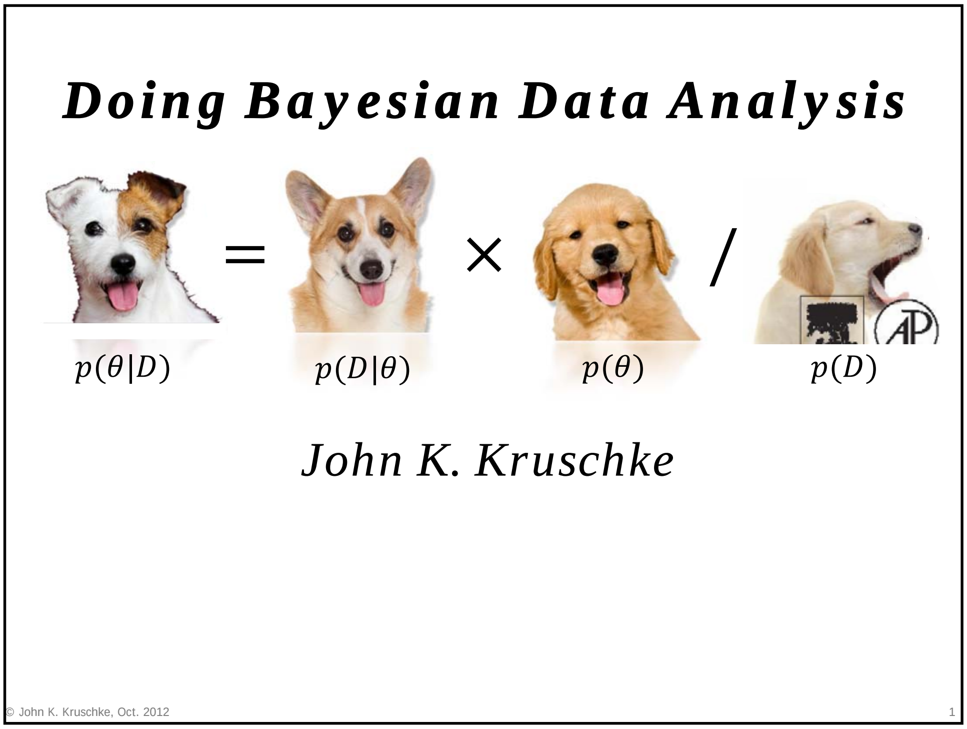
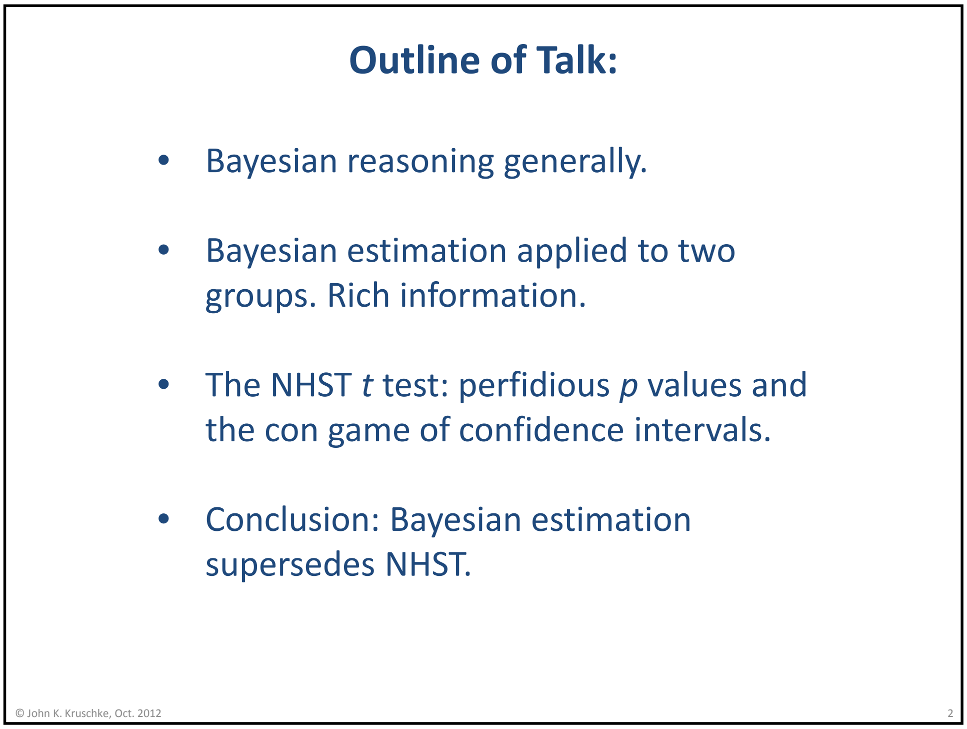


Doing Bayesian Data Analysis

$$p(\theta|D) = p(D|\theta) \times p(\theta) \,/\, p(D)$$

John K. Kruschke



Outline of Talk:

- Bayesian reasoning generally.
- Bayesian estimation applied to two groups. Rich information.
- The NHST *t* test: perfidious *p* values and the con game of confidence intervals.
- Conclusion: Bayesian estimation supersedes NHST.

Bayesian Reasoning

The role of data is to re-allocate credibility:

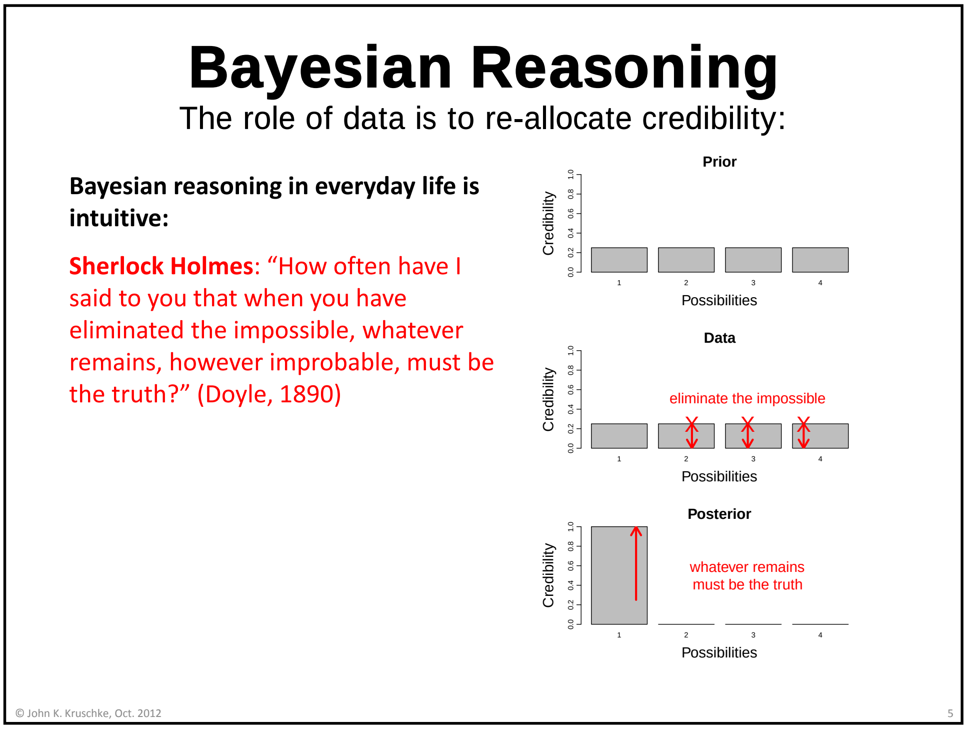
Prior Credibility with **New Data**
→ **Posterior Credibility**

via Bayes' rule

Bayesian Reasoning

The role of data is to re-allocate credibility:

Bayesian reasoning in everyday life is intuitive:

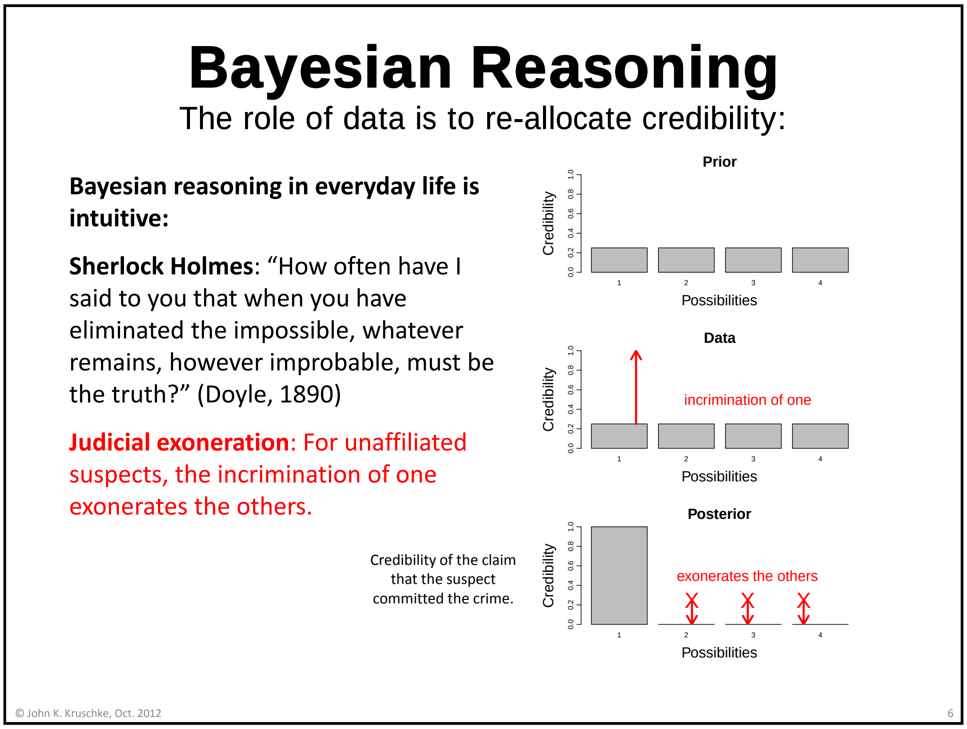


Bayesian Reasoning

The role of data is to re-allocate credibility:

Bayesian reasoning in everyday life is intuitive:

Sherlock Holmes: “How often have I said to you that when you have eliminated the impossible, whatever remains, however improbable, must be the truth?” (Doyle, 1890)



Bayesian Reasoning

The role of data is to re-allocate credibility:

Bayesian reasoning in everyday life is intuitive:

Sherlock Holmes: “How often have I said to you that when you have eliminated the impossible, whatever remains, however improbable, must be the truth?” (Doyle, 1890)

Judicial exoneration: For unaffiliated suspects, the incrimination of one exonerates the others.

Credibility of the claim that the suspect committed the crime.

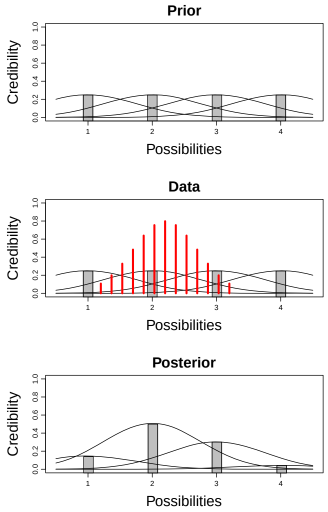
Bayesian Data Analysis

The role of data is to re-allocate credibility:

Bayesian reasoning in data analysis is intuitive:

Possibilities are *parameter values* in a model, such as the *mean* of a normal distribution.

We reallocate credibility to parameter values that are consistent with the data.



Bayesian Data Analysis

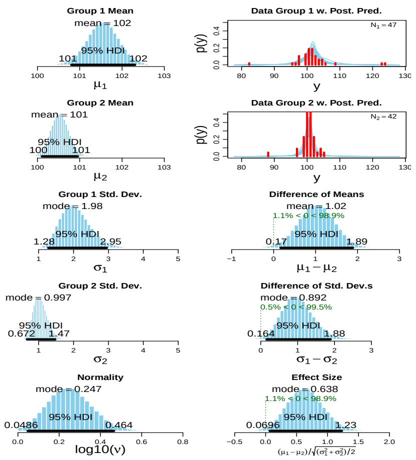
The role of data is to re-allocate credibility:

1. **Define a meaningful descriptive model.**
2. **Establish prior credibility regarding parameter values in the model. The prior credibility must be acceptable to a skeptical scientific audience.**
3. **Collect data.**
4. **Use Bayes' rule to re-allocate credibility to parameter values that are most consistent with the data.**

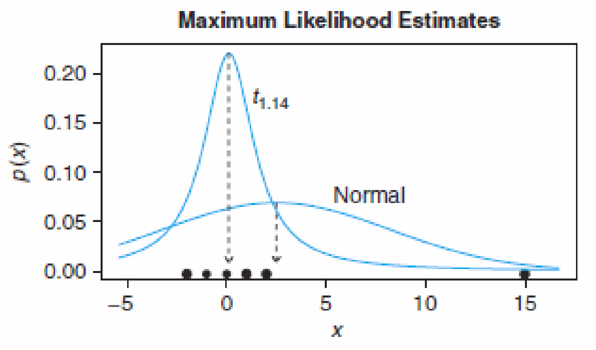
Robust Bayesian estimation for comparing two groups

Consider two groups; e.g.,
IQ of “smart drug” group and of control group.

Step 1: Define a model for describing the data.



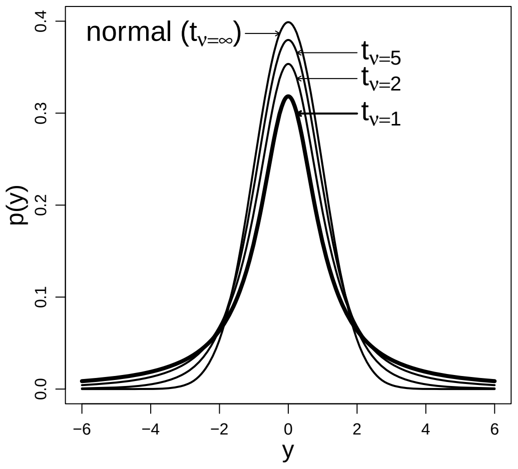
Descriptive distribution for data with outliers



Normal is pulled by outliers, but *t* distribution is not.

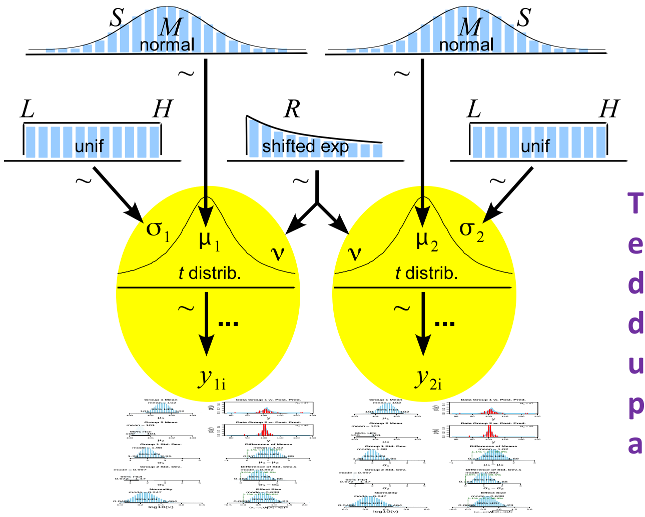
t distribution is used here as a description of data, NOT as a sampling distribution for *p* values!

Descriptive distribution for data with outliers



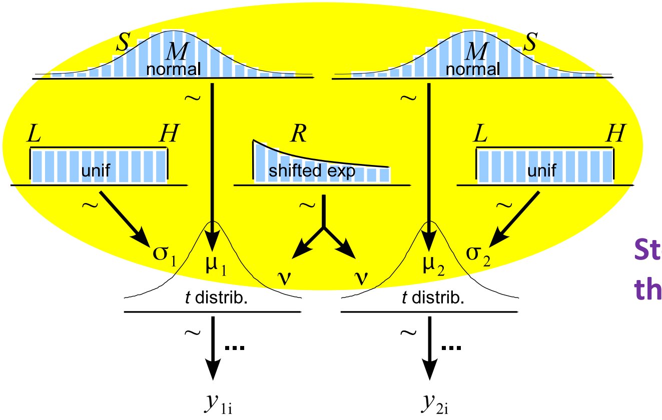
The *t* distribution has normality controlled by the parameter ν.

Robust Bayesian estimation for comparing two groups



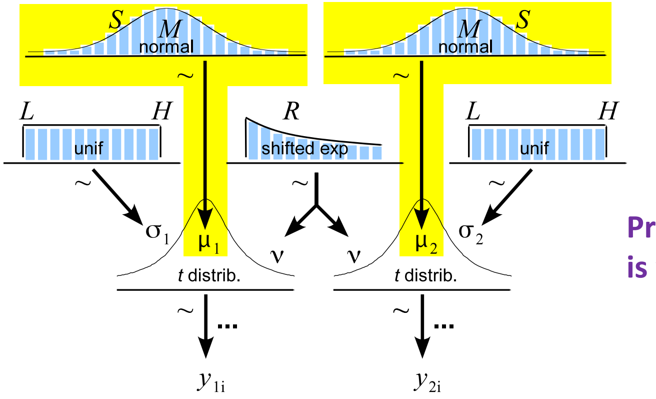
The data from each group are described by *t* distributions, using five parameters altogether.

Robust Bayesian estimation for comparing two groups



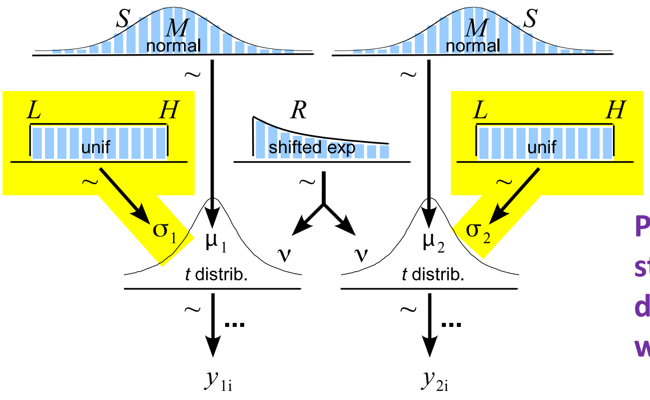
Step 2: Specify the prior.

Robust Bayesian estimation for comparing two groups



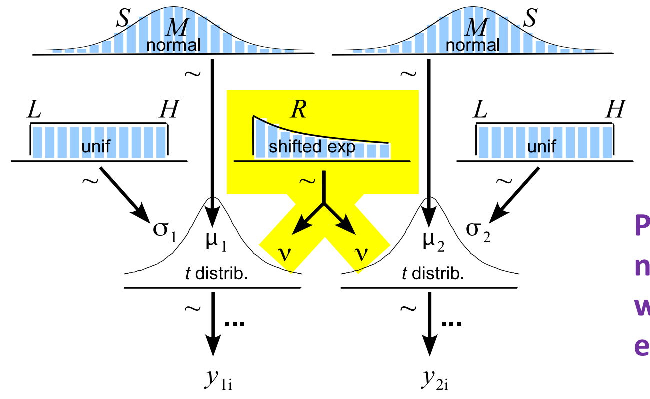
Prior on means is wide normal.

Robust Bayesian estimation for comparing two groups



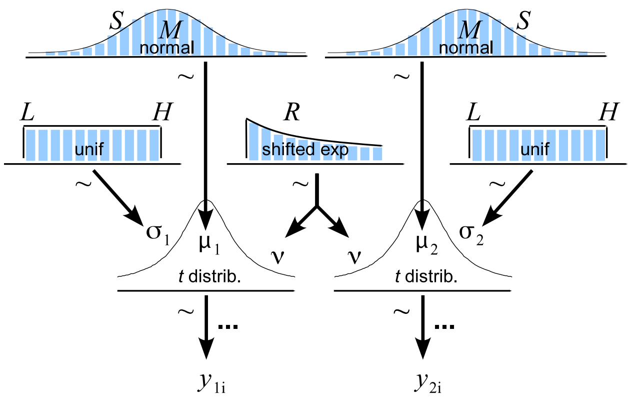
Prior on standard deviations is wide uniform.

Robust Bayesian estimation for comparing two groups



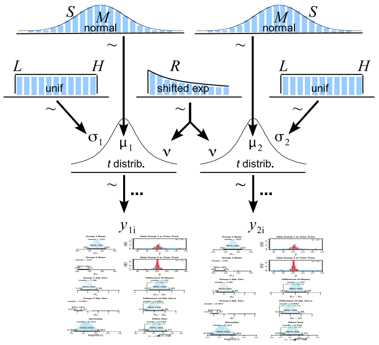
Prior on normality is wide exponential.

Robust Bayesian estimation for comparing two groups

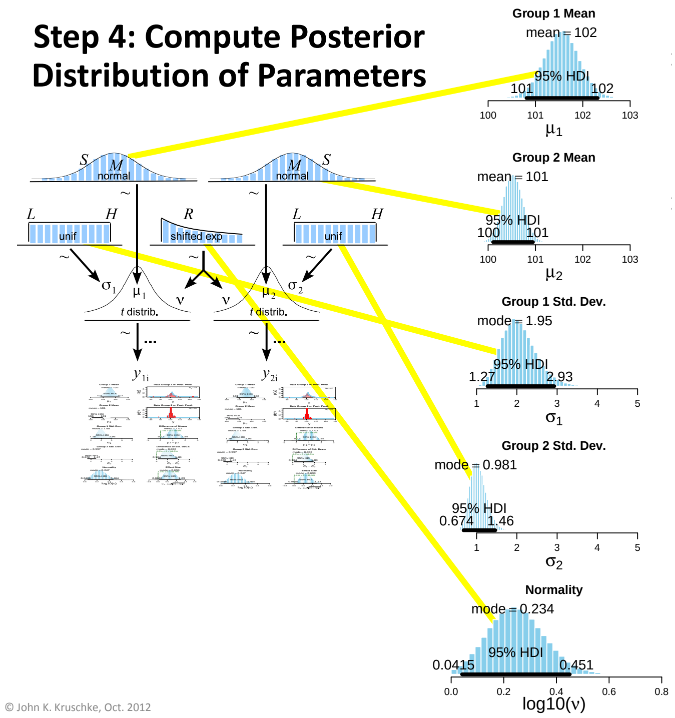


Parameter distributions will be represented by histograms: A huge number of representative parameter values.

Step 3: Collect Data.

