Supplementary Material for
Gonosomal function defines sexual dimorphism in immune cell abundance and cancer survival

Alejandro Caceres, Luis A. Perez Juarado and Juan R Gonzalez

Contents

List of Figures

Supplementary Methods

We analyzed publicly available data in the GEO repository, using the R/Bioconductor packages that can be found at https://www.bioconductor.org/. Main results are obtained from the application of the package teff (https://github.com/teff-package/teff). The results discussed in the manuscript can be entirely reproduced with the following code.

1 Profiling groups with high immune sexual dimorphism

We used GTEx data to infer the groups f high sexual dimorphism.

1.1 GTEx Data

We downloaded GTEx data ('SRP012682'), using `recount`. We extracted whole blood RNAseq count data and, for genes mapped with mutiple count data, we used the maximum count for each individual.

```
#download_study('SRP012682', outdir="./data/")
load("./data/rse_gene.RData")

rse_gene <- scale_counts(rse_gene)

#count data
recountCounts <- assays(rse_gene)$counts

#count info
recountMap <-rowRanges(rse_gene)

#gene ids
geneids <- unlist(recountMap$symbol)
gtexPd <- colData(rse_gene)

#sample ids
sampid <- gtexPd$sampid

#gene names
geneIncounts <- sapply(strsplit(rownames(recountCounts), "\\."),
                       function(x) x[[1]])
gensymbolscounts <- geneids[geneIncounts]

#counts with mapped gene ID
rmna <- !is.na(gensymbolscounts)
recountCounts<- recountCounts[rmna,]
gensymbolscounts <- gensymbolscounts[rmna]

#identify genes with multiple count data
```

```
dup <- names(table(gensymbolscounts))[table(gensymbolscounts)>1]
recountdup <- recountCounts[which(gensymbolscounts%in%dup),]
symbolsdup <- gensymbolscounts[which(gensymbolscounts%in%dup)]

#slect maximum count data per gene
recountmaxdup <- lapply(unique(symbolsdup), function(x)
{
  sapply(1:ncol(recountdup), function(ss) max(recountdup[symbolsdup%in%x,ss]))
})


recountmaxdup <- do.call(rbind, recountmaxdup)

#identify genes with single (not duplicated) count data
recountCountsnodup <- recountCounts[which(!gensymbolscounts%in%dup),]

#re-join both data
recount2 <- rbind(recountCountsnodup, recountmaxdup)
rownames(recount2) <- c(gensymbolscounts[which(!gensymbolscounts%in%dup)],
                        unique(symbolsdup))
colnames(recount2) <- sampid

#select whole-blood
mask1 <- which(gtexPd$smtsd=="Whole Blood")
counts <- recount2[,mask1]
filt <- rowSums(counts)

#select genes with at least 100 count across individuals
counts <-  counts[filt > 100,]
counts <- counts[!is.na(rownames(counts)),]

#save gene info
geneids <- data.frame(recountMap)

save(geneids, file="./data/geneids.RData")
```

Phenotype data was downloaded from https://gtexportal.org/home/ and matched them to RNA-seq count data

```
#load covariates
cov <- read.table("./data/covGTEx.txt",
                  as.is=TRUE, header=TRUE, sep="\t", fill=TRUE)

idscov <- sapply(strsplit(cov[,2],"GTEX-"), function(x) x[[2]])
rownames(cov) <- idscov

whichpheno <- sapply(idscov, function(x) grep(x,colnames(counts))[1])

matchphenoexp<-whichpheno[complete.cases(whichpheno)]

#match pheno and count data for the same individuals
counts <- counts[,matchphenoexp]
cov <- cov[names(matchphenoexp),]
```

We transformed count data to TMP, and applied MPC Counter of this data to compute immune cell composition for inter subject comparisons, as implemented in the `immunedeconv`.

```
#compute TMP count data
exprTPM <- lapply(1:ncol(counts),
                  function(ss) counts[,ss]/sum(counts[,ss])*1e6)
exprTPM <- do.call(cbind,exprTPM)
exprTPM <- as.matrix(exprTPM)
colnames(exprTPM) <- colnames(counts)
rownames(exprTPM) <- rownames(counts)

oo <- order(rownames(exprTPM))
counts <- counts[oo,]
exprTPM <- exprTPM[oo,]
exprTPM <- exprTPM[!is.na(rownames(exprTPM)),]

cellcomp1 <- deconvolute(exprTPM, "mcp_counter")
cellnames <- cellcomp1$cell_type
cm <- matrix(as.numeric(t(cellcomp1)[-1,]),
             ncol=length(cellnames))
colnames(cm) <- cellnames
rownames(cm) <- colnames(cellcomp1)[-1]

pheno <- data.frame(cov, data.frame(cm,check.names = FALSE),
                    check.names = FALSE)

pdf("./figure/celldeconv.pdf")

cellcomp1 %>%
  gather(sample, score, -cell_type) %>%

  ggplot(aes(x=sample, y=score, color=cell_type)) +
    geom_point(size=4) +
    facet_wrap(~cell_type, scales="free_x", ncol=3) +
    scale_color_brewer(palette="Paired", guide="none") +
    coord_flip() +
    theme_bw() +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))

dev.off()

save(counts, pheno, file="./data/gtexBlood.RData")
```

