

Genome-wide LD change as a signature of selection in experimental populations

Moi Exposito-Alonso

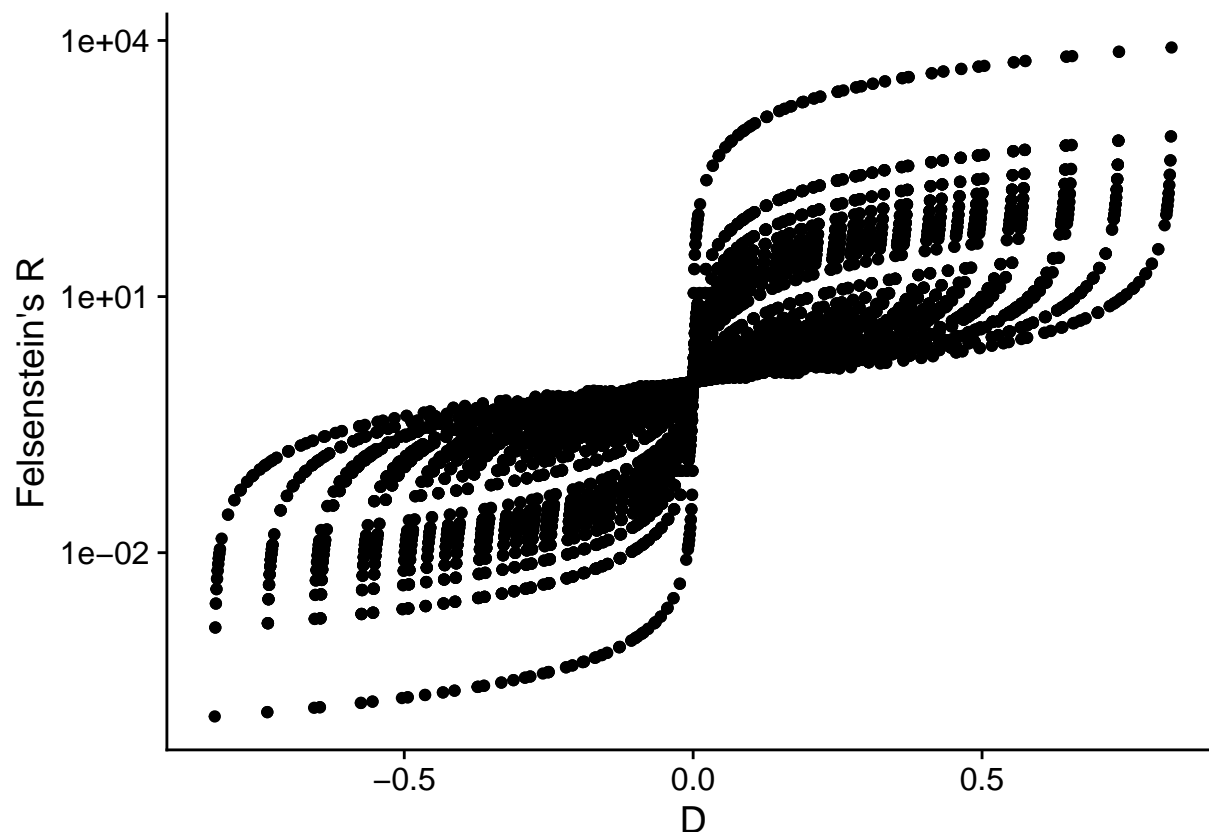
	B	b
A	$(1 + s_A)(1 + s_B)$	$(1 + s_A)$
a	$(1 + s_B)$	1

The D and R do not have an 1 to 1 relationship

```
x1<-seq(0.01,0.99,0.1)
x2<-seq(0.01,0.99,0.1)
x3<-seq(0.01,0.99,0.1)
x4<-seq(0.01,0.99,0.1)

allxs<-expand.grid(x1,x2,x3,x4)

ggdotscolor(getD(allxs),getR(allxs), xlab='D',ylab="Felsenstein's R") + scale_y_log10()
```



LD (r^2 , D nor R) does not change with multiplicative selection without epistasis

