Mathematical Ability and Socio-economic Background: IRT Modeling to Estimate Genotype by Environment Interaction: Supplementary material

Inga Schwabe Dorret I. Boomsma
Stéphanie M. van den Berg

This document contains supplementary material for the article "Mathematical Ability and Socio-economic Background: IRT Modeling to Estimate Genotype by Environment Interactiona".

Proof of the indeterminacy of the Purcell A×M parametrization

Purcell's univariate model for genotype-environment interaction resembles the general ANOVA model with a two-way interaction deceptively close:

$$P_{ij} = \beta_0 + \beta_1 M_{ij} + \beta_2 A_{ij} + \beta_3 A_{ij} M_{ij} + eE_{ij} \quad (1)$$

$$E_{ij} \sim N(0,1) \quad (2)$$

$$A_{ij} \sim N(0,1) \quad (3)$$

We have an intercept β_0, a main effect of the measured covariate M_{ij}, β_1, a main effect of the unmeasured genotypic value A_{ij}, β_2, an interaction effect of the measured covariate and the genotypic value, β_3, and a residual term with variance e^2. This model can be extended with a main effect of the shared environment, $\beta_4 C_{ij}$, plus an additional interaction effect, $\beta_5 C_{ij} M_{ij}$, but for our purpose here it suffices to discuss the model with additive genetic effects alone.

The two things that are different from the general ANOVA model is that variable A is unobserved, and that we have data on twin pairs, where phenotypes are correlated. In case of additive genetic effects, the genetic correlation equals 1 for monozygotic twin pairs and $\frac{1}{2}$ for dizygotic twin pairs, that is, for MZ twin pairs we have $Cov(A_{i1}, A_{i2}) = 1$ and for DZ twin pairs we have $Cov(A_{i1}, A_{i2}) = \frac{1}{2}$.

If we assume that M is a dichotomously scored covariate, the sufficient statistics for our model are the variance of phenotype P, the observed covariance in

MZ twin pairs and the observed phenotypic covariance in DZ twin pairs, under the two conditions $M = 0$ and $M = 1$, where we assume that M has equal values for the two twins in each pair. This consists of 6 different statistics. Thus, the following set of equations needs to be solved for the regression coefficients:

M=0

$$Cov_{MZ}(P_{i1}, P_{i2}) = \beta_2^2 \tag{4}$$

$$Cov_{DZ}(P_{i1}, P_{i2}) = \frac{1}{2}\beta_2^2 \tag{5}$$

$$Var(P_{i1}) = Var(P_{i2}) = \beta_2^2 + e^2 \tag{6}$$

M=1

$$Cov_{MZ}(P_{i1}, P_{i2}) = (\beta_2 + \beta_3)^2 \tag{7}$$

$$Cov_{DZ}(P_{i1}, P_{i2}) = \frac{1}{2}(\beta_2 + \beta_3)^2 \tag{8}$$

$$Var(P_{i1}) = Var(P_{i2}) = (\beta_2 + \beta_3)^2 + e^2 \tag{9}$$

Since β_2 is only linked to observed statistics through a quadratic function, it is easily seen that the probability of the data under $M = 0$ given a value $\beta_2 = a$ is equal to the probability of the data given $\beta_2 = -a$.

Under $M = 1$, we immediately see that combinations of values for $(\beta_2+\beta_3) = b + c$ are equally likely as $(\beta_2 + \beta_3) = -b - c$. Thus, any combination of values for β_2 and β_3 is equally likely as the combination of their negatives. Taking the negative of the value for β_2 does, as we saw, not affect the probability of the data under $M = 0$, so that for any data set, there are always two combinations that have the exact same likelihood.

