

Data simulations: GxEMM versus GE random slope model

Do they give the same estimates?

Simon Forsberg

2020-05-14

Generative model in simulations

$$y_1 = G * g_{add1} + e_1$$

$$y_2 = G * g_{add2} + (G * g_{cov} + e_{cov}) * y_1 + e_2$$

Where:

- G is the genotype matrix
- $g_{add1} \sim N(0, h_{add1}^2/n_{loci})$ is the per SNP additive effect on y_1
- $g_{add2} \sim N(0, h_{add2}^2/n_{loci})$ is the per SNP additive effect on y_2
- $g_{cov} \sim N(0, h_{cov}^2/n_{loci})$ is the per SNP effect on the slope
- $e_{cov} \sim N(0, \sigma_{cov}^2)$ is the per individual environmental effect on the slope
- $e_1 \sim N(0, 1 - h_{add1}^2)$
- $e_2 \sim N(0, 1 - h_{add2}^2 - h_{cov}^2 - \sigma_{cov}^2)$
- r_g describes the fraction of g_{add1} and g_{add2} that are the same, eg pleiotropy. ($r_g = 0$ here)

Estimated Models

GE random slope model

Both environmental and genetic random slopes:

$$y_2 = \mu + b_{g,int} + (\beta + b_{g,slope} + b_{e,slope}) * y_1 + e$$

Where:

- μ and β are fixed effects
- $b_{g,int} \sim N(0, K\sigma_{g,int}^2)$
- $b_{g,slope} \sim N(0, K\sigma_{g,slope}^2)$
- $b_{e,slope} \sim N(0, I\sigma_{e,slope}^2)$
- $e \sim N(0, I\sigma_e^2)$
- K is the kinship matrix, estimated from G

GxEMM

See [here](#)

Parameters

```

testing <- F

n_loci <- 400 #number of loci
nrPops <- 10 #number of subpopulations
Fst <- .3 #degree of stratification
models <- c("slopeGandE_add_12", "gxemm_iid")

#Sample sizes and number of simulations
nrep <- 1 #n observations
vals_n <- 1000 #N Individuals
vals_sim <- 1:3 #Simulations per parameter combination

h2_cov <- c(0, 0.2, 0.4)
h2_add_t1 <- c(0, .2)
h2_add_t2 <- c(0, .2)
sigma_cov <- c(0, 0.2, 0.4)
rhog <- 0

if(testing) {
  nrep <- 1
  vals_n <- 100
  vals_sim <- 1:2
}

```

Simulation: $h_{cov}^2 + \sigma_{cov}^2 + h_{add1}^2 + h_{add2}^2$

```

resFile <- '../results/sim1_gxemm.RData'

if(!file.exists(resFile)){
  grid <- expand.grid(n = as.integer(vals_n), sim = vals_sim, h2_add_t1 = h2_add_t1, h2_add_t2 = h2_add_t2,
    SimRes <- data.frame(grid, runtime_slopeGE = NA, runtime_gxemm = NA,
      h2.add.t2_slopeMod_GE = NA,
      h2.cov_slopeMod_GE = NA,
      sigma.cov_slopeMod_GE = NA,
      gxemm_h2.hom = NA,
      gxemm_h2.het = NA)

  for(i in 1:nrow(grid)){
    n <- grid$n[i]
    sim <- grid$sim[i]
    h2.cov <- grid$h2_cov[i]
    h2.add1 <- grid$h2_add_t1[i]
    h2.add2 <- grid$h2_add_t2[i]
    sigma.cov <- grid$sigma_cov[i]
    message(i, "/", nrow(grid), "\n")

    #Run simulation
    simdat <- genBivarCov.simulatePair_outbred(n_ind = n, n_perInd = nrep,
      h2.add.t1 = h2.add1, h2.add.t2 = h2.add2,
      h2.cov = h2.cov, rhog = rhog, e.cov = sigma.cov,
      nrPops = nrPops, Fst = Fst, postproc = T)
  }
}