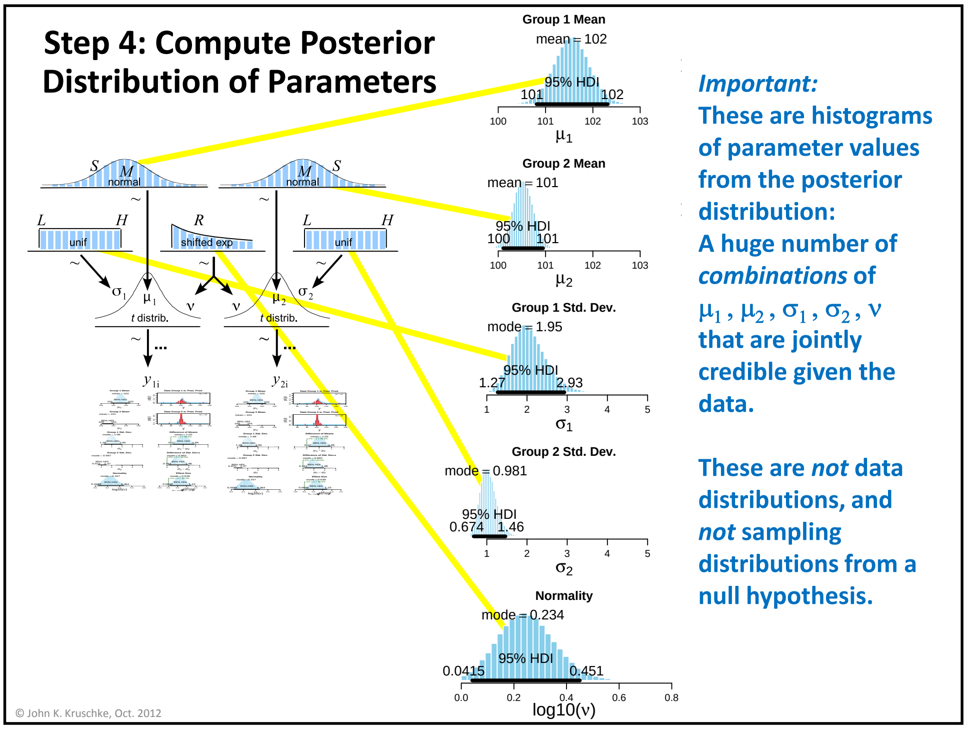


One fixed data set, shown as red histograms.



Step 4: Compute Posterior Distribution of Parameters

© John K. Kruschke, Oct. 2012



Step 4: Compute Posterior Distribution of Parameters

Important:
These are histograms of parameter values from the posterior distribution:
A huge number of *combinations* of
μ_1 , μ_2 , σ_1 , σ_2 , ν
that are jointly credible given the data.

These are *not* data distributions, and *not* sampling distributions from a null hypothesis.

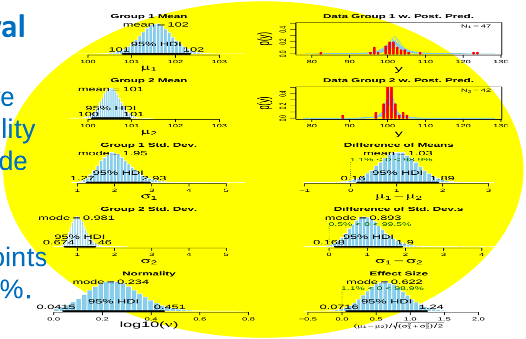
© John K. Kruschke, Oct. 2012

95% HDI:
Highest density interval

Points within the HDI have higher credibility (probability density) than points outside the HDI.

The total probability of points within the 95% HDI is 95%.

Points outside the HDI may be deemed not credible.

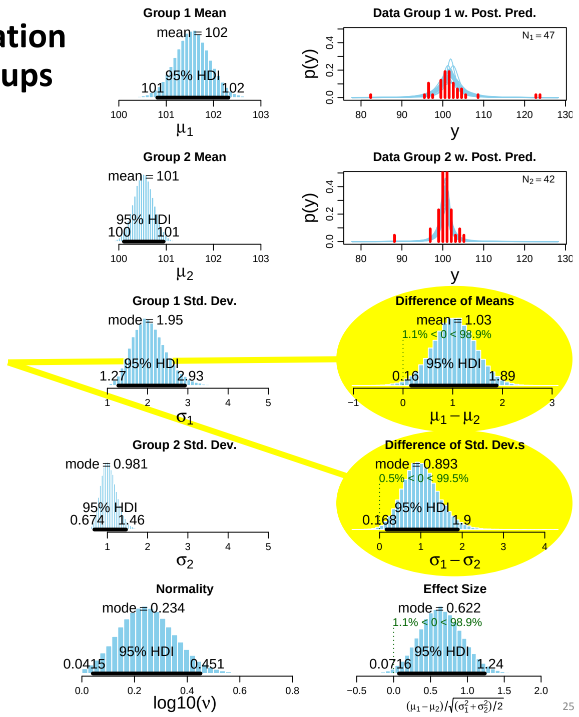


Robust Bayesian estimation for comparing two groups

Differences between groups?
Compute $\mu_1 - \mu_2$
and $\sigma_1 - \sigma_2$
at each of the many credible combinations.

Here, both differences are credibly non-zero.

(NHST would require *two* tests…)

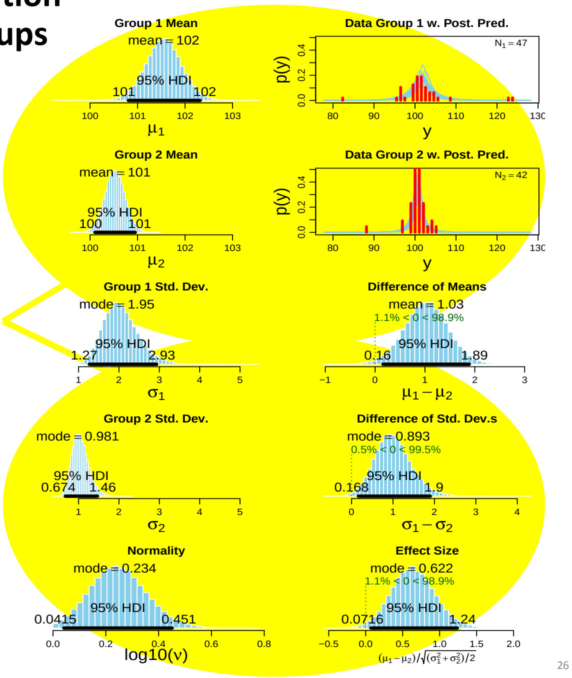


Robust Bayesian estimation for comparing two groups

Differences between groups? Compute $\mu_1 - \mu_2$ and $\sigma_1 - \sigma_2$ at each of the many credible combinations.

Here, both differences are credibly non-zero.

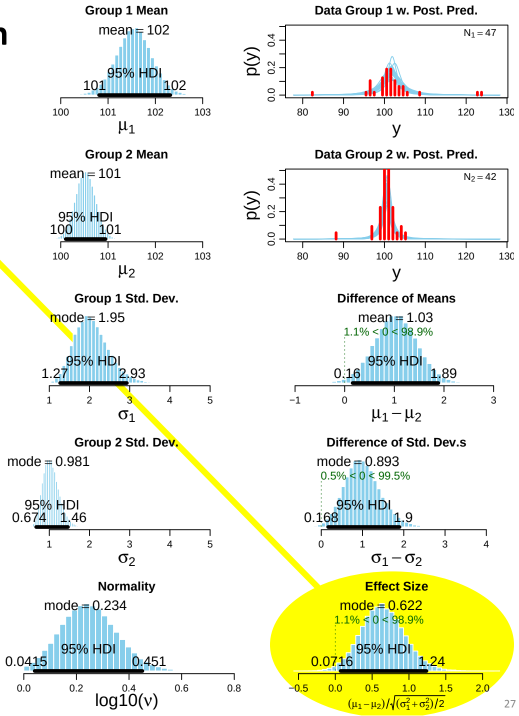
(NHST would require *two* tests…)



26

Robust Bayesian estimation for comparing two groups

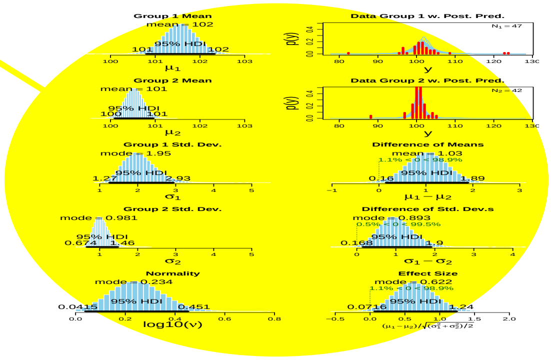
Complete distribution on effect size!



27

Robust Bayesian estimation for comparing two groups

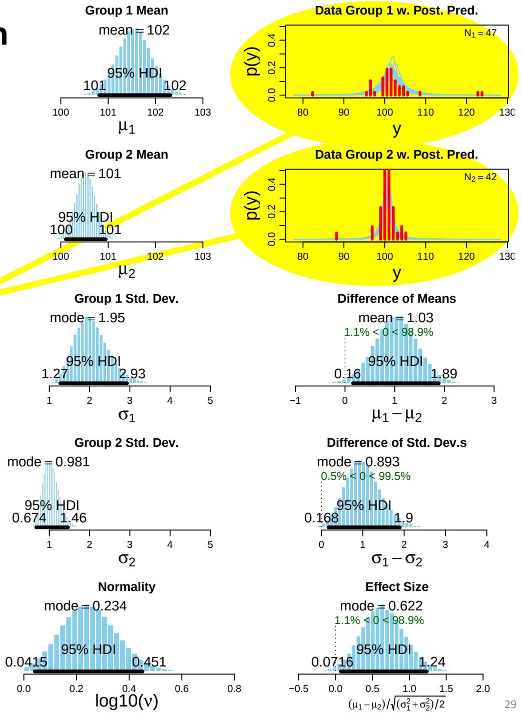
Complete distribution on effect size!



Robust Bayesian estimation for comparing two groups

Are the data described well by the model?

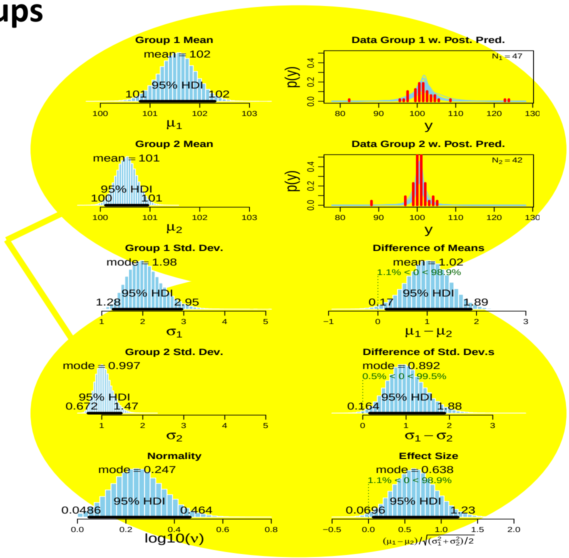
Superimpose a smattering of credible descriptive distributions on data.
= "posterior predictive check"



Robust Bayesian estimation for comparing two groups

Are the data described well by the model?

Superimpose a smattering of credible descriptive distributions on data.
= "posterior predictive check"



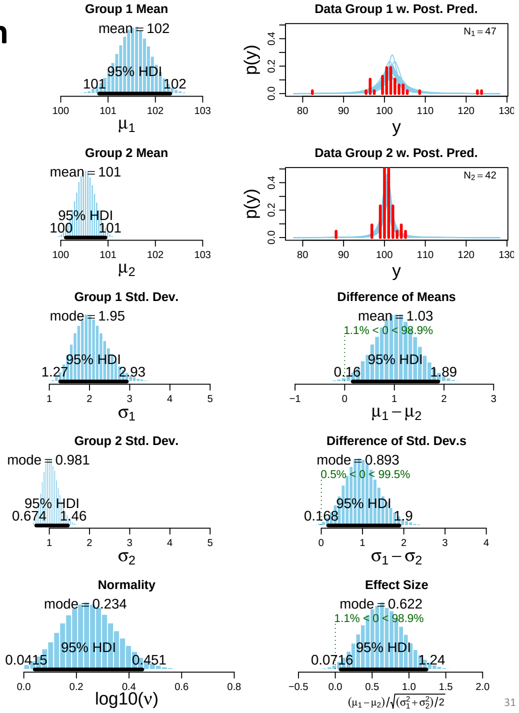
Robust Bayesian estimation for comparing two groups

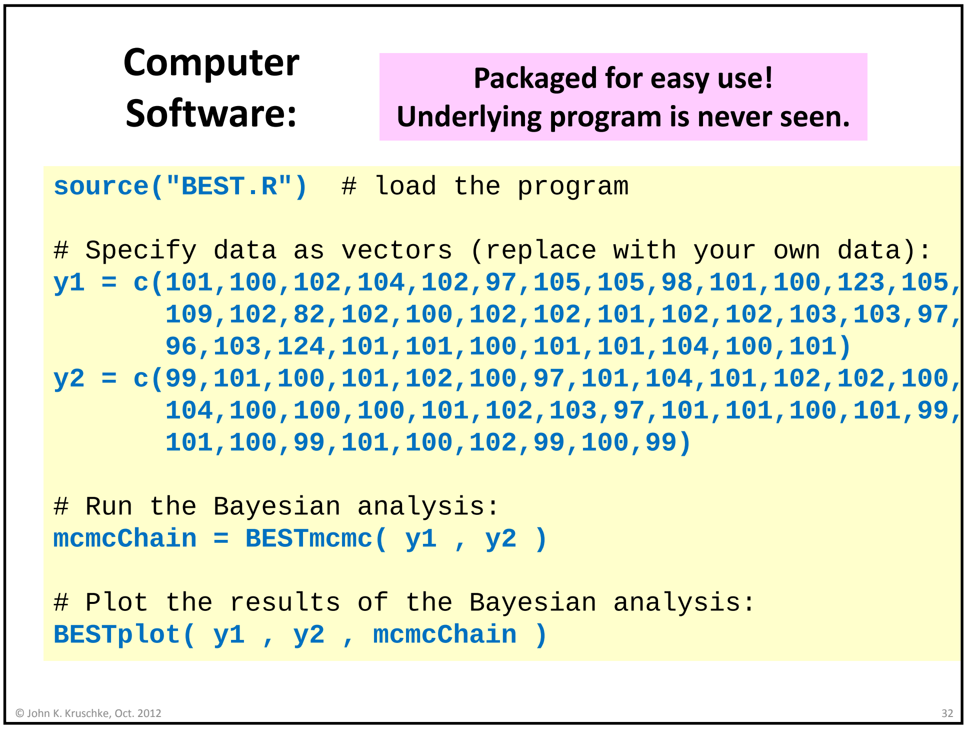
Summary:

→ Complete distribution of credible parameter values (not merely point estimate with ends of confidence interval)**.**

→ Decisions about multiple aspects of parameters (without reference to *p* values)**.**

→ Flexible descriptive model, robust to outliers (unlike NHST *t* test)**.**





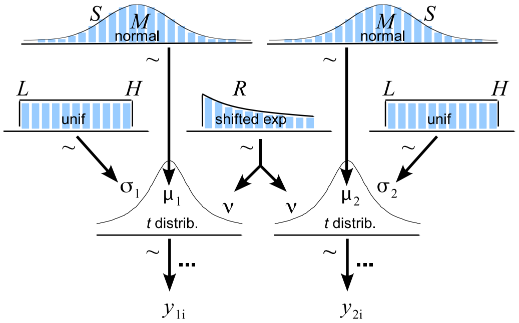
Computer Software:

Packaged for easy use!
Underlying program is never seen.

```
source("BEST.R")  # load the program

# Specify data as vectors (replace with your own data):
y1 = c(101,100,102,104,102,97,105,105,98,101,100,123,105,
       109,102,82,102,100,102,102,101,102,102,103,103,97,
       96,103,124,101,101,100,101,101,104,100,101)
y2 = c(99,101,100,101,102,100,97,101,104,101,102,102,100,
       104,100,100,100,101,102,103,97,101,101,100,101,99,
       101,100,99,101,100,102,99,100,99)

# Run the Bayesian analysis:
mcmcChain = BESTmcmc( y1 , y2 )

# Plot the results of the Bayesian analysis:
BESTplot( y1 , y2 , mcmcChain )
```

Robust Bayesian estimation for comparing two groups

Download the programs from http://www.indiana.edu/~kruschke/BEST/BEST.zip

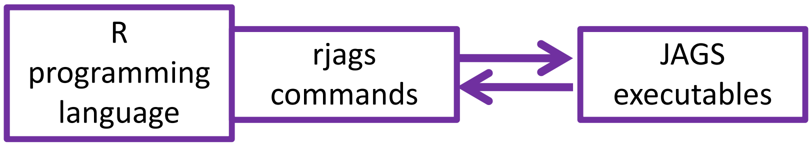


Now for a look under the hood

http://www.autonationconnect.com/2010/07/backseat-mechanic-under-the-hoo/

Doing it with JAGS

"JAGS" = Just Another Gibbs Sampler
but other sampling methods are incorporated.

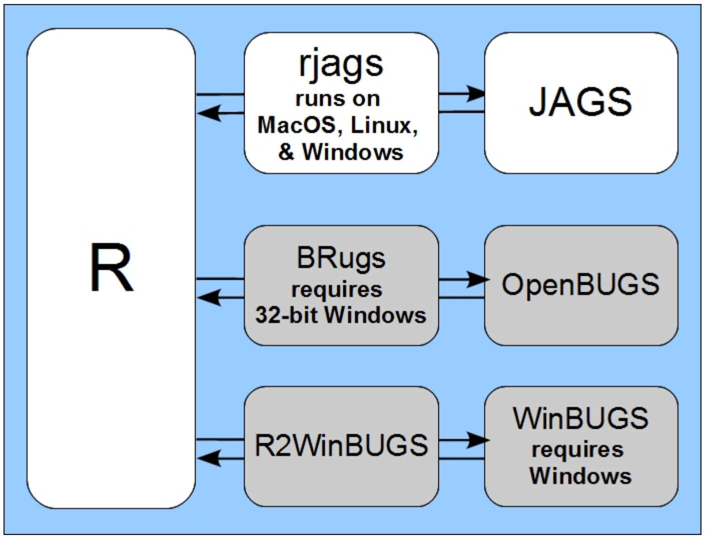


JAGS makes it easy. You specify only the

- prior function
- likelihood function

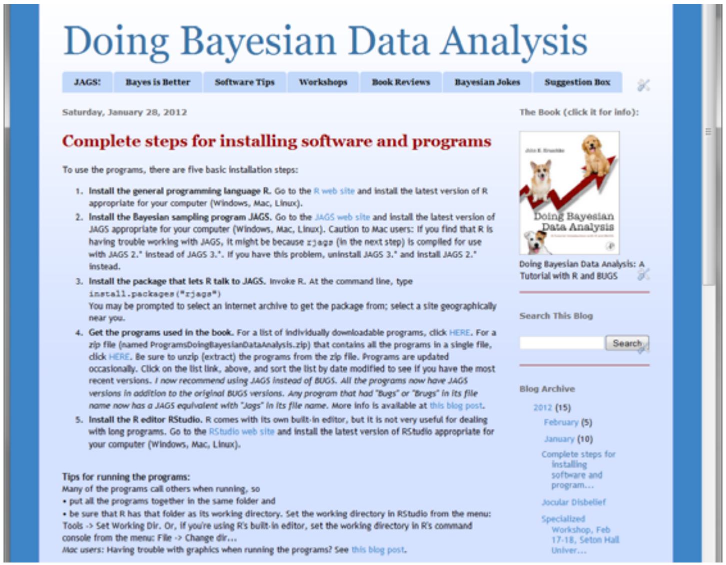
and JAGS does the rest! You do no math, no selection of sampling methods.

JAGS and BUGS



Installation: See Blog Entry

http://doingbayesiandataanalysis.blogspot.com/2012/01/complete-steps-for-installing-software.html



Doing Bayesian Data Analysis

Saturday, January 28, 2012

Complete steps for installing software and programs

To use the programs, there are five basic installation steps:

1. **Install the general programming language R.** Go to the R web site and install the latest version of R appropriate for your computer (Windows, Mac, Linux).
2. **Install the Bayesian sampling program JAGS.** Go to the JAGS web site and install the latest version of JAGS appropriate for your computer (Windows, Mac, Linux). Caution to Mac users: If you find that R is having trouble working with JAGS, it might be because `rjags` (in the next step) is compiled for use with JAGS 2.* instead of JAGS 3.*. If you have this problem, uninstall JAGS 3.* and install JAGS 2.* instead.
3. **Install the package that lets R talk to JAGS.** Invoke R. At the command line, type
   ```
   install.packages("rjags")
   ```
   You may be prompted to select an internet archive to get the package from; select a site geographically near you.
4. **Get the programs used in the book.** For a list of individually downloadable programs, click HERE. For a zip file (named ProgramsDoingBayesianDataAnalysis.zip) that contains all the programs in a single file, click HERE. Be sure to unzip (extract) the programs from the zip file. Programs are updated occasionally. Click on the list link, above, and sort the list by date modified to see if you have the most recent versions. *I now recommend using JAGS instead of BUGS. All the programs now have JAGS versions in addition to the original BUGS versions. Any program that had "Bugs" or "Brugs" in its file name now has a JAGS equivalent with "Jags" in its file name.* More info is available at this blog post.
5. **Install the R editor RStudio.** R comes with its own built-in editor, but it is not very useful for dealing with long programs. Go to the RStudio web site and install the latest version of RStudio appropriate for your computer (Windows, Mac, Linux).

Tips for running the programs:
Many of the programs call others when running, so
• put all the programs together in the same folder and
• be sure that R has that folder as its working directory. Set the working directory in RStudio from the menu: Tools -> Set Working Dir. Or, if you're using R's built-in editor, set the working directory in R's command console from the menu: File -> Change dir...
Mac users: Having trouble with graphics when running the programs? See this blog post.

