Hypothesis Testing and Model Comparison

Peter Hoff
Duke STA 610

NELS data

```
nels[1:10,]

##    school enroll flp public urbanicity hwh   ses mscore
## 1    1011      5   3      1      urban   2 -0.23  52.11
## 2    1011      5   3      1      urban   0  0.69  57.65
## 3    1011      5   3      1      urban   4 -0.68  66.44
## 4    1011      5   3      1      urban   5 -0.89  44.68
## 5    1011      5   3      1      urban   3 -1.28  40.57
## 6    1011      5   3      1      urban   5 -0.93  35.04
## 7    1011      5   3      1      urban   1  0.36  50.71
## 8    1011      5   3      1      urban   4 -0.24  66.17
## 10   1011      5   3      1      urban   8 -1.07  46.17
## 11   1011      5   3      1      urban   2 -0.10  58.76
```

Macro predictors

flp: percent category of students on the flp

flp=1 0-5% students on flp;

flp=2 5-30% students on flp;

flp=3 $>$ 30% students on flp.

```
table(tapply(nels$flp,nels$school,mean))

##
##   1   2   3
## 226 257 201
```

enroll: roughly the number of grade-10 students, in hundreds.

```
table(tapply(nels$enroll,nels$school,mean))

##
##   0   1   2   3   4   5
## 149 112 118  98 108  99
```

Macro predictors

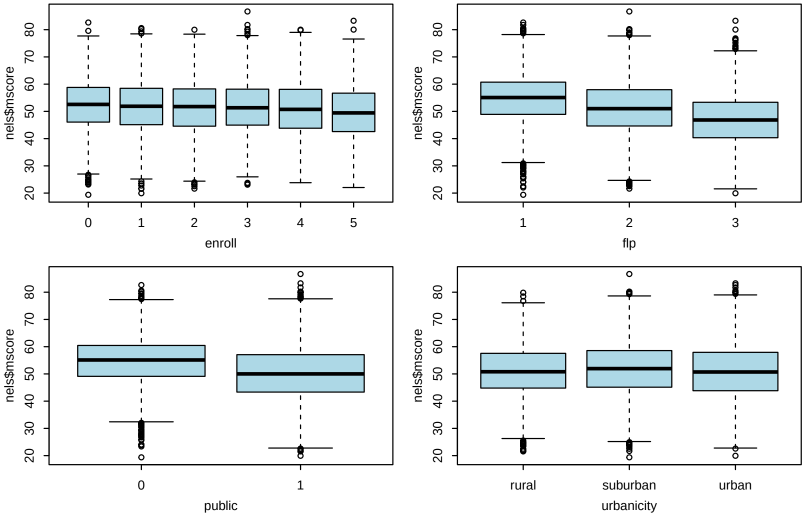
`public`: public or private school.

```
table(tapply(nels$public,nels$school,mean))

##
##   0   1
## 168 516
```

`urbanicity`: rural, suburban or urban.

```
table(tapply(nels$urbanicity,nels$school,function(x){x[1]} ))

##
##   1   2   3
## 125 324 235
```

Macro effects on `mscore`



What is wrong with the following?

Heterogeneity due to `enroll`:

```
anova(lm(mscore~as.factor(enroll),data=nels))

## Analysis of Variance Table
##
## Response: mscore
##                      Df  Sum Sq Mean Sq F value    Pr(>F)
## as.factor(enroll)     5    8660 1732.02   18.14 < 2.2e-16 ***
## Residuals         12968 1238175   95.48
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Heterogeneity due to `urbanicity`:

```
anova(lm(mscore~as.factor(urbanicity),data=nels))

## Analysis of Variance Table
##
## Response: mscore
##                          Df  Sum Sq Mean Sq F value    Pr(>F)
## as.factor(urbanicity)     2    2652 1325.87  13.823 1.008e-06 ***
## Residuals             12971 1244184   95.92
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What is wrong with the following?

Problem 1: The analyses ignore grouping/assume independence.

Problem 2: Variables are not balanced across predictors:

```
table(nels$urbanicity,nels$enroll)

##
##               0    1    2    3    4    5
##   rural     959  449  369  264  215   93
##   suburban  922 1046 1215 1054  991  886
##   urban     790  659  772  590  782  918
```

"Controlling" for covariates

```
anova(lm(mscore~as.factor(enroll) +
                as.factor(flp) +
                as.factor(public) +
                as.factor(urbanicity)  ,data=nels) )

## Analysis of Variance Table
##
## Response: mscore
##                          Df  Sum Sq Mean Sq F value    Pr(>F)
## as.factor(enroll)         5    8660    1732  20.054 < 2.2e-16 ***
## as.factor(flp)            2  111662   55831 646.433 < 2.2e-16 ***
## as.factor(public)         1    3455    3455  39.998 2.626e-10 ***
## as.factor(urbanicity)     2    3471    1735  20.093 1.937e-09 ***
## Residuals             12963 1119588      86
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

"Controlling" for covariates

```
anova(lm(mscore~as.factor(urbanicity) +
                as.factor(public) +
                as.factor(flp) +
                as.factor(enroll) ,data=nels) )

## Analysis of Variance Table
##
## Response: mscore
##                          Df  Sum Sq Mean Sq  F value    Pr(>F)
## as.factor(urbanicity)     2    2652    1326  15.3514 2.192e-07 ***
## as.factor(public)         1   61162   61162 708.1572 < 2.2e-16 ***
## as.factor(flp)            2   61253   30627 354.6062 < 2.2e-16 ***
## as.factor(enroll)         5    2181     436   5.0493 0.0001261 ***
## Residuals             12963 1119588      86
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model comparison

Often we are interested in evaluating the effects of a variable *after* accounting for effects of others.

```
### model fits
fit.add<-lm(mscore~as.factor(enroll) +
                as.factor(flp) +
                as.factor(public) +
                as.factor(urbanicity) ,data=nels)

fit.menroll<-lm(mscore~as.factor(flp) +
                as.factor(public) +
                as.factor(urbanicity) ,data=nels)
```

```
### evaluating enroll - not controlling for other effects
anova(fit.add)

## Analysis of Variance Table
##
## Response: mscore
##                          Df  Sum Sq Mean Sq F value    Pr(>F)
## as.factor(enroll)         5    8660    1732  20.054 < 2.2e-16 ***
## as.factor(flp)            2  111662   55831 646.433 < 2.2e-16 ***
## as.factor(public)         1    3455    3455  39.998 2.626e-10 ***
## as.factor(urbanicity)     2    3471    1735  20.093 1.937e-09 ***
## Residuals             12963 1119588      86
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model comparison

Often we are interested in evaluating the effects of a variable *after* accounting for effects of others.