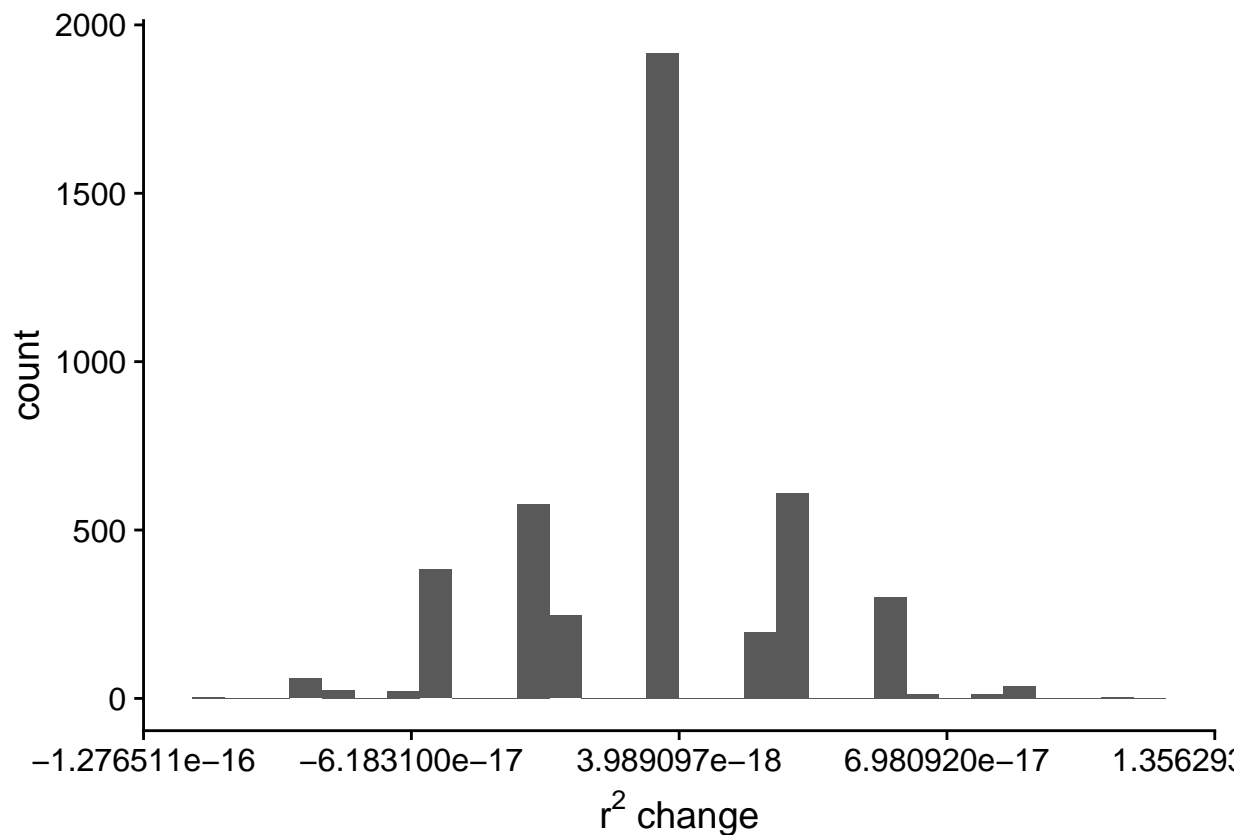
```
s<-seq(-0.5,+0.5,0.05)
t<-seq(-0.5,+0.5,0.05)
E<-0

ts<-expand.grid(t,s,E)

w1<-(1+ts[,1]+ts[,2])
w1<-(1+ts[,1])*(1+ts[,2])
w1<-(1+ts[,1])*(1+ts[,2]) *(1+ts[,3])
w2<-(1+ts[,2])
w3<-(1+ts[,1])
w4<-rep(1,nrow(ts))
allw<-cbind(w1,w2,w3,w4)

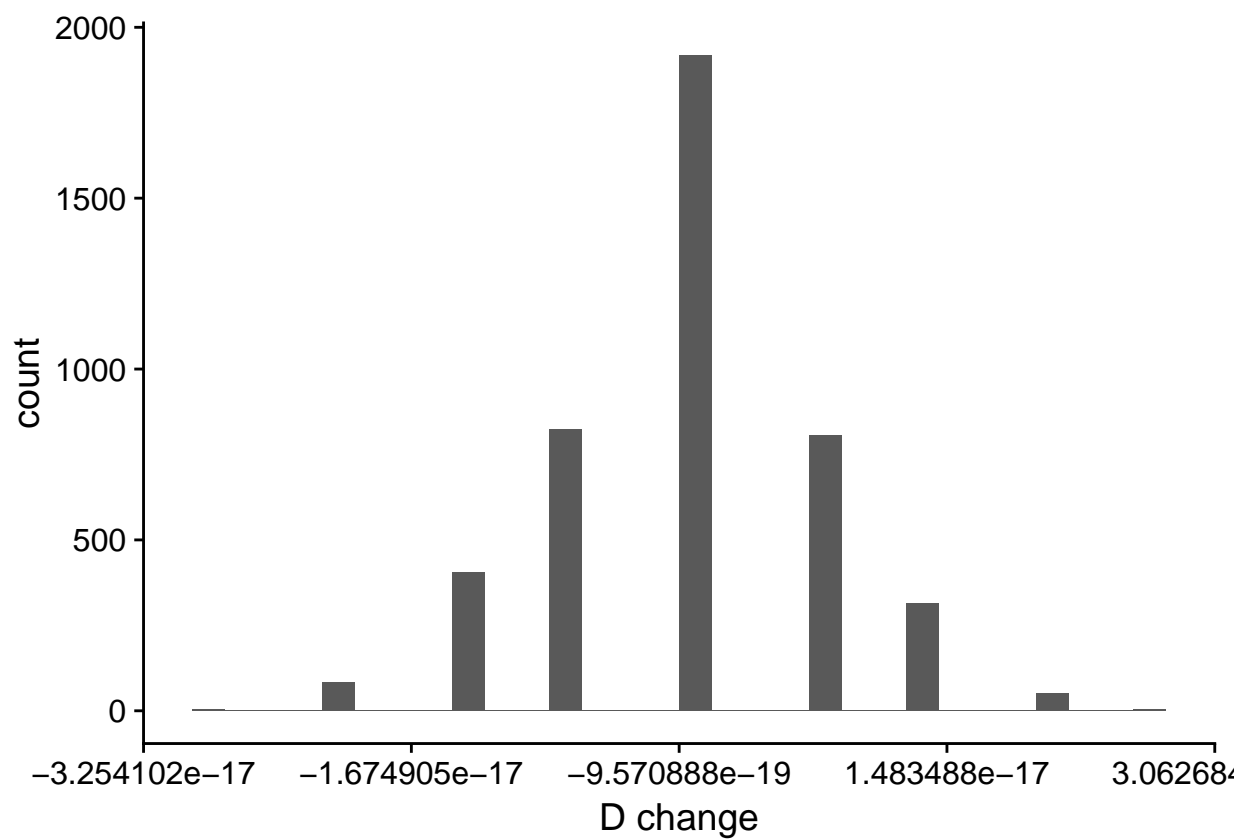
