Coordinated Interaction

This document provides a simple simulation framework for exploring Coordinated Interaction and the Even-Odd test.

Simulating polygenic phenotypes

We let N, M, h^2 be the number of individuals, SNPs, and additive heritability of the phenotype:

```
N   <- 5000 #number of individuals
M   <- 100 #number of SNPs
h2  <- 0.4 #heritabilty
h2x <- 0.2 #heritability due to epistasis
```

We draw allele frequencies from a uniform distribution, and draw SNPs from corresponding binomial distributions:

```
set.seed(123)
freqs = runif(M,0.05,0.95)
genos = matrix(nrow = N, ncol = M, rbinom(N*M,2,freqs),byrow=T)
genos = scale(genos) # scale genotypes to mean 0 and variance 1, for simplicity
```

Additive Model

We begin with a simple additive phenotypic model, where the genetic contribution is simply the product of the scaled genotypes and their additive effect sizes:

```
betas = rnorm(M,0,sqrt(h2/M)) #marginal SNP effects
y_add = genos %*% betas + rnorm(N,0,sqrt(1-h2))
```

Isotropic Interaction Model

We call a polygenic epistasis model isotropic if the pairwise SNP interaction effects are i.i.d. and mean 0. As an example, we draw K causal SNP pairs (uniformly at random) with i.i.d. Gaussian effects, with variance chosen such that the pairwise epistasis heritability is h^2_{iso}:

```
K <- .05*M^2 # 5% of all SNP pairs interact
omega_iso <- rnorm(K,0,sqrt(h2x/K))
causpairs <- sapply( 1:K, function(k) sample(M,2) )
gprods    <- sapply( 1:K, function(k) genos[,causpairs[1,k]]*genos[,causpairs[2,k]] )
y_iso = genos %*% betas + gprods %*% omega_iso + rnorm(N,0,sqrt(1-h2-h2x))
```

Here, we define $\omega_{iso} \in \mathbb{R}^K$ as the vector of pairwise epistasis effects for the K causal epistatic SNP pairs.

Coordinated Interaction Model

The Coordinated Interaction, γ, is the correlation among pairwise interactions and products of main effects:

$$\gamma := \text{Corr}(\beta_s \beta'_s, \Omega_{ss'})$$

where $s \neq s'$ index SNPs and $\Omega \in \mathbb{R}^{M \times M}$ is the matrix of all pairwise interaction effects.

There are many plausible models that induce CI (ie $\gamma \neq 0$), which we discuss in our Supplement. Here, we simulate CI in a simple, direct way:

```
y_ci = genos %*% betas +  sqrt(h2x/h2^2) * ( genos %*% betas * genos %*% betas ) + rnorm(N,0,sqrt(1-h2-
```

Evaluating Coordinated Interaction

To calculate CI for a particular polygenic model, we need to know the main effects β and interaction effects in Ω. Throughout this document, β is fixed, but the different models correspond to different choices for Ω.

Under the additive model, there is no epistasis ($\Omega_{add} = 0$), so γ is formally set to 0.

Our sparse isotropic epistasis model implicitly defines the nonzero entries in Ω_{iso} via the vector ω_{iso}:

```
Omega_iso <- matrix(0,M,M)
for(k in 1:K)
  Omega_iso[causpairs[1,k],causpairs[2,k]] <- omega_iso[k]
(gamma_iso = cor( betas %x% betas, c(Omega_iso) ))
```

```
## [1] -0.009186699
```

The CI is roughly 0, as expected since Ω_{iso} and β are independent. This clearly demonstrates that the mere existence of polygenic epistasis is insufficient to induce CI.

Likewise, our simulation model for CI implicitly defines Ω_{ci}:

```
Omega_ci <-  sqrt(h2x) * (betas %o% betas)/sum(betas^2)
(gamma_ci <- cor( betas %x% betas, c(Omega_ci) ))
```

```
## [1] 1
```

The CI is 1, as expected because the epistasis effects are perfectly coordinated with main effects.