This problem cannot be easily solved by constraining β_2 to be positive, $\beta_2 > 0$. This is because for any positive value of β_2, there are two values for β_3 that result in the same expected variances and covariances for $M = 1$, since using the square root formula for quadratic functions we get:

$$Var(P) = (\beta_2 + \beta_3)^2 = \beta_2^2 + 2\beta_2\beta_3 + \beta_3^2 + e^2 \tag{10}$$

$$\beta_3^2 + 2\beta_2\beta_3 + (\beta_2^2 + e^2 - Var(P)) = 0 \tag{11}$$

$$\beta_3 = -\beta_2 \pm \sqrt{\beta_2^2 - (\beta_2^2 + e^2 - Var(P))} = -\beta_2 \pm \sqrt{Var(P) - e^2} \tag{12}$$

$$Cov_{MZ}(P_1, P_2) = (\beta_2 + \beta_3)^2 = \beta_2^2 + 2\beta_2\beta_3 + \beta_3^2 \tag{13}$$

$$\beta_3^2 + 2\beta_2\beta_3 + (\beta_2^2 - Cov_{MZ}(P, P)) = 0 \tag{14}$$

$$\beta_3 = -\beta_2 \pm \sqrt{\beta_2^2 - (\beta_2^2 - Cov_{MZ}(P_1, P_2))} = -\beta_2 \pm \sqrt{Cov_{MZ}(P_1, P_2)} \tag{15}$$

Thus, there is no unique Maximum Likelihood solution for a given data set. The problem lies in the fact that the genotypic value is unobserved (so

that there is not observed covariance between A and P), and that therefore all observed statistics are related only quadratically with the parameters that need to be estimated. The proof can be extended to ACE models and continuous measured covariates M in a similar manner.

Estimation procedure

We applied Gibbs sampling (Geman & Geman, 1984; Gelfand & Smith, 1990; Gelman, Carlin, Stern, & Rubin, 2004), a Markov chain Monte Carlo (MCMC) algorithm that works by iteratively drawing samples from the full conditional distributions of all unobserved parameters of a model. The full conditional distribution refers to the distribution of a parameter given the current or known values of all other relevant parameters in the model (see e.g., Gilks, Richardson, & Spiegelhalter, 1996). A sample from the full conditional distribution is taken in every iteration of the Gibbs sampling. After a number of burn-in" iterations, subsequent draws can be regarded as draws from the joint posterior distribution. For the MCMC estimation, we used the freely available software package JAGS (Plummer, 2003). The JAGS script that was used for the model described in the manuscript can be found in the online supplementary material.

Prior distributions

As the above described model was estimated using Bayesian statistics, prior distributions had to be specified. We used independent normal distribution for all intercepts (β_{0a}, β_{0c} and $\beta_{0e} \sim N(-1, \sigma^2 = 2)$ and interaction effects (β_{1a},β_{1c} and $\beta_{1e} \sim N(0, \sigma^2 = 10)$). Also for the phenotypic population mean and the regression coefficient that expresses the main effect of the moderator variable, independent normal distributions were chosen as prior distributions ($\mu \sim N(0, \sigma^2 = 10$ and $\beta_{1m} \sim N(0, \sigma^2 = 10)$). As non-informative priors were used in the biometric part of the model, the Bayesian approach will yield comparable results as the maximum likelihood framework.

Simulation study: Differing number of items

In order to assess the impact of the psychometric information, we varied the number of items (20, 100 and 250 items) while fixing the number of twin pairs to 1000. Cronbach's alpha was 0.90 when responses to 60 items were simulated and 0.75 in case of 20 items.

150 datasets were simulated in each condition, μ was fixed to 0, $\exp(\beta_{0a}$ was set to 0.25, $\exp(\beta_{0c})$ was 0.25, $\exp(\beta_{0e})$ was fixed to 0.5 and β_{1m} was set to 0.7. The data was simulated without any ACE×M interaction effects. Furthermore, item difficulty parameters were assumed to be known and simulated equally spaced within the interval [-3.2;3.2] and the Rasch model was used to simulate responses to phenotypic dichotomous items. The minimum and maximum value

of this interval were based on (minus) three times the standard deviation of all phenotypic values.