qplot(x=fn(getR2change(allw)), xlab=TeX('r^2 change'))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



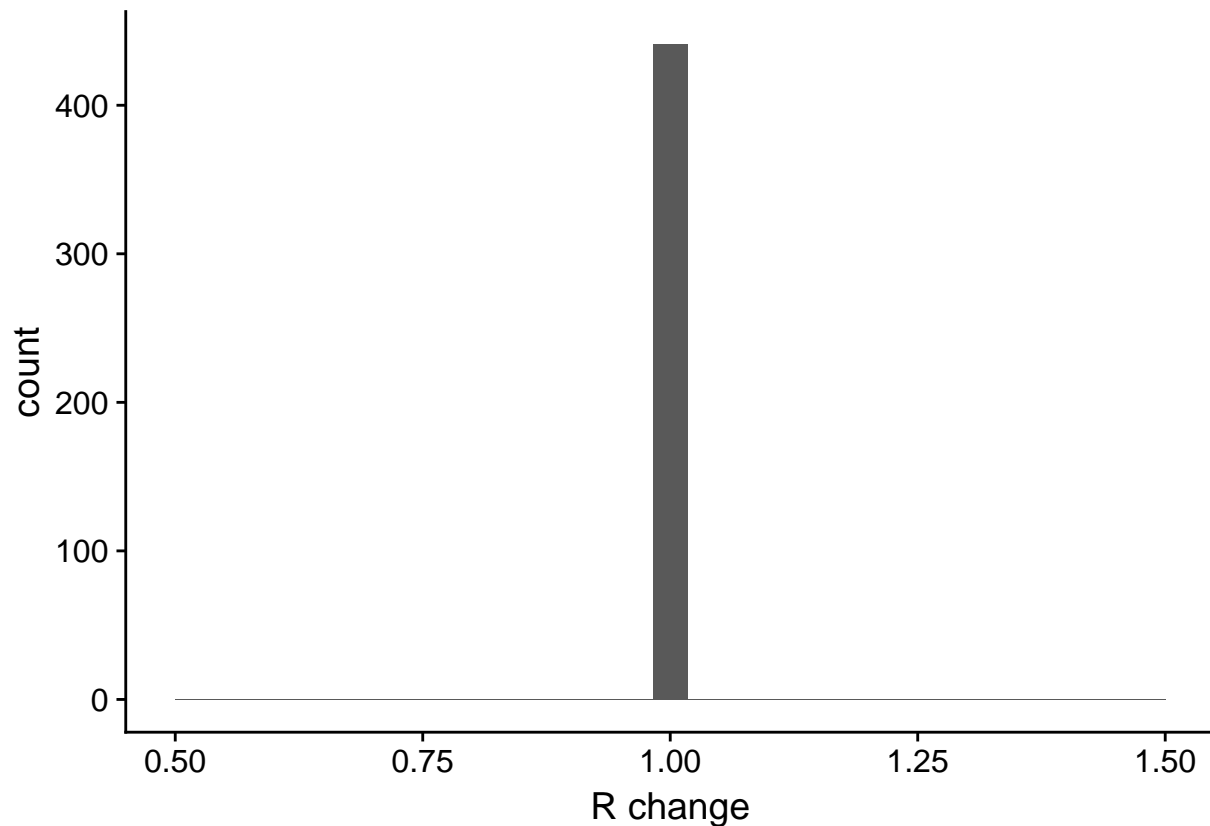
```
qplot(x=fn(getDchange(allw)), xlab=TeX('D change'))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
qplot(x=fn(getRchange(allw)), xlab=TeX('R change'))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