```
### model fits
fit.add<-lm(mscore~as.factor(enroll) +
                as.factor(flp) +
                as.factor(public) +
                as.factor(urbanicity) ,data=nels)

fit.menroll<-lm(mscore~as.factor(flp) +
                as.factor(public) +
                as.factor(urbanicity) ,data=nels)
```

```
### evaluating enroll - controlling for other effects
anova(fit.menroll,fit.add)

## Analysis of Variance Table
##
## Model 1: mscore ~ as.factor(flp) + as.factor(public) + as.factor(urbanicity)
## Model 2: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(public) +
##     as.factor(urbanicity)
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1  12968 1121768
## 2  12963 1119588  5    2180.5 5.0493 0.0001261 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Type III sums of squares

To evaluate effects *after controlling for others*,

- put in the term of interest last, or
- use type III sums of squares tests.

```
library(car)
Anova(fit.add,type=3)

## Anova Table (Type III tests)
##
## Response: mscore
##                         Sum Sq    Df    F value    Pr(>F)
## (Intercept)            3206322     1 37123.9724 < 2.2e-16 ***
## as.factor(enroll)         2181     5     5.0493 0.0001261 ***
## as.factor(flp)           57424     2   332.4354 < 2.2e-16 ***
## as.factor(public)         5121     1    59.2872 1.461e-14 ***
## as.factor(urbanicity)     3471     2    20.0932 1.937e-09 ***
## Residuals              1119588 12963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model comparison

Alternatively, without the car package, you can use drop1:

```
drop1(fit.add,test="F")

## Single term deletions
##
## Model:
## mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(public) +
##     as.factor(urbanicity)
##                        Df Sum of Sq     RSS   AIC  F value    Pr(>F)
## <none>                              1119588 57857
## as.factor(enroll)      5      2181 1121768 57872   5.0493 0.0001261 ***
## as.factor(flp)         2     57424 1177012 58502 332.4354 < 2.2e-16 ***
## as.factor(public)      1      5121 1124708 57914  59.2872 1.461e-14 ***
## as.factor(urbanicity)  2      3471 1123059 57893  20.0932 1.937e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Questionable assumptions of macro F-tests

The ANOVA model above can be expressed as

$$y_{i,j} = \mu + a_{e(j)} + b_{f(j)} + c_{p(j)} + d_{u(j)} + \epsilon_{i,j}$$

$a_{e(j)} \in \{a_1, \ldots, a_5\}$, $e(j)$ is enrollment category of j

$b_{f(j)} \in \{b_1, b_2, b_3\}$, $f(j)$ is flp category of j

etc.

The previous tests all assumed $\{\epsilon_{i,j}\} \sim$ *iid* $N(0, \sigma^2)$, and specifically,

$$\text{Cov}\left[\begin{pmatrix}\epsilon_{1,j} \\ \vdots \\ \epsilon_{n,j}\end{pmatrix}\right] = \begin{pmatrix}\sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \cdots & \sigma^2\end{pmatrix}$$

Why, in general, might we question this assumption?

Why might responses within a school be more similar than across schools?

Attempted solution with fixed effects

To account for school heterogeneity, we could fit a school-specific intercept:

$$y_{i,j} = (\mu + a_j) + a_{e(j)} + b_{f(j)} + c_{p(j)} + d_{u(j)} + \epsilon_{i,j}$$

In the absence of macro effects, OLS/ANOVA was a reasonable approach:

$$y_{i,j} = \mu + a_j + \epsilon_{i,j}$$

- $\bar{y}_j$ provides an unbiased estimate of $\mu_j = \mu + a_j$
- F-test from ANOVA is a valid test of heterogeneity across groups.

Could we use OLS/ANOVA in the presence of macro effects?

Attempted solution with fixed effects

```
fit_ols<-lm(mscore~as.factor(school) +
                   as.factor(enroll) +
                   as.factor(flp)    +
                   as.factor(public) +
                   as.factor(urbanicity)  ,data=nels)
```

```
anova(fit_ols)

## Analysis of Variance Table
##
## Response: mscore
##                      Df Sum Sq Mean Sq F value    Pr(>F)
## as.factor(school)   683 342385  501.30  6.8118 < 2.2e-16 ***
## Residuals         12290 904450   73.59
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

School-specific fixed effects explain *all* heterogeneity in means across schools.

There is nothing left for the other factors to explain.

HLM solution

$$y_{i,j} = (\mu + a_j) + a_{e(j)} + b_{f(j)} + c_{p(j)} + d_{u(j)} + \epsilon_{i,j}$$
$$a_1, \ldots, a_m \sim \; iid \; N(0, \tau^2)$$

As we've discussed, the random intercept induces a covariance within schools, and the above model is *equivalent to*

$$y_{i,j} = \mu + a_{e(j)} + b_{f(j)} + c_{p(j)} + d_{u(j)} + \epsilon_{i,j}$$

where

$$\text{Cov}\left[\begin{pmatrix} \epsilon_{1,j} \\ \vdots \\ \epsilon_{n,j} \end{pmatrix}\right] = \begin{pmatrix} \sigma^2 + \tau^2 & \tau^2 & \cdots & \tau^2 \\ \tau^2 & \sigma^2 + \tau^2 & \cdots & \tau^2 \\ \vdots & & & \vdots \\ \tau^2 & \tau^2 & \cdots & \sigma^2 + \tau^2 \end{pmatrix}$$

$$\text{Cor}[y_{i,j}, y_{i,k}] = \frac{\tau^2}{\tau^2 + \sigma^2}$$

Across school heterogeneity

```
fit0<-lmer( mscore ~ 1 + (1|school),data=nels)

fit0

## Linear mixed model fit by REML ['lmerMod']
## Formula: mscore ~ 1 + (1 | school)
##    Data: nels
## REML criterion at convergence: 93914.62
## Random effects:
##  Groups   Name        Std.Dev.
##  school   (Intercept) 4.866
##  Residual             8.585
## Number of obs: 12974, groups:  school, 684
## Fixed Effects:
## (Intercept)
##       50.94

s2.hat<-sigma(fit0)^2
t2.hat<-as.numeric(VarCorr(fit0)$school)

s2.hat

## [1] 73.70822

t2.hat

## [1] 23.6768

### ICC
t2.hat/(t2.hat+s2.hat)

## [1] 0.2431257
```

Across school heterogeneity

```
fit1<-lmer( mscore ~ as.factor(enroll) + (1|school),data=nels)


s2.hat<-sigma(fit1)^2
t2.hat<-as.numeric(VarCorr(fit1)$school)

s2.hat

## [1] 73.71874

t2.hat

## [1] 23.3493

### ICC
t2.hat/(t2.hat+s2.hat)

## [1] 0.2405457
```

Across school heterogeneity

```
fit2<-lmer( mscore ~ as.factor(enroll) +as.factor(flp) + (1|school),data=nels)


s2.hat<-sigma(fit2)^2
t2.hat<-as.numeric(VarCorr(fit2)$school)

s2.hat

## [1] 73.76314

t2.hat

## [1] 13.73191

### ICC
t2.hat/(t2.hat+s2.hat)

## [1] 0.156945
```

Across school heterogeneity

```
fit3<-lmer( mscore ~ as.factor(enroll) +as.factor(flp) + as.factor(public) +
   (1|school),data=nels)


s2.hat<-sigma(fit3)^2
t2.hat<-as.numeric(VarCorr(fit3)$school)

s2.hat

## [1] 73.77206

t2.hat

## [1] 13.4839

### ICC
t2.hat/(t2.hat+s2.hat)

## [1] 0.1545327
```

Across school heterogeneity

```
fit4<-lmer( mscore ~ as.factor(enroll) +as.factor(flp) + as.factor(public) +
  as.factor(urbanicity)  + (1|school),data=nels)


s2.hat<-sigma(fit4)^2
t2.hat<-as.numeric(VarCorr(fit4)$school)