The Book (click it for info):

Doing Bayesian Data Analysis: A Tutorial with R and BUGS

Search This Blog

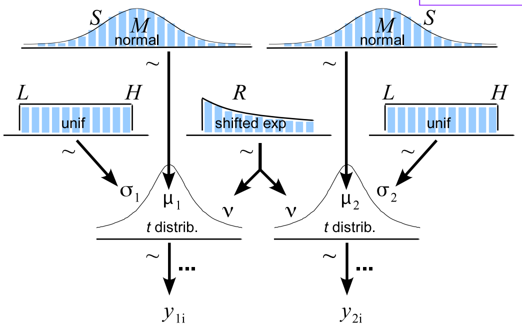
Blog Archive

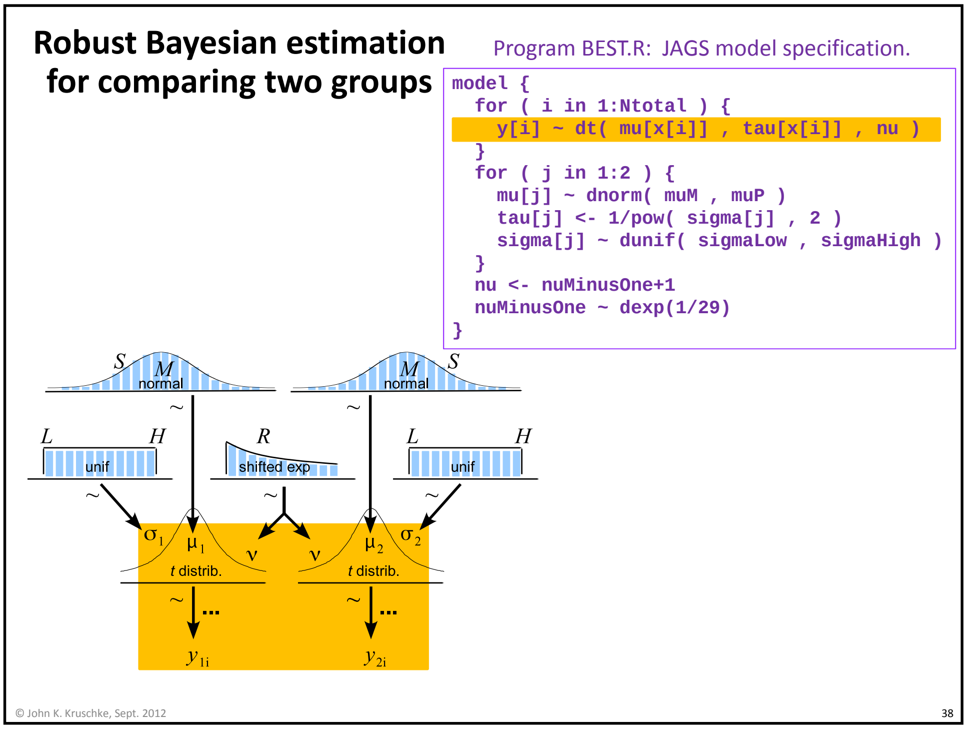
2012 (15)
February (5)
January (10)
Complete steps for installing software and program...
Jocular Disbelief
Specialized Workshop, Feb 17-18, Seton Hall Univer...

Robust Bayesian estimation for comparing two groups

Program BEST.R: JAGS model specification.

```
model {
  for ( i in 1:Ntotal ) {
    y[i] ~ dt( mu[x[i]] , tau[x[i]] , nu )
  }
  for ( j in 1:2 ) {
    mu[j] ~ dnorm( muM , muP )
    tau[j] <- 1/pow( sigma[j] , 2 )
    sigma[j] ~ dunif( sigmaLow , sigmaHigh )
  }
  nu <- numMinusOne+1
  nuMinusOne ~ dexp(1/29)
}
```



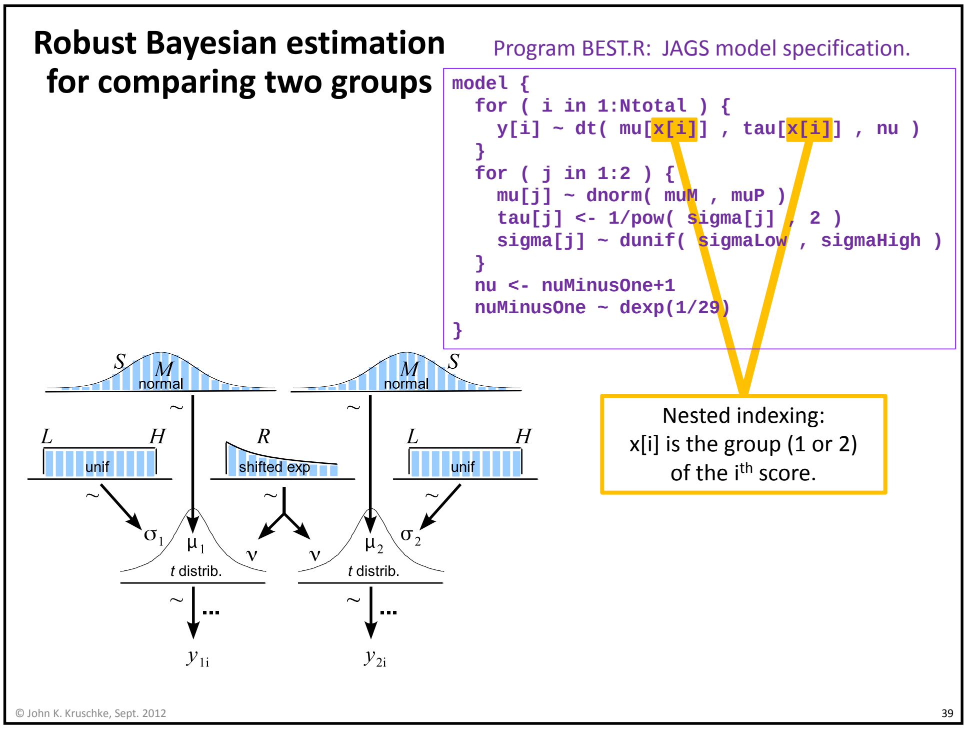


Robust Bayesian estimation for comparing two groups

Program BEST.R: JAGS model specification.

```
model {
  for ( i in 1:Ntotal ) {
    y[i] ~ dt( mu[x[i]] , tau[x[i]] , nu )
  }
  for ( j in 1:2 ) {
    mu[j] ~ dnorm( muM , muP )
    tau[j] <- 1/pow( sigma[j] , 2 )
    sigma[j] ~ dunif( sigmaLow , sigmaHigh )
  }
  nu <- nuMinusOne+1
  nuMinusOne ~ dexp(1/29)
}
```

38



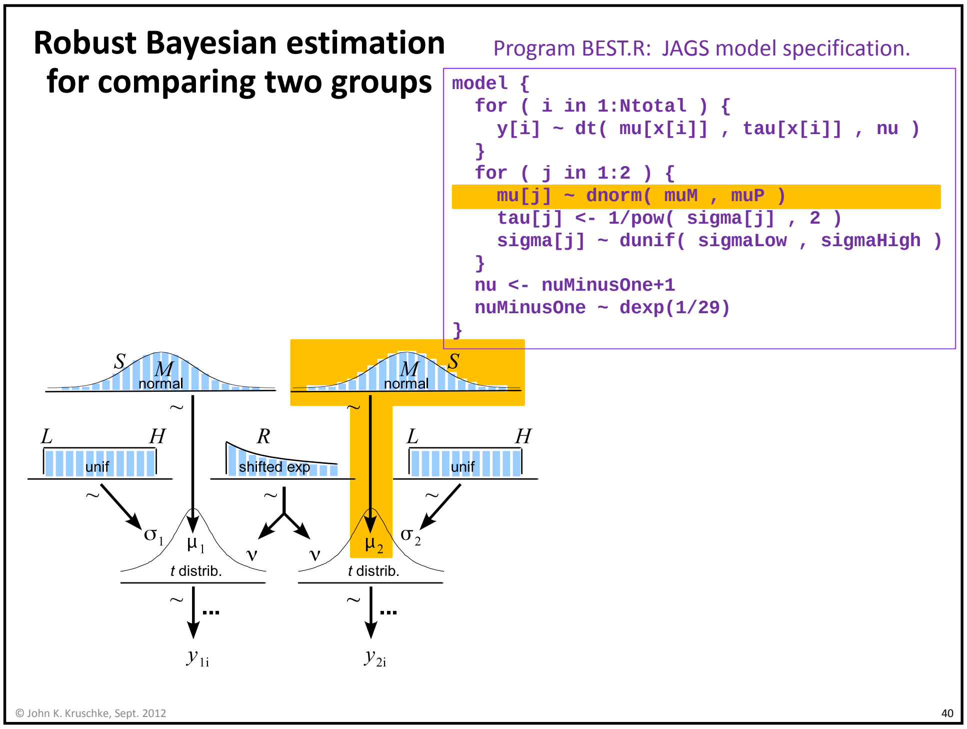
Robust Bayesian estimation for comparing two groups

Program BEST.R: JAGS model specification.

```
model {
  for ( i in 1:Ntotal ) {
    y[i] ~ dt( mu[x[i]] , tau[x[i]] , nu )
  }
  for ( j in 1:2 ) {
    mu[j] ~ dnorm( muM , muP )
    tau[j] <- 1/pow( sigma[j] , 2 )
    sigma[j] ~ dunif( sigmaLow , sigmaHigh )
  }
  nu <- nuMinusOne+1
  nuMinusOne ~ dexp(1/29)
}
```

Nested indexing:
x[i] is the group (1 or 2)
of the ith score.

39



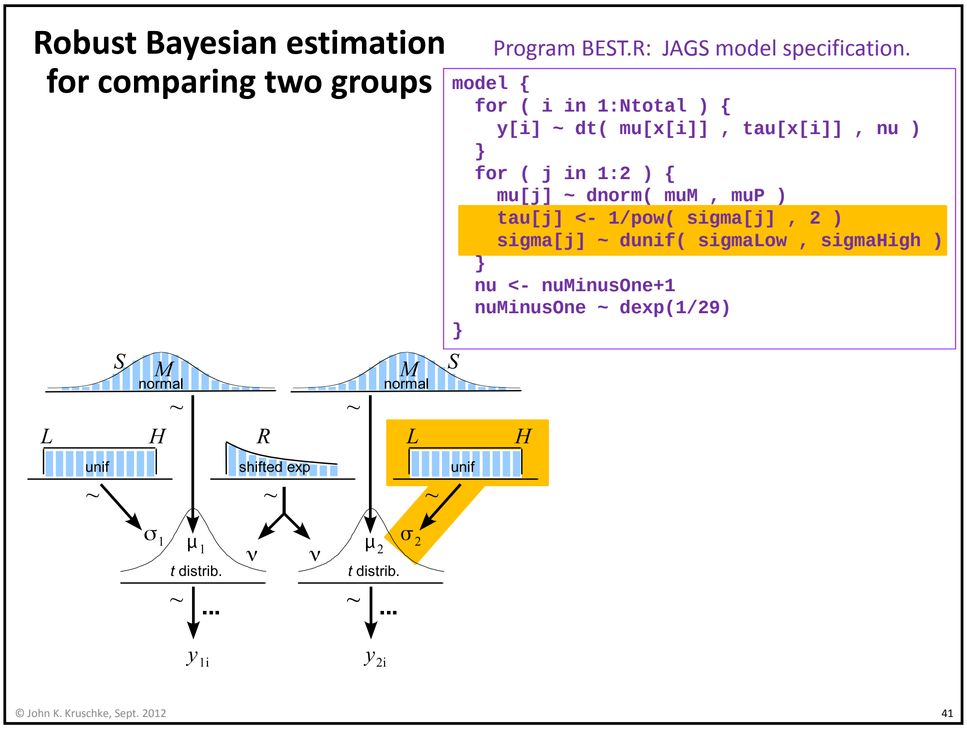
Robust Bayesian estimation for comparing two groups

Program BEST.R: JAGS model specification.

```
model {
  for ( i in 1:Ntotal ) {
    y[i] ~ dt( mu[x[i]] , tau[x[i]] , nu )
  }
  for ( j in 1:2 ) {
    mu[j] ~ dnorm( muM , muP )
    tau[j] <- 1/pow( sigma[j] , 2 )
    sigma[j] ~ dunif( sigmaLow , sigmaHigh )
  }
  nu <- nuMinusOne+1
  nuMinusOne ~ dexp(1/29)
}
```

© John K. Kruschke, Sept. 2012

40



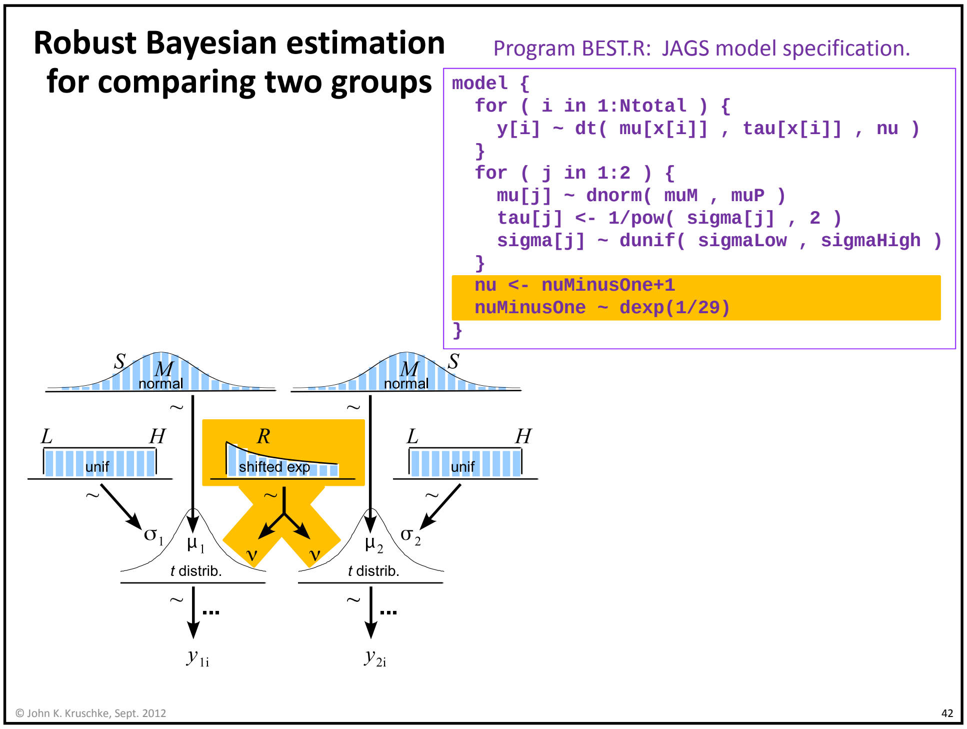
Robust Bayesian estimation for comparing two groups

Program BEST.R: JAGS model specification.

```
model {
  for ( i in 1:Ntotal ) {
    y[i] ~ dt( mu[x[i]] , tau[x[i]] , nu )
  }
  for ( j in 1:2 ) {
    mu[j] ~ dnorm( muM , muP )
    tau[j] <- 1/pow( sigma[j] , 2 )
    sigma[j] ~ dunif( sigmaLow , sigmaHigh )
  }
  nu <- nuMinusOne+1
  nuMinusOne ~ dexp(1/29)
}
```

© John K. Kruschke, Sept. 2012

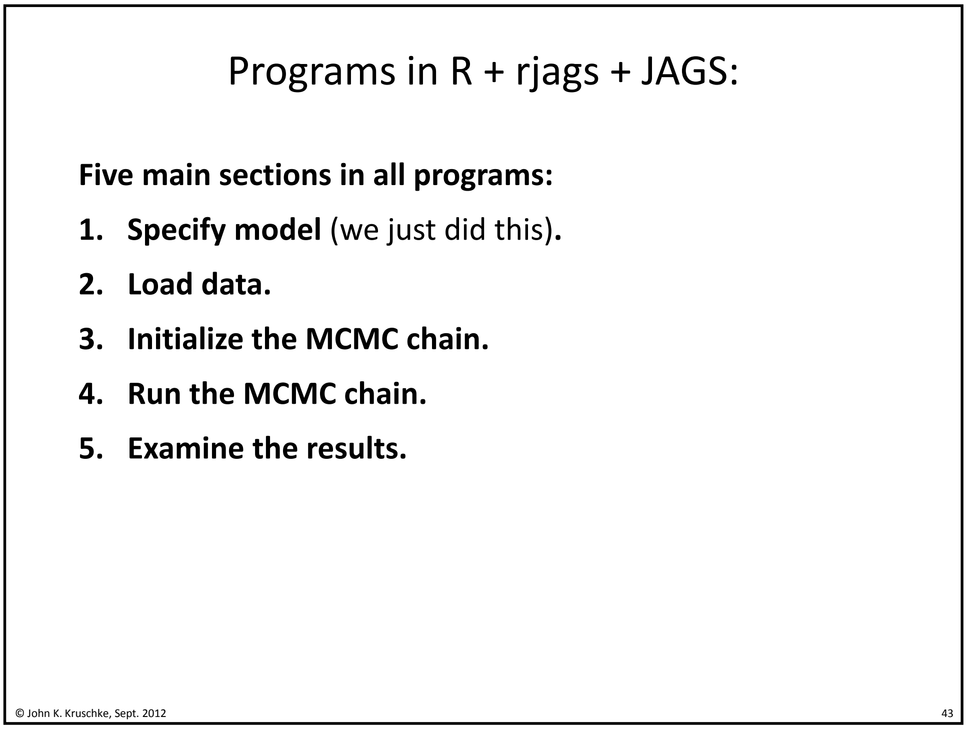
41



Robust Bayesian estimation for comparing two groups

Program BEST.R: JAGS model specification.

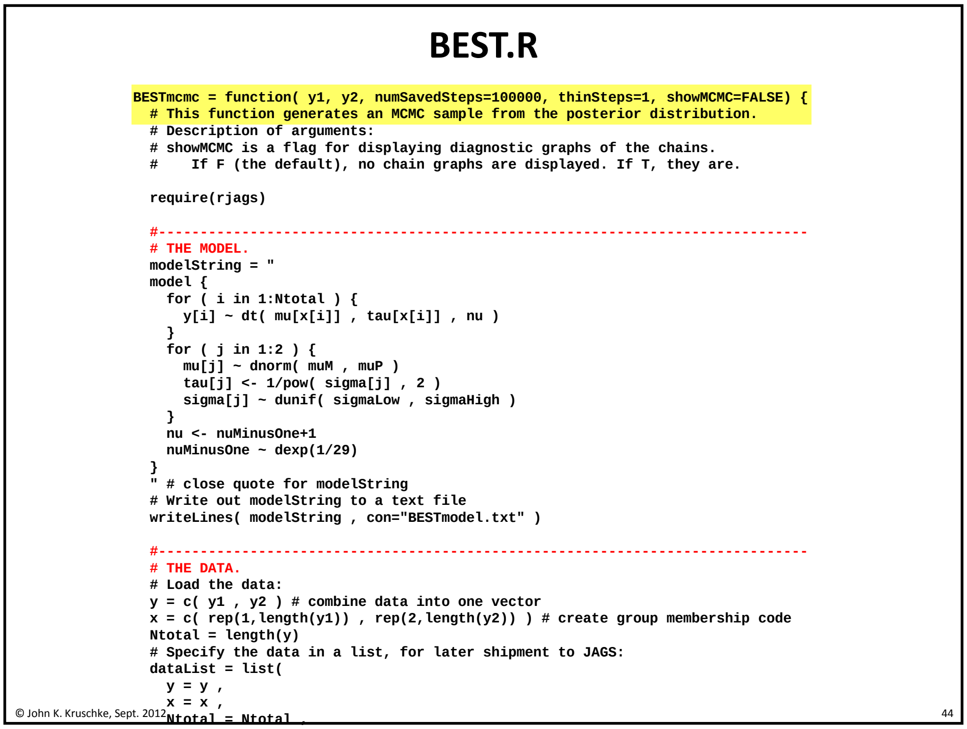
```
model {
  for ( i in 1:Ntotal ) {
    y[i] ~ dt( mu[x[i]] , tau[x[i]] , nu )
  }
  for ( j in 1:2 ) {
    mu[j] ~ dnorm( muM , muP )
    tau[j] <- 1/pow( sigma[j] , 2 )
    sigma[j] ~ dunif( sigmaLow , sigmaHigh )
  }
  nu <- nuMinusOne+1
  nuMinusOne ~ dexp(1/29)
}
```



Programs in R + rjags + JAGS:

Five main sections in all programs:

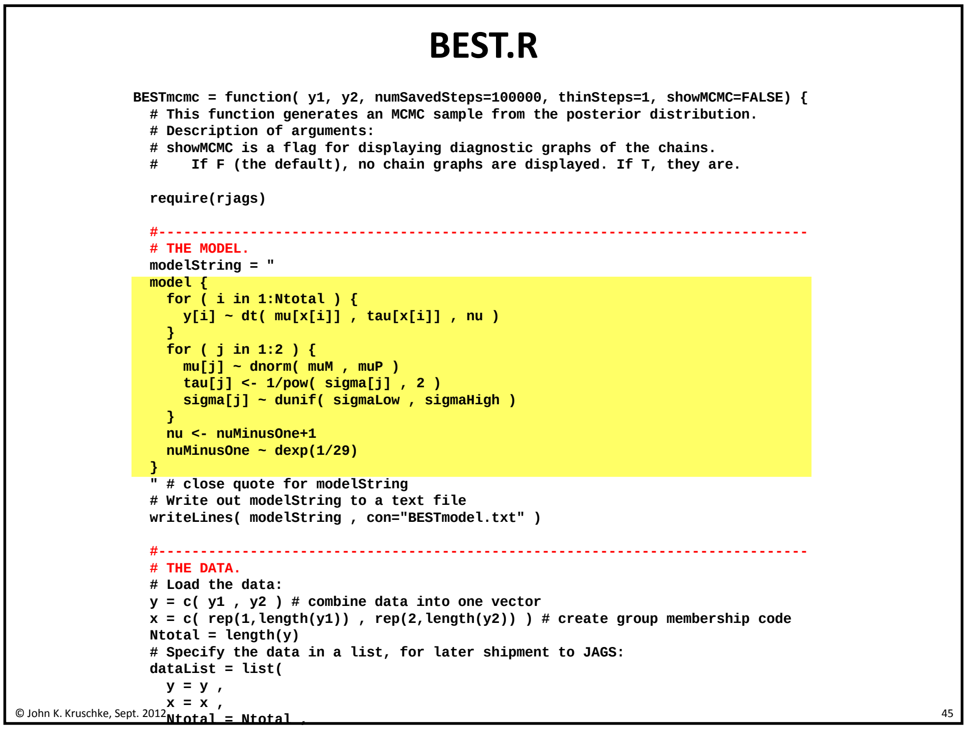
1. **Specify model** (we just did this).
2. **Load data.**
3. **Initialize the MCMC chain.**
4. **Run the MCMC chain.**
5. **Examine the results.**



