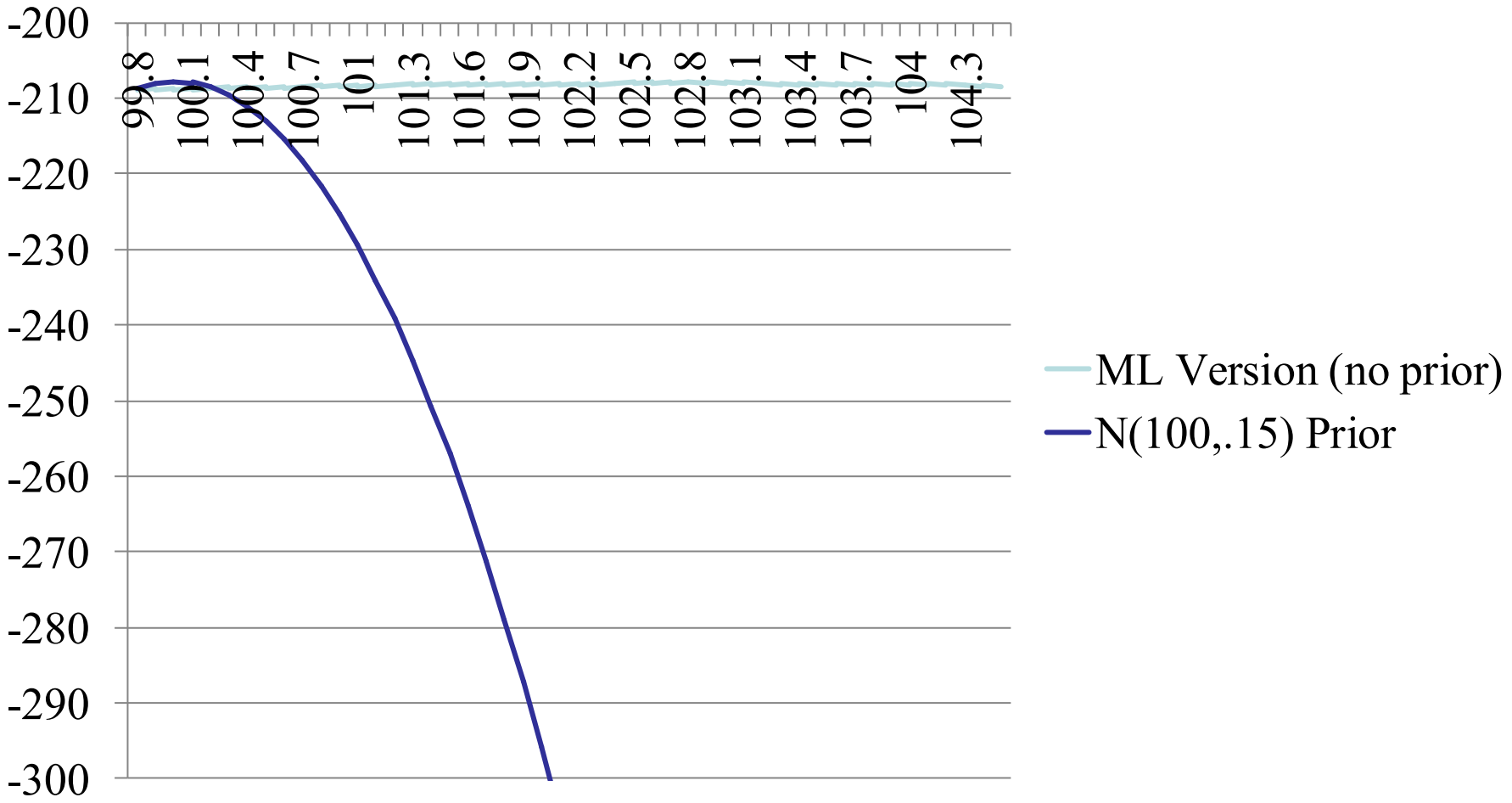
Maximum for Bayes: 102.8



ML vs. Prior for β_0 of N(100, 0.15^2)

