Supplementary Section 1: More comparisons among the $R^2$s.

To give a comprehensive assessment of the $R^2$s, this supplement discusses and plots a complete set of simulations, from which a subset was presented in the main text. The different models (LMM, PGLS, GLMM, and PLOG) are presented in turn. In the comparisons for LMMs and GLMMs, I compare the partial $R^2$s of $R^2_{\text{resid}}$, $R^2_{\text{lik}}$ and $R^2_{\text{pred}}$ for the fixed effect to $R^2_{\text{glmm(m)}}$ and the partial $R^2$s for the random effect to $R^2_{\text{glmm(r)}}$. This illustrates the differences between using partial $R^2$s and marginal $R^2$s.

Figure Captions

Figure S1: Simulation results for a Linear Mixed Model (LMM) giving $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, $R^2_{\text{pred}}$, $R^2_{\text{glmm}}$, and $R^2_{\text{ols}}$ versus the log likelihood ratio (LLR) between full and reduced models. The simulation model (equation 18) contained both a fixed effect $\beta$ for a continuous variable $x$ and a random effect $b$ for a categorical variable $u$. For (a), (b) and (c), data were simulated without the random effect ($\beta = 1$, $\sigma = 0$), and for (d), (e) and (f), data were simulated without the fixed effect ($\beta = 0$, $\sigma = 1.5$). Simulations for (g), (h), and (i) contained both fixed and random effects. Columns give different partial $R^2$s for each method. Specifically, (a), (d), and (g) give the partial $R^2$s in which the reduced model removes the fixed effect for $x$: therefore, these give partial $R^2$s for the fixed effect. Panels (b), (e), and (h) give the partial $R^2$s in which the reduced model removes the random effect for $u$: therefore, these give partial $R^2$s for the random effect. In panels (c), (f) and (i), the reduced model removes both fixed and random effects, giving the total $R^2$s. Each data set consisted of 100 simulated points, $x$ was simulated as a normal (0, 1) random variable, and $u$ had
10 levels with \( b \) is simulated as a normal \((0, \sigma)\). All analyses were performed with the function `lmer()`.

Figure S2: Simulation results for a Linear Mixed Model (LMM) giving associations between \( R^2_{resid}, R^2_{lik}, R^2_{pred}, R^2_{glmm}, \) and \( R^2_{ols} \). Data are the same as presented in figure S1.

Figure S3: Simulation results for a Linear Mixed Model (LMM) showing means and standard deviations of \( R^2_{resid}, R^2_{lik}, R^2_{pred}, R^2_{glmm}, \) and \( R^2_{ols} \) versus sample size. The simulation model (equation 18) contained both a fixed effect \( \beta \) for a continuous variable \( x \) and a random effect \( b \) for a categorical variable \( u \). For each level of \( u \), from 4 to 16 replicates were simulated. (a), (b), and (c) give means of each \( R^2 \) were calculated for 500 simulations at each sample, and (d), (e), and (f) give standard deviations. Columns give different partial \( R^2 \)s, with (a) and (d) giving the partial \( R^2 \)s for the fixed effect, (b) and (e) giving the partial \( R^2 \)s for the random effect, and (c) and (f) giving the total \( R^2 \)s. In the simulations, \( x \) is simulated as a normal \((0, 1)\) random variable with \( \beta = 1; u \) has 10 levels and \( b \) is simulated as a normal \((0, \sigma = 1.5)\) random variable; and residuals \( e \) are independent \((0, 1)\) random variables. All analyses were performed with the function `lmer()`.

