

Appendix III

```
**Stata code for fitting the various models described in the manuscript
**The commands can be run interactively or in batch mode (i.e. in a do-file)
**The mvmeta and gllamm packages are required in order to run the code
**Fixed effects models are not considered here, for simplicity
**Data input for the discrete outcome meta-analysis
**Data are taken from the study of Kato et al, 1999
**Data are listed in the form of Table 1 of the manuscript

clear
set more off
input study aa0 ab0 bb0 aa1 ab1 bb1
1 3 34 44 2 20 83
2 4 30 49 3 17 62
3 17 84 48 3 30 31
4 5 32 63 5 23 52
5 12 62 119 8 39 133
6 9 48 47 6 34 68
7 18 134 363 20 214 483
end

** calculation of the log(Odds Ratios)
gen b1=log(( ab1/ aa1)/( ab0/aa0))
gen b2=log(( bb1/ aa1)/( bb0/aa0))

** calculation of the variances
gen V11=1/aa0 +1/ab0+1/aa1 +1/ab1
gen V22=1/aa0 +1/aa1+1/bb0 +1/bb1

** calculation of the covariance
gen V12=1/aa0 +1/aa1

** fitting the model of Equation 3.7 using REML
mvmeta b V,vars(b1 b2)

** retrieve the covariance matrix
matrix v=e(V)

**calculation of lambda
di _b[b1]/_b[b2]

**calculation of standard error of lambda - Equation 5.6
di sqrt( v[1,1]/_b[b2]^2 + v[2,2]*_b[b1]^2/_b[b2]^4 -2*v[1,2]*_b[b1]/_b[b2]^3 )

**automatic calculation of confidence intervals for lambda
nlcom _b[b1]/_b[b2]

**calculation of d
di _b[b2] - _b[b1]

**calculation of standard error of d - Equation 5.2
di sqrt( v[1,1] + v[2,2] - 2*v[1,2] )

**automatic calculation of confidence intervals for d
lincom _b[b2] - _b[b1]

**data re-arrangement in a format suitable for logistic regression
reshape long aa ab bb, i(study) j(group)
gen id = _n
rename aa count1
rename ab count2
```

```

rename bb count3
reshape long count, i(id) j(gen)
drop id
gen wt1=count
xi i.study i.gen

**fitting the model based on the prospective likelihood - Equation 4.1
eq slope2:  _Igen_2
eq slope3:  _Igen_3
gllamm group  _Igen_2 _Igen_3  _Istudy_*,fam(binom) link(logit) i(study ) nrf(2)
eqs(slope2 slope3) w(wt) adapt nip(8)

**automatic calculation of confidence intervals for lambda
nlcom _b[_Igen_2]/_b[_Igen_3]

**automatic calculation of confidence intervals for d
lincom  _b[_Igen_2] - _b[_Igen_3]

**data re-arrangement in a format suitable for polytomous (multinomial) logistic
regression
sort study group gen
gen patt=_n
expand 3
sort patt
qui by patt: gen alt=_n
gen chosen=alt==gen
tab alt,gen(a)
gen a2group=a2*group
gen a3group=a3*group

**fitting the model based on the retrospective likelihood - Equation 4.4
eq slope2:a2group
eq slope3:a3group

gllamm alt group  _Istudy_*, expand(patt chosen m) i(study) link(mlogit)
family(binom) nrf(2) eqs(slope2 slope3) nip(8) weight(wt) adapt

**automatic calculation of confidence intervals for d
lincom [c3]group - [c2]group

**automatic calculation of confidence intervals for lambda
nlcom [c2]group/[c3]group

**alternative parameterization of the retrospective likelihood model
xi i.study i.study*a2 i.study*a3

gllamm alt a2 a3  _IstuXa* a2group a3group, nocons expand(patt chosen o) i(study)
link(mlogit) family(binom) nrf(2) eqs(slope2 slope3 ) nip(8) weight(wt) adapt

**automatic calculation of confidence intervals for lambda
nlcom _b[a2group]/_b[a3group]

**automatic calculation of confidence intervals for d
lincom _b[a3group] -_b[a2group]