```

```

#Fit models
simdat.modFit <- run_models(models, simdat$pheno, simdat$G, ldak_loc = ldak_loc)

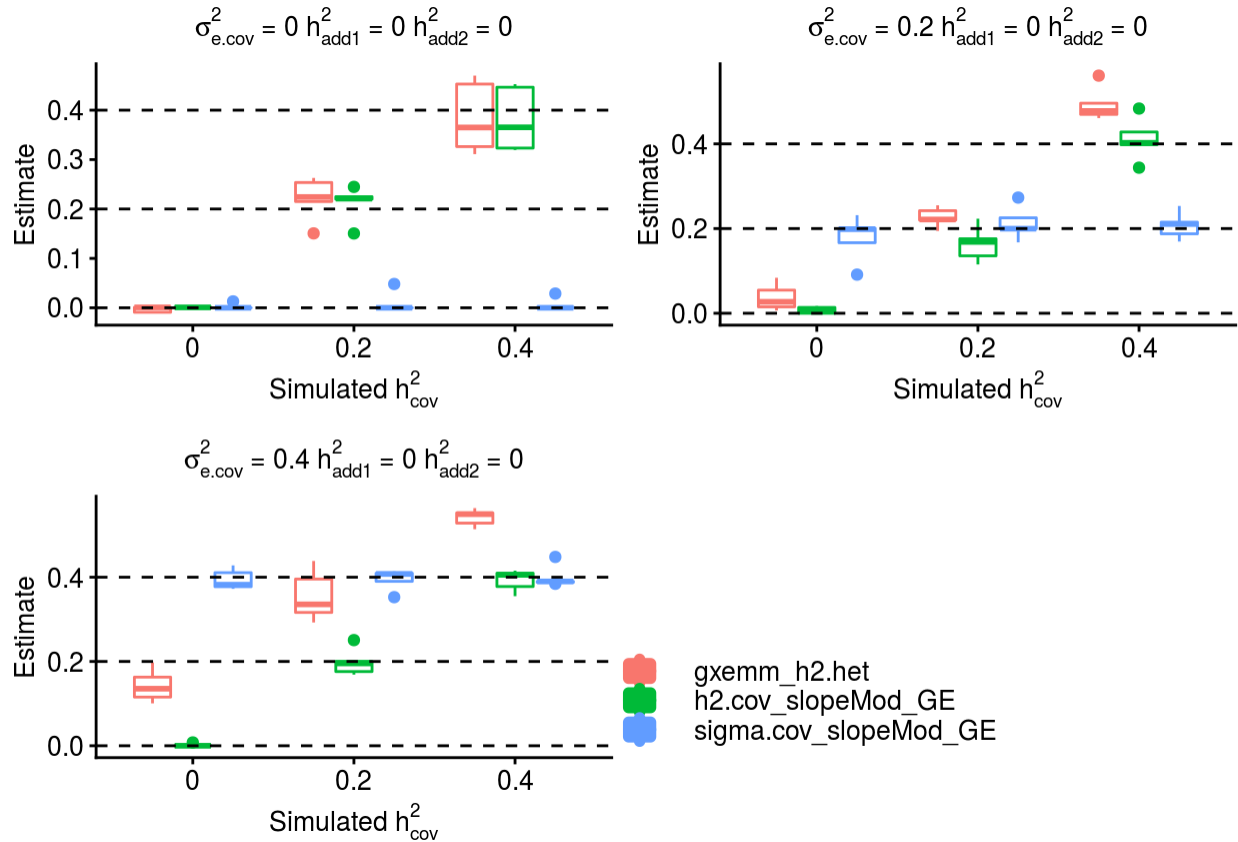
#Store results
SimRes$runtime_slopeGE[i] <- simdat.modFit$runtime[simdat.modFit$model == 'slopeGandE_add_12']
SimRes$runtime_gxemm[i] <- simdat.modFit$runtime[simdat.modFit$model == 'gxemm_iid']
SimRes$h2.add.t2_slopeMod_GE[i] <- simdat.modFit$h2.add.t2[simdat.modFit$model == 'slopeGandE_add_12']
SimRes$h2.cov_slopeMod_GE[i] <- simdat.modFit$h2.cov_12[simdat.modFit$model == 'slopeGandE_add_12']
SimRes$sigma.cov_slopeMod_GE[i] <- simdat.modFit$e.cov_12[simdat.modFit$model == 'slopeGandE_add_12']
SimRes$gxemm_h2.hom[i] <- simdat.modFit$h2.hom_gxemm[simdat.modFit$model == 'gxemm_iid']
SimRes$gxemm_h2.het[i] <- simdat.modFit$h2.het_gxemm[simdat.modFit$model == 'gxemm_iid']
}
save(list = 'SimRes', file = resFile)
}else
load(resFile)