Figure S4: Simulation results for the phylogenetic model with a continuous predictor variable \( x \) giving \( R^2_{resid}, R^2_{lik}, \) and \( R^2_{pred} \) versus the log likelihood ratio (LLR) between full and reduced models. For each simulation, a phylogenetic tree was first simulated, and the values of \( x \) were simulated up the phylogeny assuming Brownian Motion evolution. Data were simulated using equation (18) with \( b = 0 \), and residuals \( e_i \) were simulated from a multivariate normal distribution with mean zero and covariance matrix \( \Sigma(\lambda) = (1 - \lambda)I + \lambda \Sigma_{BM} \). For (a), (b) and (c), data were simulated without phylogenetic signal \((\lambda = 0, \beta = 1)\), and for (d), (e) and (f), data were simulated without the fixed effect \((\lambda = 0.5, \beta = 0)\). Simulations for (g), (h), and (i) contained both fixed and phylogenetic effects \((\beta = 1, \lambda = 0.5)\). (a), (d), and (g) give the partial \( R^2 \)s for the fixed effect. Panels (b), (e), and (h) give the partial \( R^2 \)s for the phylogenetic effect. In panels (c), (f) and (i), the reduced model removes both fixed and phylogenetic effects, giving the total \( R^2 \)s. All analyses were performed with the function `phylolm()`.
Figure S5: Simulation results for a PGLS model giving associations between $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, and $R^2_{\text{pred}}$. Data are the same as presented in figure S3.

Figure S6: Simulation results for the phylogenetic model with a continuous response variable showing means and standard deviations of $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, and $R^2_{\text{pred}}$ versus sample size. For each simulation, a phylogenetic tree was first simulated, and the values of the predictor variable $x$ were simulated up the phylogeny assuming Brownian Motion evolution. Residuals $e_i$ were simulated from a multivariate normal distribution with mean zero and covariance matrix $\Sigma(\lambda) = (1 - \lambda)I + \lambda \Sigma_{\text{BM}}$, and the parameter values were $\lambda = 0.5$, $\beta = 1$, and $b = 0$. (a), (b), and (c) give means of each $R^2$ were calculated for 500 simulations at each sample, and (d), (e), and (f) give standard deviations. Columns give different partial $R^2$s, with (a) and (d) giving the partial $R^2$s for $x$, (b) and (e) giving the partial $R^2$s for phylogenetic signal $\lambda$, and (c) and (f) giving the total $R^2$s. All analyses were performed with the function phylolm().

Figure S7: Simulation results for a binary Generalized Linear Mixed Model (GLMM) giving $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, $R^2_{\text{pred}}$, $R^2_{\text{glmm}}$, and $R^2_{\text{ols}}$ versus the log likelihood ratio (LLR) between full and reduced models. The simulation model (equation 18) contained both a fixed effect $\beta$ for a continuous variable $x$ and a random effect $b$ for a categorical variable $u$. The scaling $\sigma^2_{\text{d.rNS}} = 0.8768809 \pi^2/3$ was used for $R^2_{\text{resid}}$ (see Appendix 1). For (a), (b) and (c), data were simulated without the random effect ($\beta = 1.8, \sigma = 0$), and for (d), (e) and (f), data were simulated without the fixed effect ($\beta = 0, \sigma = 1.8$). Simulations for (g), (h), and (i) contained both fixed and random effects. (a), (d), and (g) give the partial $R^2$s for the fixed effect, and panels (b), (e), and (h) give the partial $R^2$s for the random effect. In panels (c), (f) and (i) give total $R^2$s. In the simulations, $x$ is simulated as a normal (0, 1) random variable and $u$ has 10 levels and $b$ is simulated as a normal (0, $\sigma$). All analyses were performed with the function glmer().

Figure S8: Simulation results for a Generalized Linear Mixed Model (GLMM) giving associations between $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, $R^2_{\text{pred}}$, $R^2_{\text{glmm}}$, and $R^2_{\text{ols}}$. Data are the same as presented in figure S7.
Figure S9: Simulation results for a binary Generalized Linear Mixed Model (GLMM) showing means and standard deviations of $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, $R^2_{\text{pred}}$, $R^2_{\text{glmm}}$, and $R^2_{\text{ols}}$ versus sample size. The scaling $\sigma_{d.rNS}^2 = 0.8768809 \pi^2/3$ was used for $R^2_{\text{resid}}$ (see Appendix 1). The simulation model (equation 18) contained both a fixed effect $\beta$ for a continuous variable $x$ and a random effect $b$ for a categorical variable $u$. For each level of $u$, from 4 to 16 replicates were simulated. (a), (b), and (c) give means of each $R^2$ were calculated for 1000 simulations at each sample, and (d), (e), and (f) give standard deviations. Columns give different partial $R^2$s, with (a) and (d) giving the partial $R^2$s for the fixed effect, (b) and (e) giving the partial $R^2$s for the random effect, and (c) and (f) giving the total $R^2$s. In the simulations, $x$ is simulated as a normal $(0, 1)$ random variable with $\beta = 1.8$ and $u$ has 10 levels and $b$ is simulated as a normal $(0, \sigma = 1.8)$ random variable. All analyses were performed with the function glmer().