All simulations were carried out using the software package R (R DevelopmentCore Team, 2008), an open-source language and environmental for statistical com-puting. As an interface from R to JAGS, the R package rjags was used (Plummer,2013). After an adaption phase of 5,000 iterations and a burn-in phase of 50,000iterations, the characterisation of the posterior distribution for the model parame-ters was based on an additional 25,000 iterations from 1 Markov chain.The average posterior means of all model parameters as well as the standard deviation of posterior means and the means of all posterior standard deviations were calculated.The mean of posterior standard deviations can be seen as the Bayesian version of the standard error.

Table 1: Results of the simulation study, part II (differing number of items). Posterior means (SD) averaged over 250 replications. Second line: Mean of posterior standard deviations. Ni refers to the number of items.

	β_{1m}	$\exp(\beta_{0a})$	$\exp(\beta_{0c})$	$\exp(\beta_{0e})$	β_{1a}	β_{1c}	β_{1e}
True value	0.70	0.25	0.25	0.50	0.00	0.00	0.00
Ni = 20	0.70 (0.06)	0.24 (0.12)	0.20 (0.09)	0.49 (0.06)	0.02 (1.47)	-0.01 (1.24)	-0.02 (0.25)
	0.06	0.09	0.07	0.06	0.99	0.90	0.22
Ni = 100	0.70 (0.05)	0.23 (0.10)	0.21 (0.08)	0.50 (0.04)	-0.03 (1.29)	-0.02 (1.07)	0.01 (0.17)
	0.05	0.09	0.07	0.04	0.95	0.83	0.17
Ni = 250	0.70 (0.05)	0.21 (0.10)	0.22 (0.08)	0.50 (0.04)	-0.02 (1.32)	0.02 (0.97)	0.01 (0.16)
	0.05	0.08	0.06	0.04	0.92	0.80	0.14

The results can be found in Table 1. It can be seen that, with only 20 items, average posterior means of most parameters were close to their true values with a slight bias in $\exp(\beta_{0c})$. This precision is comparable to the results of the first simulation study (1000 twin pairs) where the number of items was fixed to 60. There was only a small decrease in standard deviations and standard errors with increasing number of items. Also the increase in precision with increasing sample size was small, suggesting that as much as 20 items are sufficient to fit the ACE×M model.