```

```

**Data input for the continuous outcome meta-analysis
**Data are taken from the study of Boekholdt, et al, 2005
**Data are listed in the form of Table 2 of the manuscript

clear
input study gen11 gen12 gen22 x11 sd11 x12 sd12 x22 sd22
1 181 296 88 1.18 .32 1.24 .33 1.33 .36
2 500 896 317 .79 .25 .84 .25 .9 .27
3 328 596 210 1.09 .31 1.12 .29 1.25 .4
4 845 1236 409 1.19 .37 1.22 .37 1.3 .41
5 168 256 100 1.36 .36 1.38 .38 1.48 .42
6 237 380 150 1.13 .21 1.19 .24 1.27 .22
7 284 369 139 1.16 .35 1.2 .39 1.22 .41
8 256 335 115 .89 .21 .92 .21 1.03 .27
9 1121 1693 575 .97 .22 1 .23 1.07 .24
10 500 797 299 1.07 .256 1.13 .24 1.17 .24
end

** calculation of the mean differences
gen b1 = x12 - x11
gen b2 = x22 - x11

** calculation of the variances
gen V11 = (sd12)^2/gen12 + (sd11)^2/gen11
gen V22 = (sd22)^2/gen22 + (sd11)^2/gen11

** calculation of the covariance
gen V12 = (sd11)^2 /gen11

** fitting the model of Equation 3.7 using REML
mvmeta b V,vars(b1 b2)

** retrieve the covariance matrix
matrix v=e(V)

**calculation of lambda
di _b[b1]/_b[b2]

**calculation of standard error of lambda - Equation 5.6
di sqrt( v[1,1]/_b[b2]^2 + v[2,2]*_b[b1]^2/_b[b2]^4 -2*v[1,2]*_b[b1]/_b[b2]^3 )

**automatic calculation of confidence intervals for lambda
nlcom _b[b1]/_b[b2]

**calculation of d
di _b[b2] - _b[b1]

**calculation of standard error of d - Equation 5.2
di sqrt( v[1,1] + v[2,2] - 2*v[1,2] )

**automatic calculation of confidence intervals for d
lincom _b[b2] - _b[b1]

**data re-arrangement in a format suitable for fitting the linear mixed model
rename gen11 gen1
rename gen12 gen2
rename gen22 gen3
rename x11 x1
rename sd11 sd1
rename x12 x2
rename sd12 sd2
rename x22 x3

```

```

rename sd22 sd3
reshape long gen x sd,i(study ) j(z)

xi i.study i.z

**calculation of the variance
gen v=sd^2/ gen

**gllamm for stability parameterizes the residual variance using the logarithm of
the standard error
gen s=log(sqrt(v))

**setting up one equation for each study
eq wgt: s

**creating constraints needed for gllamm in order to perform inverse-variance
weighting
constraint define 1 [lns1]s=1

** fitting the model of Equation 3.10
eq s2: _Iz_2
eq s3: _Iz_3

gllamm x _Iz_2 _Iz_3 _Istudy* , i(study) nrf(2) eqs(s2 s3) s(wgt) constraint(1 )
nlp(8) noest

matrix A=M_initf

gllamm x _Iz_2 _Iz_3 _Istudy* , i(study) nrf(2) eqs(s2 s3) s(wgt) constraint(1 )
nlp(8) from(A) long adapt

**automatic calculation of confidence intervals for lambda
nlcom _b[ _Iz_2]/_b[ _Iz_3]

**automatic calculation of confidence intervals for d
lincom _b[ _Iz_3] - _b[ _Iz_2]

*****

```

```

**Stata program for performing bivariate meta-analysis using ML
**The program should be run in batch mode (i.e. in a do-file)
**When there is a small number of studies, the between studies correlation is
**poorly estimated (i.e. -1 or +1), in such a case the program may not reach
**convergence and the user will have to run it again
**Data input for the discrete outcome meta-analysis
**Data are taken from the study of Kato et al, 1999
**Data are listed in the form of Table 1 of the manuscript

```

```

clear
set more off
input study aa0 ab0 bb0 aa1 ab1 bb1
1 3 34 44 2 20 83
2 4 30 49 3 17 62
3 17 84 48 3 30 31
4 5 32 63 5 23 52
5 12 62 119 8 39 133
6 9 48 47 6 34 68
7 18 134 363 20 214 483
end

```

```

** calculation of the log(Odds Ratios)
gen b1=log(( ab1/ aa1)/( ab0/aa0))
gen b2=log(( bb1/ aa1)/( bb0/aa0))

```

```

** calculation of the variances
gen V11=1/aa0 +1/ab0+1/aa1 +1/ab1
gen V22=1/aa0 +1/aa1+1/bb0 +1/bb1

```

```

** calculation of the covariance
gen V12=1/aa0 +1/aa1

```

```

** initial values for input to the bivariate model are obtained from the
** fixed-effects estimates
gen _wz1=1/V11
gen _wz2=1/V22

```

```

gen _z1wz1=b1*_wz1
gen _z2wz2=b2*_wz2

```

```

summarize _z1wz1
local muz1=r(sum)

```

```

summarize _wz1
local swz1=r(sum)
local muz1=`muz1'/'swz1'
local tauz1=1/'swz1'

```

```

summarize _z2wz2
local muz2=r(sum)

```

```

summarize _wz2
local swz2=r(sum)
local muz2=`muz2'/'swz2'
local tauz2=1/'swz2'

```

```

corr _z1wz1 _z2wz2
local r=r(rho)