LD does change with multiplicative selection and multiplicative epistasis

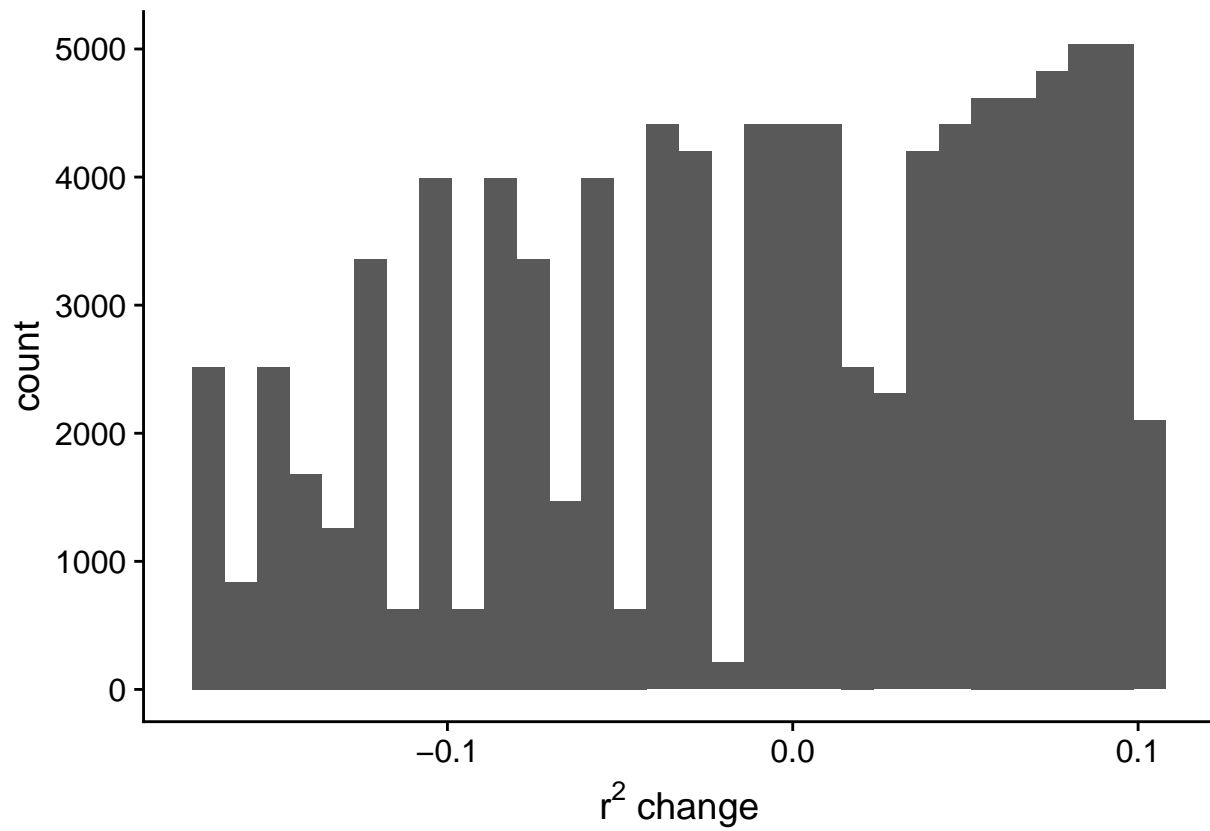
```
s<-seq(-0.5,+0.5,0.05)
t<-seq(-0.5,+0.5,0.05)
E<-seq(-0.5,+0.5,0.05)

ts<-expand.grid(t,s,E)

w1<-(1+ts[,1]+ts[,2])
w1<-(1+ts[,1])*(1+ts[,2])
w1<-(1+ts[,1])*(1+ts[,2]) *(1+ts[,3])
w2<-(1+ts[,2])
w3<-(1+ts[,1])
w4<-rep(1,nrow(ts))
allw<-cbind(w1,w2,w3,w4)

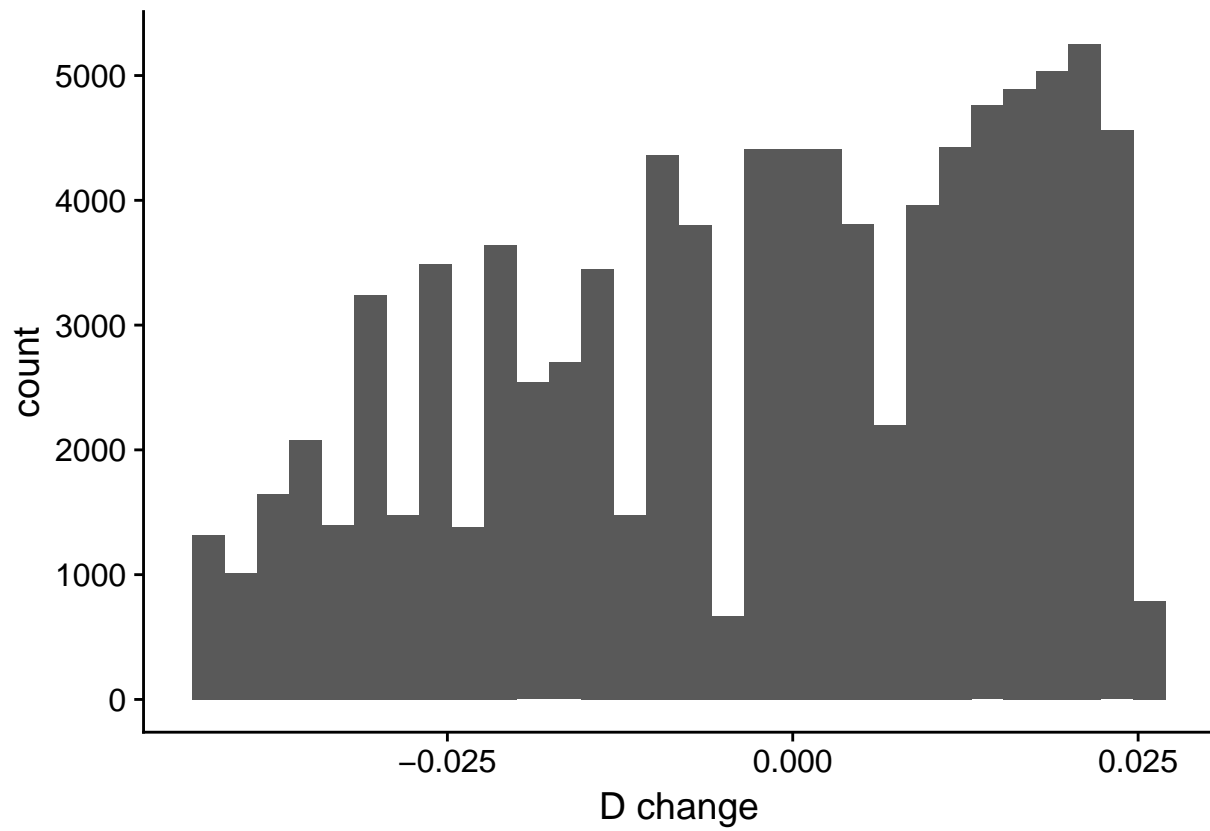
qplot(x=fn(getR2change(allw)), xlab=TeX('r^2 change'))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



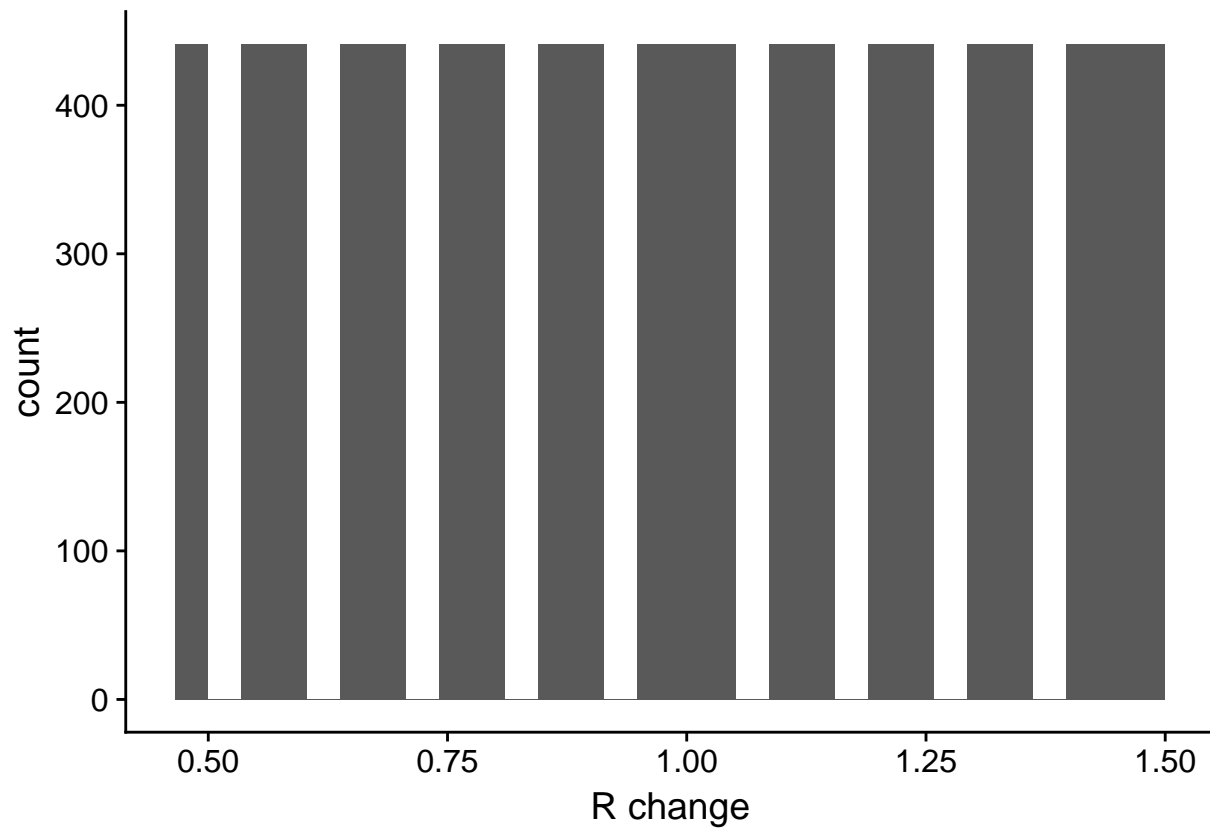
```
qplot(x=fn(getDchange(allw)), xlab=TeX('D change'))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
qplot(x=fn(getRchange(allw)), xlab=TeX('R change'))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



