Figure 3 of the GERP 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 get a pdf file. Or simplely click `Knit PDF` icon on `RStudio`. Note, to produce panels into seperate pdf files, we should set `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
fsize = 14 # absolute font size
getpdf = TRUE # get figures in seperated pdf [TRUE] or not [FALSE]
```

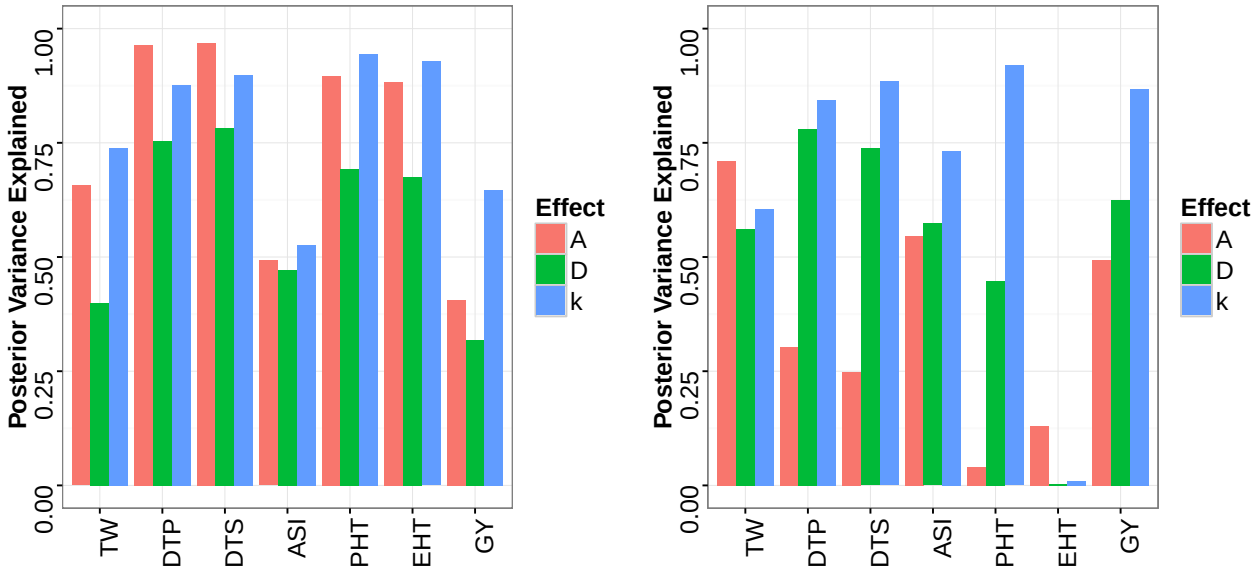
Figure 3a and 3b

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.2.4
```

```
source("lib/multiplot.R")

plot_fig3ab <- function(outfile, getpdf){
  res1 <- read.csv("cache/gerpsnp_wholeset_perse.csv")
  res2 <- read.csv("cache/gerpsnp_wholeset_bph.csv")

  res1$trait <- toupper(res1$trait)
  res2$trait <- toupper(res2$trait)

  h <- read.csv("cache/loh_pBPHmax_median.csv")
  med2 <- h
  med2$traitlw <- med2$trait
  #bymed2 <- with(trait, reorder(trait, pBPHmax, median))
  bymed2 <- med2[order(med2$h),]


p1 <- ggplot(res1, aes(x=factor(trait, levels=bymed2$traitlw), y=h2,
                       fill=factor(mode, levels=c("a2", "d2", "h2"), labels=c("A", "D", "k")))) +
  geom_bar(position=position_dodge(), stat="identity") +
  xlab("") +
  ylim(c(0,1)) +
  ylab("Posterior Variance Explained") +
```

```
  ggtitle("") + theme_bw() +
  labs(fill="Effect") +
  theme(axis.text = element_text(angle = 90, hjust = 1, size=fsize),
          axis.title=element_text(size=fsize, face="bold"),
          legend.title = element_text(size=fsize, face="bold"),
          legend.text = element_text(size=fsize))

p2 <- ggplot(res2, aes(x=factor(trait, levels=bymed2$traitlw), y=h2,
                       fill=factor(mode, levels=c("a2", "d2", "h2"), labels=c("A", "D", "k")))) +
  geom_bar(position=position_dodge(), stat="identity") +
  xlab("") +
  ylim(c(0,1)) +
  ylab("Posterior Variance Explained") +
  ggtitle("") + theme_bw() +
  labs(fill="Effect") +
  theme(axis.text = element_text(angle = 90, hjust = 1, size=fsize),
          axis.title=element_text(size=fsize, face="bold"),
          legend.title = element_text(size=fsize, face="bold"),
          legend.text = element_text(size=fsize))

  multiplot(p1, p2, cols=2)
  if(getpdf){
    pdf(outfile, width=10, height=4.5)
    multiplot(p1, p2, cols=2)
    dev.off()
  }
}
######
plot_fig3ab(outfile="graphs/Fig_post_var.pdf", getpdf)
```




```
## pdf
##   2
```