s2.hat

## [1] 73.77562

t2.hat

## [1] 13.20577

### ICC
t2.hat/(t2.hat+s2.hat)

## [1] 0.151823
```

Model selection and testing

Notice: As we add macro predictors,

- $\hat{\tau}^2$ decreases, $\hat{\sigma}^2$ remains roughly the same;
- the within-group correlation decreases.

Questions: For a given set of macro variables,

- Is there evidence of (strong) within class correlation?
  - If not, we can test for macro variables with ANOVA.
  - If so, how do we evaluate the effects of the macro variables?

Goals:

1. Develop tests of within-class correlation *in the presence of macro variables*
   equivalently, test of *excess across-school heterogeneity*
2. Develop tests of macro effects *in the presence of within-class correlation*
3. More generally, select appropriate model from among LMs and HLMs.

Testing for excess heterogeneity

Consier two competing models:

M_0: No excess heterogeneity

$$y_{i,j} = \boldsymbol{\beta}^T \mathbf{x}_{i,j} + \epsilon_{i,j}$$
$$\{\epsilon_{i,j}\} \sim \ iid \ N(0, \sigma^2)$$

M_1: Excess heterogeneity

$$y_{i,j} = \boldsymbol{\beta}^T \mathbf{x}_{i,j} + a_j + \epsilon_{i,j}$$
$$\{\epsilon_{i,j}\} \sim \ iid \ N(0, \sigma^2)$$
$$\{a_j\} \sim \ iid \ N(0, \tau^2)$$

Model comparisons via tests

Suppose you would like a model selection procedure such that

if model M_0 were true,

you have a 95% chance of saying it is true.

If this is what you want, then a *level .05 hypothesis test* is for you.

H_0**:** No excess heterogeneity - model M_0 is true.

H_1**:** Excess heterogeneity - model M_1 is true.

Objective: A level α test of H_0 versus H_1.

Likelihood ratio tests

A popular tool for comparing nested models is the *likelihood ratio test (LRT)*:

$$\text{Reject } H_0 \text{ if } \Lambda(\mathbf{y}) = \frac{p(\mathbf{y}|\hat{\theta}_1)}{p(\mathbf{y}|\hat{\theta}_0)} \text{ is large.}$$

- $p(\mathbf{y}|\hat{\theta}_1)$ is the maximized prob density of data under H_1
- $p(\mathbf{y}|\hat{\theta}_0)$ is the maximized prob density of data under H_0
- $\Lambda(\mathbf{y})$ is the likelihood ratio statistic.

For a variety of reasons, the LRT is often expressed as

$$\text{Reject } H_0 \text{ if } \lambda(\mathbf{y}) = 2 \times \left(\log p(\mathbf{y}|\hat{\theta}_1) - \log p(\mathbf{y}|\hat{\theta}_0) \right) \text{ is large.}$$

- $\log p(\mathbf{y}|\hat{\theta}_1)$ is the maximized log likelihood for M_1
- $p(\mathbf{y}|\hat{\theta}_0)$ is the maximized log likelhiood for M_0
- $\lambda(\mathbf{y})$ is the log-likelihood ratio statistic.

Example: NELS data

```
### model 0
fit0<-lm(mscore ~ as.factor(flp) , data=nels)
logLik(fit0)

## 'log Lik.' -47375.64 (df=4)

### model 1
fit1<-lmer(mscore ~ as.factor(flp) + (1|school), data=nels,REML=FALSE)
logLik(fit1)

## 'log Lik.' -46811.34 (df=5)

### log liklihood statistic
lrt.stat<- 2*( logLik(fit1) - logLik(fit0) )
lrt.stat

## 'log Lik.' 1128.586 (df=5)
```

The LRT statistic seems pretty big!

Example: NELS data

```
### model 0
fit0<-lm(mscore ~ as.factor(flp) +
                  as.factor(enroll) +
                  as.factor(public) +
                  as.factor(urbanicity) , data=nels)
logLik(fit0)

## 'log Lik.' -47326.85 (df=12)

### model 1
fit1<-lmer(mscore ~ as.factor(flp) +
                    as.factor(enroll) +
                    as.factor(public) +
                    as.factor(urbanicity) + (1|school)  , data=nels,REML=FALSE)
logLik(fit1)

## 'log Lik.' -46797.62 (df=13)

### log liklihood statistic
lrt.stat<- 2*( logLik(fit1) - logLik(fit0) )
lrt.stat

## 'log Lik.' 1058.465 (df=13)
```

Still pretty big!

Null distributions

How big is big? A level α test is one where we

$$\text{reject } H_0 \text{ if } \lambda(\mathbf{y}) = 2 \times \left(\log p(\mathbf{y}|\hat{\theta}_1) - \log p(\mathbf{y}|\hat{\theta}_0) \right) \text{ is bigger than } \lambda_\alpha$$

where λ_α is a *critical value*, determined by

- the distribution of $\lambda(\mathbf{y})$ under H_0,
- the desired type I error rate α.

Null distribution example: *t*-test

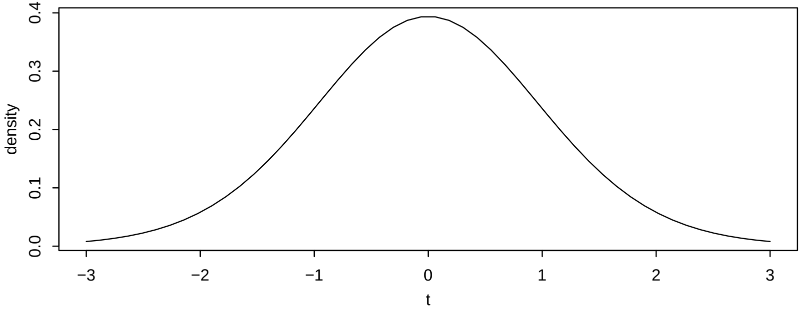
If

$$y_{1,A}, \ldots, y_{n_A,A} \sim \text{ } iid \text{ } N(\mu, \sigma^2)$$
$$y_{1,B}, \ldots, y_{n_B,B} \sim \text{ } iid \text{ } N(\mu, \sigma^2)$$

then the distribution of the t-statistic

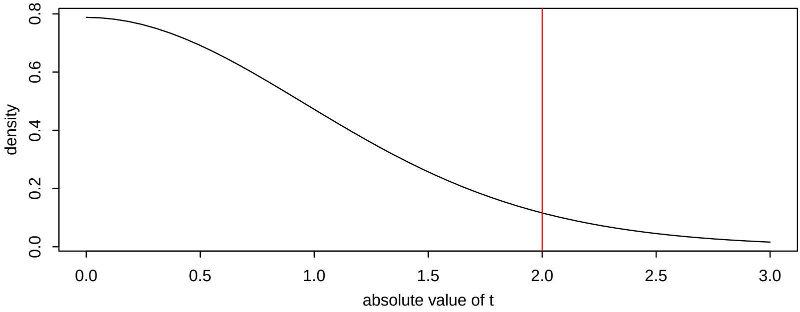
$$t(\mathbf{y}_A, \mathbf{y}_B) = \frac{\bar{y}_B - \bar{y}_A}{s_p \sqrt{1/n_A + 1/n_B}}$$

has a *t*-distribution.



Null distribution example: t-test

A typical t-test rejects if $|t(\mathbf{y}_A, \mathbf{y}_B)| > 2$.



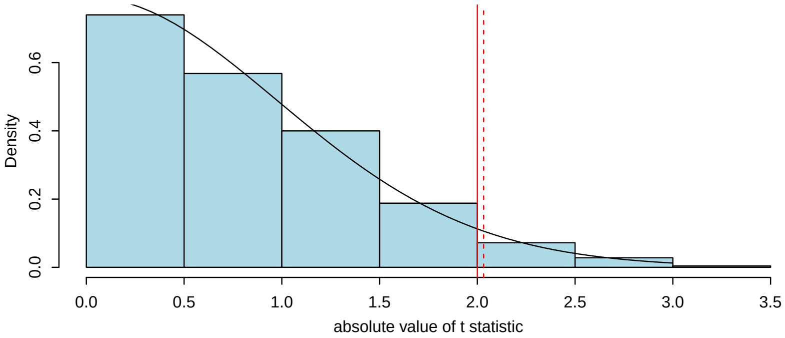
$$\Pr(|t(\mathbf{y}_A, \mathbf{y}_B)| > 2) \approx 0.05$$

- 2 is the critical value of the test;
- 0.05 is the (approximate) level of the test.

Null distribution example: *t*-test empirical validation