BEST.R

```
BESTmcmc = function( y1, y2, numSavedSteps=100000, thinSteps=1, showMCMC=FALSE) {
  # This function generates an MCMC sample from the posterior distribution.
  # Description of arguments:
  # showMCMC is a flag for displaying diagnostic graphs of the chains.
  #    If F (the default), no chain graphs are displayed. If T, they are.

  require(rjags)

  #------------------------------------------------------------------------------
  # THE MODEL.
  modelString = "
  model {
    for ( i in 1:Ntotal ) {
      y[i] ~ dt( mu[x[i]] , tau[x[i]] , nu )
    }
    for ( j in 1:2 ) {
      mu[j] ~ dnorm( muM , muP )
      tau[j] <- 1/pow( sigma[j] , 2 )
      sigma[j] ~ dunif( sigmaLow , sigmaHigh )
    }
    nu <- nuMinusOne+1
    nuMinusOne ~ dexp(1/29)
  }
  " # close quote for modelString
  # Write out modelString to a text file
  writeLines( modelString , con="BESTmodel.txt" )

  #------------------------------------------------------------------------------
  # THE DATA.
  # Load the data:
  y = c( y1 , y2 ) # combine data into one vector
  x = c( rep(1,length(y1)) , rep(2,length(y2)) ) # create group membership code
  Ntotal = length(y)
  # Specify the data in a list, for later shipment to JAGS:
  dataList = list(
    y = y ,
    x = x ,
    Ntotal = Ntotal ,
```

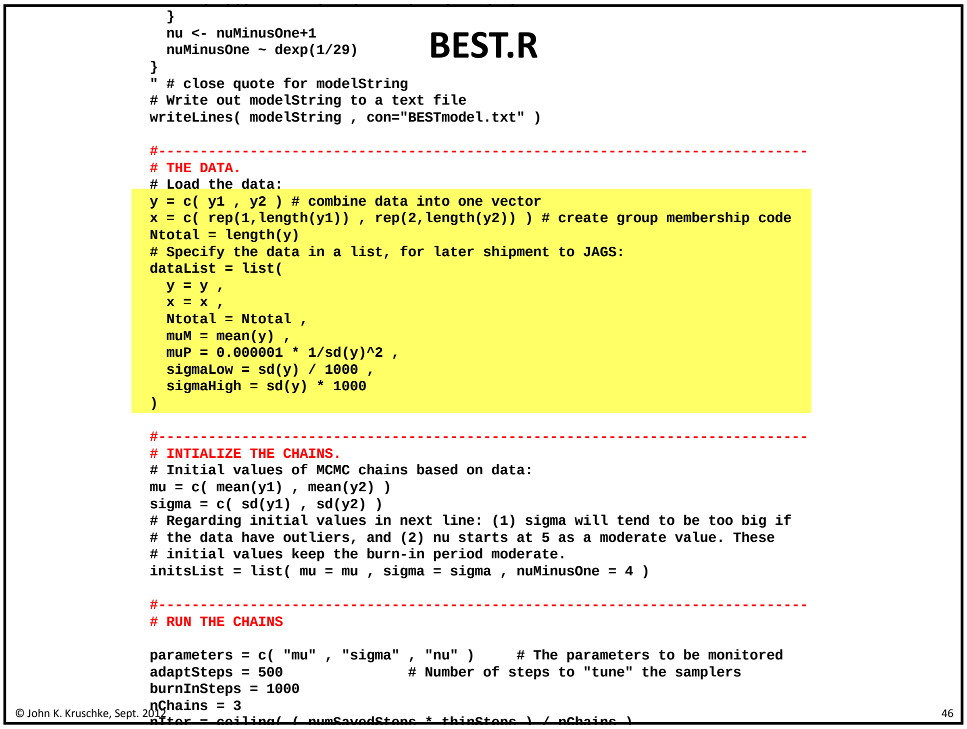
© John K. Kruschke, Sept. 2012



BEST.R

```
BESTmcmc = function( y1, y2, numSavedSteps=100000, thinSteps=1, showMCMC=FALSE) {
  # This function generates an MCMC sample from the posterior distribution.
  # Description of arguments:
  # showMCMC is a flag for displaying diagnostic graphs of the chains.
  #    If F (the default), no chain graphs are displayed. If T, they are.

  require(rjags)

  #------------------------------------------------------------------------------
  # THE MODEL.
  modelString = "
  model {
    for ( i in 1:Ntotal ) {
      y[i] ~ dt( mu[x[i]] , tau[x[i]] , nu )
    }
    for ( j in 1:2 ) {
      mu[j] ~ dnorm( muM , muP )
      tau[j] <- 1/pow( sigma[j] , 2 )
      sigma[j] ~ dunif( sigmaLow , sigmaHigh )
    }
    nu <- nuMinusOne+1
    nuMinusOne ~ dexp(1/29)
  }
  " # close quote for modelString
  # Write out modelString to a text file
  writeLines( modelString , con="BESTmodel.txt" )

  #------------------------------------------------------------------------------
  # THE DATA.
  # Load the data:
  y = c( y1 , y2 ) # combine data into one vector
  x = c( rep(1,length(y1)) , rep(2,length(y2)) ) # create group membership code
  Ntotal = length(y)
  # Specify the data in a list, for later shipment to JAGS:
  dataList = list(
    y = y ,
    x = x ,
    Ntotal = Ntotal ,
```

© John K. Kruschke, Sept. 2012

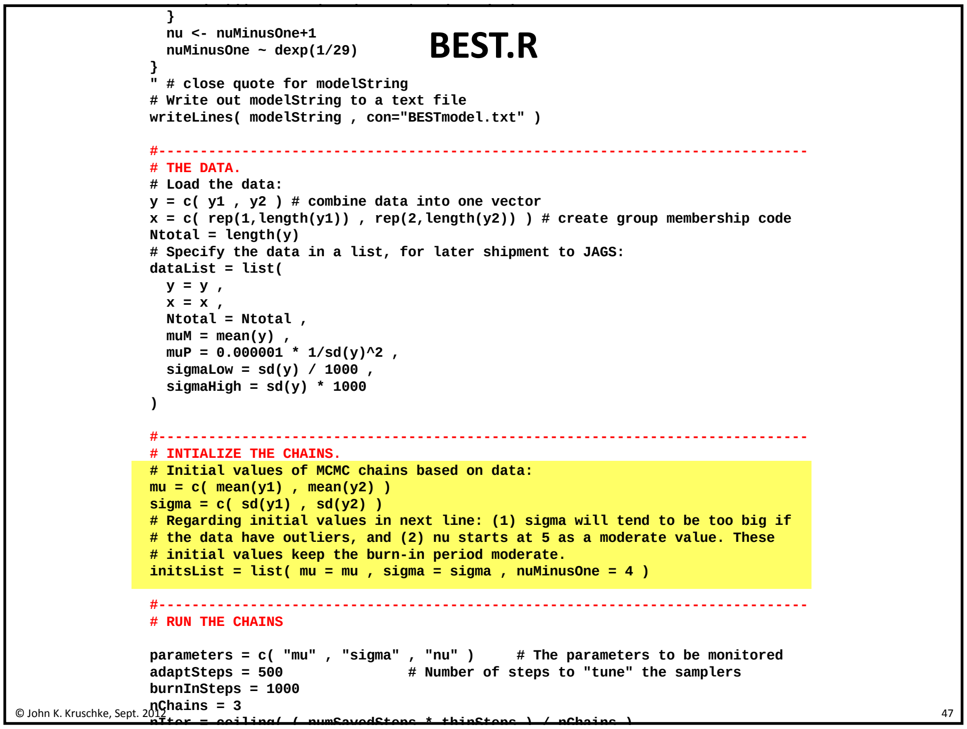


BEST.R

```
  }
  nu <- numMinusOne+1
  nuMinusOne ~ dexp(1/29)
}
" # close quote for modelString
# Write out modelString to a text file
writeLines( modelString , con="BESTmodel.txt" )

#------------------------------------------------------------------------------
# THE DATA.
# Load the data:
y = c( y1 , y2 ) # combine data into one vector
x = c( rep(1,length(y1)) , rep(2,length(y2)) ) # create group membership code
Ntotal = length(y)
# Specify the data in a list, for later shipment to JAGS:
dataList = list(
  y = y ,
  x = x ,
  Ntotal = Ntotal ,
  muM = mean(y) ,
  muP = 0.000001 * 1/sd(y)^2 ,
  sigmaLow = sd(y) / 1000 ,
  sigmaHigh = sd(y) * 1000
)

#------------------------------------------------------------------------------
# INTIALIZE THE CHAINS.
# Initial values of MCMC chains based on data:
mu = c( mean(y1) , mean(y2) )
sigma = c( sd(y1) , sd(y2) )
# Regarding initial values in next line: (1) sigma will tend to be too big if
# the data have outliers, and (2) nu starts at 5 as a moderate value. These
# initial values keep the burn-in period moderate.
initsList = list( mu = mu , sigma = sigma , nuMinusOne = 4 )

#------------------------------------------------------------------------------
# RUN THE CHAINS

parameters = c( "mu" , "sigma" , "nu" )     # The parameters to be monitored
adaptSteps = 500               # Number of steps to "tune" the samplers
burnInSteps = 1000
nChains = 3
nIter = ceiling( ( numSavedSteps * thinSteps ) / nChains )
```

© John K. Kruschke, Sept. 2012

46

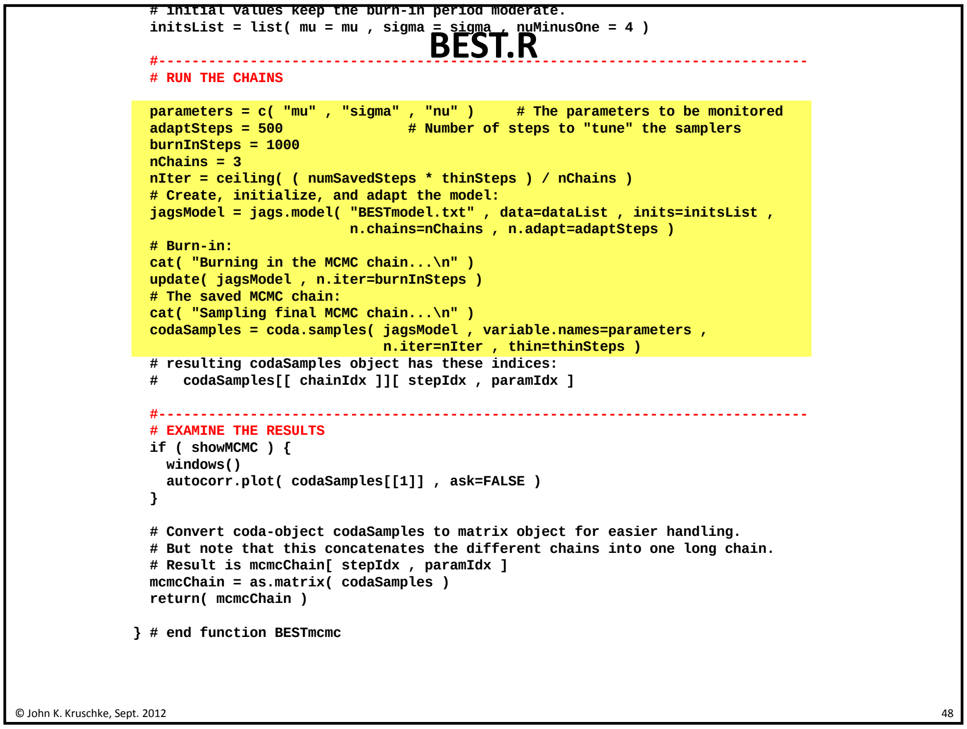


BEST.R

```
  }
  nu <- numMinusOne+1
  nuMinusOne ~ dexp(1/29)
}
" # close quote for modelString
# Write out modelString to a text file
writeLines( modelString , con="BESTmodel.txt" )

#------------------------------------------------------------------------------
# THE DATA.
# Load the data:
y = c( y1 , y2 ) # combine data into one vector
x = c( rep(1,length(y1)) , rep(2,length(y2)) ) # create group membership code
Ntotal = length(y)
# Specify the data in a list, for later shipment to JAGS:
dataList = list(
  y = y ,
  x = x ,
  Ntotal = Ntotal ,
  muM = mean(y) ,
  muP = 0.000001 * 1/sd(y)^2 ,
  sigmaLow = sd(y) / 1000 ,
  sigmaHigh = sd(y) * 1000
)

#------------------------------------------------------------------------------
# INTIALIZE THE CHAINS.
# Initial values of MCMC chains based on data:
mu = c( mean(y1) , mean(y2) )
sigma = c( sd(y1) , sd(y2) )
# Regarding initial values in next line: (1) sigma will tend to be too big if
# the data have outliers, and (2) nu starts at 5 as a moderate value. These
# initial values keep the burn-in period moderate.
initsList = list( mu = mu , sigma = sigma , nuMinusOne = 4 )

#------------------------------------------------------------------------------
# RUN THE CHAINS

parameters = c( "mu" , "sigma" , "nu" )     # The parameters to be monitored
adaptSteps = 500               # Number of steps to "tune" the samplers
burnInSteps = 1000
nChains = 3
nIter = ceiling( ( numSavedSteps * thinSteps ) / nChains )
```

© John K. Kruschke, Sept. 2012

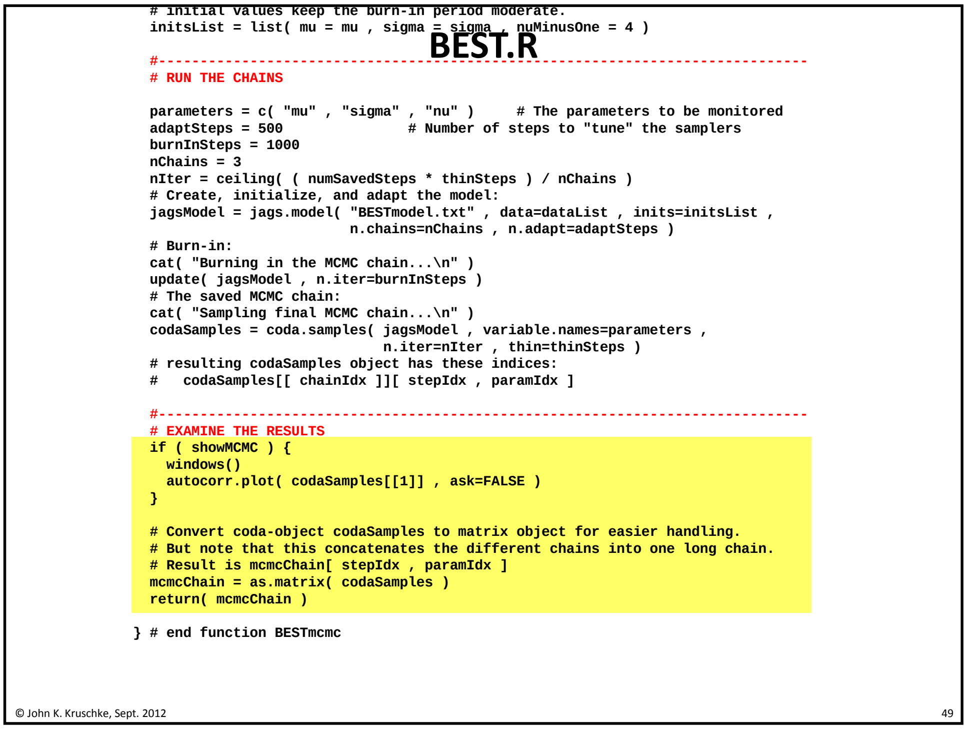
47



BEST.R

```
  # initial values keep the burn-in period moderate.
  initsList = list( mu = mu , sigma = sigma , nuMinusOne = 4 )

  #------------------------------------------------------------------------------
  # RUN THE CHAINS

  parameters = c( "mu" , "sigma" , "nu" )     # The parameters to be monitored
  adaptSteps = 500               # Number of steps to "tune" the samplers
  burnInSteps = 1000
  nChains = 3
  nIter = ceiling( ( numSavedSteps * thinSteps ) / nChains )
  # Create, initialize, and adapt the model:
  jagsModel = jags.model( "BESTmodel.txt" , data=dataList , inits=initsList ,
                          n.chains=nChains , n.adapt=adaptSteps )
  # Burn-in:
  cat( "Burning in the MCMC chain...\n" )
  update( jagsModel , n.iter=burnInSteps )
  # The saved MCMC chain:
  cat( "Sampling final MCMC chain...\n" )
  codaSamples = coda.samples( jagsModel , variable.names=parameters ,
                              n.iter=nIter , thin=thinSteps )
  # resulting codaSamples object has these indices:
  #   codaSamples[[ chainIdx ]][ stepIdx , paramIdx ]

  #------------------------------------------------------------------------------
  # EXAMINE THE RESULTS
  if ( showMCMC ) {
    windows()
    autocorr.plot( codaSamples[[1]] , ask=FALSE )
  }

  # Convert coda-object codaSamples to matrix object for easier handling.
  # But note that this concatenates the different chains into one long chain.
  # Result is mcmcChain[ stepIdx , paramIdx ]
  mcmcChain = as.matrix( codaSamples )
  return( mcmcChain )

} # end function BESTmcmc
```

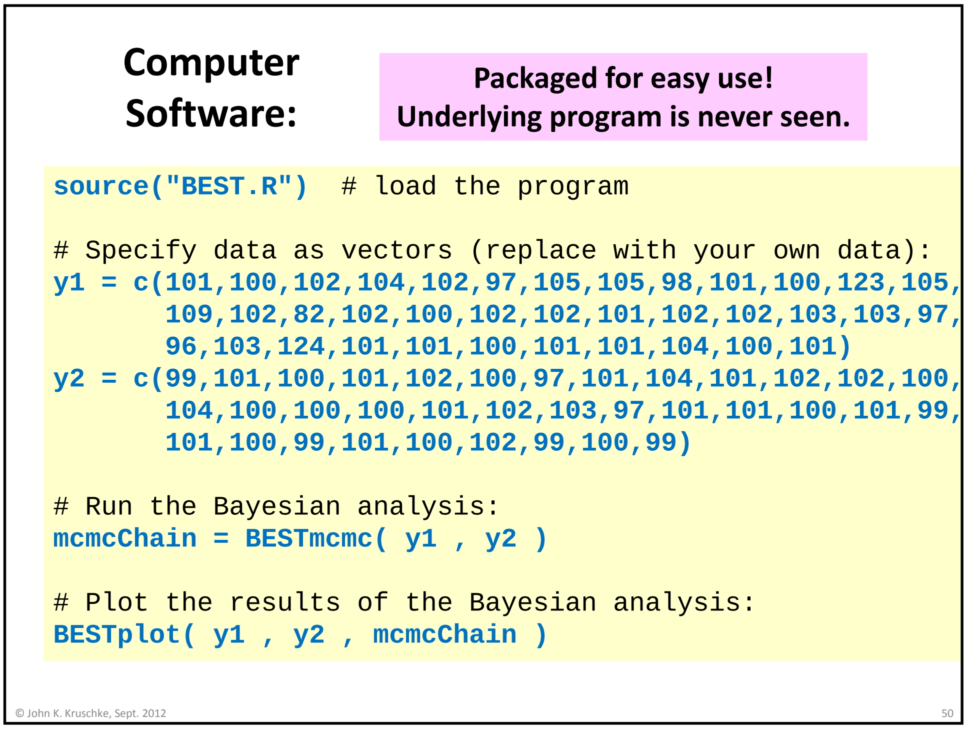
© John K. Kruschke, Sept. 2012 48



BEST.R

```
  # initial values keep the burn-in period moderate.
  initsList = list( mu = mu , sigma = sigma , nuMinusOne = 4 )

  #------------------------------------------------------------------------------
  # RUN THE CHAINS

  parameters = c( "mu" , "sigma" , "nu" )     # The parameters to be monitored
  adaptSteps = 500               # Number of steps to "tune" the samplers
  burnInSteps = 1000
  nChains = 3
  nIter = ceiling( ( numSavedSteps * thinSteps ) / nChains )
  # Create, initialize, and adapt the model:
  jagsModel = jags.model( "BESTmodel.txt" , data=dataList , inits=initsList ,
                          n.chains=nChains , n.adapt=adaptSteps )
  # Burn-in:
  cat( "Burning in the MCMC chain...\n" )
  update( jagsModel , n.iter=burnInSteps )
  # The saved MCMC chain:
  cat( "Sampling final MCMC chain...\n" )
  codaSamples = coda.samples( jagsModel , variable.names=parameters ,
                              n.iter=nIter , thin=thinSteps )
  # resulting codaSamples object has these indices:
  #   codaSamples[[ chainIdx ]][ stepIdx , paramIdx ]

  #------------------------------------------------------------------------------
  # EXAMINE THE RESULTS
  if ( showMCMC ) {
    windows()
    autocorr.plot( codaSamples[[1]] , ask=FALSE )
  }

  # Convert coda-object codaSamples to matrix object for easier handling.
  # But note that this concatenates the different chains into one long chain.
  # Result is mcmcChain[ stepIdx , paramIdx ]
  mcmcChain = as.matrix( codaSamples )
  return( mcmcChain )

} # end function BESTmcmc
```

© John K. Kruschke, Sept. 2012 49



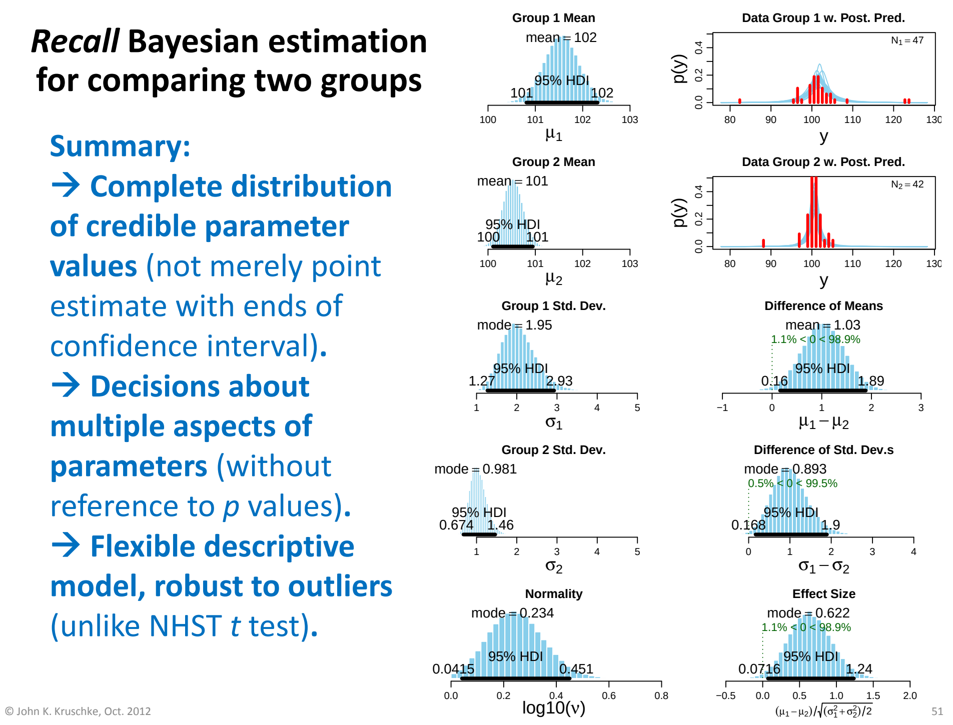
Computer Software:

Packaged for easy use!
Underlying program is never seen.

```
source("BEST.R")  # load the program

# Specify data as vectors (replace with your own data):
y1 = c(101,100,102,104,102,97,105,105,98,101,100,123,105,
       109,102,82,102,100,102,102,101,102,102,103,103,97,
       96,103,124,101,101,100,101,101,104,100,101)
y2 = c(99,101,100,101,102,100,97,101,104,101,102,102,100,
       104,100,100,100,101,102,103,97,101,101,100,101,99,
       101,100,99,101,100,102,99,100,99)

# Run the Bayesian analysis:
mcmcChain = BESTmcmc( y1 , y2 )

# Plot the results of the Bayesian analysis:
BESTplot( y1 , y2 , mcmcChain )
```

© John K. Kruschke, Sept. 2012

50



Recall Bayesian estimation for comparing two groups

Summary:

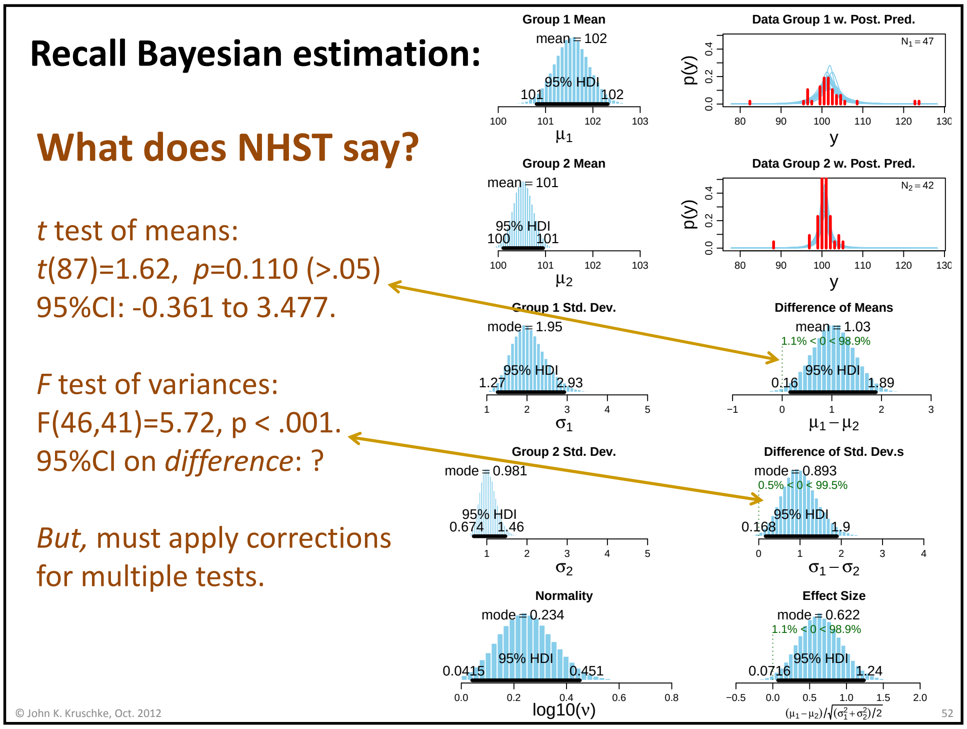
→ Complete distribution of credible parameter values (not merely point estimate with ends of confidence interval)**.**

→ Decisions about multiple aspects of parameters (without reference to *p* values)**.**

→ Flexible descriptive model, robust to outliers (unlike NHST *t* test)**.**

© John K. Kruschke, Oct. 2012

51



Recall Bayesian estimation:

What does NHST say?

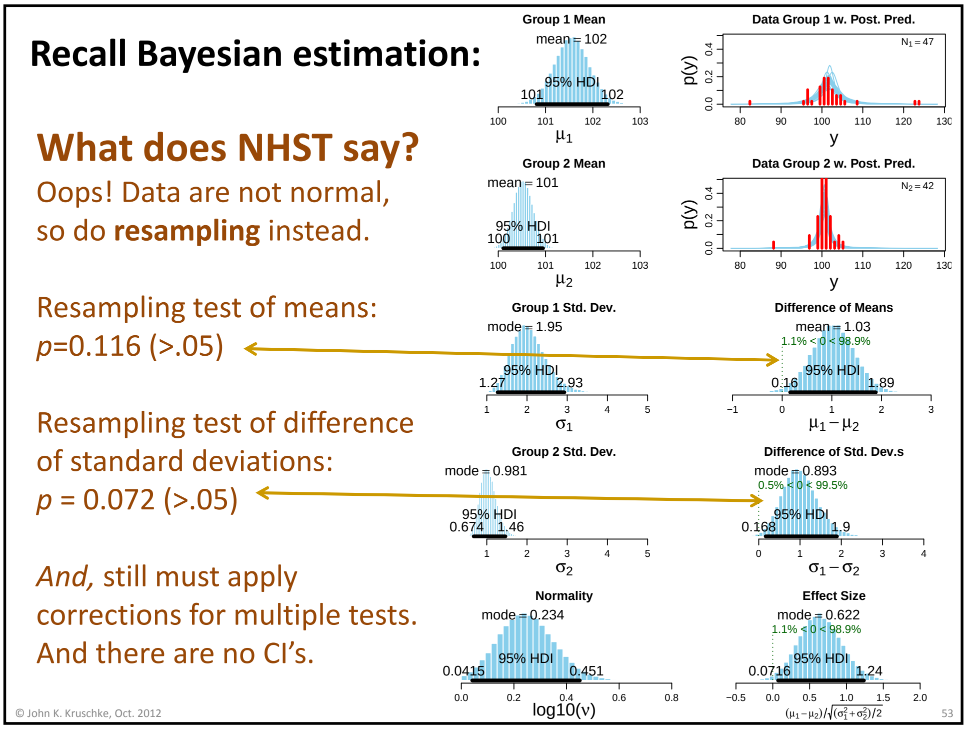
t test of means:
t(87)=1.62, *p*=0.110 (>.05)
95%CI: -0.361 to 3.477.

F test of variances:
F(46,41)=5.72, p < .001.
95%CI on *difference*: ?

But, must apply corrections for multiple tests.

© John K. Kruschke, Oct. 2012

52



Recall Bayesian estimation:

What does NHST say?

Oops! Data are not normal, so do **resampling** instead.

Resampling test of means:
p=0.116 (>.05)

Resampling test of difference of standard deviations:
p = 0.072 (>.05)

And, still must apply corrections for multiple tests. And there are no CI's.

© John K. Kruschke, Oct. 2012

53

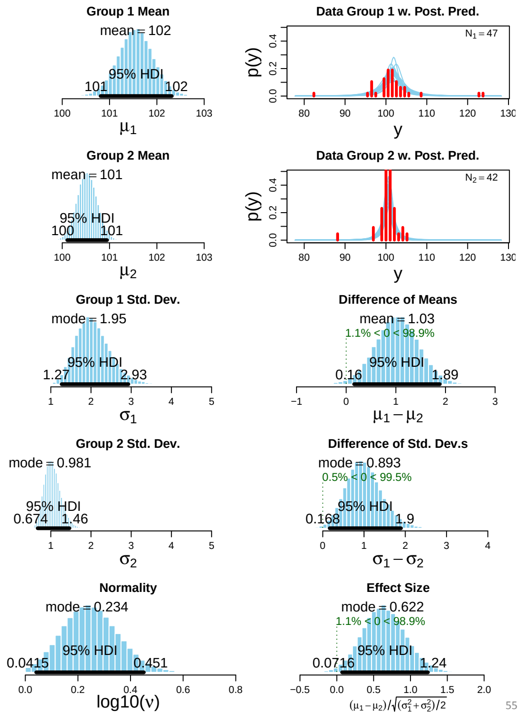
Example with outliers: BESTexample.R

Bayesian estimation:

- Credible differences between means and standard deviations.
- Complete distributional information on effect size and everything else.
- Non-normality indicated.

NHST *t* test:

- Outliers invalidate classic test.
- Resampling shows p>.05 for difference of means, p>.05 for difference of standard deviations.
- Need correction for multiple tests.
- No CI's. (And CI's would have no distributional info and fickle end points linked to fickle *p* values.)



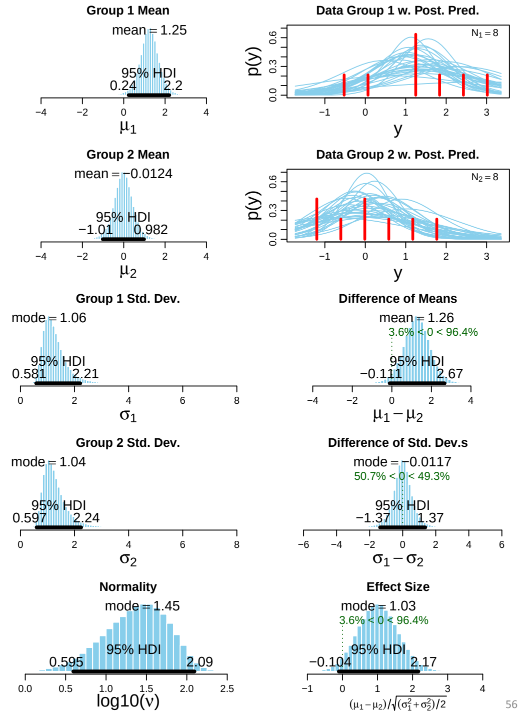
Example with small N

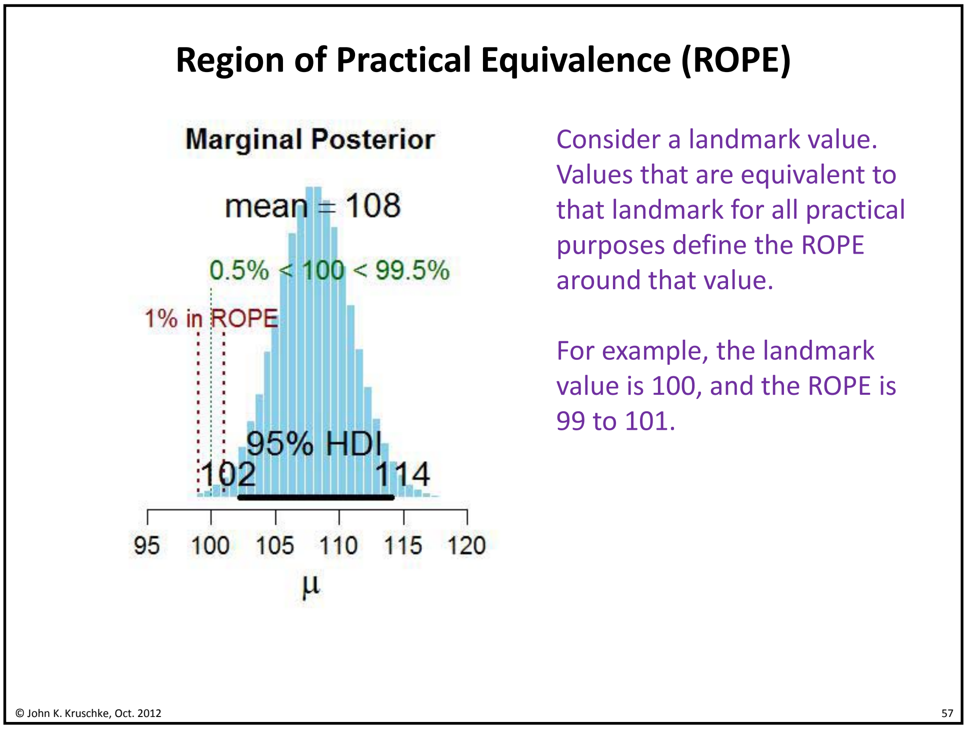
Bayesian estimation:

- **Zero is among credible differences between means** and standard deviations, and for effect size.
- Complete distributional information on effect size and everything else.
- Normality is credible.

NHST *t* test:

- *t*(14)=2.33, ***p*=0.035**, 95% CI: 0.099, 2.399. (*F*(7,7)=1.00, *p*=.999, CI on ratio: 0.20, 5.00.)
- Need correction for multiple tests, if intended.
- CI's have no distributional info and fickle end points linked to fickle *p* values.
- ***t* test fails to reveal true uncertainty in parameter estimates when simultaneously estimating SD's and normality.**



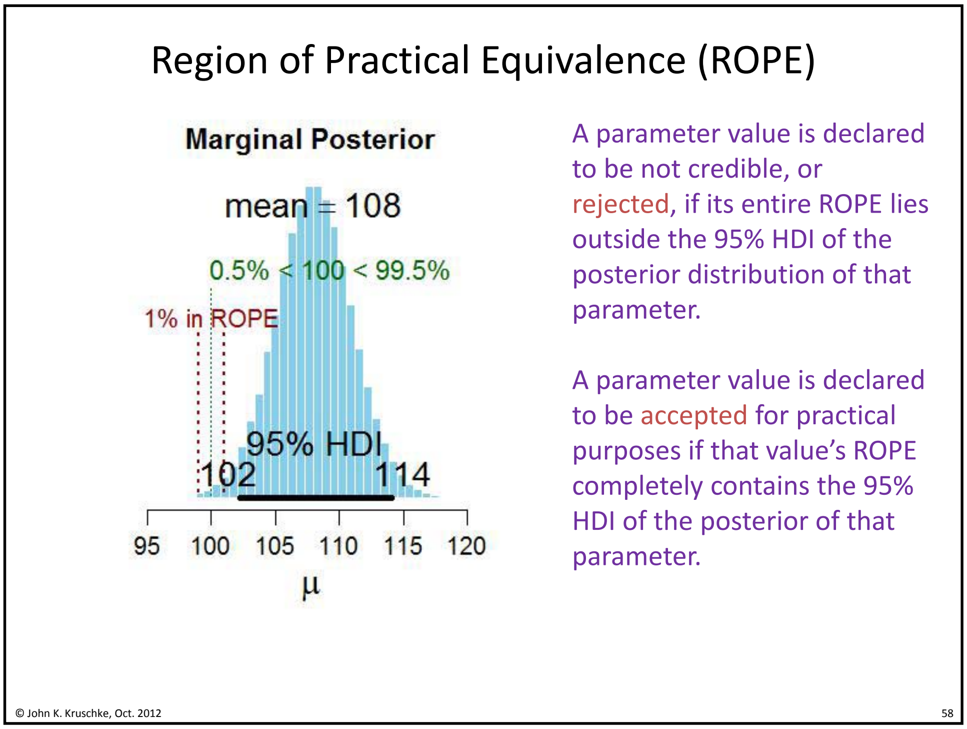


Region of Practical Equivalence (ROPE)

Consider a landmark value. Values that are equivalent to that landmark for all practical purposes define the ROPE around that value.

For example, the landmark value is 100, and the ROPE is 99 to 101.

57



Region of Practical Equivalence (ROPE)

A parameter value is declared to be not credible, or rejected, if its entire ROPE lies outside the 95% HDI of the posterior distribution of that parameter.

A parameter value is declared to be accepted for practical purposes if that value's ROPE completely contains the 95% HDI of the posterior of that parameter.

58

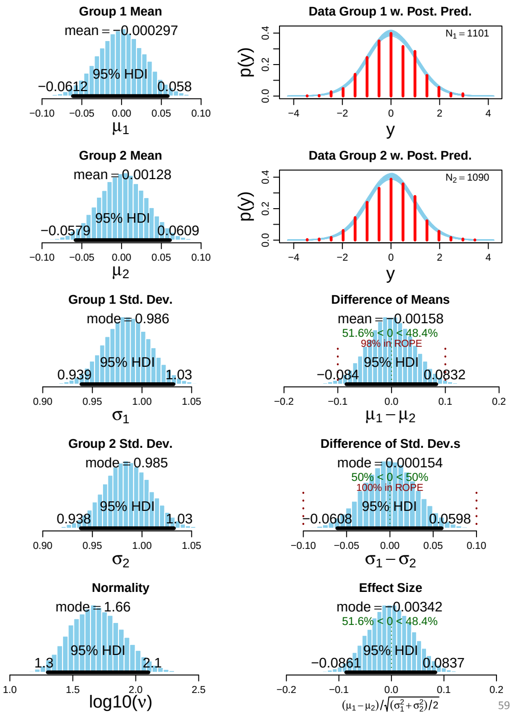
Example of accepting null value

Bayesian estimation:

- 95% HDI for difference on means falls within ROPE; same for SD's **(enlarged in next slide)**.
- Complete distributional information on effect size and everything else.
- Normality is credible.

NHST *t* test:

- *p* is large for both *t* and *F* tests, but NHST cannot accept null hypothesis.
- Need correction for multiple tests, if intended.
- CI's have no distributional info and fickle end points linked to fickle *p* values, and CI does not indicate probability of parameter value. Hence, **cannot use ROPE method in NHST.**



© John K. Kruschke, Oct. 2012

59

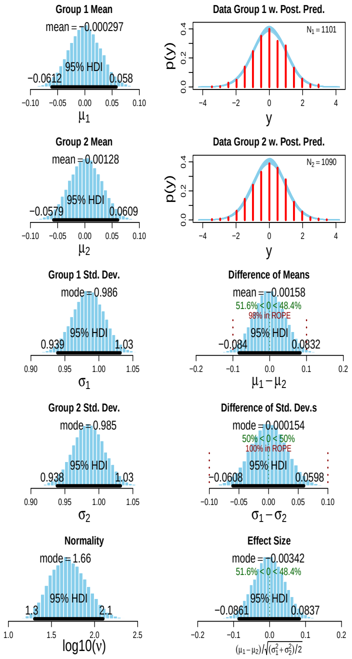
Example of accepting null value

Bayesian estimation:

- 95% HDI for difference on means falls within ROPE; same for SD's.
- Complete distributional information on effect size and everything else.
- Normality is credible.

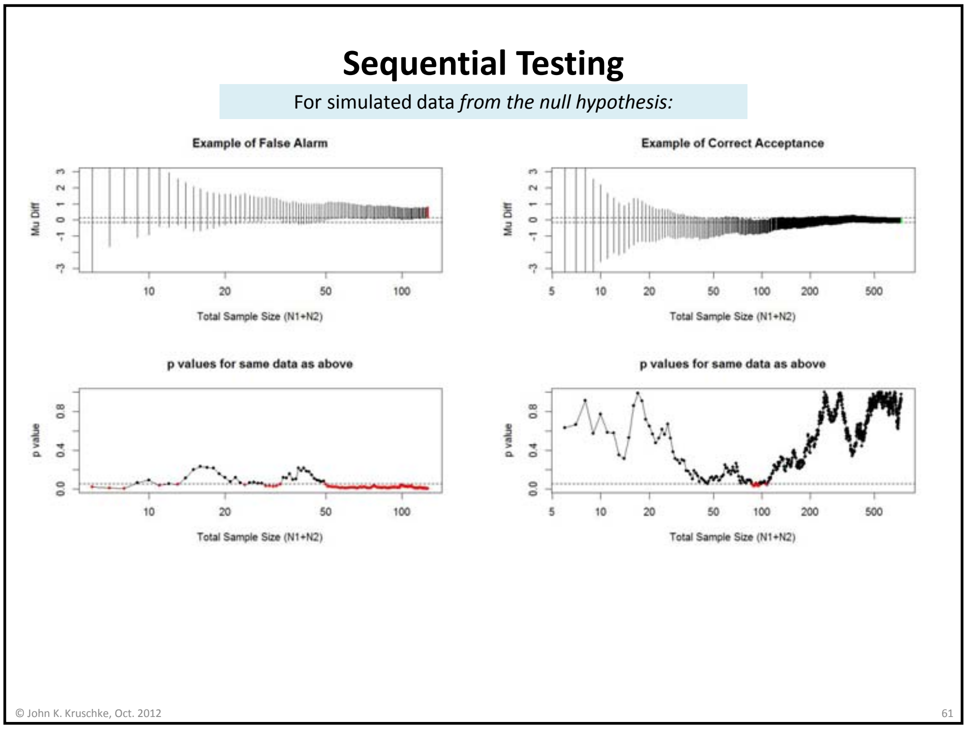
NHST *t* test:

- *p* is large for both *t* and *F* tests, but NHST cannot accept null hypothesis.
- Need correction for multiple tests, if intended.
- CI's have no distributional info and fickle end points linked to fickle *p* values, and CI does not indicate probability of parameter value. Hence, **cannot use ROPE method in NHST.**



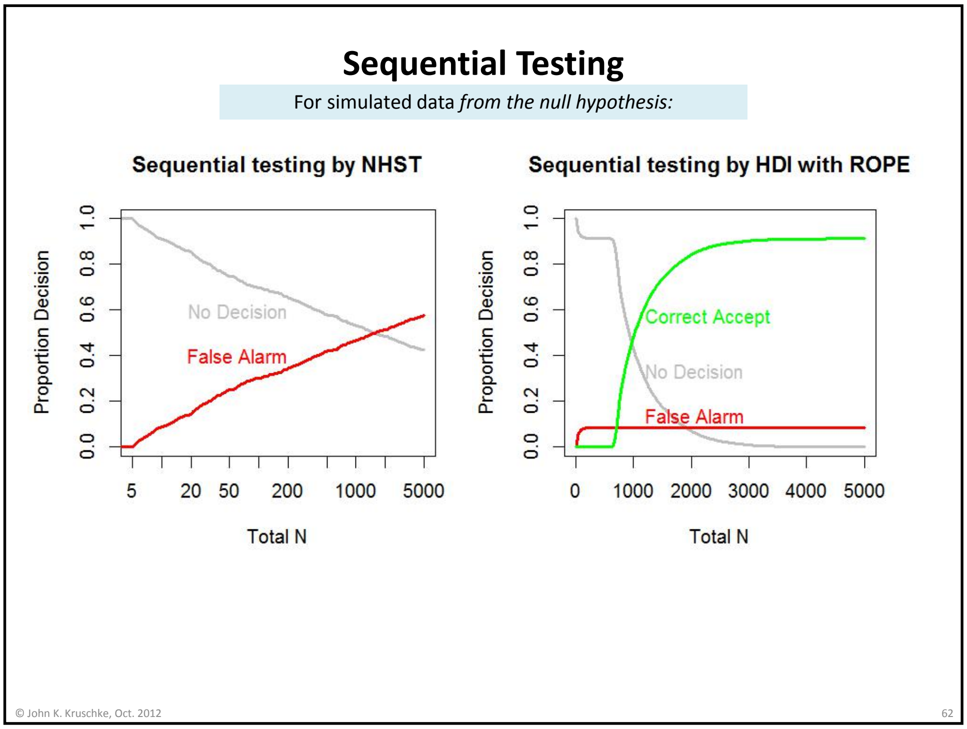
© John K. Kruschke, Oct. 2012

60



Sequential Testing

For simulated data *from the null hypothesis:*

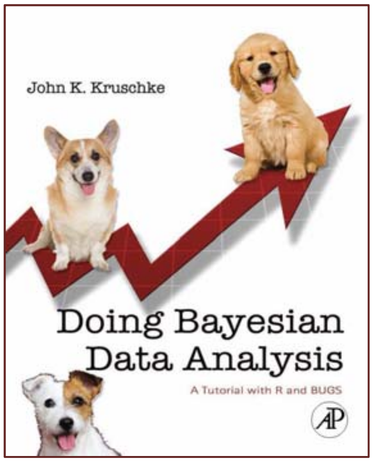


Sequential Testing

For simulated data *from the null hypothesis:*

Many other topics are in the book, e.g.