Finally, we can mix together isotropic and coordinated effects to get γ values in between 0 and 1:

```
(gamma_mix <- cor( betas %x% betas, .5*c(Omega_iso) + .5*c(Omega_ci) )^2 )
```

```
## [1] 0.5211547
```

Estimating CI: the Even-Odd Test

We propose the Even-Odd test to specifically identify CI by testing the interaction between polygenic risk scores constructed from two independent parts of the genomes (eg even- and odd-indexed chromosomes):

```
EOtest  <- function( betas, genos, y, E=1:(length(betas)/2), O=setdiff( 1:length(betas), E ) ){
  PE    <- scale(genos[,E] %*% betas[E])
  PO    <- scale(genos[,O] %*% betas[O])
  out <- summary( lm( y ~ PE * PO   ) )$coef['PE:PO',c(1,2,4)]
  out[1] <- out[1] / sqrt( h2x )
  names(out) <- c( 'gammaEO', 'se', 'pv' )
  out
}
```

The estimated interaction term, $\hat{\gamma}_{EO}$, is an unbiased estimate of γ (under regularity conditions). Note that the unbiased estimate requires knowledge of h^2 and h^2_{CI}. This can be estimated with mixed models, or it can be ignored if only the sign and significance of γ are of interest.

As explained above, the true CI is 0 under additive and isotropic interaction models. Therefore, the Even-Odd test should be null:

```
EOtest( betas, genos, y_add )
```

```
##    gammaEO         se         pv
## 0.01956703 0.01116437 0.43319410
```

```
EOtest( betas, genos, y_iso )
```

```
##     gammaEO          se          pv
## -0.01753861  0.01103016  0.47705807
```

Under our simulated CI model, where $\gamma = 1$, the $\hat{\gamma}_{EO}$ is:

```
EOtest( betas, genos, y_ci )
```

```
##    gammaEO         se         pv
## 1.12742584 0.01190553 0.00000000
```

Finally, we can mix together both isotropic and coordinated effects to get γ values in between 0 and 1:

```
y_mix <- .5*y_iso + .5*y_ci
EOtest( betas, genos, y_mix )
```

```
##       gammaEO            se            pv
##  5.549436e-01  8.217041e-03 3.303525e-184
```

We get similar answers if we use feasible estimates of β (built out-of-sample) in place of the true β:

```
betahatfxn <- function(x,y) 1/length(y) * t(x) %*% y
trn <- 1:(N/2)     # estimate genetic effects with first half of samples
tst <- N/2+1:(N/2) # estimate CI with second half
EOtest( betahatfxn(genos[trn,],y_add[trn]), genos[tst,], y_add[tst] )
```

```
##      gammaEO           se           pv
## -0.001814886  0.016469142  0.960698045
```

```
EOtest( betahatfxn(genos[trn,],y_iso[trn]), genos[tst,], y_iso[tst] )
```

```
##    gammaEO         se         pv
## 0.01426394 0.01577000 0.68587672
```

```
EOtest( betahatfxn(genos[trn,],y_ci [trn]), genos[tst,], y_ci [tst] )
```

```
##       gammaEO            se            pv
##  9.505514e-01  1.725730e-02 4.091499e-120
```

```
EOtest( betahatfxn(genos[trn,],y_mix[trn]), genos[tst,], y_mix[tst] )
```

```
##      gammaEO           se           pv
## 5.157179e-01 1.170932e-02 2.146613e-80
```

These feasible estimates of γ are deflated toward 0 because the noise in $\hat{\beta}$ is not coordinated with the epistasis effects. Nonetheless, the tests remain well-powered to detect CI.