LD does change with multiplicative selection and power epistasis

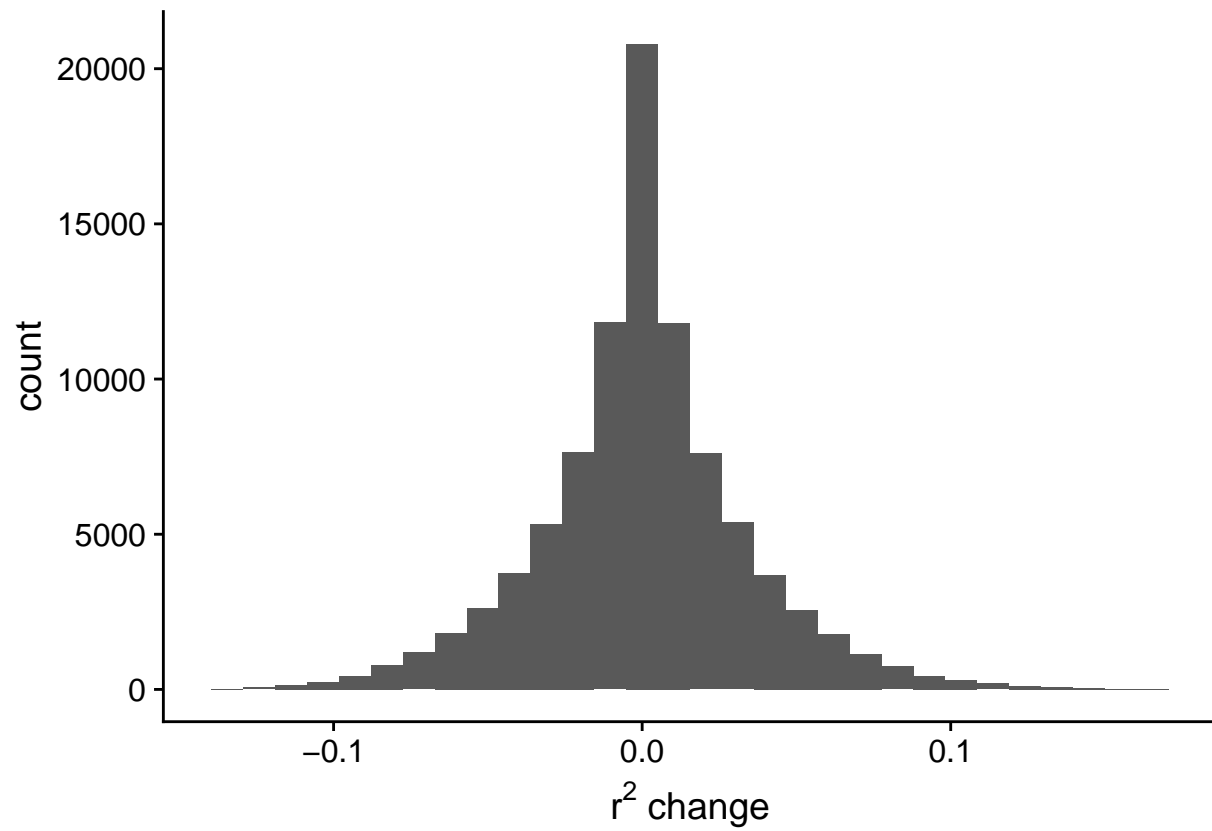
```
s<-seq(-0.5,+0.5,0.05)
t<-seq(-0.5,+0.5,0.05)
E<-seq(-0.5,+0.5,0.05)

ts<-expand.grid(t,s,E)

w1<-((1+ts[,1])*(1+ts[,2]))^(1+ts[,3])
w2<-(1+ts[,2])
w3<-(1+ts[,1])
w4<-rep(1,nrow(ts))
allw<-cbind(w1,w2,w3,w4)

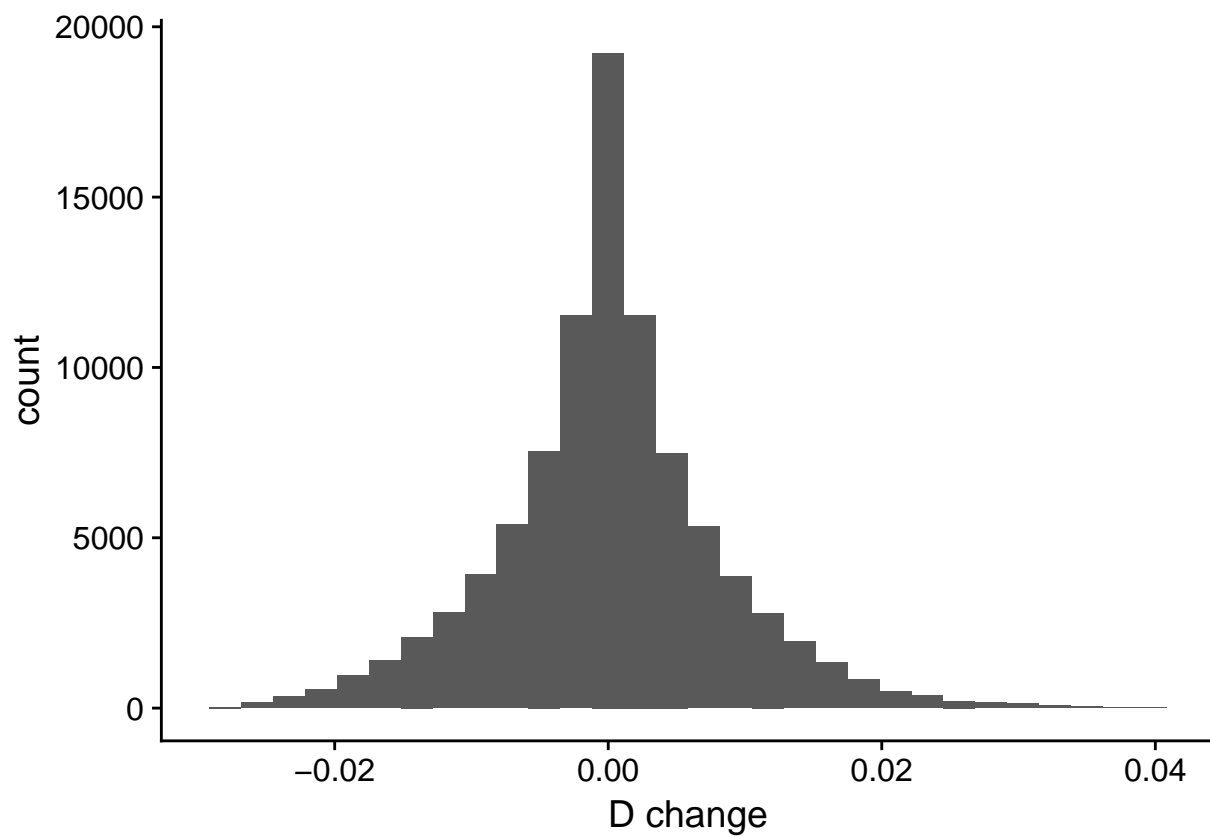
qplot(x=fn(getR2change(allw)), xlab=TeX('r^2 change'))

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



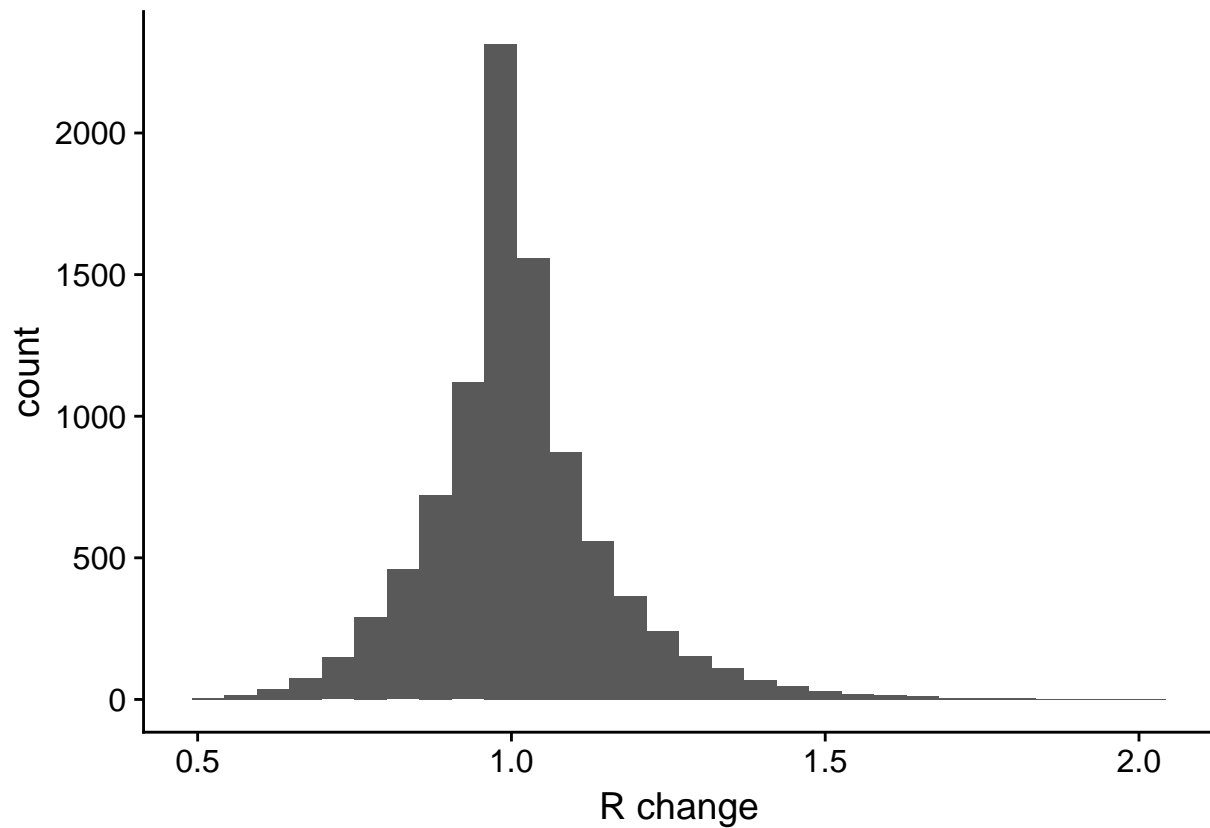
```
qplot(x=fn(getDchange(allw)), xlab=TeX('D change'))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
qplot(x=fn(getRchange(allw)), xlab=TeX('R change'))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Does LD change with a 3rd locus under selection?

```
Dc<-0.02
pc<-0.1
sc<-0.8

s<-seq(-0.5,+0.5,0.05)
t<-seq(-0.5,+0.5,0.05)
E<-seq(-0.5,+0.5,0.05)
E<-0

ts<-expand.grid(t,s,E)

w1<-(((1+ts[,1])*(1+ts[,2]))^(1+ts[,3])) [1]
w2<-(1+ts[,2]) [1]
w3<-(1+ts[,1]) [1]
w4<-rep(1,nrow(ts)) [1]
allw<-cbind(w1,w2,w3,w4)

a=0.2
b=0.3
c=0.6

Dab<-0.05
Dbc<-0.03
```