❖ Bayesian hierarchical **ANOVA, oneway and twoway with interaction contrasts**.
❖The **generalized linear model**.
❖ Many types of **regression**, including multiple linear regression, logistic regression, ordinal regression.
❖ **Log-linear models vs chi-square test**.
❖ **Power**: Probability of achieving the goals of research.
❖ All preceded by **extensive introductory chapters** covering notions of probability, Bayes' rule, MCMC, model comparison, etc.



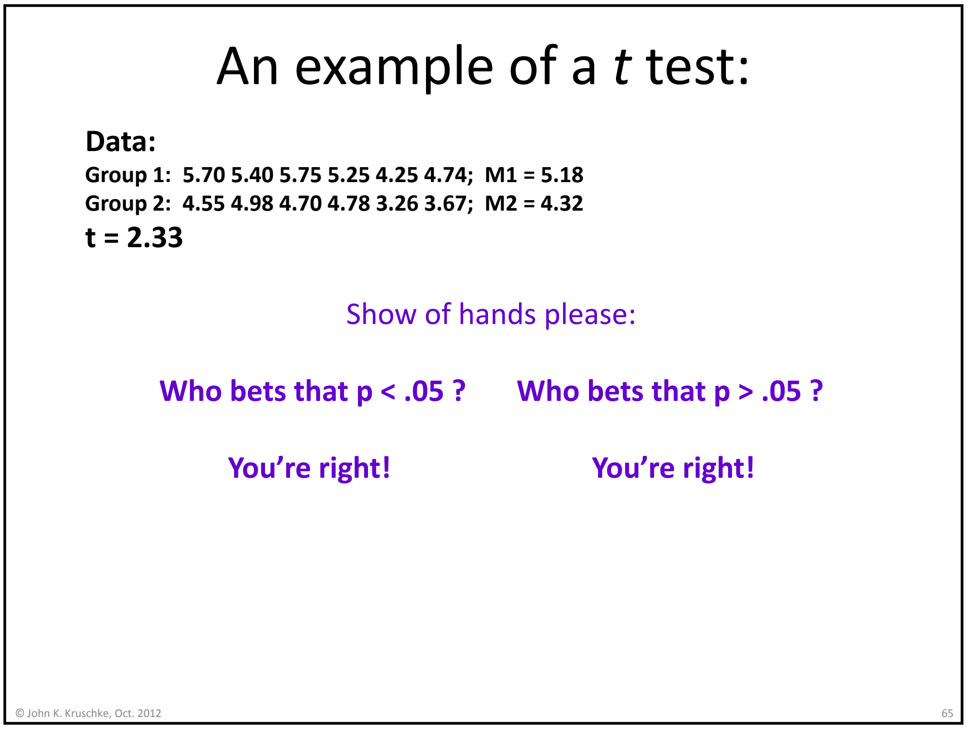
An example of a *t* test:

Data:

Group 1: 5.70 5.40 5.75 5.25 4.25 4.74; M1 = 5.18
Group 2: 4.55 4.98 4.70 4.78 3.26 3.67; M2 = 4.32
t = 2.33

Show of hands please:

Who bets that p < .05 ? **Who bets that p > .05 ?**



An example of a *t* test:

Data:

Group 1: 5.70 5.40 5.75 5.25 4.25 4.74; M1 = 5.18

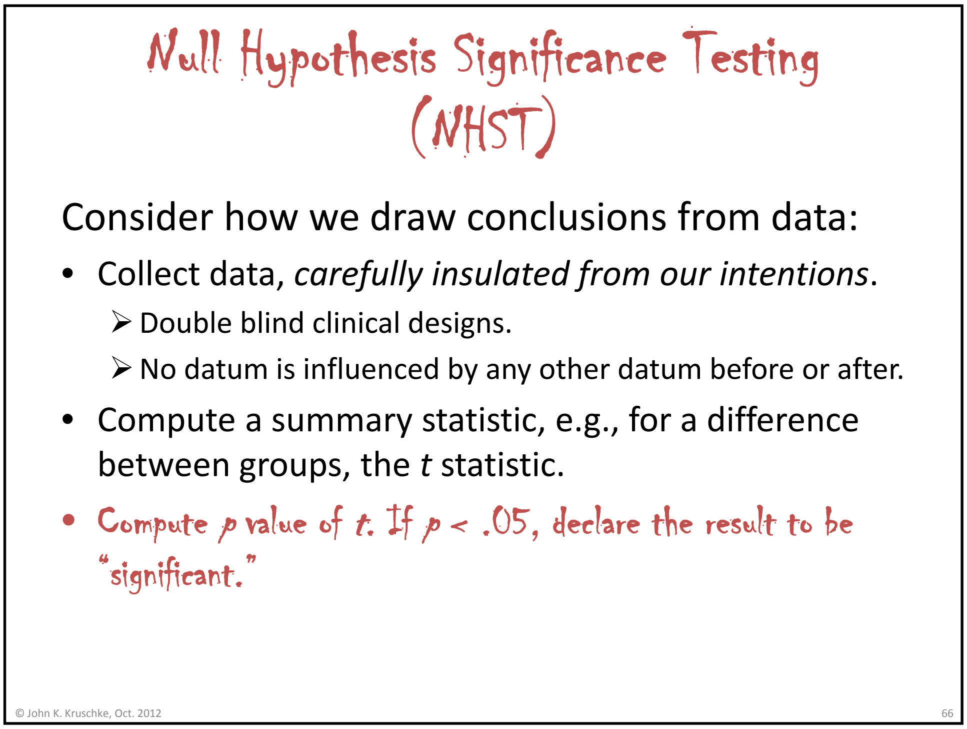
Group 2: 4.55 4.98 4.70 4.78 3.26 3.67; M2 = 4.32

t = 2.33

Show of hands please:

Who bets that p < .05 ? **Who bets that p > .05 ?**

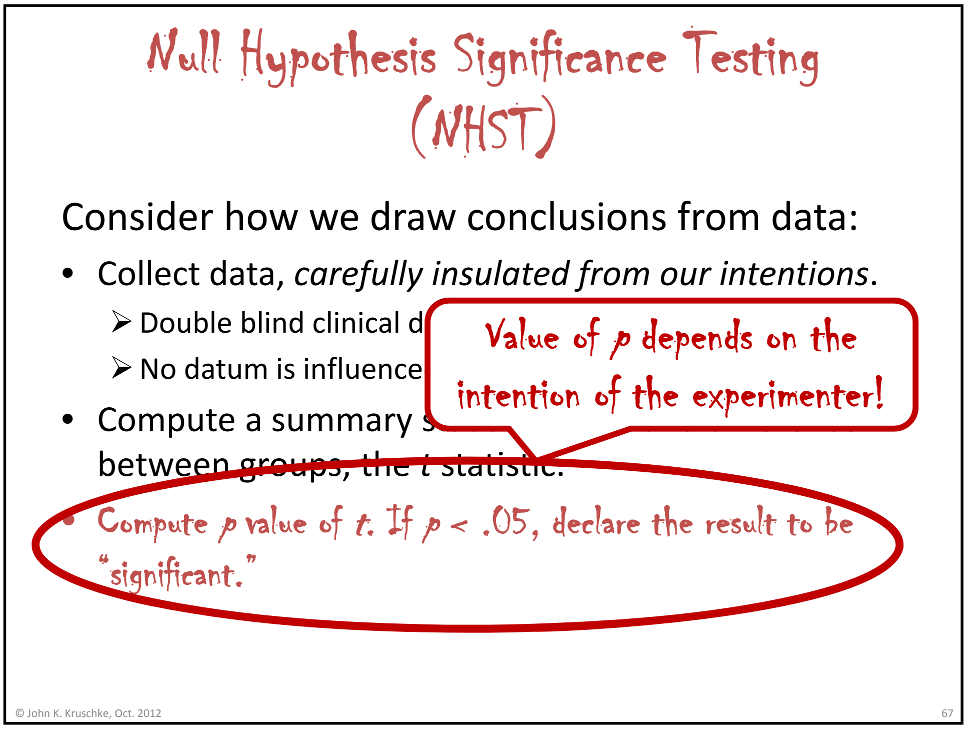
You're right! **You're right!**



Null Hypothesis Significance Testing (NHST)

Consider how we draw conclusions from data:

- Collect data, *carefully insulated from our intentions*.
  - ➢ Double blind clinical designs.
  - ➢ No datum is influenced by any other datum before or after.
- Compute a summary statistic, e.g., for a difference between groups, the *t* statistic.
- Compute *p* value of *t*. If *p* < .05, declare the result to be "significant."

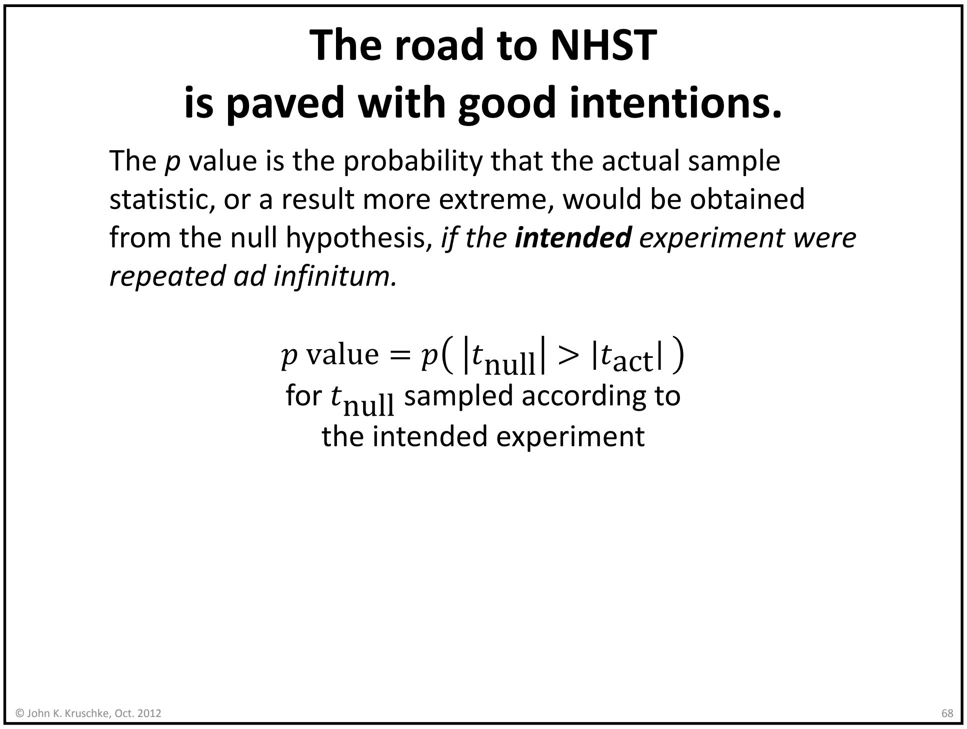


Null Hypothesis Significance Testing (NHST)

Consider how we draw conclusions from data:

- Collect data, *carefully insulated from our intentions*.
  - ➢ Double blind clinical d
  - ➢ No datum is influence
- Compute a summary s between groups, the *t* statistic.
- Compute *p* value of *t*. If $p < .05$, declare the result to be "significant."

Value of *p* depends on the intention of the experimenter!

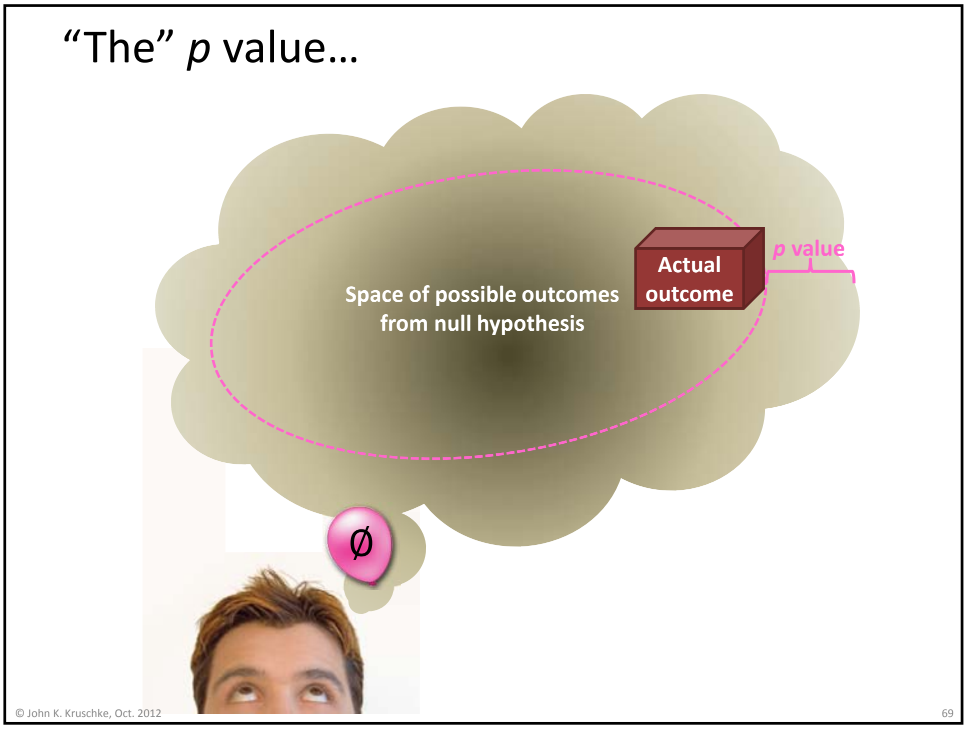


The road to NHST is paved with good intentions.

The *p* value is the probability that the actual sample statistic, or a result more extreme, would be obtained from the null hypothesis, *if the **intended** experiment were repeated ad infinitum.*

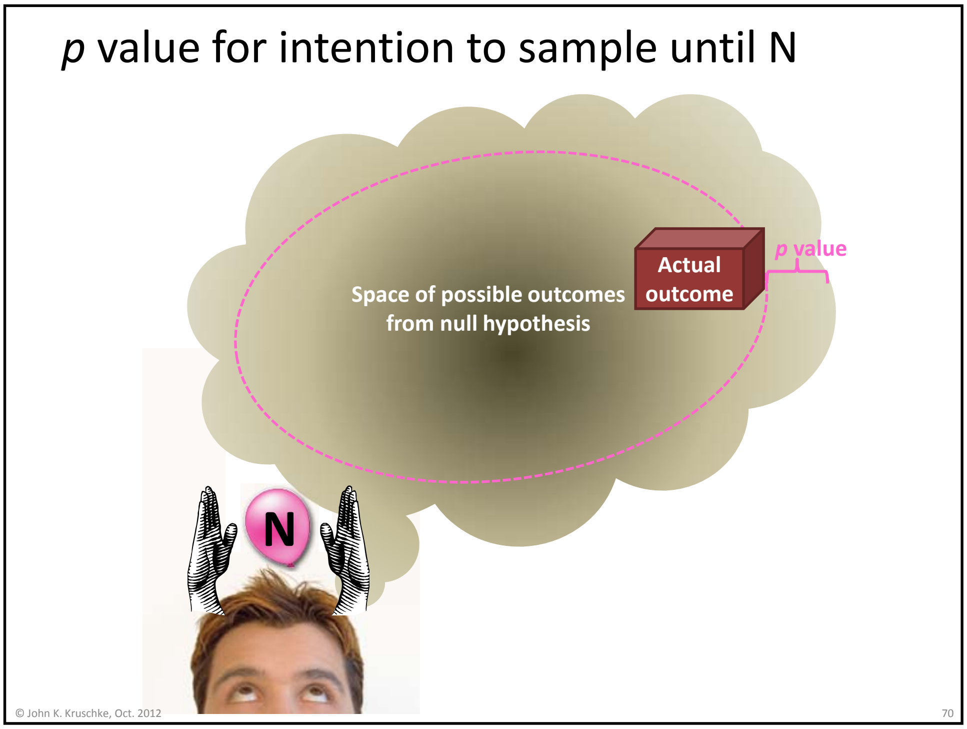
$$p \text{ value} = p\left(|t_{\text{null}}| > |t_{\text{act}}| \right)$$

for t_{null} sampled according to the intended experiment



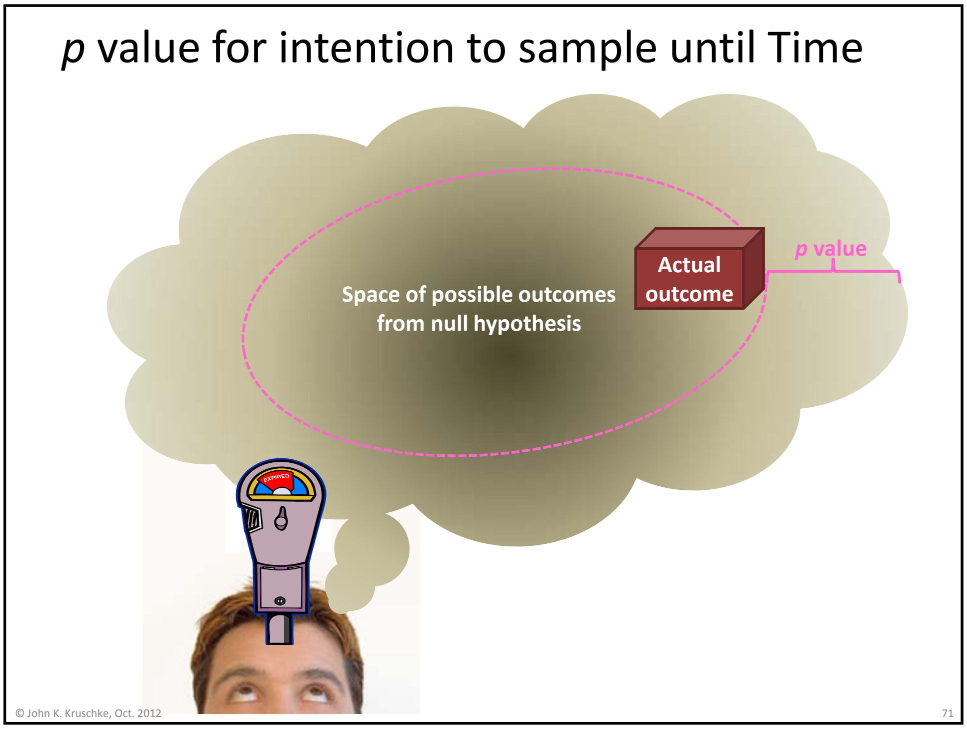
“The” *p* value…

69



p value for intention to sample until N

70



p value for intention to sample until Time

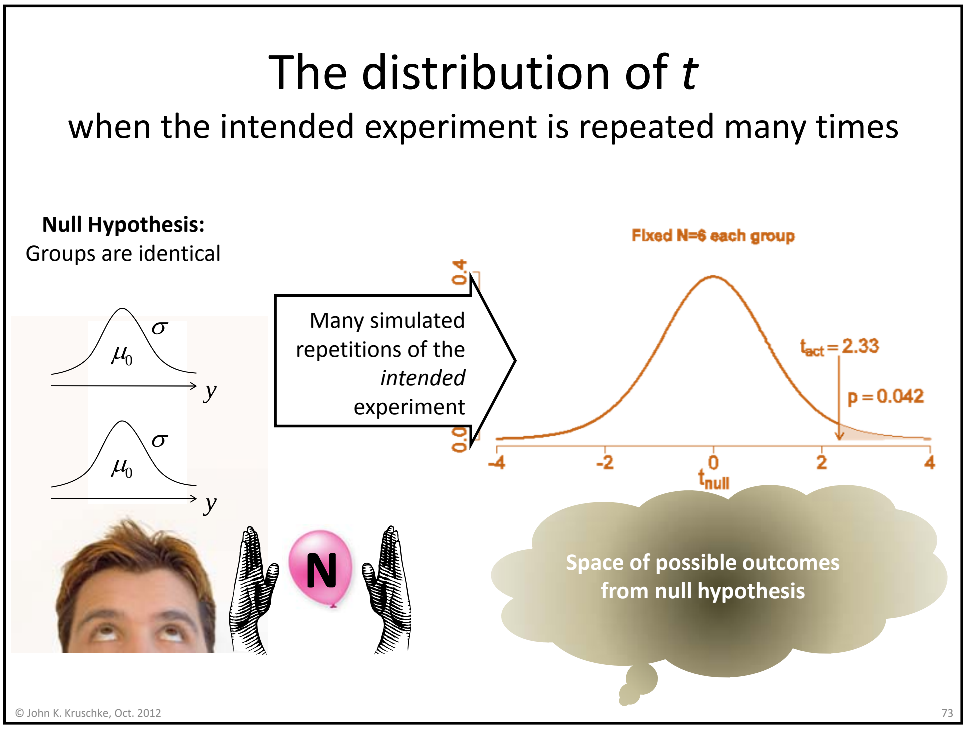
Space of possible outcomes from null hypothesis