Figure S10: Simulation results for the phylogenetic model with a continuous predictor variable $x$ giving $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, and $R^2_{\text{pred}}$ versus the log likelihood ratio (LLR) between full and reduced models. The scaling $\sigma_{d.rNS}^2 = 0.8768809 \pi^2/3$ was used for $R^2_{\text{resid}}$ (see Appendix 1). For each simulation, a phylogenetic tree was first simulated, and the values of $x$ were simulated up the phylogeny assuming Brownian Motion evolution. Data were simulated using equation (18) with $b = 0$, and residuals $e_i$ were simulated from a multivariate normal distribution with mean zero and covariance matrix $\Sigma(\lambda) = (1 - \lambda)I + \lambda \Sigma_{BM}$. For (a), (b) and (c), data were simulated without phylogenetic signal ($\lambda = 0$, $\beta = 1.5$), and for (d), (e) and (f), data were simulated without the fixed effect ($\lambda = 2$, $\beta = 0$). Simulations for (g), (h), and (i) contained both fixed and phylogenetic effects ($\lambda = 2$, $\beta = 1.5$). (a), (d), and (g) give the partial $R^2$s for the fixed effect. Panels (b), (e), and (h) give the partial $R^2$s for the phylogenetic effect. In panels (c), (f) and (i), the reduced model removes both fixed and phylogenetic effects, giving the total $R^2$s. All analyses were performed with the function phylolm().

Figure S11: Simulation results for a phylogenetic logistic regression model giving associations between $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, and $R^2_{\text{pred}}$. Data are the same as presented in figure S10.

Figure S12: Simulation results for the phylogenetic model with a binary response variable showing means and standard deviations of $R^2_{\text{resid}}$, $R^2_{\text{lik}}$, and $R^2_{\text{pred}}$ versus sample size. The scaling
$\sigma_{d.rNS}^2 = 0.8768809 \pi^2/3$ was used for $R^2_{\text{resid}}$ (see Appendix 1). For each simulation, a phylogenetic tree was first simulated, and residuals $e_i$ (equation 18) were simulated from a multivariate normal distribution with mean zero and covariance matrix $\Sigma(\lambda) = \lambda \Sigma_{\text{BM}}$. Values of the predictor variable $x$ were assumed to be independently distributed by a $(0,1)$ normal distribution, and the parameter values were $\lambda = 2$, $\beta = 1.5$, and $b = 0$. (a), (b), and (c) give means of each $R^2$ were calculated for 500 simulations at each sample, and (d), (e), and (f) give standard deviations. Columns give different partial $R^2$'s, with (a) and (d) giving the partial $R^2$'s for $x$, (b) and (e) giving the partial $R^2$'s for phylogenetic signal $\lambda$, and (c) and (f) giving the total $R^2$'s. Calculations of $R^2_{\text{lik}}$ were performed with a modified version of the function phyloglm() and the function glm(). Calculations of $R^2_{\text{resid}}$ and $R^2_{\text{pred}}$ were performed with the function binaryPGLMM().
Fig. S1

Partial $R^2$ for $\beta_1$

Partial $R^2$ for $\theta$

Total $R^2$

(d)

Partial $R^2$ for $\theta$

(f)

Total $R^2$

(g)

(resid lik pred glmm ols)

(h)

