Figure 1 for Deleterious Alleles paper

```
#setwd("~/Documents/Github/pvpDiallel/")
knitr::opts_knit$set(root.dir=normalizePath('../'))
```

To reproduce the figures, we should set `pvpDiallel` as the root path, i.e. `setwd("~/Documents/Github/pvpDiallel/")`. And then use `knitr` package to knit a pdf file. Or simply click `Knit PDF` icon in `RStudio`. Note, to generate each panel into seperate pdf files, we should turn `getpdf` into `TRUE` (i.e. `getpdf=TRUE`) when calling the plotting functions.

First of all, determine fond size and set the getpdf option:

```
#par(mar=c(5,4,4,2))
par(font=2, font.lab=2, font.axis=2)
fs <- 1.6 # times bigger than default
ht=4.5; wt=4.5 #figure height and weight
getpdf <- FALSE # get figures in seperated pdf [TRUE] or not [FALSE]
```

Figure 1a

```
plotH2 <- function(getpdf, outfile, ...){
  H2 <- read.csv("data/DIalleleHeritability.csv")
  h <- read.csv("cache/loh_pBPHmax_median.csv")
  H2 <- merge(H2, h, by.x="Traits", by.y="trait")
  H2 <- H2[order(H2$h), ]

  barplot(H2[,4], names.arg = H2$Traits, ...)
  #text(cex=fs, x=x-.25, y=-1.25, H2$Traits, xpd=TRUE, srt=45, pos=2)
  if(getpdf == TRUE){
    pdf(outfile, width=ht, height=wt)
    par(mar=c(5,5,4,2))
    barplot(H2[,4], names.arg = H2$Traits, ...)
    box()
    dev.off()
  }
}

plotH2(getpdf, outfile="graphs/Fig1a_v3.pdf",
       ylim=c(0, 1), col="antiquewhite3",  ylab="Heritability",
          cex.axis=fs, cex.names=fs, cex.lab=fs, las=2)
```
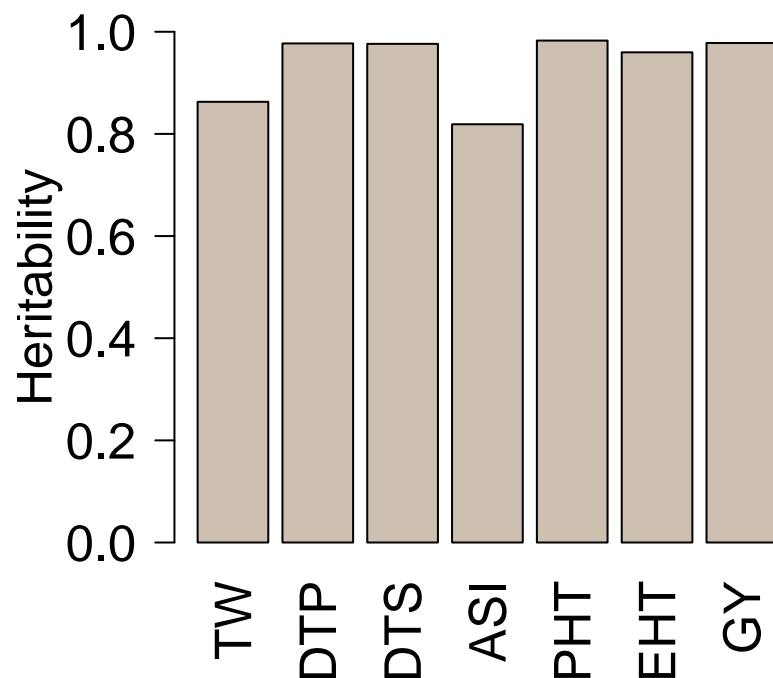
Figure 1b

```r
plotloh <- function(getpdf, outfile, ...){
  trait <- read.csv("cache/STable_heterosis.csv")
  bymed2 <- with(trait, reorder(trait, pBPH, median))
  boxplot(pBPH ~ bymed2, data=trait,
          xlab = "", ylab= "BPH (100%)", col="antiquewhite3",
          ...)
  if(getpdf == TRUE){
    pdf(outfile, width=ht, height=wt)
    par(mar=c(5,5,4,2))
    boxplot(pBPH ~ bymed2, data=trait,
            xlab = "", ylab= "BPH (100%)", col="antiquewhite3",
            ...)
    dev.off()
  }
}

plotloh(getpdf, outfile="graphs/Fig1b_v3.pdf",
        main="", cex.axis=fs, cex.lab=fs, las=2)
```
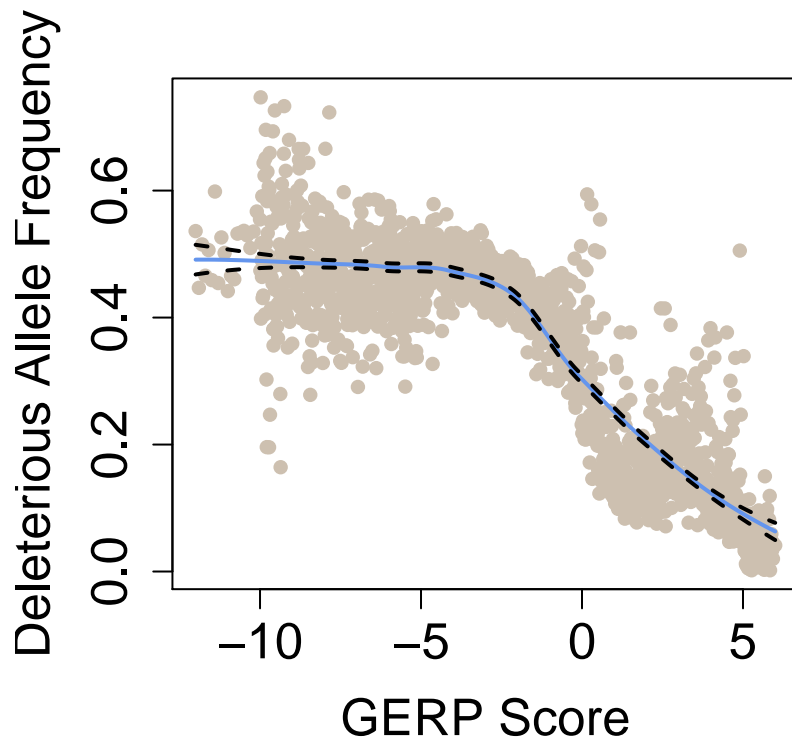
Figure 1c

```r
plotReg <- function(getpdf, outfile, ...){
  snptab <- read.csv("cache/daf_gerp2.csv")
  snptab <- snptab[order(snptab$GERP2),]
  plx <- predict(loess(snptab$meandaf ~ snptab$GERP2), se=T)
  x <- snptab$GERP2
  y <- snptab$meandaf

  plot(x, y, ...)
  lines(snptab$GERP2, plx$fit, col="cornflowerblue", lwd=2)
  lines(snptab$GERP2, plx$fit - qt(0.975,plx$df)*plx$se, lty=2, lwd=2, col="black")
  lines(snptab$GERP2, plx$fit + qt(0.975,plx$df)*plx$se, lty=2, lwd=2, col="black")

  if(getpdf == TRUE){
    pdf(outfile, width=wt, height=ht)
    par(mar=c(5,5,4,2))
    plot(x, y, ...)
    lines(snptab$GERP2, plx$fit, col="cornflowerblue", lwd=2)
    lines(snptab$GERP2, plx$fit - qt(0.975,plx$df)*plx$se, lty=2, lwd=2, col="black")
    lines(snptab$GERP2, plx$fit + qt(0.975,plx$df)*plx$se, lty=2, lwd=2, col="black")

    dev.off()
  }
}

plotReg(getpdf, outfile="graphs/Fig1c_v3.pdf",
        pch=16, col="antiquewhite3", xlab="GERP Score", ylab="Deleterious Allele Frequency",
        main="", cex.axis=fs, cex.lab=fs)
```