```
n<-20 ; ATSTAT<-NULL

for(i in 1:S)
{
  yA<-rnorm(n)
  yB<-rnorm(n)
  ATSTAT<-c(ATSTAT, abs(t.test(yA,yB,pooled=TRUE)$stat))
}
```

Null distribution example: t-test empirical validation




```
quantile(ATSTAT,probs=.95)

##      95%
## 2.032179

qt(.975,2*(n-1))

## [1] 2.024394
```

Null distribution for LRT

LRT:

$$\text{Reject } H_0 \text{ if } \lambda(\mathbf{y}) = 2 \times \left(\log p(\mathbf{y}|\hat{\theta}_1) - \log p(\mathbf{y}|\hat{\theta}_0)\right) \text{ is greater than } c,$$

where c is the value such that

$$\Pr(\lambda(\mathbf{y}) > c | H_0) = 0.05.$$

To figure out what c is, we need the distribution of $\lambda(\mathbf{y})$ when H_0 is true. That is, we need to know the *null distribution*.

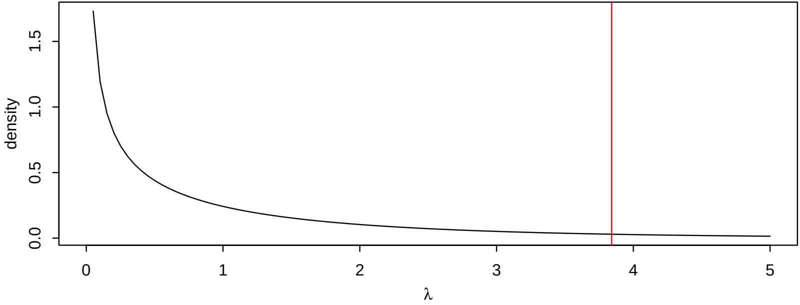
Null distribution for LRT

Statistical folklore says the following: If

- M_0 is *nested* in M_1 (M_0 is a special case of M_1), and
- M_0 is true, then

$$\lambda(\mathbf{y}) \dot{\sim} \chi^2_d$$

where d is the difference in the number of parameters between M_1 and M_0.




```
qchisq(.95,1)

## [1] 3.841459
```

Null distribution for LRT: Fixed effects

M_0**:** No fixed effect of $x_{i,j}$

$$y_{i,j} = \beta_0 + a_j + \epsilon_{i,j}$$
$$a_j \sim N(0, \tau^2)$$

M_1**:** Yes fixed effect of $x_{i,j}$

$$y_{i,j} = \beta_0 + \beta_1 x_{i,j} + a_j + \epsilon_{i,j}$$
$$a_j \sim N(0, \tau^2)$$

Distribution of LRT: The change in the number of parameters is $d = 1$.

Presumably,

$$\lambda(\mathbf{y}) \dot{\sim} \chi^2_1$$

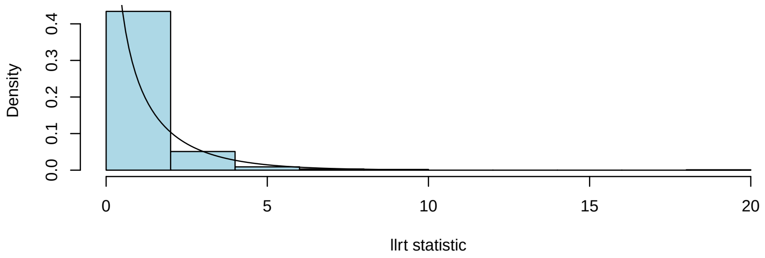
The $\dot{\sim}$ means “approximately distributed as.”

The approximation improves as sample size increases.

Null distribution for LRT: Empirical evaluation

```
m<-20 ; n<-10
beta0<-1 ; beta1<-0

g<-rep(1:m,times=rep(n,m))

LAMBDA.H0<-NULL
for(s in 1:S)
{
  a<-rnorm(m)
  x<-rnorm(m*n)

  y<-a[g] + beta0 + beta1*x + rnorm(m*n)

  fit0<-lmer(y ~ 1 + (1|g), REML=FALSE )
  fit1<-lmer(y ~ x + (1|g), REML=FALSE )

  lambda<-2*( logLik(fit1) - logLik(fit0) )

  LAMBDA.H0<-c(LAMBDA.H0,lambda)
}
```

Null distribution for LRT: Empirical evaluation




```
quantile(LAMBDA.H0,.95)

##      95%
## 3.258501

qchisq(.95,1)

## [1] 3.841459
```

LRT for HLM

M_0:

$$\mathbf{y}_j = \mathbf{X}_j\boldsymbol{\beta} + \boldsymbol{\epsilon}_j \ , \quad \text{Cov}\left[\begin{pmatrix}\epsilon_{1,j}\\ \vdots \\ \epsilon_{n,j}\end{pmatrix}\right] = \begin{pmatrix}\sigma^2 & 0 & \cdots & 0\\ 0 & \sigma^2 & \cdots & 0\\ \vdots & & & \vdots\\ 0 & 0 & \cdots & \sigma^2\end{pmatrix}$$

M_1:

$$\mathbf{y}_j = \mathbf{X}_j\boldsymbol{\beta} + \epsilon_j \ , \quad \text{Cov}\left[\begin{pmatrix}\epsilon_{1,j}\\ \vdots \\ \epsilon_{n,j}\end{pmatrix}\right] = \begin{pmatrix}\sigma^2+\tau^2 & \tau^2 & \cdots & \tau^2\\ \tau^2 & \sigma^2+\tau^2 & \cdots & \tau^2\\ \vdots & & & \vdots\\ \tau^2 & \tau^2 & \cdots & \sigma^2+\tau^2\end{pmatrix}$$

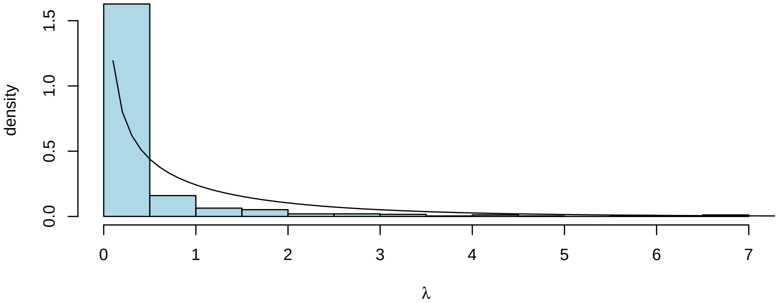
Q: What is the difference in the number of parameters?

A: $d = 1$

Simulation study

```
m<-20 ; n<-10
beta0<-1 ; beta1<-1

g<-rep(1:m,times=rep(n,m))

LAMBDA.H0<-NULL
for(s in 1:S)
{
  x<-rnorm(m*n)

  y<-beta0 + beta1*x + rnorm(m*n)

  fit0<-lm(y ~ x )

  fit1<-lmer(y ~ x + (1|g), REML=FALSE)

  lambda<-2*( logLik(fit1) - logLik(fit0) )

  LAMBDA.H0<-c(LAMBDA.H0,lambda)
}
```

Simulation study




```
mean( LAMBDA.H0>= qchisq(.95,1) )

## [1] 0.02
```

Simulation study

```
zapsmall(LAMBDA.H0[1:20])

##  [1] 0.000000 0.891508 0.497324 0.177651 0.000000 0.417878 0.000000 0.000000
##  [9] 0.000138 0.040075 0.000000 4.920390 0.000000 0.000000 0.387080 0.000000
## [17] 0.000000 0.000000 0.281322 0.052502
```

```
mean( zapsmall(LAMBDA.H0[1:20]) == 0 )