Maximum for ML: 102.8

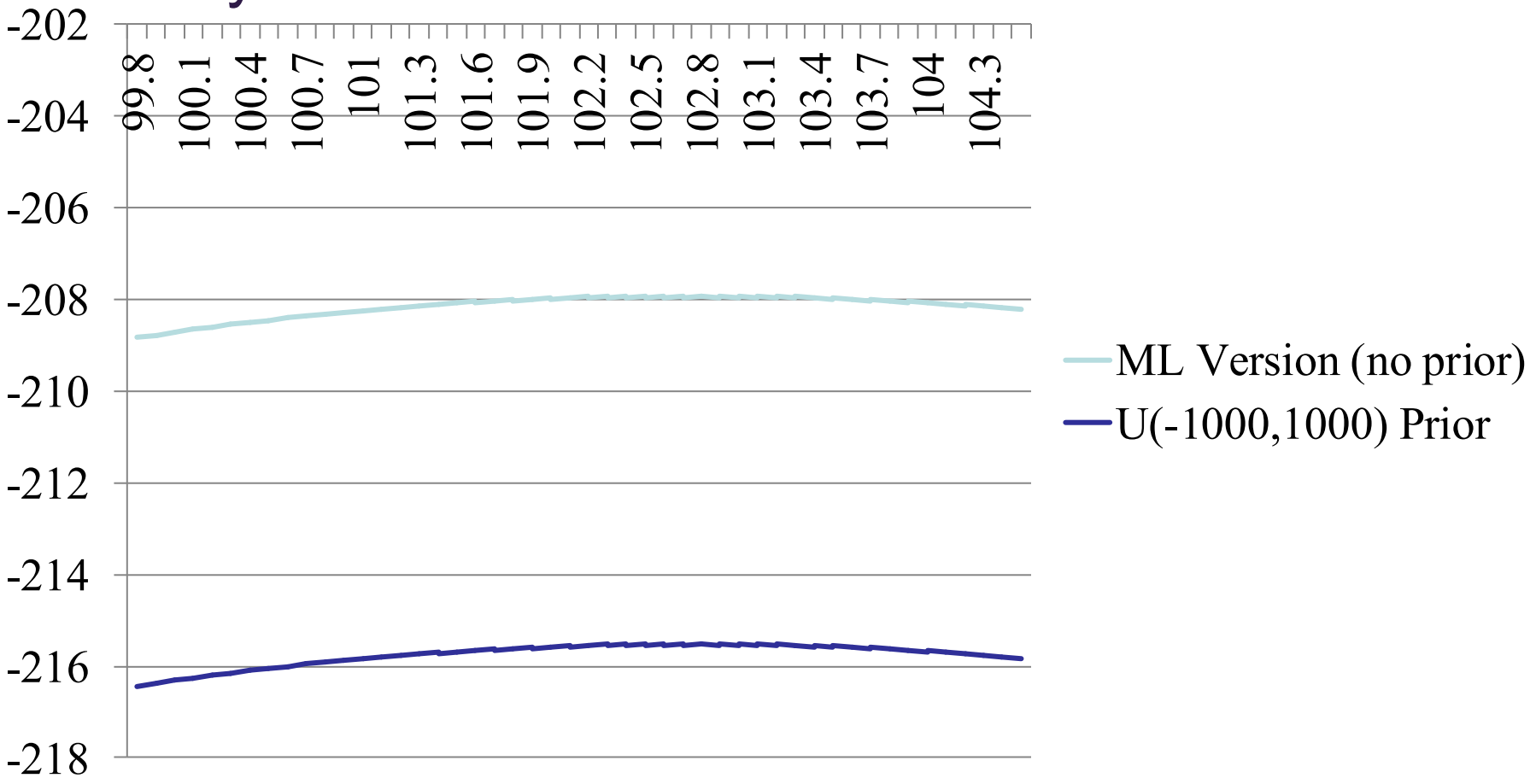
Maximum for Bayes: 100



ML vs. Prior for β_0 of U(-1000,1000)

Maximum for ML: 102.8

Maximum for Bayes: 102.8



Summarizing Bayesian So Far

Bayesian → parameters have prior distributions

Estimation in Bayesian → MAP estimation is much like estimation in ML, only instead of likelihood of data, now have to add in likelihood for prior of all parameters

- But…MAP estimation may be difficult as figuring out derivatives for gradient function (for Newton-Raphson) are not always easy
- Where they are easy: **Conjugate** priors → prior distributions that are the same as the posterior distribution (think multilevel with normal outcomes)

Priors can be **informative** (highly peaked) or **uninformative** (not peaked)

- Some uninformative priors will give MAP estimates that are equal to ML

Up next: estimation by brute force: Markov Chain Monte Carlo

MARKOV CHAIN MONTE CARLO ESTIMATION: THE BASICS

How Estimation Works (More or Less)

Most estimation routines do one of three things:

1. **Minimize Something:** Typically found with names that have “least” in the title. Forms of least squares include “Generalized”, “Ordinary”, “Weighted”, “Diagonally Weighted”, “WLSMV”, and “Iteratively Reweighted.” Typically the estimator of last resort…

2. **Maximize Something:** Typically found with names that have “maximum” in the title. Forms include “Maximum likelihood”, “ML”, “Residual Maximum Likelihood” (REML), “Robust ML”. Typically the gold standard of estimators (and we now know why).

3. **Use Simulation to Sample from Something:** more recent advances in simulation use resampling techniques. Names include “Bayesian Markov Chain Monte Carlo”, “Gibbs Sampling”, “Metropolis Hastings”, “Metropolis Algorithm”, and “Monte Carlo”. Used for complex models where ML is not available or for methods where prior values are needed.

How MCMC Estimation Works

MCMC estimation works by taking samples from the posterior distribution of the data given the parameters:

$$f\left(\beta_0, \sigma_e^2 \middle| \boldsymbol{y}_p\right) = \frac{f\left(\boldsymbol{y}_p \middle| \beta_0, \sigma_e^2\right) f(\beta_0) f(\sigma_e^2)}{f\left(\boldsymbol{y}_p\right)}$$

- How is that possible? We don't know $f(\boldsymbol{y}_p)$…but…we'll see…

After enough values are drawn, a rough shape of the distribution can be formed

- From that shape we can take summaries and make them our parameters (i.e., mean)

How the sampling mechanism happens comes from several different algorithms that you will hear about, the most popular being:

- **Gibbs Sampling**: used when $f\left(\beta_0, \sigma_e^2 \middle| \boldsymbol{y}_p\right)$ is known
  - Parameter values are drawn and kept throughout the chain
- **Metropolis-Hastings (within Gibbs):** used when $f\left(\beta_0, \sigma_e^2 \middle| \boldsymbol{y}_p\right)$ is unknown
  - Parameter values are proposed, then either kept or rejected
  - SAS PROC MCMC uses the latter
  - TRIVIA NOTE: The Metropolis algorithm comes from Chemistry (in 1950)
- **Hybrid MC:** Newer versions (1980s; implemented in Stan)

In some fields (Physics in particular), MCMC estimation is referred to as Monte Carlo estimation

MCMC Estimation with MHG

The Metropolis-Hastings algorithm works a bit differently than Gibbs sampling:

1. Each parameter (here β_0 and σ_e^2) is given an initial value
2. In order, a new value is proposed for each model parameter from some distribution:

$$\beta_0^* \sim Q(\beta_0^*|\beta_0); \sigma_e^{2^*} \sim Q\left(\sigma_e^{2^*}\middle|\sigma_e^2\right)$$

3. The proposed value is then accepted as the current value with probability $\max(r_{MHG}, 1)$:

$$r_{MHG} = \frac{f\left(\boldsymbol{y}_p\middle|\beta_0^*, \sigma_e^{2^*}\right) f(\beta_0^*) f\left(\sigma_e^{2^*}\right) Q\left(\beta_0\middle|\beta_0^*\right) Q\left(\sigma_e^2\middle|\sigma_e^{2^*}\right)}{f\left(\boldsymbol{y}_p\middle|\beta_0, \sigma_e^2\right) f(\beta_0) f(\sigma_e^2) Q(\beta_0^*|\beta_0) Q\left(\sigma_e^{2^*}\middle|\sigma_e^2\right)}$$

4. The process continues for a pre-specified number of iterations (more is better)

Notes About MHG

The constant in the denominator of the posterior distribution:

$$f(\beta_0, \sigma_e^2 | \boldsymbol{y}_p) = \frac{f(\boldsymbol{y}_p | \beta_0, \sigma_e^2) f(\beta_0) f(\sigma_e^2)}{f(\boldsymbol{y}_p)}$$

…cancels when the ratio is formed

The proposal distributions $Q(\beta_0^* | \beta_0)$ and $Q(\sigma_e^{2^*} | \sigma_e^2)$ can literally be any statistical distribution

- The trick is picking ones that make the chain "converge" quickly
- Want to find values that lead to moderate number of accepted parameters
- Most Bayesian programs don't make you pick these

Given a long enough chain, the final values of the chain will come from the posterior distribution