Testing CI in other epistasis models

Third-Order Isotropic Interaction

We call a polygenic epistasis model isotropic if the pairwise SNP interaction effects are i.i.d. and mean 0. As an example, we draw K causal SNP pairs (uniformly at random) with i.i.d. Gaussian effects, with variance chosen such that the pairwise epistasis heritability is h^2_{iso}:

```
Ktrip <- .005*M^3 # .5% of all SNP triples interact
omega_trip <- rnorm(Ktrip,0,sqrt(h2x/Ktrip))
caustrips <- sapply( 1:Ktrip, function(k) sample(M,3) )

gprods <- sapply( 1:Ktrip, function(k) genos[,caustrips[1,k]]*genos[,caustrips[2,k]]*genos[,caustrips[3
y_trip = genos %*% betas + gprods %*% omega_trip + rnorm(N,0,sqrt(1-h2-h2x))
```

Under the triple model, there is no pairwse epistasis ($\Omega_{add} = 0$), so γ is formally set to 0. And, as expected, both γ_{EO} and $\hat{\gamma}_{EO}$ are roughly zero:

```
EOtest( betas, genos, y_trip )
```

```
##      gammaEO           se           pv
## -0.008018333  0.011281087  0.750597623
```

```
EOtest( betahatfxn(genos[trn,],y_trip[trn]), genos[tst,], y_trip[tst] )
```

```
##     gammaEO          se          pv
## 0.008680758 0.016321253 0.812010168
```

Polygenic Dominance Effects

Our polygenic dominance model adds a quadratic effect to each causal SNP:

```
gdom = (genos^2) %*% (betas^2)
y_dom = genos %*% betas + sqrt(h2x)*scale(gdom) + rnorm(N,0,sqrt(1-h2-h2x))
```

Under this model, there is no inter-SNP epistasis (Ω_{dom} is diagonal), so γ is formally 0. As expected, both γ_{EO} and $\hat{\gamma}_{EO}$ are near zero:

```
EOtest( betas, genos, y_dom  )
```

```
##     gammaEO          se          pv
## -0.04145607  0.01108423  0.09446452
```

```
EOtest( betahatfxn(genos[trn,],y_dom[trn]), genos[tst,], y_dom[tst] )
```

```
##     gammaEO          se          pv
## -0.04357146  0.01431839  0.17367136
```

Unlike the Even-Odd test, the direct interaction test for the PRS is positive under this dominance model:

```
EOtest( betas, genos, y_dom, E=1:M, O=1:M )
```

```
##      gammaEO           se           pv
## 1.070838e-01 7.722771e-03 6.059801e-10
```

This direct test provides a true positive result for epistasis in the sense that this model has true nonlinear genetic effects. However, it is a false positive for CI, which focuses specifically on interactions between SNPs.

Pathway-Level Interaction

Our pathway-level interaction model assumes that two additive pathways, P_1 and P_2, as well as their interaction:

```
betas1 <- rnorm(M/2,0,sqrt(h2/M)) #marginal SNP effects via pathway 1
betas2 <- rnorm(M/2,0,sqrt(h2/M)) #marginal SNP effects via pathway 2
P1 <- c(genos[,1:(M/2)*2-1] %*% betas1)
P2 <- c(genos[,1:(M/2)*2  ] %*% betas2)
```

```
h2xpath<- .1
y1 = P1 + P2 + P1*P1 * sqrt(h2xpath/var(P1*P1)) + rnorm(N,0,sqrt(1-h2-h2xpath))
y2 = P1 + P2 + P1*P2 * sqrt(h2xpath/var(P1*P2)) + rnorm(N,0,sqrt(1-h2-h2xpath))
```

The two phenotypes, y_1 and y_2, are driven by biologically distinct types of epistasis. The first phenotype, y_1, is driven by the intra-pathway interaction of pathway 1 with itself. The second phenotype, y_2, is instead driven by the inter-pathway interaction between the two independent genetic risk pathways. We derive the CI induced by inter- and inter-pathway interaction in our Supplementary Material.