Actual outcome

p value

© John K. Kruschke, Oct. 2012

71



The distribution of *t*

when the intended experiment is repeated many times

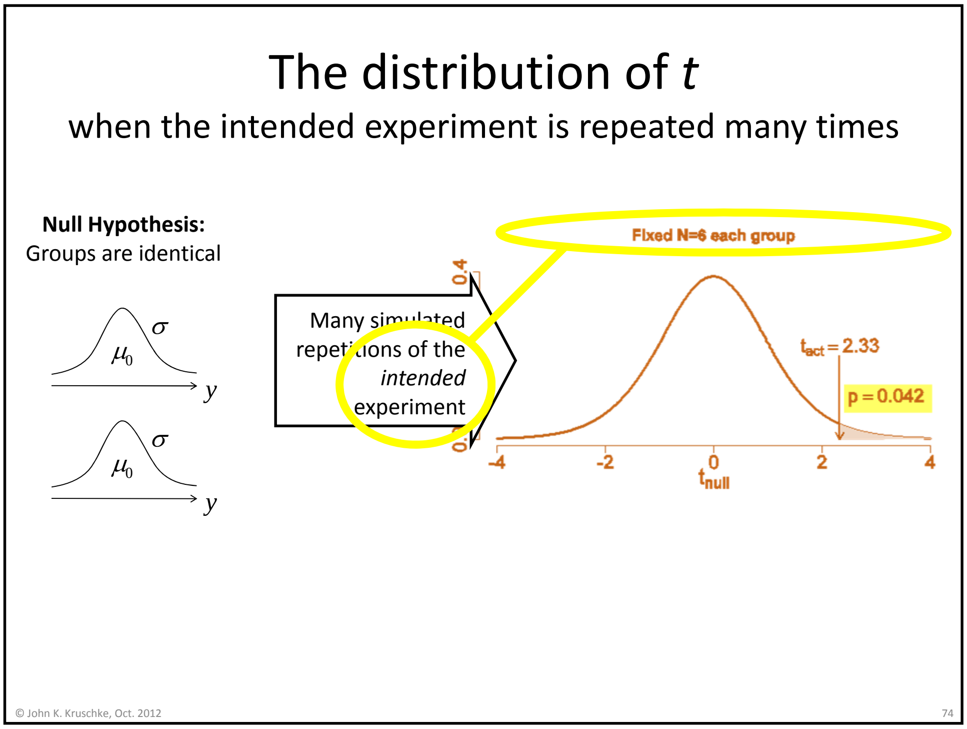
Null Hypothesis:
Groups are identical

Many simulated repetitions of the *intended* experiment

Space of possible outcomes from null hypothesis

© John K. Kruschke, Oct. 2012

73



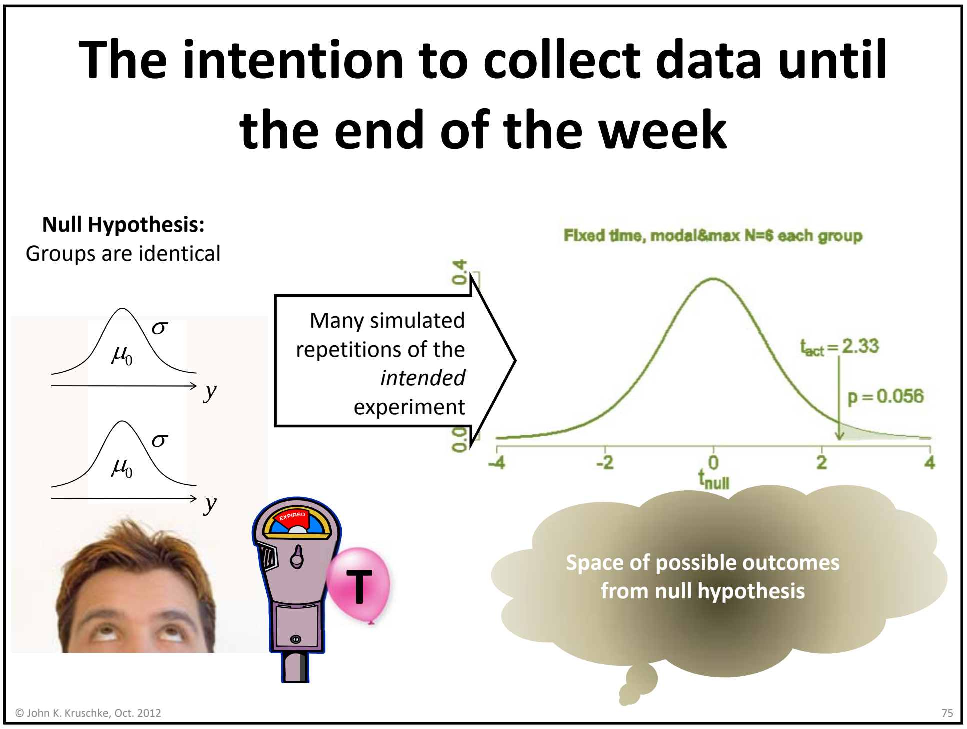
The distribution of *t*

when the intended experiment is repeated many times

Null Hypothesis:
Groups are identical

Many simulated repetitions of the *intended* experiment

74



The intention to collect data until the end of the week

Null Hypothesis:
Groups are identical

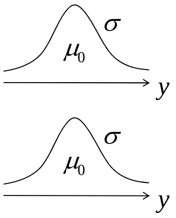
Many simulated repetitions of the *intended* experiment

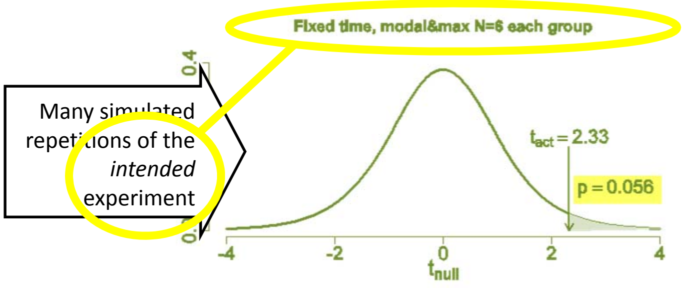
Space of possible outcomes from null hypothesis

75

The intention to collect data until the end of the week

Null Hypothesis:
Groups are identical




Many simulated repetitions of the *intended* experiment

An example of a *t* test:

Data:

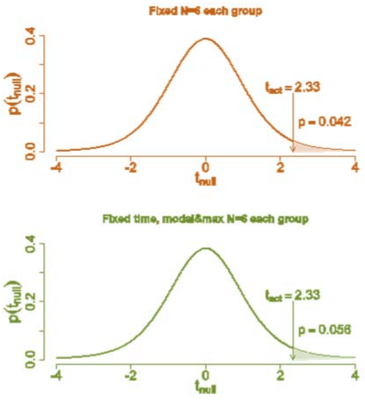
Group 1: 5.70 5.40 5.75 5.25 4.25 4.74; M1 = 5.18

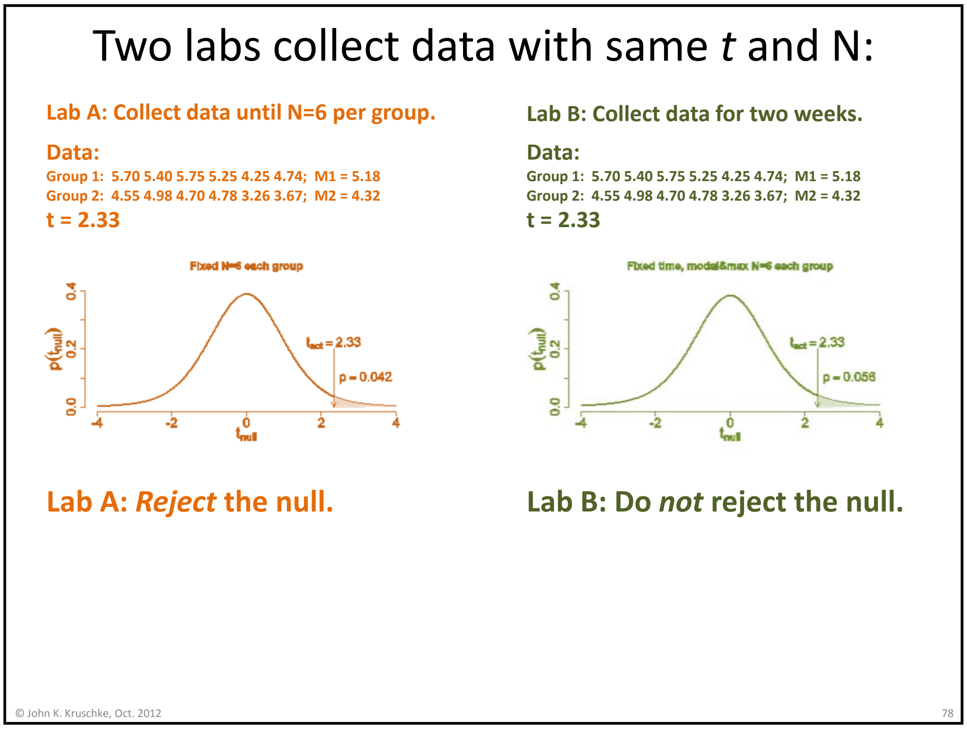
Group 2: 4.55 4.98 4.70 4.78 3.26 3.67; M2 = 4.32

t = 2.33

Can the null hypothesis be rejected? *To answer, we must know the intention of the data collector.*

- We ask the research assistant who collected the data. The assistant says, "I just collected data for two weeks. It's my job. I happened to get 6 subjects in each group."
- We ask the graduate student who oversaw the assistant. The student says, "I knew we needed 6 subjects per group, so I told the assistant to run for two weeks, because we usually get about 6 subjects per week."
- We ask the lab director, who says, "I told my graduate student to collect 6 subjects per group."
- Therefore, for the lab director, t = 2.33 **rejects the null hypothesis** (because p < .05), but for the research assistant who actually collected the data, t = 2.33 **fails to reject** the null hypothesis (because p > .05).





Two labs collect data with same *t* and N:

Lab A: Collect data until N=6 per group.

Data:

Group 1: 5.70 5.40 5.75 5.25 4.25 4.74; M1 = 5.18
Group 2: 4.55 4.98 4.70 4.78 3.26 3.67; M2 = 4.32

t = 2.33

Lab B: Collect data for two weeks.

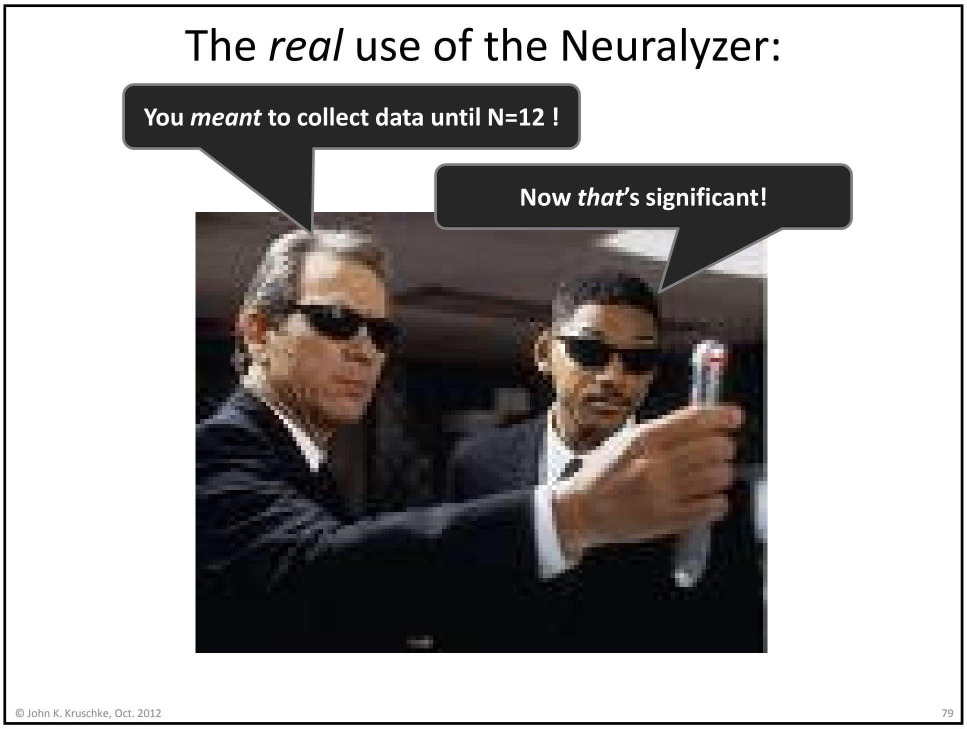
Data:

Group 1: 5.70 5.40 5.75 5.25 4.25 4.74; M1 = 5.18
Group 2: 4.55 4.98 4.70 4.78 3.26 3.67; M2 = 4.32

t = 2.33

Lab A: *Reject* the null.

Lab B: Do *not* reject the null.



The *real* use of the Neuralyzer:

You *meant* to collect data until N=12 !
Now *that*'s significant!

Problem is not solved by “fixing” the intention

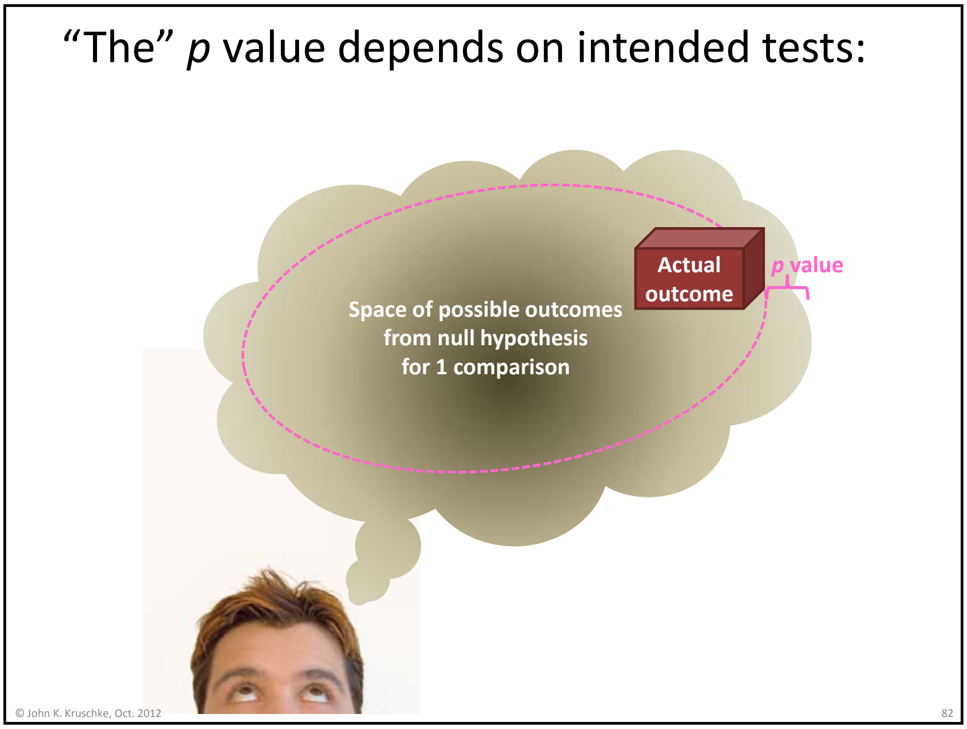
- All we need to do is decide in advance exactly what our intention is (or use a Neuralyzer after the fact), and have everybody chant a mantra to keep that intention fixed in their minds while the experiment is being conducted. Right?
- Wrong. The data don’t know our intention, and the same data could have been collected under many other intentions.

The intention to examine data thoroughly

Many experiments involve multiple groups, and **multiple comparisons** of means.

Example: Consider 2 different drugs from chemical family A, 2 different drugs from chemical family B, and a placebo group. Lots of possible comparisons…

Problem: With every test, there is possibility of false alarm! False alarms are bad; therefore, keep the experimentwise false alarm rate down to 5%.



“The” *p* value depends on intended tests:

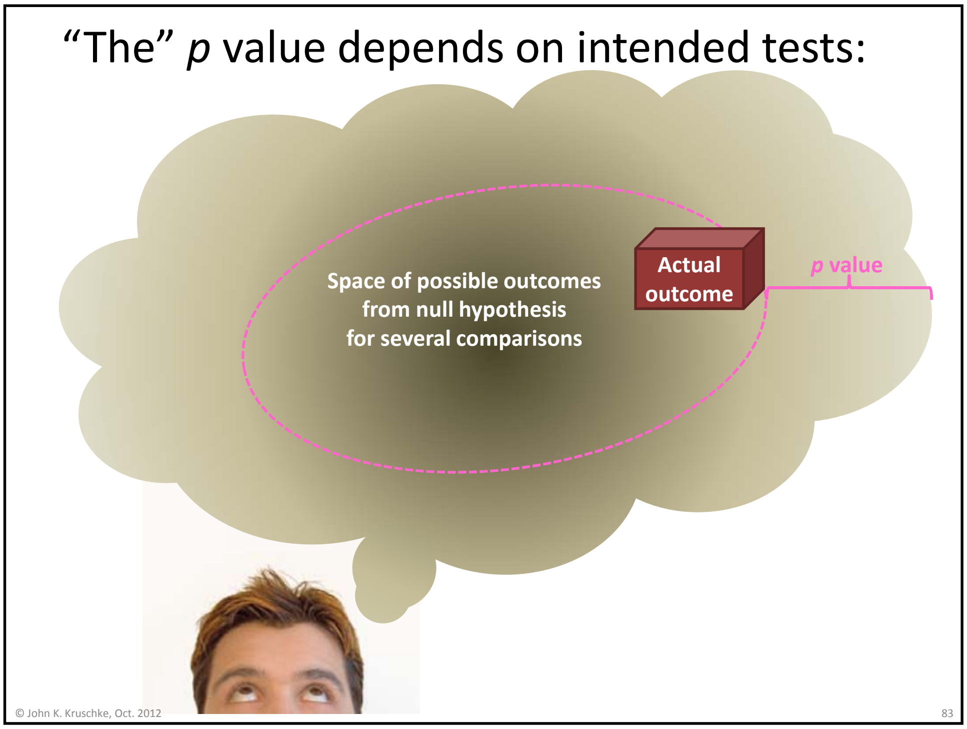
Actual outcome

p value

Space of possible outcomes
from null hypothesis
for 1 comparison

© John K. Kruschke, Oct. 2012

82



“The” *p* value depends on intended tests:

Actual outcome

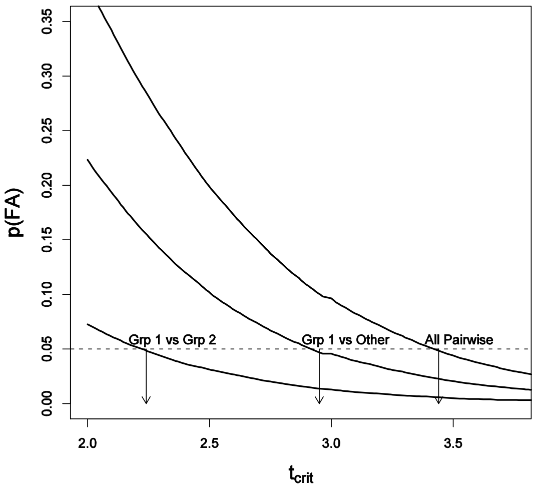
p value

Space of possible outcomes
from null hypothesis
for several comparisons

© John K. Kruschke, Oct. 2012

83

Experimentwise false alarm rate

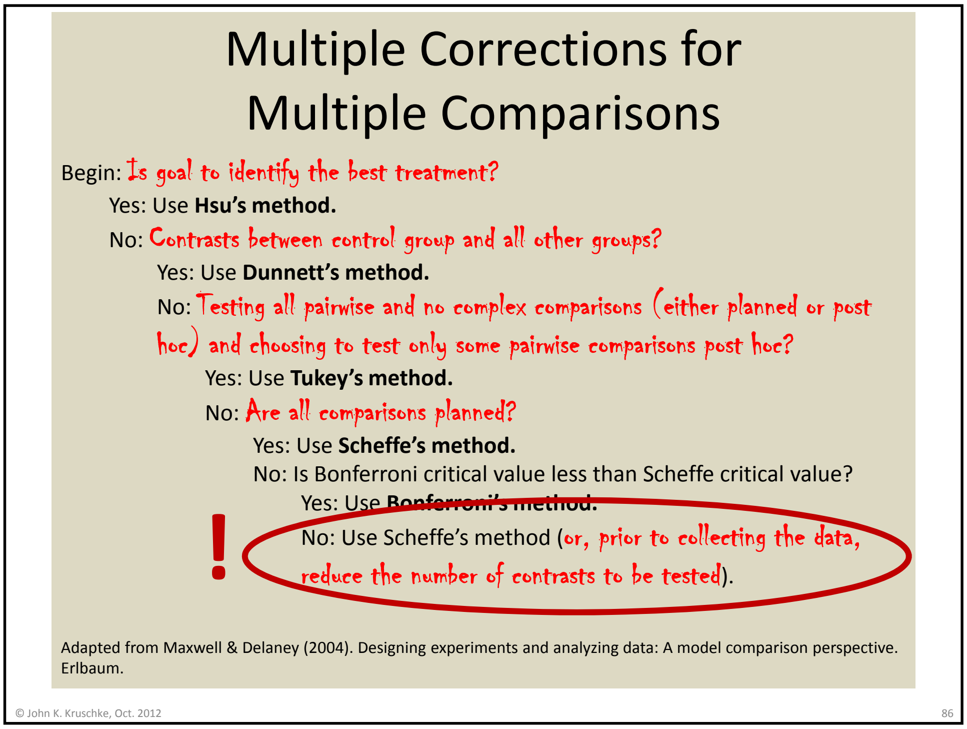


Multiple Corrections for Multiple Comparisons

Begin: Is goal to identify the best treatment?

- Yes: Use **Hsu's method.**
- No: Contrasts between control group and all other groups?
  - Yes: Use **Dunnett's method.**
  - No: Testing all pairwise and no complex comparisons (either planned or post hoc) and choosing to test only some pairwise comparisons post hoc?
    - Yes: Use **Tukey's method.**
    - No: Are all comparisons planned?
      - Yes: Use **Scheffe's method.**
      - No: Is Bonferroni critical value less than Scheffe critical value?
        - Yes: Use **Bonferroni's method.**
        - No: Use Scheffe's method (or, prior to collecting the data, reduce the number of contrasts to be tested).

Adapted from Maxwell & Delaney (2004). Designing experiments and analyzing data: A model comparison perspective. Erlbaum.