- From that you can get your parameter estimates

Introducing Jags…

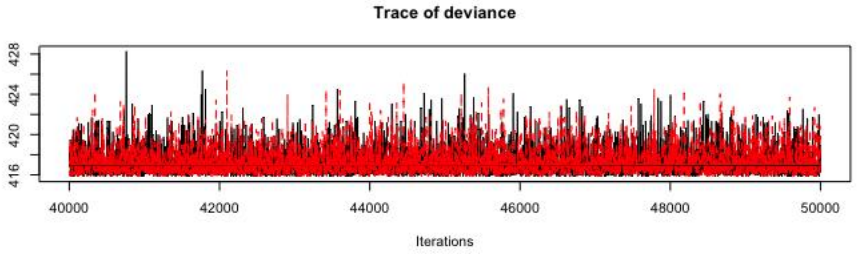
```
# estimation with Bayesian

model01Bayes = function(){

  # likelihood
  for (i in 1:n){
    y[i] ~ dnorm(mu, tau)
  }

  #priors
  mu ~ dnorm(100, 1/2.13^2)
  tau ~ dunif(1/400, 1/200)
  sigma2 = 1/tau
}

data = list(y = dataIQ$y, n = nrow(dataIQ))

jags.param = c("mu", "tau", "sigma2")

fit <- jags.parallel(data=data,
                     parameters.to.save=jags.param,
                     n.iter=50000, n.chains=2,n.thin=2,n.burnin=40000,
                     model.file=model01Bayes)
```

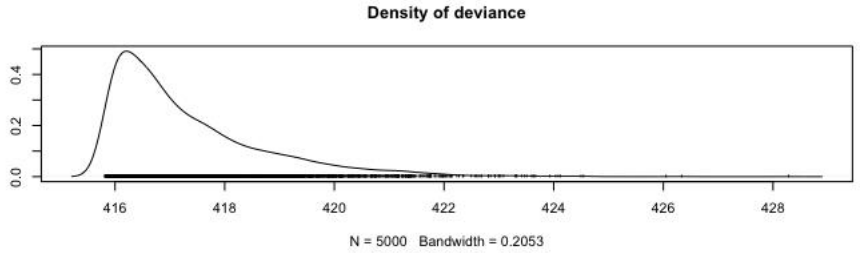
Iteration History from JAGS

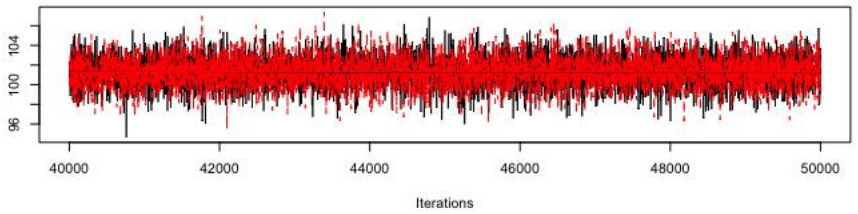
Filter

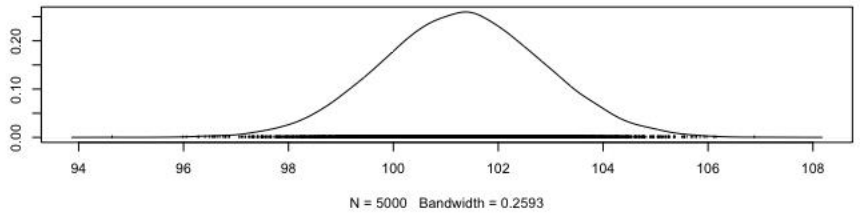
	deviance	mu	sigma2	tau
1	416.7964	100.85794	267.4554	0.003738941
2	418.1109	102.33473	328.7272	0.003042036
3	416.7720	100.62472	242.2023	0.004128781
4	416.7956	100.59997	246.9883	0.004048774
5	416.7415	100.74722	257.4887	0.003883666
6	417.8188	99.76745	230.0463	0.004346951
7	419.4729	98.87860	298.0748	0.003354863
8	415.9143	102.97544	254.1412	0.003934821
9	416.2015	101.86554	219.4816	0.004556191
10	417.2012	102.11906	303.9016	0.003290539
11	415.8802	103.19065	247.1723	0.004045762
12	417.0017	101.14455	285.1704	0.003506675
13	417.2518	100.24214	229.8218	0.004351197
14	415.8366	103.05208	240.4516	0.004158841
15	416.1867	104.09353	243.5673	0.004105641
16	416.5808	100.85335	244.6015	0.004088282
17	416.4756	101.11430	228.4360	0.004377594
18	419.4313	99.29725	314.4079	0.003180582
19	416.1548	101.89660	221.1416	0.004521989
20	416.5363	101.11042	224.6534	0.004451302
21	416.2943	101.30912	250.4290	0.003993148
22	415.8509	102.77879	248.1755	0.004029407
23	419.3382	98.95027	296.3523	0.003374363
24	415.9175	103.40843	245.3931	0.004075094
25	417.6465	102.22143	316.6267	0.003158293
26	420.4722	103.00402	381.0396	0.002624399
27	416.8376	101.28403	208.8647	0.004787788
28	417.6739	100.28041	287.1524	0.003482472
29	417.9725	104.73157	310.0394	0.003225396