Evaluating the Even-Odd test

The Even-Odd test detects CI from both inter- and intra-pathway interaction. In our Supplement (Example 3), we show the expected CI is roughtly 1/2 for both y_1 and y_2, which roughly holds both for the oracle PRS built from β:

```
EOtest( c(rbind(betas1,betas2)), genos, y1 )
```

```
##      gammaEO           se           pv
## 3.181085e-01 1.070355e-02 1.219177e-39
```

```
EOtest( c(rbind(betas1,betas2)), genos, y2 )
```

```
##      gammaEO           se           pv
## 3.451569e-01 1.058588e-02 3.390176e-47
```

The feasible PRS built from $\hat{\beta}_{OLS}$ also yields signicant Even-Odd tests, albeit with deflated estimates of γ, as expected:

```
EOtest( betahatfxn(genos[trn,],y1[trn]), genos[tst,], y1[tst] )
```

```
##      gammaEO           se           pv
## 2.508923e-01 1.582981e-02 1.762879e-12
```

```
EOtest( betahatfxn(genos[trn,],y2[trn]), genos[tst,], y2[tst] )
```

```
##      gammaEO           se           pv
## 2.980100e-01 1.507639e-02 1.770752e-18
```

Assessing phenotype rescaling

In the case of intra-pathway interaction, e.g. y_1, the CI may appear to be merely a phenotype scaling issue. However, we show in the Supplement (Section 5.2) that the sign and significance of CI are unchanged by rescaling the phenotype (under regularity conditions). For example, the CI persists if we quantile normalize the phenotype y_1:

```
q1 = as.numeric(quantnorm(as.matrix(y1)))
EOtest( betahatfxn(genos[trn,],q1[trn]), genos[tst,], q1[tst] )
```

```
##      gammaEO           se           pv
## 1.770436e-01 1.642622e-02 1.520848e-06
```

The Even-Odd test for different chromsome splits

The EO test power increases when the (arbitrary) chromosome partition used with the causal pathway partition:

```
EOtest( c(rbind(betas1,betas2)), genos, y2, E=1:(M/2)*2-1, O=1:(M/2)*2 )
```

```
##       gammaEO            se            pv
##  6.772892e-01  9.964275e-03 2.232772e-186
```

Additionally, power to detect CI vanishes completely when the EO partition perfectly misses the causal partition:

```
EOtest( c(rbind(betas1,betas2)), genos, y1, E=1:(M/2)*2-1, O=1:(M/2)*2 )
```

```
##     gammaEO          se          pv
## 0.077084888 0.010926390 0.001614182
```

Nonetheless, random choices for E and O give similar answers for large enough M. But large M also makes PRS estimates noisier, reducing $\hat{\gamma}$:

```
N <- 1e4
Mlist <- c( 3e1, 1e2, 3e2 )
nit <- 1e3
gams <- array( NA, dim=c(length(Mlist),nit) )
for( M in Mlist ){
  genos = scale(matrix(rbinom(N*M,2,runif(M,0.05,0.95)),N,M,byrow=T))
  betas <- rnorm(M,0,sqrt(h2/M))
  P1 <- c(genos[,1:(M/2)*2-1] %*% betas[1:(M/2)*2-1])
  P2 <- c(genos[,1:(M/2)*2  ] %*% betas[1:(M/2)*2  ])
  y = P1 + P2 + P1*P2 * sqrt(h2x/var(P1*P2)) + rnorm(N,0,sqrt(1-h2-h2x))
  for( i in 1:nit )
  gams[which(M==Mlist),i] <- EOtest( betahatfxn(genos[trn,],y[trn]), genos[tst,], y[tst], E=sample(M,M/
}
xs <- seq( min(gams, na.rm=T), max(gams, na.rm=T), len=31)
ys <- apply( gams, 1, function(x) hist( x, breaks=xs, plot=F )$counts )
plot(range(xs),range(ys,na.rm=T),type='n',xlab=expression(hat(gamma)['EO']),ylab='Count')
for( k in 1:length(Mlist) )
  lines((xs[-1]+xs[-length(xs)])/2,ys[,k],col=k)
legend( 'topright', bty='n', fill=1:length(Mlist), paste0( 'M=', Mlist ) )
```