## [1] 0.5
```

Mixture null distributions

What is going on? Suppose we are fitting M_1 in the simple HNM:

$$y_{i,j} = \mu + a_j + \epsilon_{i,j}$$
$$a_j \sim N(0, \tau^2)$$

Recall,

$$\mathrm{E}[MSW] = \sigma^2$$
$$\mathrm{E}[MSA] = \sigma^2 + n \times \tau^2$$
$$\hat{\tau}^2 = (MSA - MSW)/n$$

Mixture null distributions

If M_0 is in fact true, then $\tau^2 = 0$ and

$$\begin{aligned} \mathrm{E}[MSW] &= \sigma^2 \\ \mathrm{E}[MSA] &= \sigma^2. \end{aligned}$$

If we are fitting M_1, then sometimes (due to sampling variability)

$$MSW > MSA$$

$$(MSA - MSW)/n < 0 \;\Rightarrow\; \text{ use } \hat{\tau}^2 = 0 \text{ in practice.}$$

In these cases (roughly speaking),

- the MLE $\hat{\tau}^2$ is zero.
- the best M_0 fit is the same as the best M_1 fit.

$$\max_{\mu,\sigma^2,\tau^2} \log p(\mathbf{y}|\mu,\sigma^2,\tau^2) = \max_{\mu,\sigma^2} \log p(\mathbf{y}|\mu,\sigma^2,\tau^2 = 0)$$

Example dataset

```
set.seed(2)
y<-1 + rnorm(m*n)

anova(lm(y~as.factor(g)) )

## Analysis of Variance Table
##
## Response: y
##                Df  Sum Sq Mean Sq F value Pr(>F)
## as.factor(g)   19  14.745 0.77606  0.6503 0.8629
## Residuals     180 214.812 1.19340

MSW<-anova(lm(y~as.factor(g)) )[2,3]
MSA<-anova(lm(y~as.factor(g)) )[1,3]

MSW

## [1] 1.193401

MSA

## [1] 0.7760613

MSA-MSW

## [1] -0.4173393
```

Example dataset

```
fit0<-lm(y ~ 1 )
fit1<-lmer(y ~ 1 + (1|g), REML=FALSE)
```

```
fit0

##
## Call:
## lm(formula = y ~ 1)
##
## Coefficients:
## (Intercept)
##      0.9993
```

```
fit1

## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: y ~ 1 + (1 | g)
##       AIC       BIC    logLik  deviance  df.resid
##  601.1424  611.0374 -297.5712  595.1424       197
## Random effects:
##  Groups   Name        Std.Dev.
##  g        (Intercept) 0.000
##  Residual             1.071
## Number of obs: 200, groups:  g, 20
## Fixed Effects:
## (Intercept)
##      0.9993
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
```

```
2*( logLik(fit1) - logLik(fit0) )

## 'log Lik.' -2.273737e-13 (df=3)
```

The (asymptotic) null distribution

It turns out that *under M_0*,

$$\Pr(\lambda(\mathbf{y}) = 0) = \frac{1}{2}$$

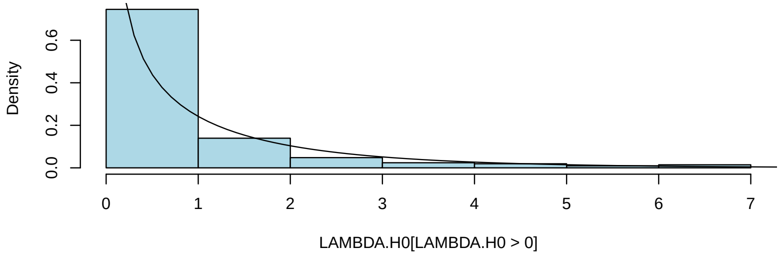
The values that are *not* equal to zero are distributed as χ^2_1:

$$\lambda(\mathbf{y}) | \{\lambda(\mathbf{y}) \neq 0\} \dot{\sim} \chi^2_1$$

This means that under M_0, $\lambda(\mathbf{y})$ has a *mixture distribution*

The empirical null distribution

```
LAMBDA.H0<-zapsmall(LAMBDA.H0)
mean(LAMBDA.H0==0)

## [1] 0.584

hist(LAMBDA.H0[LAMBDA.H0>0],col="lightblue",prob=TRUE,main="")
lines(xs,dchisq(xs,1),type="l")
```



Mixture distributions

We can represent the distribution of $\lambda(\mathbf{y})$ as follows:

$$\lambda(\mathbf{y}) = \begin{cases} X_0 & \text{with probability } 1/2 \\ X_1 & \text{with probability } 1/2 \end{cases}$$

where

- $X_0 = 0$
- X_1 has a χ^2_1 distribution.

Computing a p-value

Recall, a *p-value* is the probability under the null of getting a test statistic equal to or larger than the observed test statistic.

For a given observed value λ_{obs},

$$p - \text{value} = \Pr(\lambda(\mathbf{y}) \geq \lambda_{obs} | H_0)$$

How do we compute this for a given value λ_{obs}?

Computing a p-value

Case 1: $\lambda_{obs} = 0$.

$$\Pr(\lambda(\mathbf{y}) \geq 0) = 1$$

as X_0 and X_1 are ≥ 0.

Case 2: $\lambda_{obs} > 0$.

$$\begin{aligned}\Pr(\lambda(\mathbf{y}) \geq \lambda_{obs}) &= \Pr(\lambda(\mathbf{y}) = X_0 \text{ and } X_0 \geq \lambda_{obs}) + \Pr(\lambda(\mathbf{y}) = X_1 \text{ and } X_1 \geq \lambda_{obs}) \\ &= \tfrac{1}{2}0 + \tfrac{1}{2}\Pr(X_1 \geq \lambda_{obs}) \\ &= \tfrac{1}{2}\Pr(\chi^2_1 \geq \lambda_{obs}),\end{aligned}$$

which is $1/2$ the p-value that would be obtained using the χ^2_1 null distribution.

Folklore: "The p-value for testing … the random intercept variance is half this $[\chi^2_1]$ tail value."

(true if $\lambda_{obs} \neq 0$).

Example: NELS

Recall one of our original questions:
Can the heterogeneity across schools be ascribed to known macro covariates?

Model fits:

```
fit0<-lm(mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
                  ses + hwh, data=nels)

fit1<-lmer(mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
                    ses + hwh + (1|school) , data=nels,REML=FALSE)
```

Hypothesis test:

```
### LRT statistic
lambda<-2*(logLik(fit1)-logLik(fit0))

lambda

## 'log Lik.' 696.8672 (df=14)

### p-value
.5*(1-pchisq(c(lambda),1) )

## [1] 0
```

- `pchisq(lambda,1)` is the probability of being smaller than `lambda`
- `1-pchisq(lambda,1)` is the probability of being larger than `lambda`

The null hypothesis of no excess heterogeneity is strongly rejected.

Summary of testing

$$y_{i,j} = \boldsymbol{\beta}^T x_{i,j} + a_j + \epsilon_{i,j}$$
$$a_j \sim N(0, \tau^2)$$

For models consisting of

- fixed effects, and
- a single random intercept,

Tests involving β : Testing components of $\boldsymbol{\beta}$ equal zero can be obtained with the usual *LRT*.

- Null distribution: $\lambda_0 \sim \chi^2_d$,
- *p*-value: `1-pchisq(lambda,d)`.

Tests involving τ^2 : Testing $\tau^2 = 0$ can be obtained with the modified *LRT*.

- Null distribution: $\lambda_0 \sim \frac{1}{2}(\{0\} + \chi^2_1)$,
- *p*-value: `.5*(1-pchisq(lambda,1))` if `lambda > 0`, 1 if `lambda =0`.

Testing examples