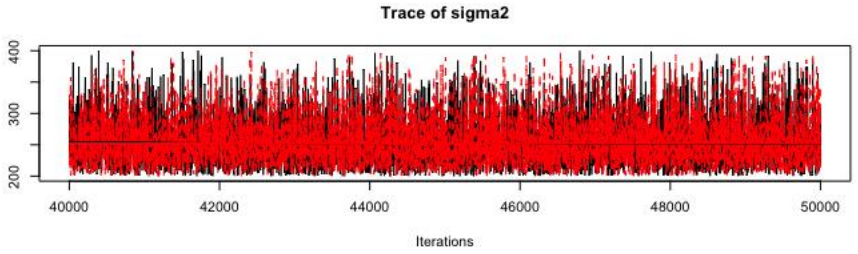
Examining the Chain and Posteriors

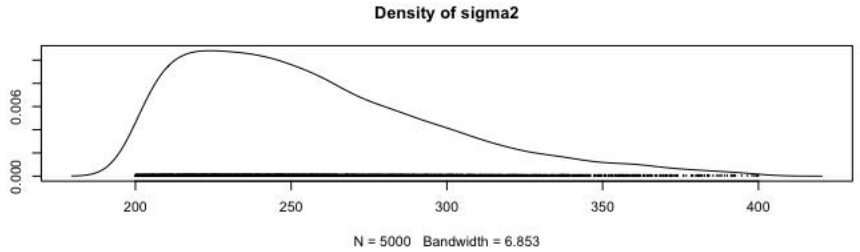


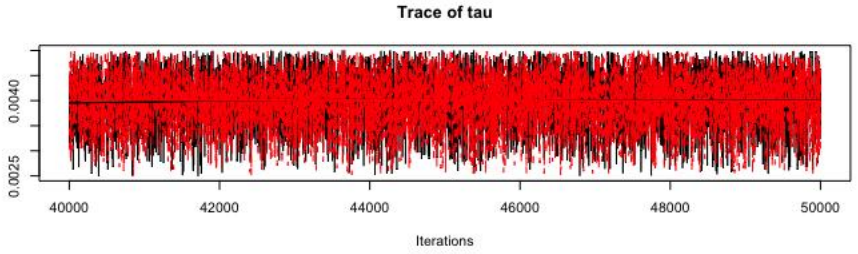


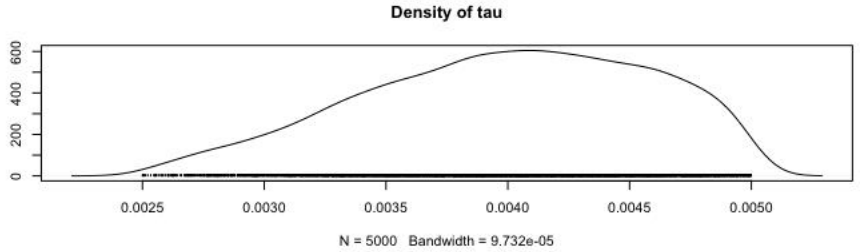












Practical Specifics in MCMC Estimation

A **burn-in** period is used where a chain is run for a set number of iterations before the sampled parameter values are used in the posterior distribution

Because of the rejection/acceptance process, any two iterations are likely to have a high correlation (called **autocorrelation**) → posterior chains use a **thinning interval** to take every Xth sample to reduce the autocorrelation

- A high autocorrelation may indicate the standard error of the posterior distribution will be smaller than it should be

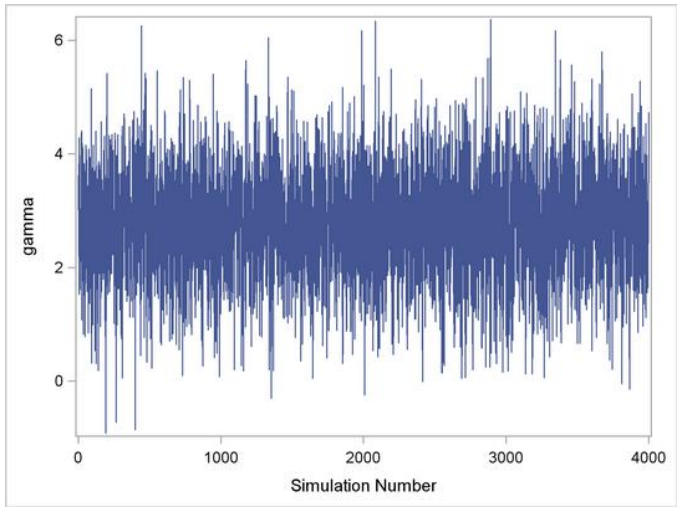
The **chain length** (and sometimes number of chains) must also be long enough so the rejection/acceptance process can reasonably approximate the posterior distribution

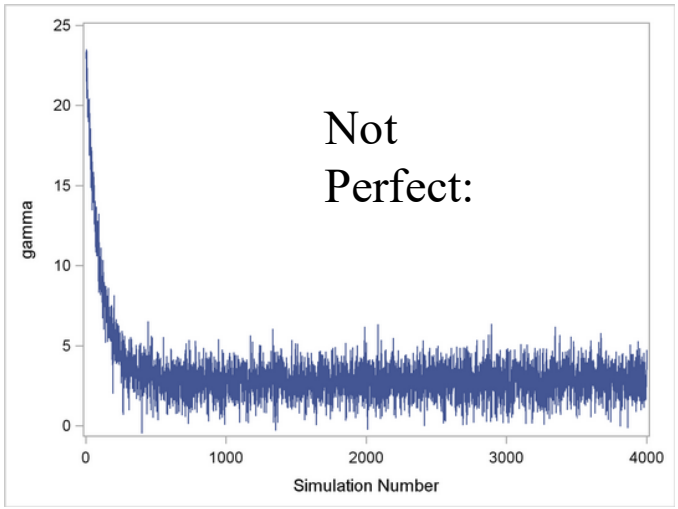
How does one what values to pick for these? Output diagnostics

- Trial. And. Error.

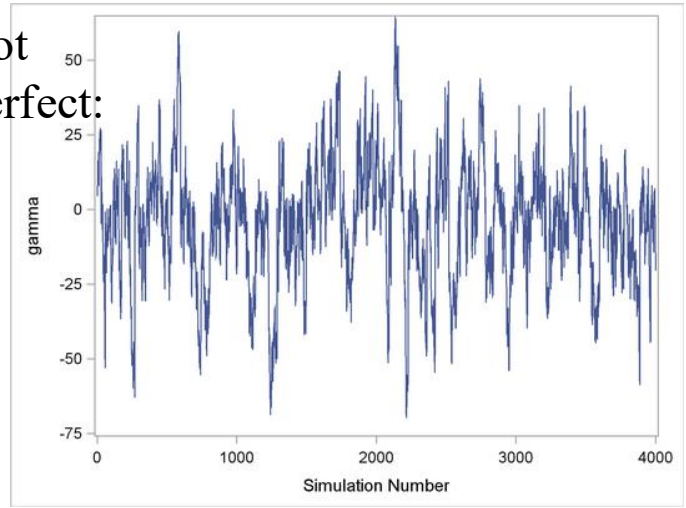
Best Output Diagnostics: the Eye Ball Test

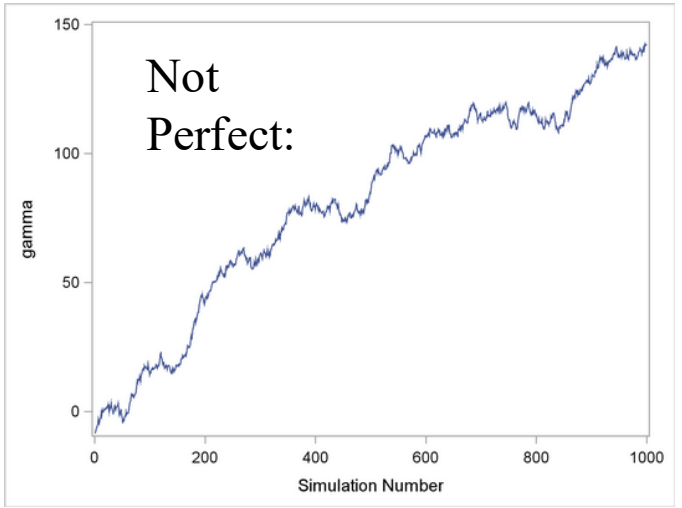
Perfect:





Not
Perfect:





Output Statistics and Diagnostics

```
> fit
Inference for Bugs model at "model01Bayes", fit using jags,
 2 chains, each with 50000 iterations (first 40000 discarded), n.thin = 2
 n.sims = 10000 iterations saved
         mu.vect sd.vect    2.5%     25%     50%     75%   97.5%  Rhat n.eff
mu       101.312   1.546  98.280 100.279 101.316 102.347 104.349 1.001 10000
sigma2   256.627  40.791 202.918 224.724 248.240 280.247 358.019 1.001 10000
tau        0.004   0.001   0.003   0.004   0.004   0.004   0.005 1.001 10000
deviance 417.317   1.404 415.862 416.284 416.869 417.922 421.059 1.001 10000

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2)
pD = 1.0 and DIC = 418.3
DIC is an estimate of expected predictive error (lower deviance is better).
```

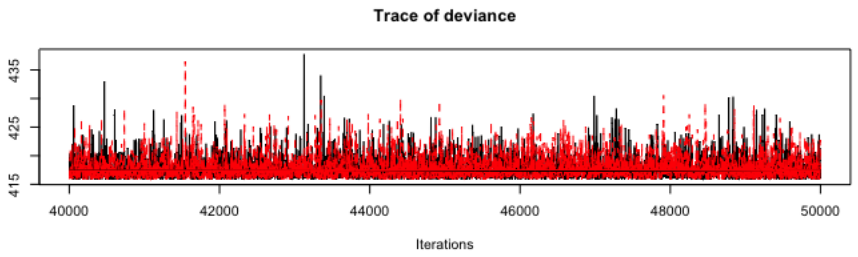
Changing Up the Prior

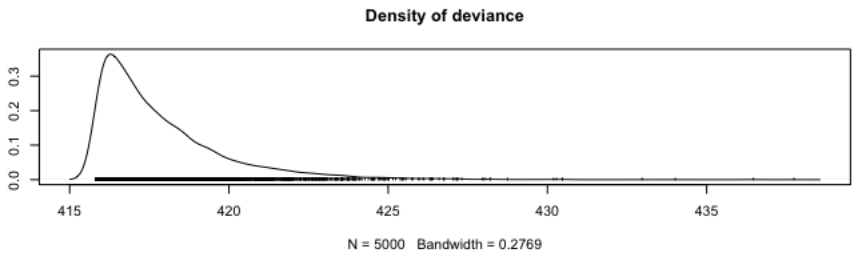
To demonstrate how changing the prior affects the analysis, we will now try a few prior distributions for our parameters

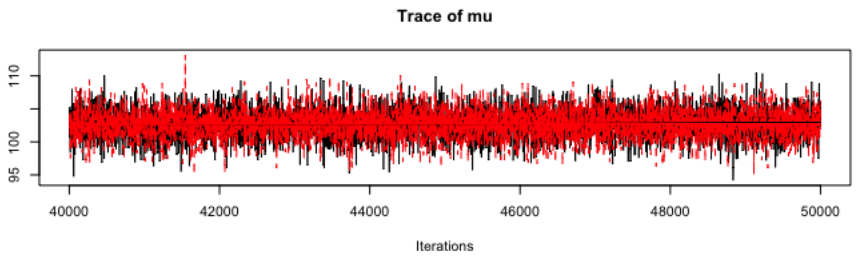
Prior: $\beta_0 \sim U(-10000,10000); \sigma_e^2 \sim U(0,5000)$

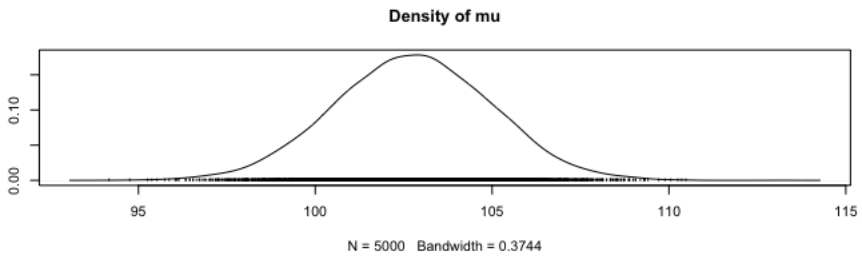
```
> fit2
Inference for Bugs model at "model02Bayes", fit using jags,
 2 chains, each with 50000 iterations (first 40000 discarded), n.thin = 2
 n.sims = 10000 iterations saved
         mu.vect sd.vect    2.5%     25%     50%     75%   97.5%  Rhat n.eff
mu       102.750   2.229  98.385 101.239 102.758 104.251 107.100 1.001 10000
sigma2   244.899  51.326 164.789 208.259 238.353 273.458 362.843 1.001 10000
tau        0.004   0.001   0.003   0.004   0.004   0.005   0.006 1.001 10000
deviance 417.856   2.028 415.869 416.415 417.241 418.624 423.285 1.001  3200

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2)
pD = 2.1 and DIC = 419.9
DIC is an estimate of expected predictive error (lower deviance is better).
```