```

Dac<-0.02

freq3<-function(G='ABC',a,b,c,Dab,Dac,Dbc){

  a*b*c * (1+Dab+Dac+Dbc - (Dab*Dac)-(Dab*Dbc)-(Dac*Dbc) +(Dab*Dac*Dbc))

}
ABC<- a*b*c + (Dab+Dac+Dbc - (Dab*Dac)-(Dab*Dbc)-(Dac*Dbc) +(Dab*Dac*Dbc))
aBC<- (1-a)*b*c + (-Dab+Dac+Dbc - (Dab*Dac)-(Dab*Dbc)-(Dac*Dbc) +(Dab*Dac*Dbc))
AbC<- a*(1-b)*c * (1+Dab)
abC<- (1-a)*(1-b)*c * (1+Dab)
ABc<- a*b*(1-c) * (1+Dab)
aBc<- (1-a)*b*(1-c) * (1+Dab)
Abc<- a*(1-b)*(1-c) * (1+Dab)
abc<- (1-a)*(1-b)*(1-c) * (1+Dab)

x1C =0.4
x4C =0.4
x2C =0.1
x3C =0.1

e=0

w1<-((1+sa)*(1+sb))^(1+e)
w2<-(1+sb)
w3<-(1+sa)
w4<-1

sc=0.8

w1<- (1-x1C) * w1 + x1C * (w1*(1+sc))
w4<- (1-x4C) * w4 + x4C * (w4*(1+sc))
w2<- (1-x2C) * w2 + x2C * (w2*(1+sc))
w3<- (1-x3C) * w3 + x3C * (w3*(1+sc))

(w1*w4) / (w2*w3)

## [1] 1.493827

# x1p = (x1*pc +Dc) * (w1+sc) + (1-(x1*pc+Dc)) *(w1)
# x2p = (x2*pc -Dc) * (w2+sc) + (1-(x2*pc-Dc)) *(w2)
# x3p = (x3*pc -Dc) * (w3+sc) + (1-(x3*pc-Dc)) *(w3)
# x4p = (x4*pc +Dc) * (w4+sc) + (1-(x4*pc+Dc)) *(w4)
#
# x1p= ()
#
# w_hat=(x1p+x2p+x3p+x4p)
# x1p=x1p/w_hat
# x2p=x2p/w_hat
# x3p=x3p/w_hat
# x4p=x4p/w_hat
#