JAGS script for an ACE×M + 1PL model

Following JAGS script fits a (univariate) ACE×M model (same moderator value for every family) with an incorporated 1 PL IRT model at the phenotypic level. Item parameters are assumed known.

```
1 #y_dz = Item responses of DZ twins (matrix)
2 #y_mz = Item responses of MZ twins (matrix)
```

```
#x_MZ = Values on moderator variable for all MZ twin pairs
#x_DZ = Values on moderator variable for all DZ twin pairs
#n_mz = Number of MZ twin pairs
#n_dz = Number of DZ twin pairs
#n_items = Number of phenotypic items administered
#b = Vector with item difficulty parameters, assumed known here

#Required structure of the y_mz/y_dz data matrix:
#y_dz[i,k] = kth datapoint from the ith DZ twin pair
#y_mz[i,k] = kth datapoint from the ith MZ twin pair

#This results in a matrix of n_mz (or, in case of y_dz, n_dz)
#rows and 2*n_items columns. e.g. y_mz[1,22] is the response
#of MZ twin 1 from family 1 to item 22 if n_items = 22

#JAGS uses precision parameters for the variance parameters.
#Therefore, after running the script, these precision parameters
#have to be inverted. For example:
#var_a <-  1/outputAnalysis$tau_a[,,1] with the rjags package

#In this script, IRT parameters are assumed known.

model{
##MZ twins
for (fam in 1:n_mz){
        c_mz[fam] ~ dnorm(mu + beta_1m * x_MZ[fam], tau_c_mz[fam])
        f_mz[fam] ~ dnorm(c_mz[fam], tau_a_mz[fam])

        tau_c_mz[fam] <- 1/exp(beta_0c + beta_1c * x_MZ[fam])
        tau_a_mz[fam] <- 1/exp(beta_0a + beta_1a * x_MZ[fam])
        tau_e_mz[fam] <- 1/exp(beta_0e + beta_1e * x_MZ[fam])

    pheno_mz[fam,1] ~ dnorm(f_mz[fam], tau_e_mz[fam])
        pheno_mz[fam,2] ~ dnorm(f_mz[fam], tau_e_mz[fam])

        #1pl model twin1
        for (k in 1:n_items){
                logit(p[fam,k]) <- pheno_mz[fam,1] - b[k]
                y_mz[fam,k] ~ dbern(p[fam,k])
        }

        #1pl model twin2
        for (k in (n_items+1):(2*n_items)){
                logit(p[fam,k]) <- pheno_mz[fam,2] - b[k-n_items]
                y_mz[fam,k] ~ dbern(p[fam,k])
        }
```

```
} #end MZ twins

##DZ twins
for (fam in 1:n_dz){
        c_dz[fam] ~ dnorm(0,tau_c_dz[fam])
        a1_dz[fam] ~ dnorm(0,2)
        a2_dz[fam,1] ~ dnorm(a1_dz[fam], 2)
    a2_dz[fam,2] ~ dnorm(a1_dz[fam], 2)

        tau_c_dz[fam] <- 1/(exp(beta_0c + beta_1c * x_DZ[fam]))
        var_a_dz[fam] <- exp(beta_0a + beta_1a * x_DZ[fam])
        tau_e_dz[fam] <- 1/(exp(beta_0e + beta_1e * x_DZ[fam]))

        a_dz_twin1[fam] <- a2_dz[fam,1] * sqrt(var_a_dz[fam])
        a_dz_twin2[fam] <- a2_dz[fam,2] * sqrt(var_a_dz[fam])

        pheno_dz[fam,1] ~ dnorm(mu + beta_1m * x_DZ[fam] +
                                c_dz[fam] + a_dz_twin1[fam], tau_e_dz[fam])
        pheno_dz[fam,2] ~ dnorm(mu + beta_1m * x_DZ[fam] +
                                c_dz[fam] + a_dz_twin2[fam], tau_e_dz[fam])

        #1pl model twin1
        for (k in 1:n_items){
                logit(p_dz[fam,k]) <- pheno_dz[fam,1] - b[k]
                y_dz[fam,k] ~ dbern(p_dz[fam,k])
        }

        #1pl model twin2
        for (k in (n_items+1):(2*n_items)){
                logit(p_dz[fam,k]) <- pheno_dz[fam,2] - b[k-n_items]
                y_dz[fam,k] ~ dbern(p_dz[fam,k])
        }
} #end DZ twins

#Priors
mu ~ dnorm(0, .1)
beta_1a ~ dnorm(0, .1)
beta_1c ~ dnorm(0, .1)
beta_1e ~ dnorm(0, .1)
beta_1m ~ dnorm(0, .1)

beta_0a ~ dnorm(-1, .5)
beta_0c ~ dnorm(-1, .5)
beta_0e ~ dnorm(-1, .5)
}
```

References

Gelfand, A. E., & Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *J Am Stat Ass*, *85*(410), 398–409.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2004). *Bayesian data analysis* (2nd ed.). London: Chapman and Hall.

Geman, S., & Geman, D. (1984). Stochastic relaxation, gibbs distributions and the bayesian restoration of images. *IEEE Trans Patt Anal Mach Intell*, *6*(6), 721–741.

Gilks, W., Richardson, S., & Spiegelhalter, D. (1996). *Markov chain monte carlo in practice*. Boca Raton, FL: Chapman & Hall/CRC.

Plummer, M. (2003). *Jags: A program for analysis of bayesian graphical models using gibbs sampling.*