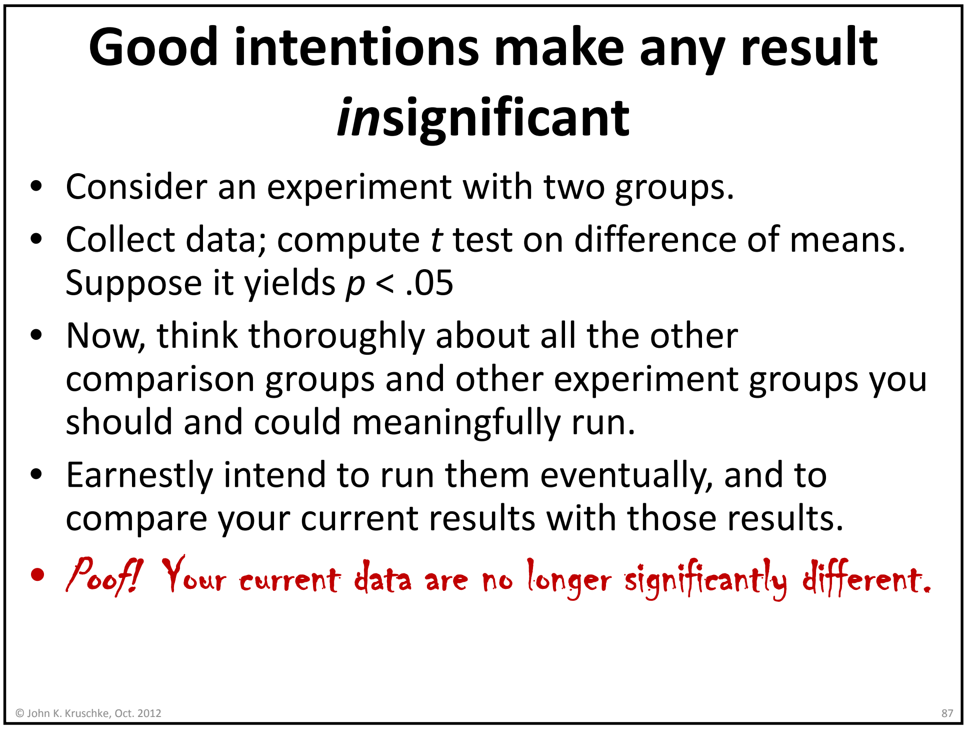


Multiple Corrections for Multiple Comparisons

Begin: Is goal to identify the best treatment?

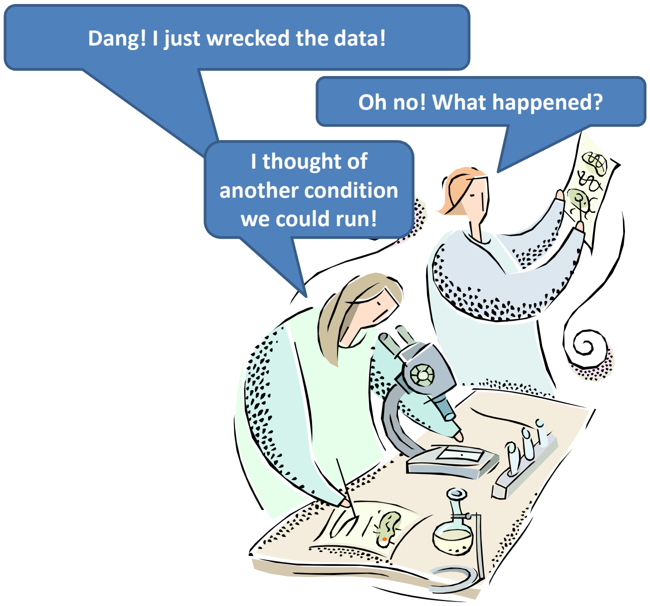
- Yes: Use **Hsu's method.**
- No: Contrasts between control group and all other groups?
  - Yes: Use **Dunnett's method.**
  - No: Testing all pairwise and no complex comparisons (either planned or post hoc) and choosing to test only some pairwise comparisons post hoc?
    - Yes: Use **Tukey's method.**
    - No: Are all comparisons planned?
      - Yes: Use **Scheffe's method.**
      - No: Is Bonferroni critical value less than Scheffe critical value?
        - Yes: Use **Bonferroni's method.**
        - ! No: Use Scheffe's method (or, prior to collecting the data, reduce the number of contrasts to be tested).

Adapted from Maxwell & Delaney (2004). Designing experiments and analyzing data: A model comparison perspective. Erlbaum.



Good intentions make any result *in*significant

- Consider an experiment with two groups.
- Collect data; compute *t* test on difference of means. Suppose it yields $p < .05$
- Now, think thoroughly about all the other comparison groups and other experiment groups you should and could meaningfully run.
- Earnestly intend to run them eventually, and to compare your current results with those results.
- Poof! Your current data are no longer significantly different.


Dang! I just wrecked the data!
Oh no! What happened?
I thought of another condition we could run!

Good intentions make many results *significant*

- Consider an experiment with two groups.
- Collect data; compute *t* test on difference of means, using df corresponding to actual N. Suppose *p* > .05, but not by much.
- *You had intended to collect a much larger sample size, but you were unexpectedly interrupted*.
- Use the larger intended N for df in the *t* test.
- Poof! Your current data are now significantly different!

Confidence Intervals provide no confidence

Data:

Group 1: 5.70 5.40 5.75 5.25 4.25 4.74; M1 = 5.18
Group 2: 4.55 4.98 4.70 4.78 3.26 3.67; M2 = 4.32

Under assumption of fixed N:

$$\boldsymbol{CI} = (M_1 - M_2) \ \pm \ t_{crit} \times se$$
$$= (5.18 - 4.32) \ \pm \ 2.23 \times 0.370$$
$$= [\, \mathbf{0.036} \,,\, \mathbf{1.68} \,]$$

which *ex*cludes zero.

Under assumption of fixed duration:

$$\boldsymbol{CI} = (M_1 - M_2) \ \pm \ t_{crit} \times se$$
$$= (5.18 - 4.32) \ \pm \ 2.45 \times 0.370$$
$$= [\, \mathbf{-0.046} \,,\, \mathbf{1.77} \,]$$

which *in*cludes zero.

95% CI constructed with fixed-N t_{crit} will span true difference *less* than 95% of time if data are sampled according to fixed duration.

95% CI constructed with fixed-duration t_{crit} will span true difference *more* than 95% of the time if data are sampled according to fixed N.

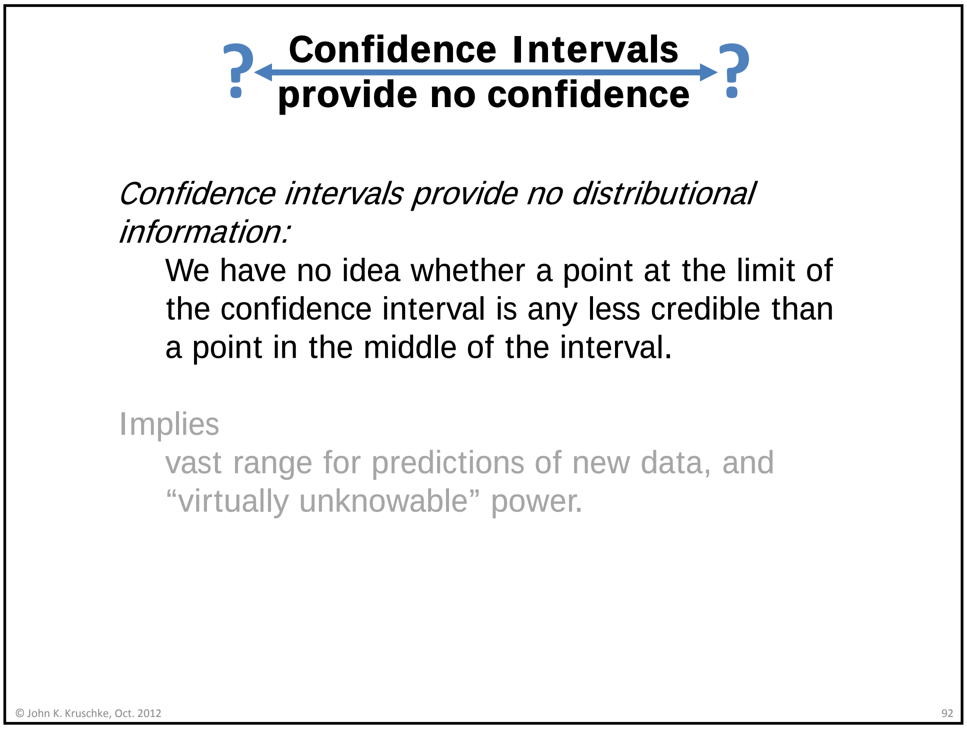
Confidence Intervals provide no confidence

General definition of CI:

95% CI is the range of parameter values (e.g., $\mu_1 - \mu_2$) that would not be rejected by $p < .05$

Hence, *the 95% CI is as ill-defined as the p value.*

We see this dramatically in confidence intervals corrected for multiple comparisons.



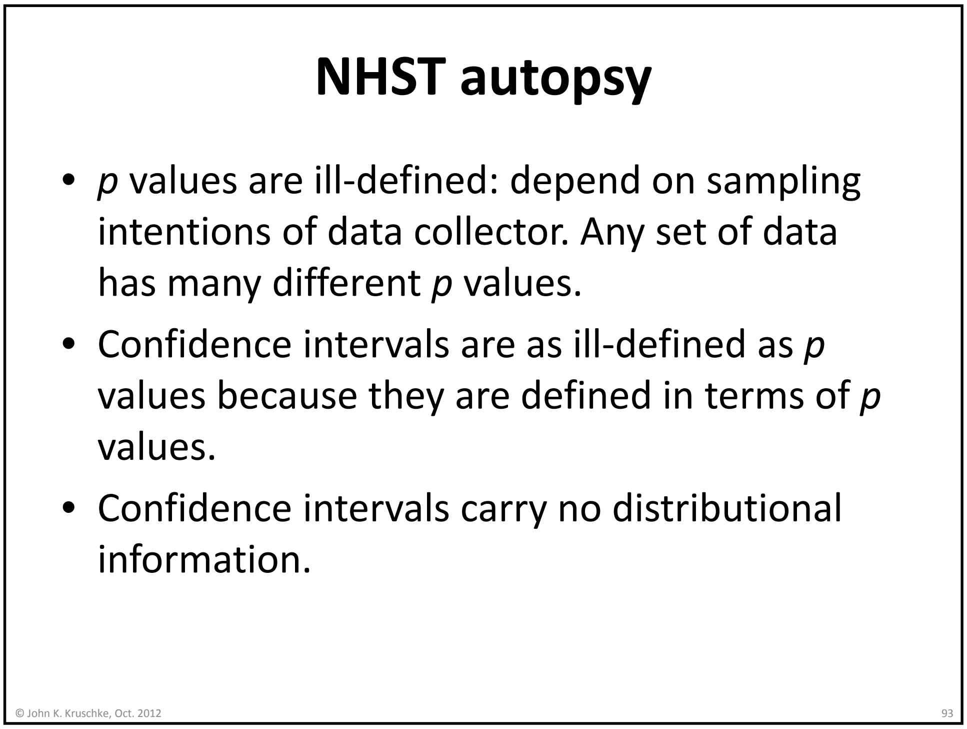
? Confidence Intervals provide no confidence ?

Confidence intervals provide no distributional information:

We have no idea whether a point at the limit of the confidence interval is any less credible than a point in the middle of the interval.

Implies

vast range for predictions of new data, and “virtually unknowable” power.



NHST autopsy

- *p* values are ill-defined: depend on sampling intentions of data collector. Any set of data has many different *p* values.
- Confidence intervals are as ill-defined as *p* values because they are defined in terms of *p* values.
- Confidence intervals carry no distributional information.

Bayesian Estimation or NHST?

When Bayesian estimation and NHST ***agree***, which should be used?

Bayesian estimation gives the most complete and informative answer. Answer from NHST is not informative and is fickle.

When Bayesian estimation and NHST ***dis****agre*e, which should be used?

Bayesian estimation gives the most complete and informative answer. Answer from NHST is not informative and is fickle.

Conclusion

- p values are not well defined, nor are the limits of confidence intervals, and confidence intervals have no distributional info.
- Bayesian data analysis is the most complete and normatively correct way to estimate parameters in any model, for all your data.
- Bayesian data analysis is taking hold in 21st century science, from astronomy to zoology. *Don't be left behind.*
- And, for more info, …

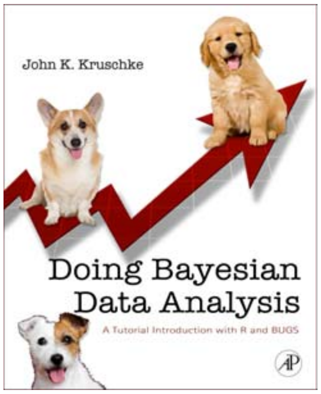
The blog: http://doingbayesiandataanalysis.blogspot.com/

Kruschke, J. K. (2012). **Bayesian estimation supersedes the *t* test.** ***Journal of Experimental Psychology: General***.

Kruschke, J. K. (2011). **Bayesian assessment of null values via parameter estimation and model comparison.** ***Perspectives on Psychological Science***, 6(3), 299-312.

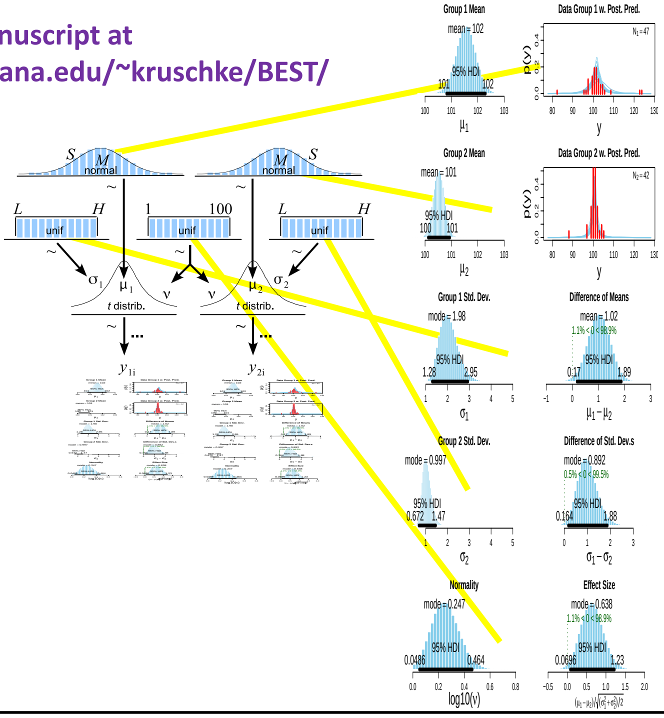
Kruschke, J. K. (2010). **What to believe: Bayesian methods for data analysis.** ***Trends in Cognitive Sciences***, 14(7), 293-300.

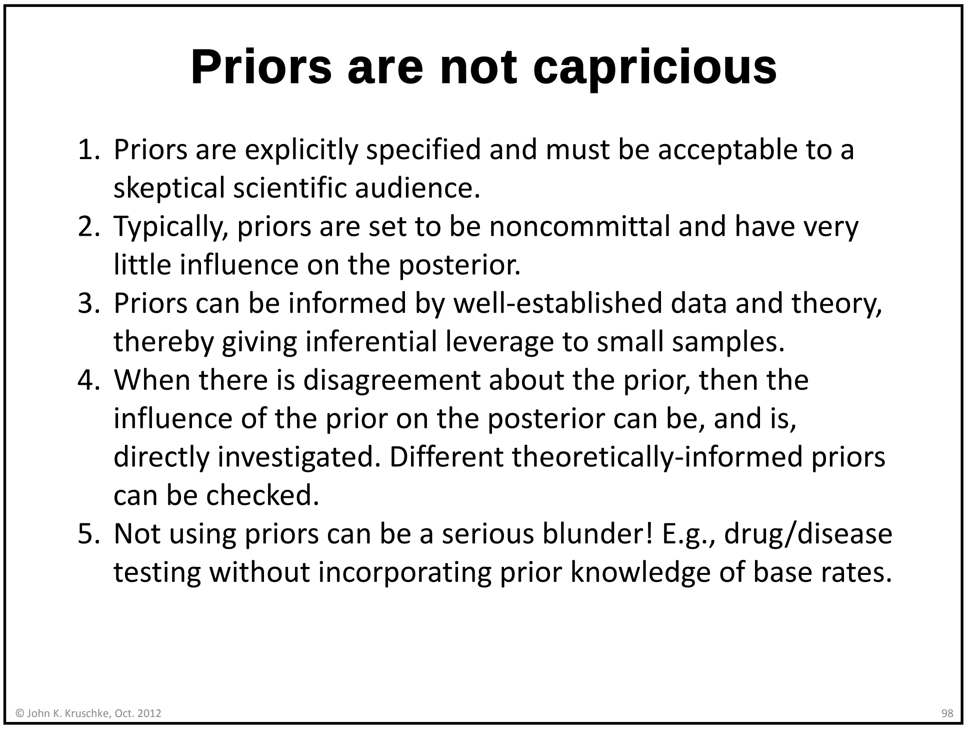
Kruschke, J. K. (2010). **Bayesian data analysis.** ***Wiley Interdisciplinary Reviews: Cognitive Science***, 1(5), 658-676.


John K. Kruschke
Doing Bayesian Data Analysis
A Tutorial Introduction with R and BUGS
AP

Kruschke, J. K. (2011). ***Doing Bayesian Data Analysis: A Tutorial with R and BUGS.*** Academic Press / Elsevier.

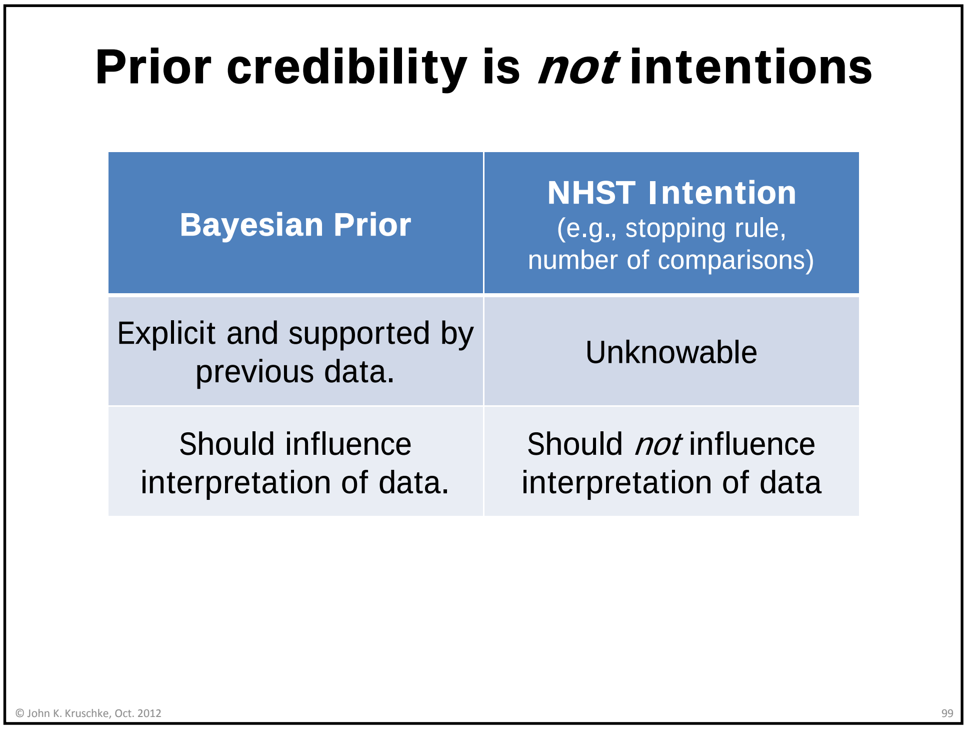
Program and manuscript at http://www.indiana.edu/~kruschke/BEST/





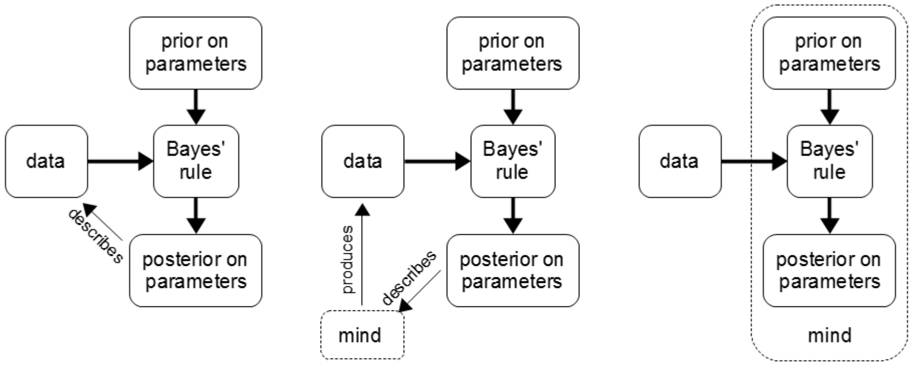
Priors are not capricious

1. Priors are explicitly specified and must be acceptable to a skeptical scientific audience.
2. Typically, priors are set to be noncommittal and have very little influence on the posterior.
3. Priors can be informed by well-established data and theory, thereby giving inferential leverage to small samples.
4. When there is disagreement about the prior, then the influence of the prior on the posterior can be, and is, directly investigated. Different theoretically-informed priors can be checked.
5. Not using priors can be a serious blunder! E.g., drug/disease testing without incorporating prior knowledge of base rates.



Prior credibility is *not* intentions

Bayesian Prior	**NHST Intention** (e.g., stopping rule, number of comparisons)
Explicit and supported by previous data.	Unknowable
Should influence interpretation of data.	Should *not* influence interpretation of data



Data analysis: Model and its parameters merely describe trends in data. The parameters need not refer to anything in the process that generated the data.

Psychometric model: Model and its parameters describe trends in *behavioral* data, and *parameters refer to mental constructs*. But there need not be anything Bayesian that generated the behavioral data.

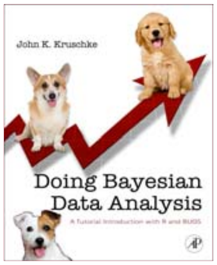
Bayesian model of mind: Model and its parameters describe mental representations, and Bayesian inference is done in/by the mind (enclosed in dashed line). Data come into the mind from the world.

http://doingbayesiandataanalysis.blogspot.com/2011/10/bayesian-models-of-mind-psychometric.html

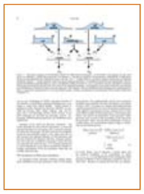
Bayesian estimation or Bayesian model comparison?

Bayesian estimation is also better than the "Bayesian *t* test," which uses the "Bayes factor" from Bayesian model comparison…

Kruschke, J. K. (2011). **Bayesian assessment of null values via parameter estimation and model comparison.** ***Perspectives on Psychological Science***, 6(3), 299-312.


John K. Kruschke
Doing Bayesian Data Analysis
A Tutorial Introduction with R and BUGS

Chapter 12 of Kruschke, J. K. (2011). ***Doing Bayesian Data Analysis: A Tutorial with R and BUGS.*** Academic Press / Elsevier.



Kruschke, J. K. (in press). **Bayesian estimation supersedes the *t* test.** ***Journal of Experimental Psychology: General***. Appendix D.