Figure 1d

To run this chunk, we need to install **beanplot** package.

```r
library("beanplot")
plotbeans <- function(getpdf, outfile, ...){

  res <- read.csv("cache/mgerp_cm.csv")
  cutoff <- quantile(res$gen, c(0.25, 0.5, 0.75))
  res$sq <- 0
  res[res$gen < cutoff[1], ]$sq <- 1
  res[res$gen >= cutoff[1] & res$gen < cutoff[2], ]$sq <- 2
  res[res$gen >= cutoff[2] & res$gen < cutoff[3], ]$sq <- 3
  res[res$gen >= cutoff[3], ]$sq <- 4

  beanplot(mgerp ~ sq, data = res, kernel="cosine", ll = 0.04, cex=fs, side = "no", cut=10, ylim=c(0.5,
          border = NA, col=list("#cd5b45", "antiquewhite3", "antiquewhite3", "antiquewhite3"),
          xaxt="n", ...)
  axis(side =1, at =1:4, labels =c("25", "50", "75", "100"), cex.axis=fs)

  if(getpdf == TRUE){
    pdf(outfile, width=wt, height=ht)
    par(mar=c(5,5,4,2))
    beanplot(mgerp ~ sq, data = res, kernel="cosine", ll = 0.04, cex=1.5, side = "no", cut=10, ylim=c(0
            border = NA, col=list("#cd5b45", "antiquewhite3", "antiquewhite3", "antiquewhite3"),
            xaxt="n", ...)
    axis(side =1, at =1:4, labels =c("25", "50", "75", "100"), cex.axis=fs)
    dev.off()
```

```
  }
}

plotbeans(getpdf, outfile="graphs/Fig1d_v3.pdf",
          ylab="GERP Score", xlab="Quantiles of cM/Mb",
          cex.axis=fs, cex.lab=fs)
```

```
## log="y" selected
```