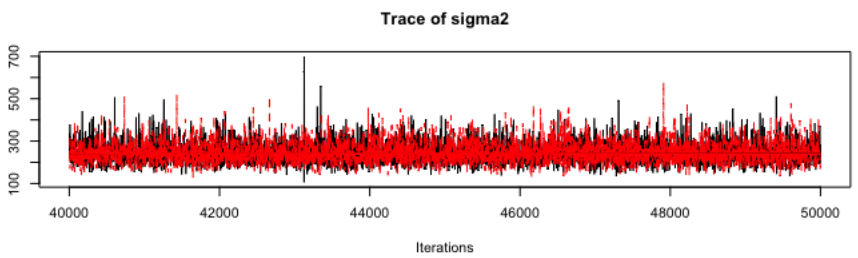
Chain Plots

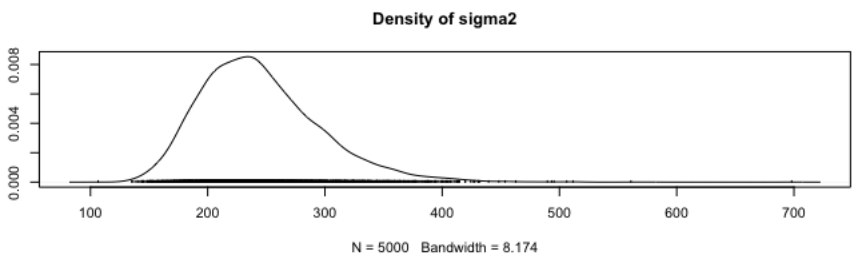

Trace of deviance
435
425
415
40000
42000
44000
46000
48000
50000
Iterations

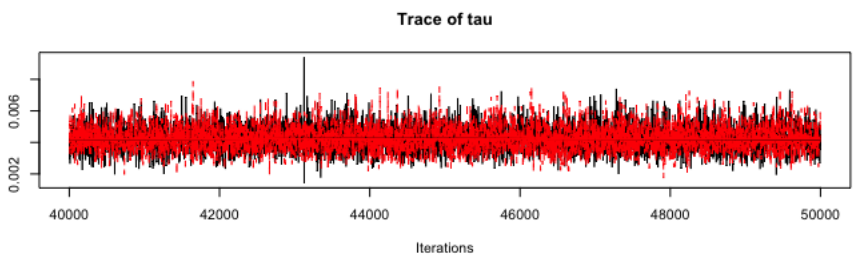

Density of deviance
0.3
0.2
0.1
0.0
415
420
425
430
435
N = 5000 Bandwidth = 0.2769

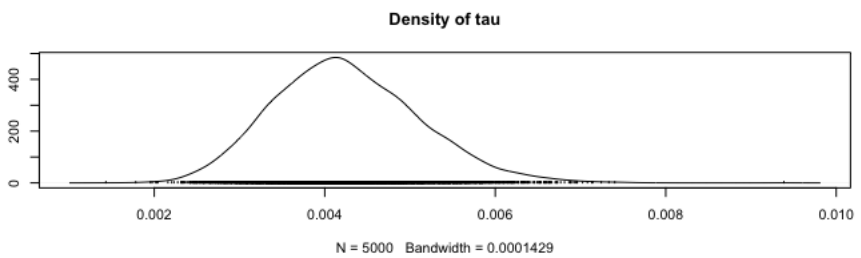

Trace of mu
110
100
95
40000
42000
44000
46000
48000
50000
Iterations


Density of mu
0.10
0.00
95
100
105
110
115
N = 5000 Bandwidth = 0.3744


Trace of sigma2
700
500
300
100
40000
42000
44000
46000
48000
50000
Iterations


Density of sigma2
0.008
0.004
0.000
100
200
300
400
500
600
700
N = 5000 Bandwidth = 8.174


Trace of tau
0.006
0.002
40000
42000
44000
46000
48000
50000
Iterations


Density of tau
400
200
0
0.002
0.004
0.006
0.008
0.010
N = 5000 Bandwidth = 0.0001429

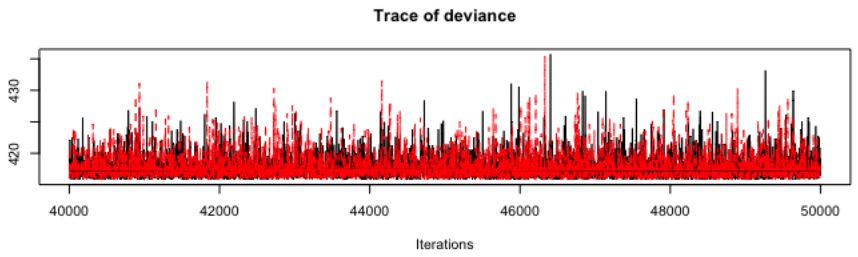
Changing Up the Prior

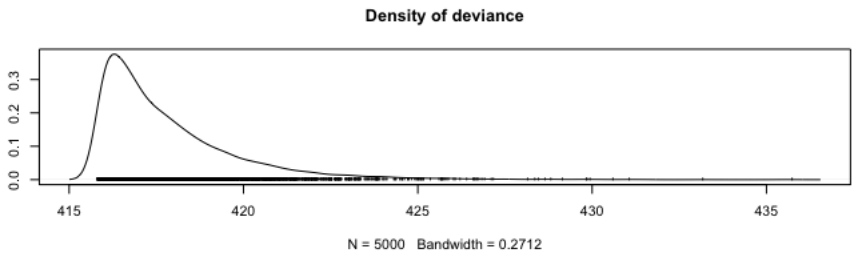
Prior: $\beta_0 \sim N(0,100{,}000)$;

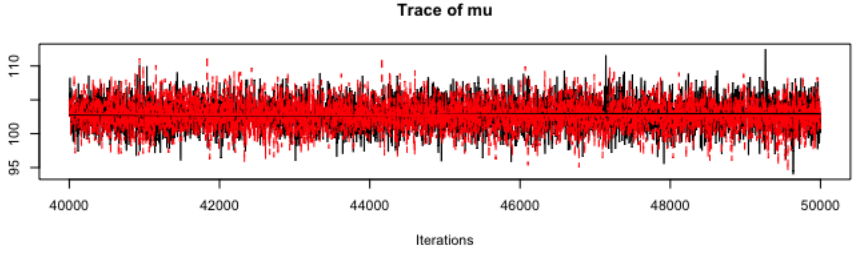
$$\sigma_e^{-2} \sim gamma(r = .01, \lambda = .01)$$

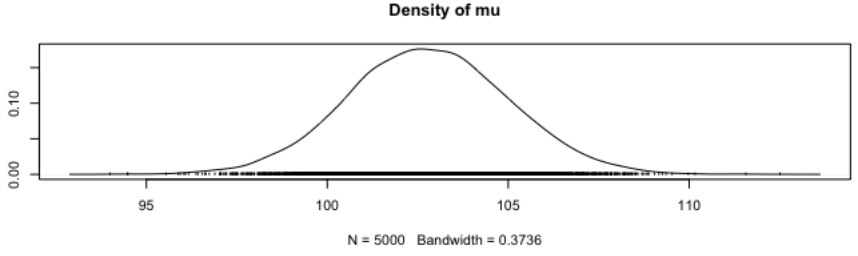
```
> fit3
Inference for Bugs model at "model03Bayes", fit using jags,
 2 chains, each with 50000 iterations (first 40000 discarded), n.thin = 2
 n.sims = 10000 iterations saved
         mu.vect sd.vect    2.5%     25%     50%     75%   97.5%  Rhat n.eff
mu       102.784   2.224  98.426 101.274 102.758 104.254 107.175 1.001  7300
sigma2   253.996  52.970 171.522 216.053 247.279 284.606 375.192 1.001  9500
tau        0.004   0.001   0.003   0.004   0.004   0.005   0.006 1.001  9500
deviance 417.812   1.994 415.872 416.409 417.203 418.572 423.105 1.003  1700

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2)
pD = 2.0 and DIC = 419.8
DIC is an estimate of expected predictive error (lower deviance is better).
```