```

```

# pp<-x1p+x2p
# qp<-x3p+x4p
# pp+qp
# p=x1+x2
# q=x3+x4
#
# ((x1p*x4p) / (x2p*x3p) ) - (x1*x4) / (x2*x3)
# ((x1p*x4p) - (x2p*x3p) ) - (x1*x4) - (x2*x3)
#
# (((x1p*x4p) - (x2p*x3p) )/sqrt(pp*qp*(1-pp)*(1-qp)) ) - ( ((x1*x4) - (x2*x3)) /sqrt(p*q*(1-p)*(1-q)) )
#
#
# x

```

```

X<-get_topSNPs(100,navalue = 0.5)
dim(X)

```

```

## [1] 515 100

```

```

X[,1] <-X[,1] * (-1)

```

```

nsnps=3

```

```

ldfocal<-c()

```

```

Dwith3rd<-c()

```

```

for(i in 1:1000){
  Xs<-X[,sample(1:100,nsnps)]
  Xs[Xs==(-1)]<-0

```

```

  s<-rep(0.5,nsnps)
  # e<- 0.8
  e<- 0

```

```

  haps<-apply(Xs,1,function(i) prod(i))
  w<- apply(1+ s*Xs, 1, function(i){prod(i)})
  w<-w^(1+(e*haps))

```

```

  ldnow_d<-ldCnow(Xs,R = FALSE,dolog = FALSE, addone=FALSE) %>% upperTmat()
  ldnow<-ldCnow(Xs,R = TRUE,dolog = FALSE, addone=FALSE) %>% upperTmat()
  ldnext<-ldCnext(Xs,w,R=TRUE,dolog = FALSE,addone=FALSE) %>% upperTmat()
  ldnext/ldnow
  Dwith3rd<-append(Dwith3rd,mean(ldnow_d[-1],na.rm=TRUE))
  ldfocal<-append(ldfocal,mean((ldnext/ldnow)[1],na.rm=TRUE))

```

```

}

```

```

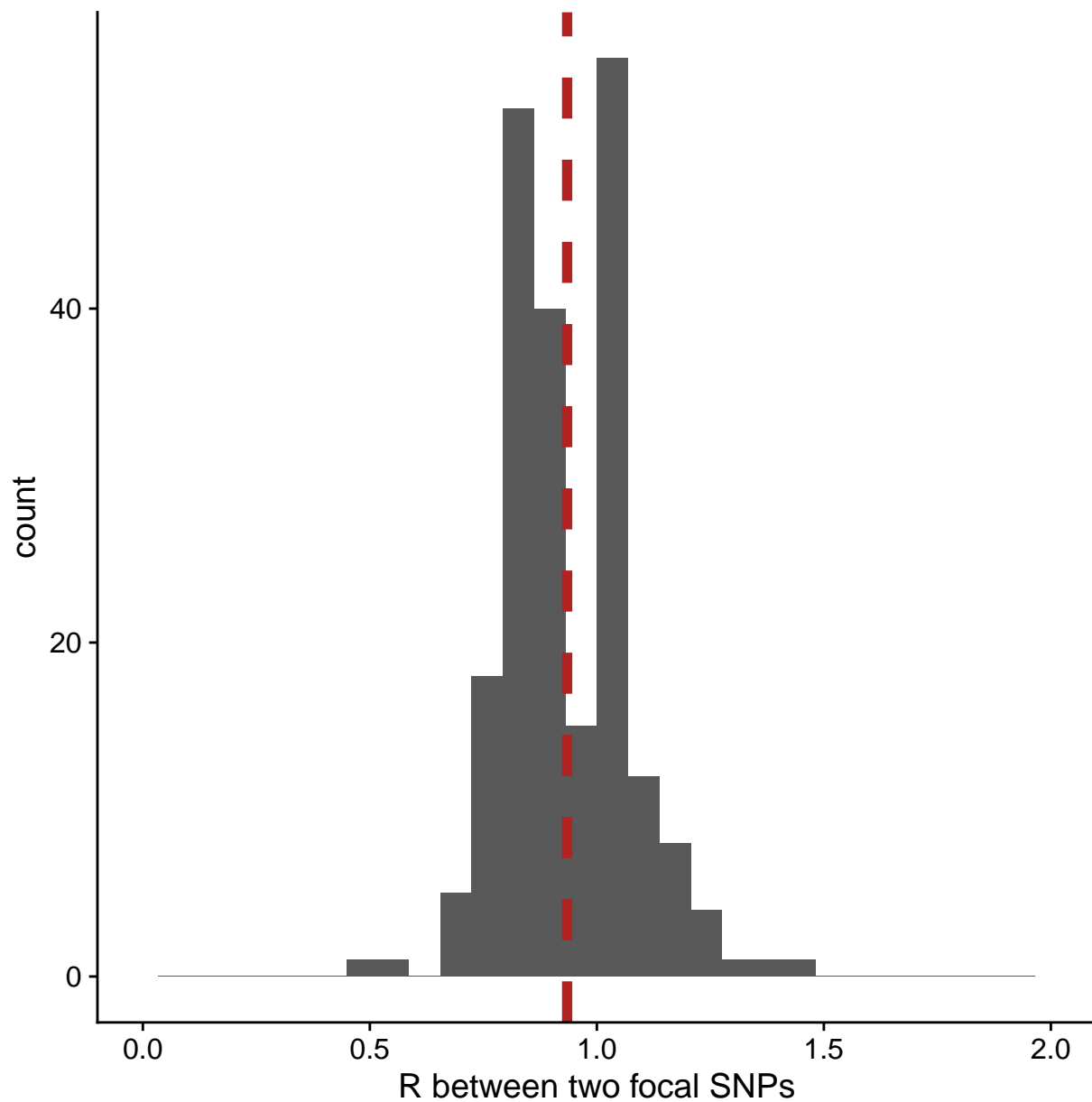
qplot(ldfocal,xlab='R between two focal SNPs',xlim=c(0,2))+
  geom_vline(xintercept = mean(ldfocal,na.rm = T), col='firebrick', lty='dashed',lwd=2)