```
fit.full<-lmer(mscore~
   as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
   hwh + ses +
  (1|school) , data=nels,REML=FALSE)

fit.full

## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
##     hwh + ses + (1 | school)
##    Data: nels
##       AIC       BIC    logLik  deviance  df.resid
##  92408.36  92512.95 -46190.18  92380.36     12960
## Random effects:
##  Groups   Name        Std.Dev.
##  school   (Intercept) 2.969
##  Residual             8.243
## Number of obs: 12974, groups:  school, 684
## Fixed Effects:
##                   (Intercept)             as.factor(enroll)1
##                      52.82676                        0.54442
##            as.factor(enroll)2             as.factor(enroll)3
##                       0.61973                        0.61739
##            as.factor(enroll)4             as.factor(enroll)5
##                       0.52867                        0.16135
##               as.factor(flp)2                as.factor(flp)3
##                      -2.09257                       -4.84231
## as.factor(urbanicity)suburban     as.factor(urbanicity)urban
##                      -0.05113                       -0.86587
##                           hwh                            ses
##                       0.01354                        4.13467
```

Testing examples

```
fit.menr<-lmer(mscore~
   as.factor(flp) + as.factor(urbanicity) +
   hwh + ses +
  (1|school) , data=nels,REML=FALSE)
```

```
fit.mflp<-lmer(mscore~
   as.factor(enroll) + as.factor(urbanicity) +
   hwh + ses +
  (1|school) , data=nels,REML=FALSE)
```

```
fit.murb<-lmer(mscore~
   as.factor(enroll) + as.factor(flp) +
   hwh + ses +
  (1|school) , data=nels,REML=FALSE)
```

Testing examples

Compute the LRT statistic:

```
lambda<-2*(logLik(fit.full) - logLik(fit.menr))

lambda

## 'log Lik.' 3.204099 (df=14)
```

Calculate *d*:

```
table(nels$enroll)

##
##    0    1    2    3    4    5
## 2671 2154 2356 1908 1988 1897
```

```
attr( logLik(fit.full),"df")

## [1] 14

attr( logLik(fit.menr),"df")

## [1] 9

d<- attr( logLik(fit.full),"df") - attr( logLik(fit.menr),"df")

d

## [1] 5
```

Testing examples

Compute the *p*-value:

```
(1-pchisq(c(lambda),d))

## [1] 0.668553
```

This is mostly automated in R:

```
anova(fit.full,fit.menr)

## Data: nels
## Models:
## fit.menr: mscore ~ as.factor(flp) + as.factor(urbanicity) + hwh + ses + (1 | school)
## fit.full: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) + hwh + se
##          npar   AIC   BIC logLik deviance  Chisq Df Pr(>Chisq)
## fit.menr    9 92402 92469 -46192    92384
## fit.full   14 92408 92513 -46190    92380 3.2041  5     0.6686
```

Testing other factors

```
anova(fit.full,fit.mflp)

## Data: nels
## Models:
## fit.mflp: mscore ~ as.factor(enroll) + as.factor(urbanicity) + hwh + ses + (1 | school)
## fit.full: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) + hwh + se
##          npar   AIC   BIC logLik deviance  Chisq Df Pr(>Chisq)
## fit.mflp   12 92564 92654 -46270    92540
## fit.full   14 92408 92513 -46190    92380 159.58  2  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit.full,fit.murb)

## Data: nels
## Models:
## fit.murb: mscore ~ as.factor(enroll) + as.factor(flp) + hwh + ses + (1 | school)
## fit.full: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) + hwh + se
##          npar   AIC   BIC logLik deviance  Chisq Df Pr(>Chisq)
## fit.murb   12 92412 92502 -46194    92388
## fit.full   14 92408 92513 -46190    92380 7.7808  2    0.02044 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Testing examples

```
fit.mhwh<-lmer(mscore~
   as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
   ses +
  (1|school) , data=nels,REML=FALSE)
```

```
fit.mses<-lmer(mscore~
   as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
   hwh +
  (1|school) , data=nels,REML=FALSE)
```

Testing examples

```
anova(fit.full,fit.mhwh)

## Data: nels
## Models:
## fit.mhwh: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) + ses + (
## fit.full: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) + hwh + se
##          npar   AIC   BIC logLik deviance  Chisq Df Pr(>Chisq)
## fit.mhwh   13 92407 92504 -46190    92381
## fit.full   14 92408 92513 -46190    92380 0.3107  1     0.5772
```

```
anova(fit.full,fit.mses)

## Data: nels
## Models:
## fit.mses: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) + hwh + (
## fit.full: mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) + hwh + se
##          npar   AIC   BIC logLik deviance Chisq Df Pr(>Chisq)
## fit.mses   13 93634 93731 -46804    93608
## fit.full   14 92408 92513 -46190    92380  1228  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Testing examples

```
summary(fit.full)$coef

##                                      Estimate Std. Error     t value
## (Intercept)                       52.82676162  0.4309192 122.5908794
## as.factor(enroll)1                 0.54442472  0.4569472   1.1914390
## as.factor(enroll)2                 0.61973124  0.4541606   1.3645642
## as.factor(enroll)3                 0.61738849  0.4828518   1.2786293
## as.factor(enroll)4                 0.52866612  0.4891502   1.0807849
## as.factor(enroll)5                 0.16135353  0.4932025   0.3271547
## as.factor(flp)2                   -2.09257387  0.3497278  -5.9834361
## as.factor(flp)3                   -4.84231161  0.3677904 -13.1659532
## as.factor(urbanicity)suburban -0.05113111  0.3932499  -0.1300219
## as.factor(urbanicity)urban    -0.86587407  0.4204572  -2.0593634
## hwh                                0.01353902  0.0242850   0.5575056
## ses                                4.13466985  0.1142795  36.1803310

2*(1-pnorm(.5575))

## [1] 0.5771859

2*(1-pnorm(36.1803))

## [1] 0
```

Testing examples

Now that you know where the numbers come from,

```
drop1(fit.full,test="Chisq")

## Single term deletions
##
## Model:
## mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
##     hwh + ses + (1 | school)
##                        npar   AIC     LRT Pr(Chi)
## <none>                      92408
## as.factor(enroll)         5 92402    3.20 0.66855
## as.factor(flp)            2 92564  159.58 < 2e-16 ***
## as.factor(urbanicity)     2 92412    7.78 0.02044 *
## hwh                       1 92407    0.31 0.57725
## ses                       1 93634 1228.01 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Summary of tests so far

$$y_{i,j} = \boldsymbol{\beta}^T x_{i,j} + a_j + \epsilon_{i,j}$$
$$a_j \sim N(0, \tau^2)$$

Fixed effects:

- enrollment : no strong evidence of effect
- flp : decreasing scores with increasing flp
- urban : urban schools have lower scores than others
- hwh : no strong evidence of an effect *on average across schools*
- ses : strong evidence of a positive effect *on average across schools*

Random effects: Strong evidence of excess across-school heterogeneity in mean score.