(i)
Fig. S2

\[ \beta_1 = 1, \theta = 1.5 \]

\[ \beta_1 = 1, \theta = 0 \]

\[ \beta_1 = 0, \theta = 1.5 \]
Fig. S3

(a) Partial $R^2$ for $\beta_1$

(b) Partial $R^2$ for $\theta$

(c) Total $R^2$

(d) Standardized SD of $R^2$

(e) Standardized SD of $R^2$

(f) Standardized SD of $R^2$
Fig. S4

(a) Partial $R^2$ for $\beta_1$

(b) Partial $R^2$ for $\theta$

(c) Total $R^2$

(d) Residuals

(e) Likelihood

(f) Predicted values

(g) Partial $R^2$ for $\beta_1$ vs. LLR

(h) Partial $R^2$ for $\theta$ vs. LLR

(i) Total $R^2$ vs. LLR
Fig. S5

\[ \beta_1=1, \quad \theta=0.5 \]

\[ \beta_1=1, \quad \theta=0 \]

\[ \beta_1=0, \quad \theta=0.5 \]
Fig. S6

(a) Partial $R^2$ for $\beta_1$

(b) Partial $R^2$ for $\theta$

(c) Total $R^2$

(d) Standardized SD $R^2$

(e) Standardized SD $R^2$

(f) Standardized SD $R^2$
Fig. S7

(a) Partial $R^2$ for $\beta_1$

(b) Partial $R^2$ for $\theta$

(c) Total $R^2$

(d) $R^2$

(e) $R^2$

(f) $R^2$

(g) $R^2$

(h) $R^2$

(i) $R^2$

Legend:
- x resid
- o lik
- * pred
- △ glmm
- □ ols

$\theta$ vs. LLR

$\beta_1$ vs. LLR
Fig. S8

$\beta_1 = 1.8, \theta = 1.8$

$\beta_1 = 1.8, \theta = 0$

$\beta_1 = 0, \theta = 1.8$
Fig. S9

**Partial $R^2$ for $\beta_1$**

- **(a)** Residuals (x), Likelihood (o), Predicted (▲), and GLMM (▲)
- **(b)** Mean $R^2$ for $\theta$
- **(c)** Total $R^2$

**Partial $R^2$ for $\theta$**

**Standardized SD $R^2$**

- **(d)** (e)** (f)** Standardized SD $R^2$ for different sample sizes.
Fig. S10

Partial $R^2$ for $\beta_1$

Partial $R^2$ for $\theta$

Total $R^2$

(d) resid

(e) lik

(f) pred

Residual Likelihood

Total Likelihood

(h) resid

(i) lik

Residual Likelihood

Total Likelihood
Fig. S11

- $\beta_1 = 1.5, \theta = 2$
- $\beta_1 = 1.5, \theta = 0$
- $\beta_1 = 0, \theta = 2$
Fig. S12

Partial $R^2$ for $\beta_1$

Partial $R^2$ for $\theta$

$R^2$

Standardized SD $R^2$

Sample size

Sample size

Sample size
library(lme4)
library(MuMIn)
library(rr2)

inv.logit <- function(x) 1/(1+exp(-x))

# find correction factor for s2d that minimizes the difference between logit and Gaussian cumulative distribution functions
optim(par = .85, function(s) {
  x <- .001*(-5000:5000)
  SS <- mean((plogis(x)-pnorm(x, sd=s^.5*pi/3^.5))^2)
  return(SS)
}, method="BFGS")
  # [1] 0.8768809

n <- 1000
p <- 10

d <- data.frame(x=0, y=0, u=rep(1:p, each=n/p))
d$u <- as.factor(d$u)
bl <- 0
sd1 <- 1

nreps <- 100
w <- data.frame(rep=1:nreps)

for(i in 1:nreps){
  d$x <- rnorm(n=n)

  # simulate data from a logit model
  d$prob <- inv.logit(bl * d$x + rep(rnorm(n=p, sd=sd1), each=n/p))

  # simulate data from a probit model
  #d$prob <- pnorm(bl * d$x + rep(rnorm(n=p, sd=sd1), each=n/p))

  d$y <- rbinom(n=n, size=1, prob=d$prob)