```

```

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

```



```
# hist(ldfocal)
mean(ldfocal,na.rm = T)

## [1] 0.9345427

for(i in 1:1000){
  Xs<-X[,sample(1:100,nsnps)]
  Xs[Xs==(-1)]<-0

  s<-rep(0.5,nsnps)
  # e<- 0.8
  e<- 1

  haps<-apply(Xs,1,function(i) prod(i))
  w<- apply(1+ s*Xs, 1, function(i){prod(i)})
  w<-w^(1+(e*haps))
}
```

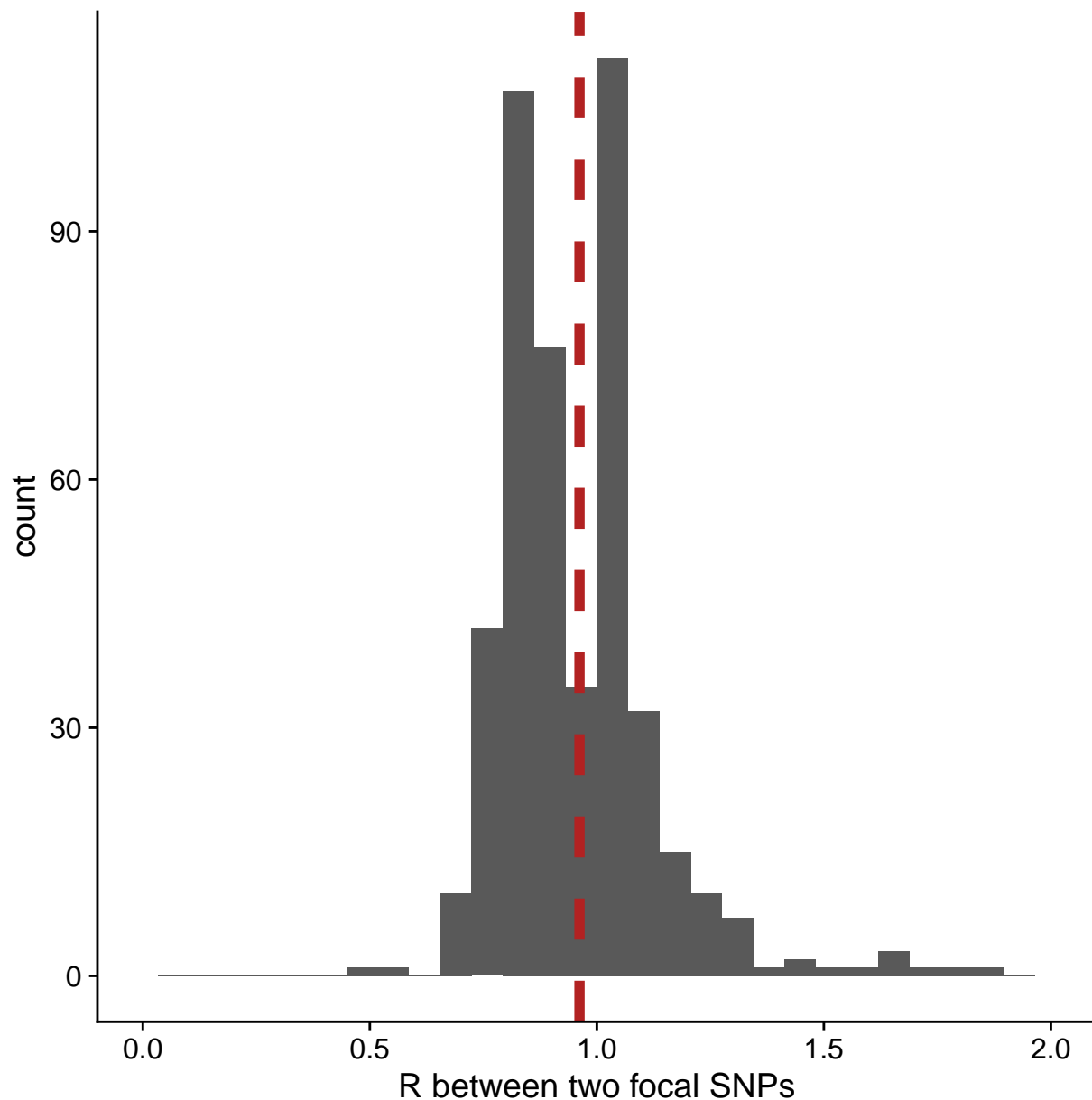
```

ldnow_d<-ldCnow(Xs,R = FALSE,dolog = FALSE, addone=FALSE) %>% upperTmat()
ldnow<-ldCnow(Xs,R = TRUE,dolog = FALSE, addone=FALSE) %>% upperTmat()
ldnext<-ldCnext(Xs,w,R=TRUE,dolog = FALSE,addone=FALSE) %>% upperTmat()
ldnext/ldnow
Dwith3rd<-append(Dwith3rd,mean(ldnow_d[-1],na.rm=TRUE))
ldfocal<-append(ldfocal,mean((ldnext/ldnow)[1],na.rm=TRUE))
}

qplot(ldfocal,xlab='R between two focal SNPs',xlim=c(0,2))+
  geom_vline(xintercept = mean(ldfocal,na.rm = T), col='firebrick', lty='dashed',lwd=2)

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

```



```

# hist(ldfocal)
mean(ldfocal,na.rm = T)

## [1] 0.9617987
# It does not change with 2 SNPs
Xs<-Xs[,1:2]
s<-s[1:2]
w<- apply(1+ (s*Xs) , 1, function(i){prod(i)})
ldnow<-ldCnow(Xs,R = TRUE,dolog = FALSE, addone=FALSE) %>% upperTmat()
ldnext<-ldCnext(Xs,w,R=TRUE,dolog = FALSE,addone=FALSE) %>% upperTmat()
ldnext/ldnow

##      [,1]
## [1,]  NaN

Xs<-Xs[,1:2]
s<-s[1:2]
e<- 0.1
e<- 0
haps<-apply(Xs,1,function(i) prod(i))
w<- apply(1+ s*Xs, 1, function(i){prod(i)})
w<-w^(1+(e*haps))
ldnow<-ldCnow(Xs,R = TRUE,dolog = FALSE, addone=FALSE) %>% upperTmat()
ldnext<-ldCnext(Xs,w,R=TRUE,dolog = FALSE,addone=FALSE) %>% upperTmat()
ldnext/ldnow

##      [,1]
## [1,]  NaN

# qplot(ldfocal,xlab='R between two focal SNPs')+
#   geom_vline(xintercept = mean(ldfocal,na.rm = T), col='firebrick', lty='dashed',lwd=2)

```