ANOVA comparison

Compare to tests that don't account for across-group heterogeneity:

```
### model fit
fit.afull<-lm(mscore~
   as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
   hwh + ses,
   data=nels )


### factor evaluation
drop1(fit.afull,test="F")

## Single term deletions
##
## Model:
## mscore ~ as.factor(enroll) + as.factor(flp) + as.factor(urbanicity) +
##     hwh + ses
##                       Df Sum of Sq     RSS   AIC   F value    Pr(>F)
## <none>                              991486 56283
## as.factor(enroll)      5       377  991863 56278    0.9863    0.4243
## as.factor(flp)         2     28135 1019621 56642  183.9096 < 2.2e-16 ***
## as.factor(urbanicity)  2      1516  993002 56298    9.9107 5.002e-05 ***
## hwh                    1       167  991653 56283    2.1819    0.1397
## ses                    1    132644 1124130 57910 1734.0918 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Testing for heterogeneous slopes

General two-level HLM:

$$y_{i,j} = \boldsymbol{\beta}^T x_{i,j} + \boldsymbol{a}_j^T z_{i,j} + \epsilon_{i,j}$$
$$\boldsymbol{a}_j \sim \; N(0, \Psi)$$

For example, maybe

$$\begin{pmatrix} z_{i,j,1} \\ z_{i,j,2} \end{pmatrix} = \begin{pmatrix} 1 \\ ses_{i,j} \end{pmatrix} \quad \Psi = \begin{pmatrix} \psi_1^2 & \psi_{1,2} \\ \psi_{2,1} & \psi_2^2 \end{pmatrix}$$

We would like to be able to test

$$H_0 : \psi_2^2 = 0 \;\; \text{(no heterogeneity in slope with } ses\text{)},$$

in the presence of heterogeneity in intercept.

Testing for heterogeneous slopes

$$H_0 : \psi_2^2 = 0 \ \ \text{(no heterogeneity in slope with } \textit{ses}\text{)}$$

If the variance of something is zero, its covariance with anything else is zero.

This means that under $H_0 : \psi_2^2 = 0$,

$$\mathbf{\Psi} = \begin{pmatrix} \psi_1^2 \end{pmatrix}$$

while under $H_1 : \psi_2^2 \neq 0$,

$$\mathbf{\Psi} = \begin{pmatrix} \psi_1^2 & \psi_{1,2} \\ \psi_{2,1} & \psi_2^2 \end{pmatrix}$$

The difference in the number of parameters is $d = 2$.

NELS data

```
fit.r1<-lmer(
   mscore~
        as.factor(flp) + as.factor(urbanicity) +
        ses +
       (ses | school) , data=nels,REML=FALSE)
```

```
summary(fit.r1)$coef

##                                  Estimate Std. Error    t value
## (Intercept)                   53.13668593  0.3943076 134.759485
## as.factor(flp)2               -2.02135574  0.3342738  -6.047006
## as.factor(flp)3               -4.81780351  0.3612673 -13.335840
## as.factor(urbanicity)suburban  0.05675027  0.3803280   0.149214
## as.factor(urbanicity)urban    -0.80937534  0.4049585  -1.998663
## ses                            4.12877819  0.1255087  32.896343
```

```
VarCorr(fit.r1)

##  Groups   Name        Std.Dev. Corr
##  school   (Intercept) 2.9673
##           ses         1.2712   -0.005
##  Residual             8.2008
```

NELS data

```
fit.r0<-lmer(
   mscore~
        as.factor(flp) + as.factor(urbanicity) +
        ses +
       (1 | school) , data=nels,REML=FALSE)
```

```
summary(fit.r0)$coef

##                                    Estimate Std. Error     t value
## (Intercept)                     53.12042202  0.3928410 135.2211600
## as.factor(flp)2                 -2.00043931  0.3324308  -6.0176108
## as.factor(flp)3                 -4.77163280  0.3596303 -13.2681609
## as.factor(urbanicity)suburban    0.06620705  0.3792811   0.1745593
## as.factor(urbanicity)urban      -0.78129077  0.4032054  -1.9376990
## ses                              4.13800015  0.1141748  36.2426730
```

```
VarCorr(fit.r0)

##  Groups   Name        Std.Dev.
##  school   (Intercept) 2.9760
##  Residual             8.2437
```

NELS data

```
logLik(fit.r1)

## 'log Lik.' -46185.14 (df=10)
```

```
logLik(fit.r0)

## 'log Lik.' -46191.93 (df=8)
```

```
lambda<-2*c( logLik(fit.r1) - logLik(fit.r0) )

lambda

## [1] 13.58696
```

What do we compare `lambda` to?

What types of values would we expect under H_0?

Null distribution

Speculation 1: Maybe under H_0, $\lambda \sim \frac{1}{2}(\{0\} + \chi^2_1)$.

Speculation 2: Maybe under H_0, $\lambda \sim \chi^2_2$, as $d = 2$.

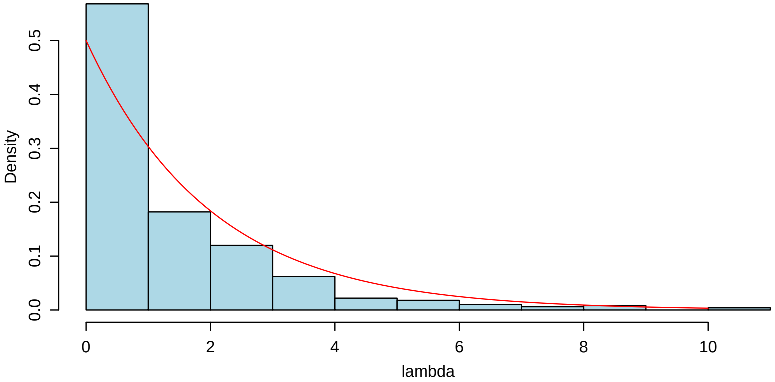
Let's investigate with a simulation study

Null distribution

```
m<-30 ; n<-10
beta0<-1 ; beta1<-1
g<-rep(1:m,times=rep(n,m))

LAMBDA.H0<-NULL
for(s in 1:S)
{
  a<-rnorm(m) # random effects

  x<-rnorm(m*n) # covariates

  y<-beta0 + a[g] + beta1*x + rnorm(m*n)  #simulated under null

  fit0<-lmer(y ~ x + (1|g), REML=FALSE )

  fit1<-lmer(y ~ x + (x|g), REML=FALSE)

  lambda<-2*( logLik(fit1) - logLik(fit0) )