```

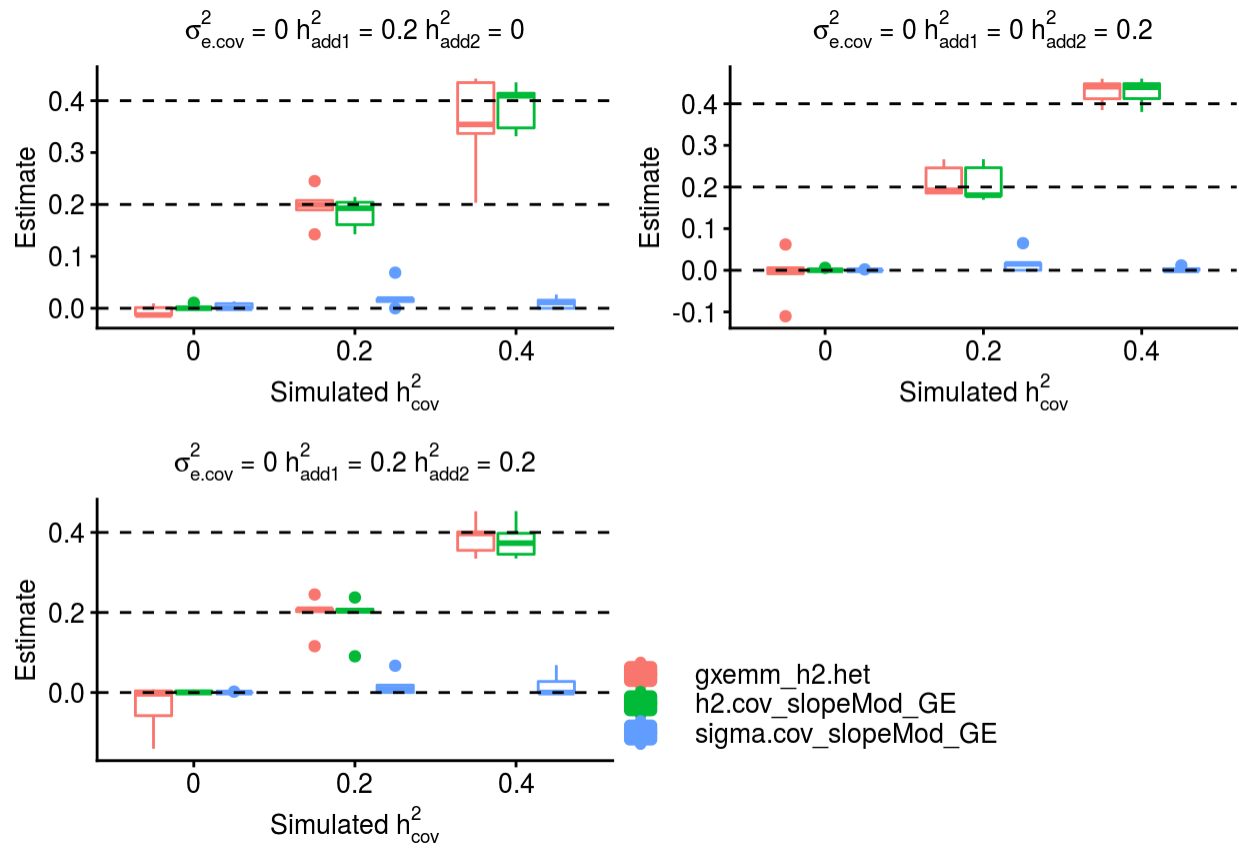
Results

- x-axes = simulated h_{cov}^2
- y-axes = estimates of h_{cov}^2 , σ_{cov}^2 (from GE random slope model), and h_{het}^2 (from GxEMM)