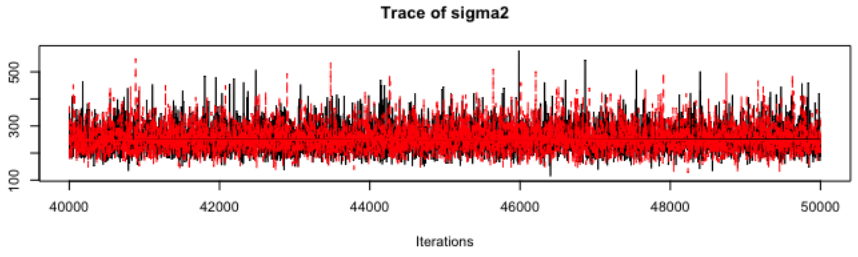
Chain Plots

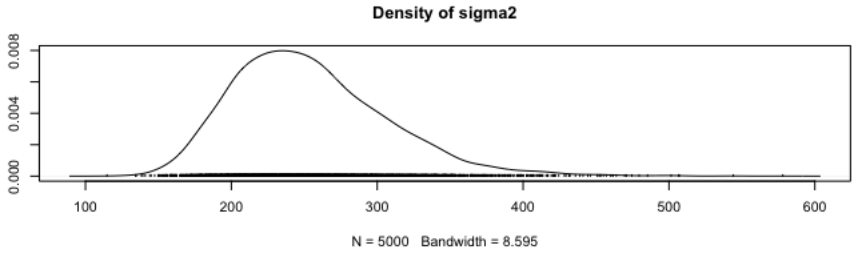

Trace of deviance
430
420
40000
42000
44000
46000
48000
50000
Iterations

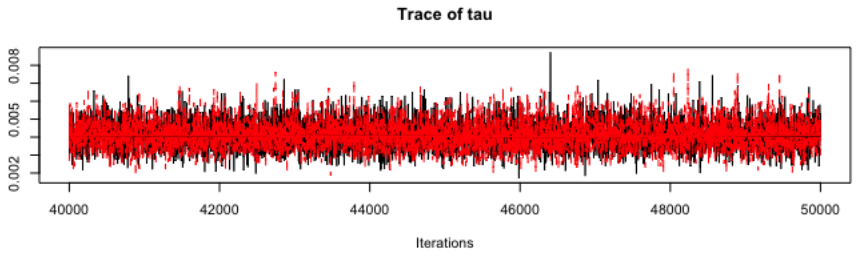

Density of deviance
0.0
0.1
0.2
0.3
415
420
425
430
435
N = 5000 Bandwidth = 0.2712

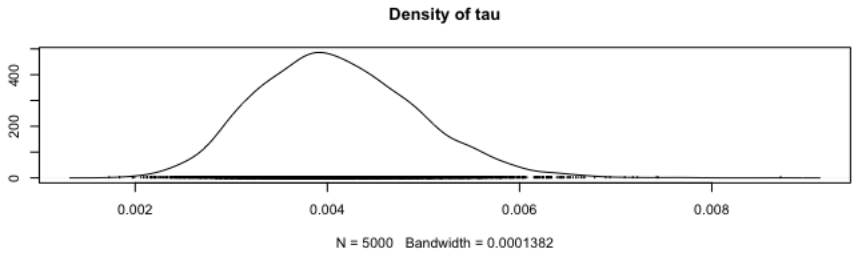

Trace of mu
110
100
95
40000
42000
44000
46000
48000
50000
Iterations


Density of mu
0.00
0.10
95
100
105
110
N = 5000 Bandwidth = 0.3736


Trace of sigma2
500
300
100
40000
42000
44000
46000
48000
50000
Iterations


Density of sigma2
0.000
0.004
0.008
100
200
300
400
500
600
N = 5000 Bandwidth = 8.595


Trace of tau
0.008
0.005
0.002
40000
42000
44000
46000
48000
50000
Iterations

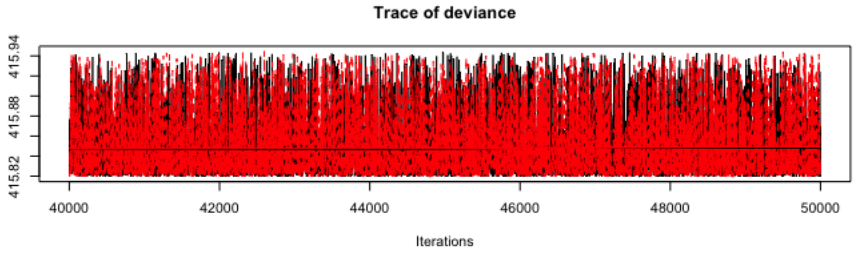

Density of tau
0
200
400
0.002
0.004
0.006
0.008
N = 5000 Bandwidth = 0.0001382

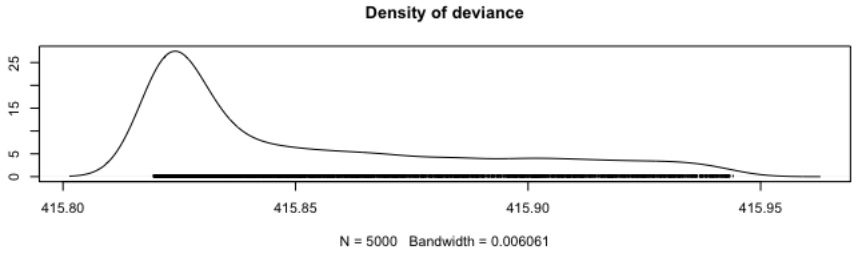
What About an Informative Prior?