```

```

local tauz1=log(`tauz1')
local tauz2=log(`tauz2')

```

```

local r=0.5*log((1+'r')/(1-'r'))

drop _wz1 _wz2 _z1wz1 _z2wz2

program drop _all

** here is the main program for evaluating the log-likelihood

program define LL
args ln1 mub1 mub2 logtb1 logtb2 r
quietly {
    scalar taub1=exp(`logtb1')
    scalar taub2=exp(`logtb2')
    scalar rho=(exp(2*'r')-1)/(exp(2*'r')+1)

    gen double vb1=taub1 + V11
    gen double vb2=taub2 + V22

    gen double corr=(rho*sqrt(taub1*taub2) + V12)/sqrt(vb1*vb2)
    gen double L=log(vb1*vb2*(1-corr^2))+ /*
        */ ( (b1-'mub1')^2/vb1 + /*
        */ (b2-'mub2')^2/vb2 - /*
        */ 2*corr*(b1-'mub1')*(b2-'mub2')/sqrt(vb1*vb2)) /*
        */ /(1-corr^2)
    replace `lnl'=-0.5*L - 0.5*log(2*_pi)
    drop vb1 vb2 corr L
}
end

** model fitting - Equation 3.7
** the results should be identical to running: mvmeta b V,vars(b1 b2) ml

ml model lf LL (b1:) (b2:) (logvar1:) (logvar2:) (z_rho:)
ml init b1:_cons='muz1' b2:_cons='muz2' logvar1:_cons='tauz1'
logvar2:_cons='tauz2' z_rho:_cons='r'
ml check
ml search
ml maximize, difficult iter(30)

```

```

**Stata program for fitting the genetic model-free approach of Minelli et al.
**The program should be run in batch mode (i.e. in a do-file)
**Data input for the discrete outcome meta-analysis
**Data are taken from the study of Kato et al, 1999
**Data are listed in the form of Table 1 of the manuscript

```

```

clear
set more off
input study aa0 ab0 bb0 aa1 ab1 bb1
1 3 34 44 2 20 83
2 4 30 49 3 17 62
3 17 84 48 3 30 31
4 5 32 63 5 23 52
5 12 62 119 8 39 133
6 9 48 47 6 34 68
7 18 134 363 20 214 483
end

```

```

** calculation of the log(Odds Ratios)
gen b1=log(( ab1/ aa1)/( ab0/aa0))
gen b2=log(( bb1/ aa1)/( bb0/aa0))

```

```

** calculation of the variances
gen V11=1/aa0 +1/ab0+1/aa1 +1/ab1
gen V22=1/aa0 +1/aa1+1/bb0 +1/bb1

```

```

** calculation of the covariance
gen V12=1/aa0 +1/aa1

```

```

** initial values for input to the bivariate model are obtained from the
** fixed-effects estimates

```

```

gen _wz1=1/V11
gen _wz2=1/V22

```

```

gen _z1wz1=b1*_wz1
gen _z2wz2=b2*_wz2

```

```

summarize _z1wz1
local muz1=r(sum)

```

```

summarize _wz1
local swz1=r(sum)
local muz1=`muz1'/'swz1'
local tauz1=1/'swz1'

```

```

summarize _z2wz2
local muz2=r(sum)

```

```

summarize _wz2
local swz2=r(sum)
local muz2=`muz2'/'swz2'
local tauz2=1/'swz2'

```

```

local lambda=`muz1'/'muz2'

```

```

drop _wz1 _wz2 _z1wz1 _z2wz2
local tauz2=log(`tauz2')

```

```

program drop _all

```

```

** here is the main program for evaluating the log-likelihood

```

```

program define LL

```

```

args ln1 lambda mub2 logtb2
quietly {
    scalar taub2=exp(`logtb2')
    scalar mub1=`mub2'*`lambda'
    gen double vb1=`lambda'^2*taub2 + V11
    gen double vb2=taub2 + V22
    gen double cov=`lambda'*taub2 + V12
    gen double corr=cov/sqrt(vb2*vb1)
    gen double L=log(vb1*vb2*(1-corr^2)) + /*
        */ ( (b1 - mub1 )^2/vb1 + /*
        */ (b2 -`mub2')^2/vb2 - /*
        */ 2*corr*(b1 - mub1)*(b2 - `mub2')/sqrt(vb2*vb1) )/*
        */ /(1-corr^2)
    replace `lnl'=-0.5*L - 0.5*log(2*_pi)
    drop vb2 vb1 corr cov L
}
end

** model fitting- Equation 3.13

ml model lf LL (lambda:) (b2:) (logvar2:)
ml init lambda:_cons=`lambda' b2:_cons=`mub2' logvar2:_cons=`tauz2'
ml check
ml search
ml maximize, difficult iter(30)

```