  LAMBDA.H0<-c(LAMBDA.H0,lambda)
}

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
Model failed to converge with max|grad| = 0.00448623 (tol = 0.002, component 1)
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
Model failed to converge with max|grad| = 0.00257302 (tol = 0.002, component 1)
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
Model failed to converge with max|grad| = 0.00344872 (tol = 0.002, component 1)
```

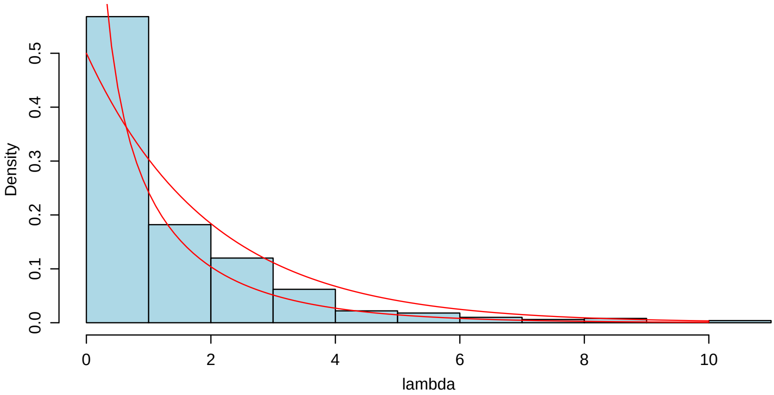
Null distribution

Compare to a χ^2_2 distribution:



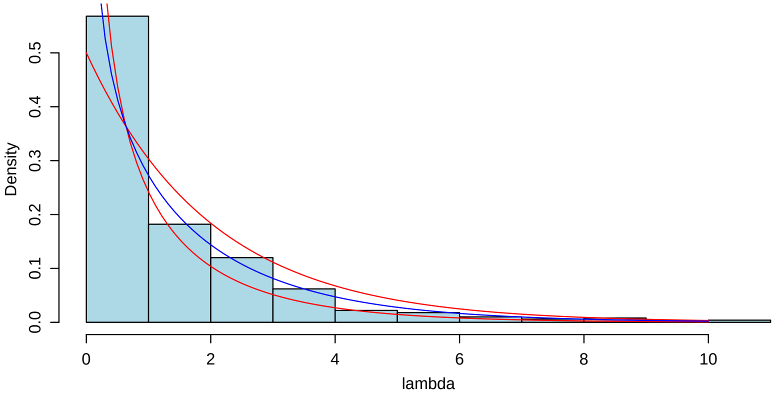
Null distribution

Compare to a χ^2_1 distribution:



Null distribution

Here is the theoretical, asymptotic null distribution: $\lambda \sim \frac{1}{2}(\chi_1^2 + \chi_2^2)$



Mixture distributions

We can represent the distribution of $\lambda(\mathbf{y})$ as follows:

$$\lambda(\mathbf{y}) = \begin{cases} X_1 & \text{with probabilty } 1/2 \\ X_2 & \text{with probabilty } 1/2 \end{cases}$$

where

- X_1 has a χ^2_1 distribution;
- X_2 has a χ^2_2 distribution.

Computing the p-value

$$
\begin{aligned}
\Pr(\lambda(\mathbf{y}) \geq \lambda_{obs}) &= \Pr(\lambda(\mathbf{y}) = X_1 \text{ and } X_1 \geq \lambda_{obs}) + \Pr(\lambda(\mathbf{y}) = X_2 \text{ and } X_2 \geq \lambda_{obs}) \\
&= \tfrac{1}{2}\Pr(X_1 \geq \lambda_{obs}) + \tfrac{1}{2}\Pr(X_1 \geq \lambda_{obs}) \\
&= \tfrac{1}{2}\left(\Pr(\chi^2_1 \geq \lambda_{obs}) + \Pr(\chi^2_2 \geq \lambda_{obs})\right)
\end{aligned}
$$

which is a 50-50 average between the naive *p*-value (based on a χ^2_2 distribution), and one based on a reduced degrees of freedom.

***p*-value**: The *p*-value can be obtained with `pchisq` as before:

- $\Pr(\chi^2_1 \geq \lambda) =$ `1-pchisq(lambda,1)`
- $\Pr(\chi^2_2 \geq \lambda) =$ `1-pchisq(lambda,2)`

The general result

$$y_{i,j} = \boldsymbol{\beta}^T \mathbf{x}_{i,j} + \mathbf{a}_j^T \mathbf{z}_{i,j} + \epsilon_{i,j}$$

If $\mathbf{a}_j \in \mathbb{R}^p$, then

$$\text{Cov}[\mathbf{a}_j] = \Psi = \begin{pmatrix} \psi_1^2 & \psi_{12} & \cdots & \psi_{1p} \\ \psi_{21} & \psi_2^2 & \cdots & \psi_{2p} \\ \vdots & & & \vdots \\ \psi_{p1} & \psi_{p2} & \cdots & \psi_p^2 \end{pmatrix}$$

Consider testing to compare the following models:

- M_1**:** Full model
- M_1**:** Reduced model with $\psi_p^2 = 0$ (and $\psi_{pk} = 0$ also)

Question: What is the change in number of parameters?

Answer: $d = p$

The null distribution in the general case

M_1 p random effects coefficients

M_0 $p-1$ random effects coefficients

Null distribution: Under M_0, the LRT statistic has is distributed as

$$\lambda(\mathbf{y}) = \begin{cases} X_{p-1} & \text{with probabilty } 1/2 \\ X_p & \text{with probabilty } 1/2 \end{cases}$$

where

- X_{p-1} has a χ^2_{p-1} distribution;
- X_p has a χ^2_p distribution.

The null distribution in the general case

Shorthand for this is

$$\lambda | M_0 \sim \frac{1}{2}(\chi^2_{p-1} + \chi^2_p).$$

- This *does not* mean that λ is the average of two χ^2 random variables,
- this *does* mean that the *density* of λ is the average of two χ^2 *densities*.

CAREFUL: Some authors say $\lambda | M_0 \sim \frac{1}{2}(\chi^2_p + \chi^2_{p+1})$.
This is because they are not counting the intercept.

Check with previous results:

Single random effect:

$$M_0 \ : \ y_{i,j} = \boldsymbol{\beta}^T \mathbf{x}_{i,j} + \epsilon_{i,j}$$
$$M_1 \ : \ y_{i,j} = \boldsymbol{\beta}^T \mathbf{x}_{i,j} + b_{1,j} + \epsilon_{i,j}$$
$$\lambda | M_0 \sim \tfrac{1}{2}(\{0\} + \chi^2_1)$$

Two random effects:

$$M_0 \ : \ y_{i,j} = \boldsymbol{\beta}^T \mathbf{x}_{i,j} + b_{1,j}\epsilon_{i,j}$$
$$M_1 \ : \ y_{i,j} = \boldsymbol{\beta}^T \mathbf{x}_{i,j} + b_{1,j} + b_{2,j} w_{i,j} + \epsilon_{i,j}$$
$$\lambda | M_0 \sim \tfrac{1}{2}(\chi^2_1 + \chi^2_2)$$

Effects on p-values and critical values

Naive critical value:

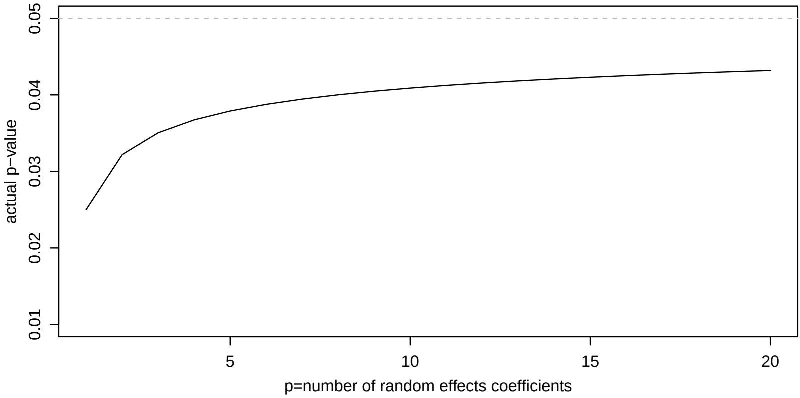
- p random effects implies $d = p$.
- The naive 0.05 critical value is λ_c=qchisq(.95,p)

Actual p-value: Suppose you observed a test statistic equal to λ_c:

- Your "naive" p-value is 0.05.
- Your actual p-value is lower.
- The naive p-value is more likely to erroneously accept the null (simpler) model.

Effects on *p*-values and critical values

```
p<-1:20
lc.naive<-qchisq(.95,p)
pval<-.5*( (1-pchisq(lc.naive,p-1)) + (1-pchisq(lc.naive,p)) )
```



Summary of testing

LRT: The LRT can be used to compare nested models:

- models with and without various fixed effects;
- models with and without various random effects.

LRT: The LRT statistic can be compared to a null distribution:

- χ^2_d for testing if d fixed effects are zero.
- $\frac{1}{2}(\chi^2_{p-1} + \chi^2_p)$ for testing if a single random effect is zero, in the presence of $p-1$ other random effects.

Cautions

Consequences of ignoring the mixture null distribution:

- The naive *p*-value will be larger than the actual *p*-value.
- The naive *p*-value will underrepresent evidence against the null.
- From a decision-theory perspective, the naive *p*-value still controls type I error below α;
- From a practical perspective, the naive *p*-value can lead to using the wrong (simpler) model more often.

Caution: null distributions and *p*-values are based on *asymptotic* results.

If you are concerned about the validity for your sample size, then simulate!