  # analyze with a logit link function
  z <- glmer(y ~ 1 + (1|u), data=d, family=binomial)
  X <- model.matrix(z)
  mu <- fitted(z)
  Yhat <- X %*% lme4::fixef(z)
  s2.logit <- VarCorr(z)[[1]][1]
  s2w.logit <- exp(mean(log(1/(mu*(1-mu)))))
  s2d <- pi^2/3
  s2d.r <- 0.8768809 * pi^2/3

  w$s2.logit[i] <- s2.logit
  w$s2w.logit[i] <- s2w.logit
  w$s2d[i] <- s2d
  w$s2d.r[i] <- s2d.r
  w$R2.logit.NS[i] <- 1 - s2d/(var(Yhat) + s2.logit + s2d)
w$R2.logit.rNS[i] <- 1 - s2d.r/(var(Yhat) + s2.logit + s2d.r)
w$R2.logit.deltaNS[i] <- r.squaredGLMM(z)[2,2]
w$R2.logit.w[i] <- 1 - s2w.logit/(var(Yhat) + s2.logit + s2w.logit)

# analyze with a probit link function
z.probit <- glmer(y ~ 1 + (1|u), data=d, family=binomial(link="probit"))
mu.probit <- fitted(z.probit)
Yhat.probit <- X %*% lme4::fixef(z.probit)
s2.probit <- VarCorr(z.probit)[[1]][1]
s2w.probit <- exp(mean(log(mu.probit)*(1-mu.probit)/dnorm(qnorm(mu.probit))^2)))

w$s2.probit[i] <- s2.probit
w$s2w.probit[i] <- s2w.probit
w$R2.probit.NS[i] <- 1 - 1/(var(Yhat.probit) + s2.probit + 1)
w$R2.probit.deltaNS[i] <- r.squaredGLMM(z.probit)[2,2]
w$R2.probit.w[i] <- 1 - s2w.probit/(var(Yhat.probit) + s2.probit + s2w.probit)

} # Fig. S1
par(mfrow=c(1,3))
xlim <- c(0,.8)
ylim <- ylim
plot(w$s2.probit, w$s2.logit/w$s2d, xlim=xlim, ylim=ylim, xlab="s2[probit]", ylab="s2[logit]/s2d.NS")
lines(c(0,10),c(0,10), col="red")

plot(w$s2.probit, w$s2.logit/w$s2d.r, xlim=xlim, ylim=ylim, xlab="s2[probit]", ylab="s2[logit]/s2d.rNS")
lines(c(0,10),c(0,10), col="red")
xlim <- c(0,4)
ylim <- ylim
plot(w$s2.probit/w$s2w.probit, w$s2.logit/w$s2w.logit, xlim=xlim, ylim=ylim, xlab="s2[probit]/s2w[probit]", ylab="s2[logit]/s2w[logit]"
lines(c(0,10),c(0,10), col="red")

# Fig. S2
par(mfrow=c(2,2))
xlim <- c(0,.45)
ylim <- ylim
plot(w$R2.probit.NS, w$R2.logit.NS, xlim=xlim, ylim=ylim, xlab="R2.NS[probit]", ylab="R2.NS[logit]"
lines(c(0,10),c(0,10), col="red")

plot(w$R2.probit.deltaNS, w$R2.logit.deltaNS, xlim=xlim, ylim=ylim, xlab="R2.deltaNS[probit]", ylab="R2.deltaNS[logit]"
lines(c(0,10),c(0,10), col="red")

plot(w$R2.probit.NS, w$R2.logit.rNS, xlim=xlim, ylim=ylim, xlab="R2.NS[probit]", ylab="R2.rNS[logit]"
lines(c(0,10),c(0,10), col="red")
plot(w$R2.probit.w, w$R2.logit.w, xlim=xlim, ylim=ylim, xlab="R2.w[probit]", ylab="R2.w[logit]")
lines(c(0,10),c(0,10), col="red")