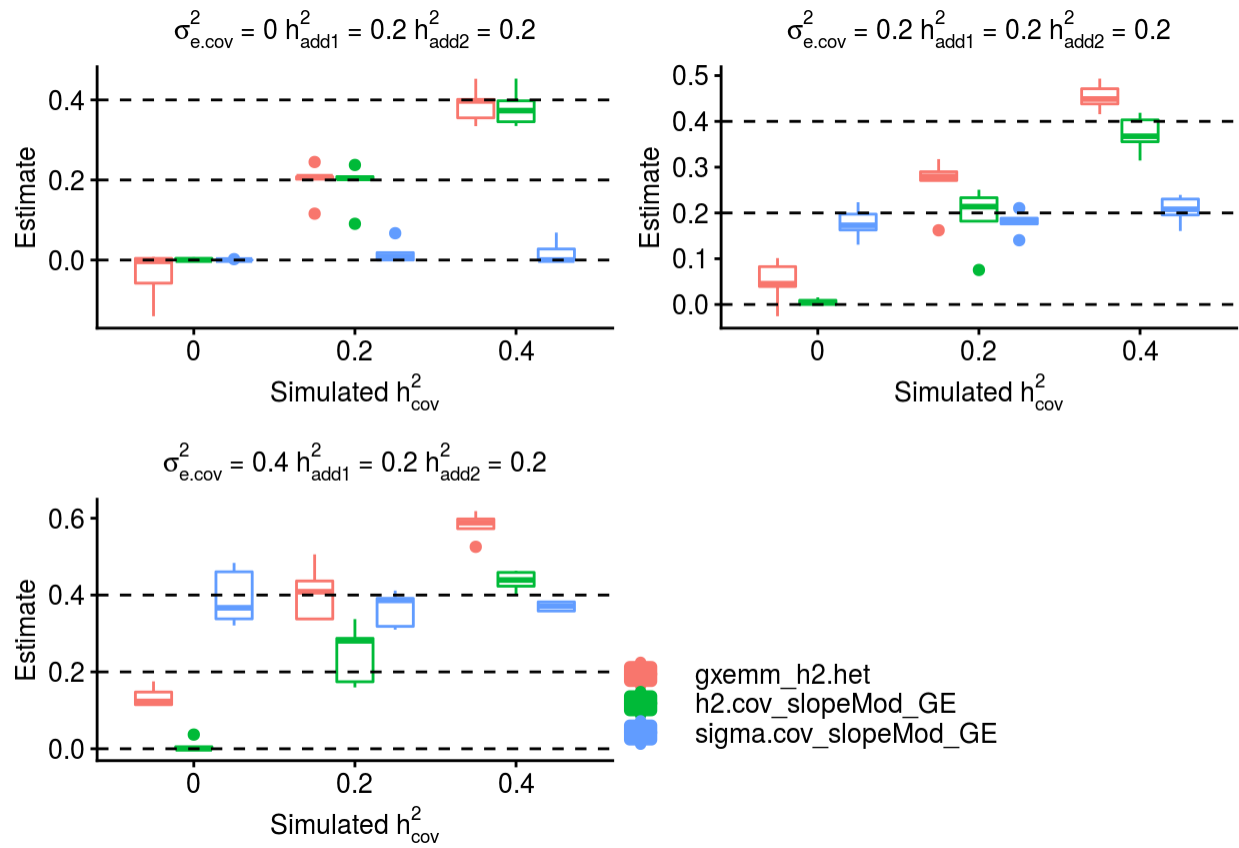
$$h_{cov}^2 + \sigma_{cov}^2$$



$$h_{cov}^2 + h_{add1}^2 + h_{add2}^2$$



$$h_{cov}^2 + \sigma_{cov}^2 + h_{add1}^2 + h_{add2}^2$$



Runtime