Figure 3c and 3d

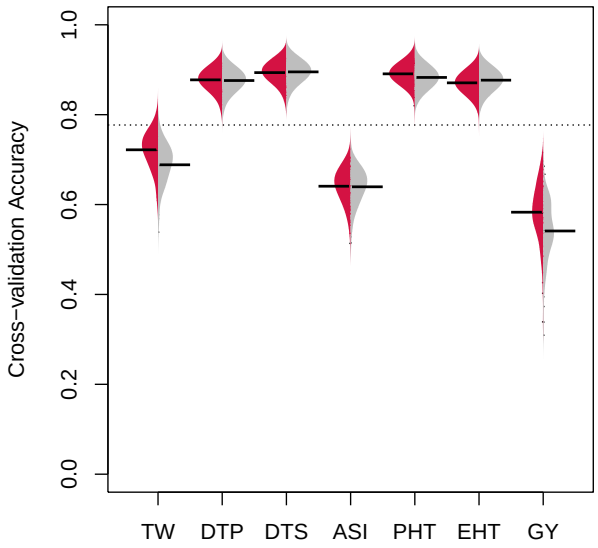
```
library(beanplot)
library(ggplot2)
library(plyr)
```

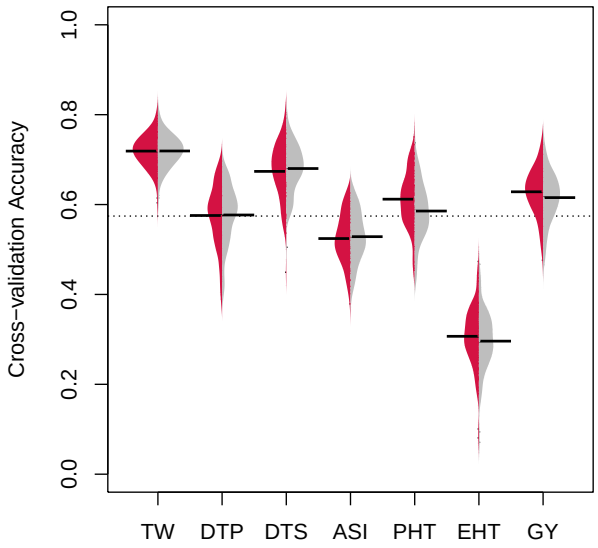
```
## Warning: package 'plyr' was built under R version 3.2.5
```

```
#http://www.jstatsoft.org/v28/c01/paper
mybean <- function(inputdf, mymode="a2", ...){
  res1 <- ddply(inputdf, .(trait, mode, sp, type), summarise,
                r = mean(r),
                m = median(r))
  res1 <- subset(res1, type != "null")

  res1$mode <- as.character(res1$mode)
  res1 <- subset(res1, mode == mymode)
  res1$trait <- toupper(res1$trait)
  #print(nrow(res1))
  par(lend = 1, mai = c(0.8, 0.8, 0.5, 0.5))
  res1$type <- factor(res1$type, levels = c("real", "cs"))
  res1$trait <- factor(res1$trait, levels = toupper(c("tw", "dtp", "dts", "asi","pht", "eht", "gy")))
  beanplot(m ~ type + trait, data = res1, kernel="cosine", ll = 0.04, cex=1.5, side = "both", bw=0.02,
           border = NA, col = list(c("#d41243", "#d41243"), c("grey", "grey")), ...)
  #legend("bottomleft", fill = c("black", "grey"),
  #       legend = c("Group 2", "Group 1"))
  #return(res0)

  out <- data.frame()
  myt <- toupper(c("tw", "dtp", "dts", "asi","pht", "eht", "gy"))
  for(myti in myt){
    real <- subset(res1, type == "real" & trait == myti)
    rand <- subset(res1, type == "cs" & trait == myti)

    message(sprintf("###>>> real [ %s ] and random [ %s ]", nrow(real), nrow(rand)))
    test <- t.test(real$r, rand$r, alternative ="greater")

    tem <- data.frame(trait=myti, realmean=mean(real$r), csmean=mean(rand$r), pval=test$p.value)
    out <- rbind(out, tem)
  }

  return(out)
}

####
plot_fig3cd <- function(outfile, getpdf){
  res01 <- read.csv("largedata/newGERPv2/res_realk_perse_42000.csv")
  res02 <- read.csv("largedata/newGERPv2/res_realk_bph_42000.csv")

  par(mfrow=c(1,2))
  #par(mar=c(5,4,4,2))
  res1 <- mybean(inputdf=res01, mymode = "h2", ylim=c(0, 1), main="", ylab="Cross-validation Accuracy")
```

```
  res2 <- mybean(inputdf=res02, mymode = "h2", ylim=c(0, 1), main="", ylab="Cross-validation Accuracy")

  if(getpdf){
    pdf(outfile, height=5, width=10)
    par(mfrow=c(1,2))
    #par(mar=c(5,4,4,2))
    par(font=2, font.lab=2, font.axis=2)
    res1 <- mybean(inputdf=res01, mymode = "h2", ylim=c(0, 1), main="", ylab="Cross-validation Accuracy
    res2 <- mybean(inputdf=res02, mymode = "h2", ylim=c(0, 1), main="", ylab="Cross-validation Accuracy
    dev.off()
  }


}

###
plot_fig3cd(outfile="graphs/Fig3_perse_BPH_2plots.pdf", getpdf)
```

```
## ###>>> real [ 100 ] and random [ 100 ]
## ###>>> real [ 100 ] and random [ 100 ]
## ###>>> real [ 100 ] and random [ 100 ]
```


Cross−validation Accuracy
0.0
0.2
0.4
0.6
0.8
1.0
TW
DTP
DTS
ASI
PHT
EHT
GY


Cross−validation Accuracy
0.0
0.2
0.4
0.6
0.8
1.0
TW
DTP
DTS
ASI
PHT
EHT
GY

```
## pdf
##   2
```