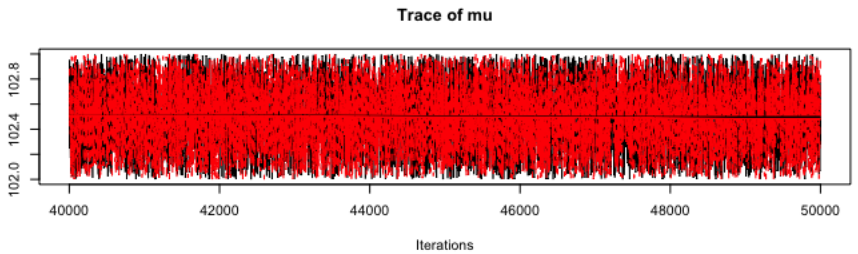
Prior: $\beta_0 \sim U(102,103); \sigma_e^2 \sim U(238,242)$

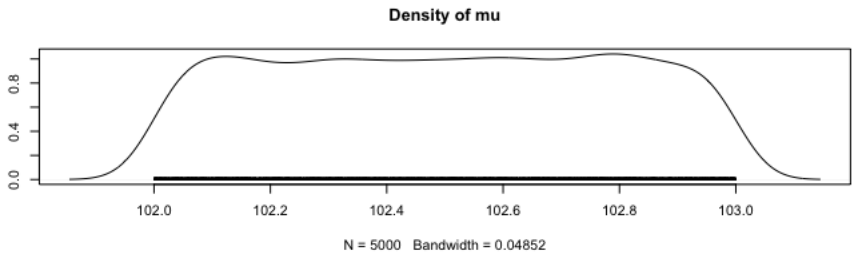
```
> fit4
Inference for Bugs model at "model04Bayes", fit using jags,
 2 chains, each with 50000 iterations (first 40000 discarded), n.thin = 2
 n.sims = 10000 iterations saved
         mu.vect sd.vect    2.5%     25%     50%     75%   97.5%  Rhat n.eff
mu       102.500   0.289 102.026 102.250 102.502 102.752 102.975 1.001  8200
sigma2   239.992   1.155 238.104 238.979 240.011 240.993 241.890 1.001 10000
tau        0.004   0.000   0.004   0.004   0.004   0.004   0.004 1.001 10000
deviance 415.853   0.036 415.820 415.823 415.835 415.876 415.935 1.000     1

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2)
pD = 0.0 and DIC = 415.9
DIC is an estimate of expected predictive error (lower deviance is better).
```

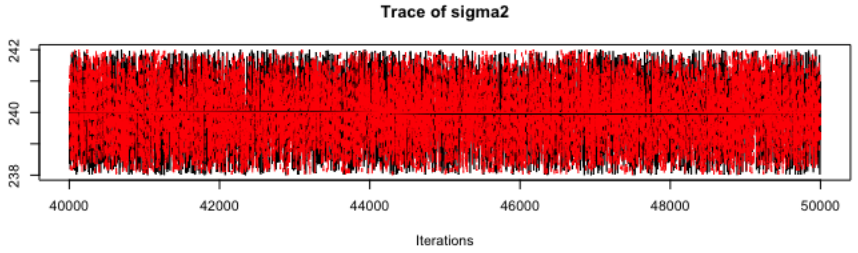
Chain Plots

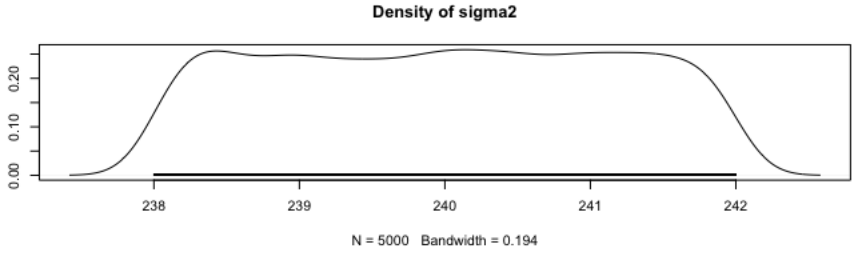

Trace of deviance
415.94
415.88
415.82
40000
42000
44000
46000
48000
50000
Iterations

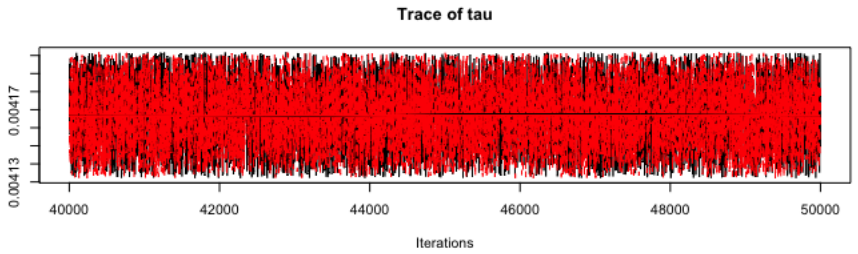

Density of deviance
25
15
5
0
415.80
415.85
415.90
415.95
N = 5000 Bandwidth = 0.006061

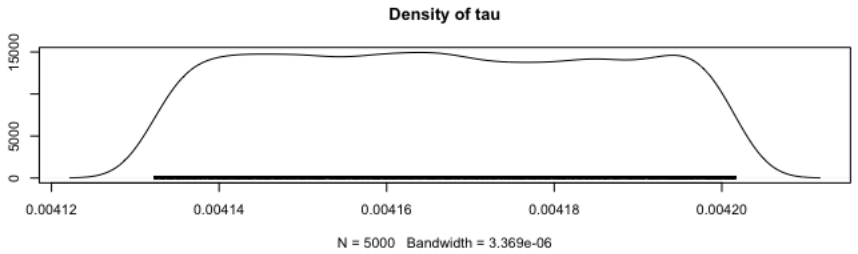

Trace of mu
102.8
102.4
102.0
40000
42000
44000
46000
48000
50000
Iterations


Density of mu
0.8
0.4
0.0
102.0
102.2
102.4
102.6
102.8
103.0
N = 5000 Bandwidth = 0.04852


Trace of sigma2
242
240
238
40000
42000
44000
46000
48000
50000
Iterations


Density of sigma2
0.20
0.10
0.00
238
239
240
241
242
N = 5000 Bandwidth = 0.194


Trace of tau
0.00417
0.00413
40000
42000
44000
46000
48000
50000
Iterations


Density of tau
15000
5000
0
0.00412
0.00414
0.00416
0.00418
0.00420
N = 5000 Bandwidth = 3.369e-06

MCMC in R

R itself does not have an MCMC engine native to the language – but there are many free versions available outside of R

For instance, if you wanted to estimate a path model with MCMC you can:

- Install the blavaan package (Bayesian lavaan)
- Run the path analysis with MCMC

I am not showing you these because I they all end up being really frustrating

- Very buggy
- Took me about an hour to just install all code

Bayesian Wrap Up

Today was an introduction to Bayesian statistics

- Bayes = use of prior distributions on parameters

We used two methods for estimation:

- MAP estimation – far less common
- MCMC estimation
  - Commonly, people will say Bayesian and mean MCMC – but Bayesian is just the addition of priors. MCMC is one way of estimating Bayesian models!

MCMC is effective for most Bayesian models:

- Model likelihood and prior likelihood are all that are needed

MCMC is estimation by brute force:

- Can be very slow, computationally intensive, and disk-space intensive