We computed the immune cell distribution in bulk trasncriptomic data of whole blood.

```
load("./data/gtexBlood.RData")
cellnames <- colnames(pheno)[13:22][-7]


par(mfrow=c(3,3))
for(xx in cellnames[cellnames%in%colnames(pheno)]){
  hist(pheno[,xx],main=xx, xlab="Abundance score")
}
```

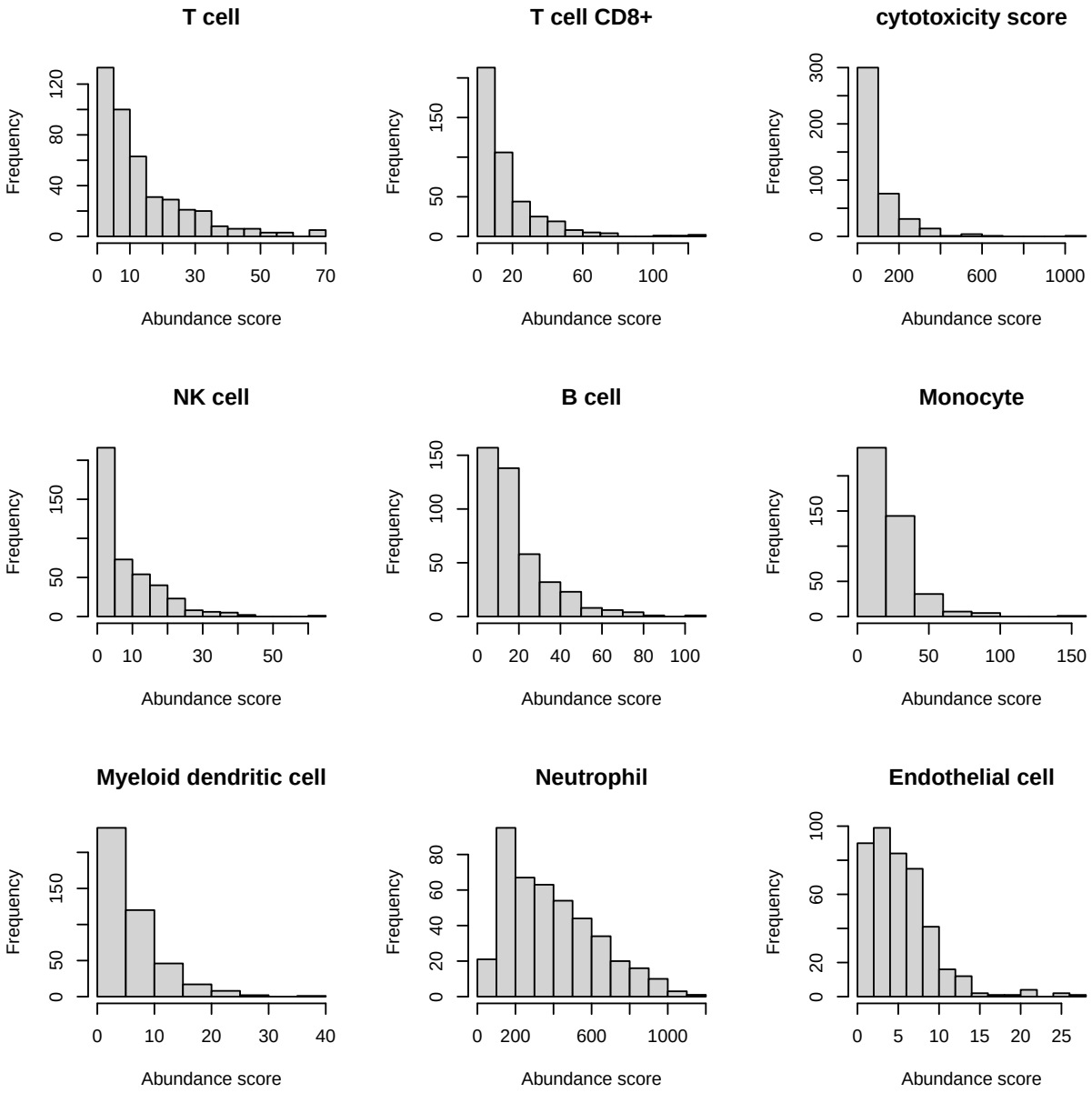
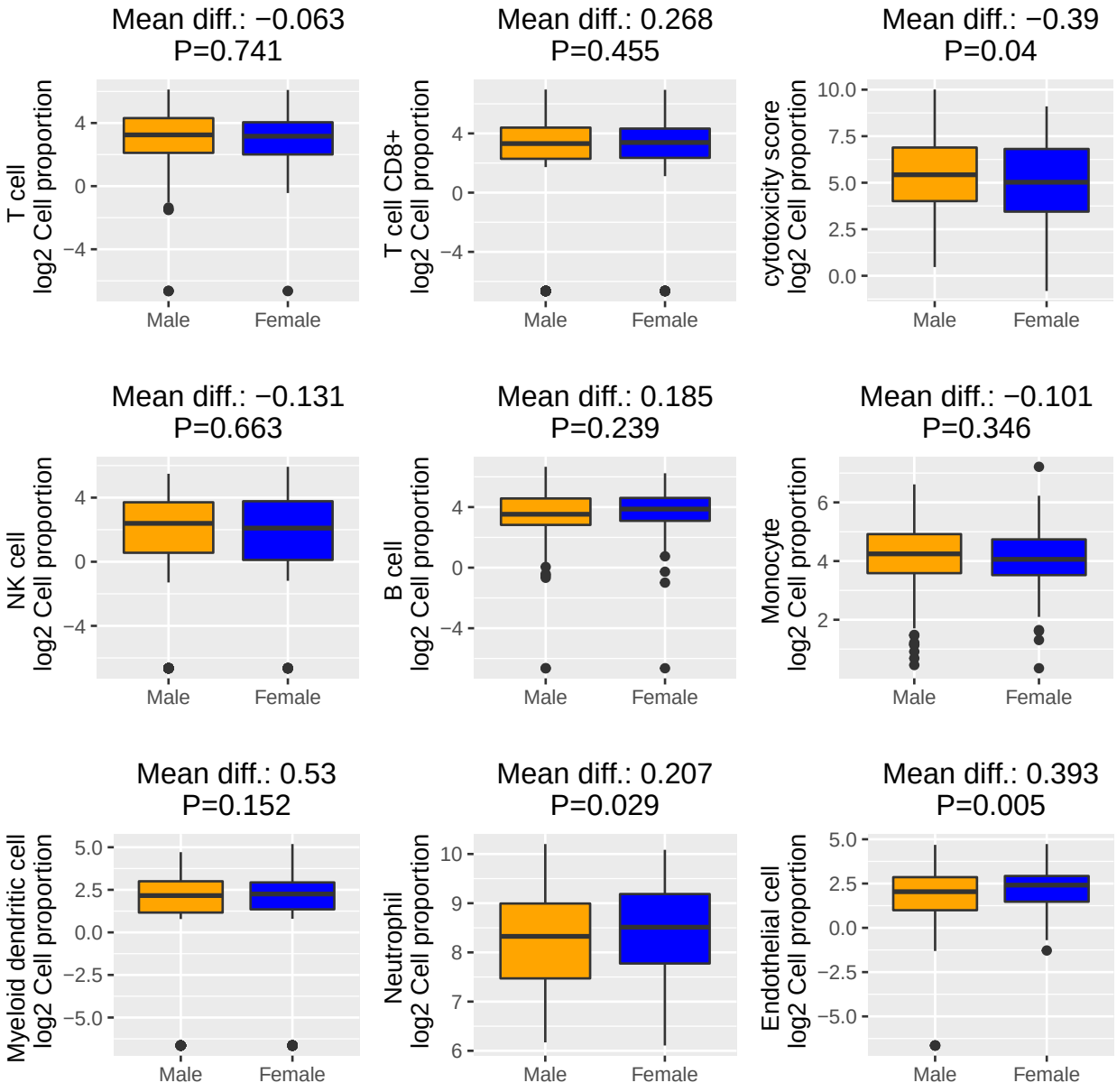

T cell
Frequency
0 40 80 120
0 10 30 50 70
Abundance score
T cell CD8+
Frequency
0 50 150
0 20 60 100
Abundance score
cytotoxicity score
Frequency
0 100 200 300
0 200 600 1000
Abundance score
NK cell
Frequency
0 50 150
0 10 30 50
Abundance score
B cell
Frequency
0 50 100 150
0 20 40 60 80 100
Abundance score
Monocyte
Frequency
0 50 150
0 50 100 150
Abundance score
Myeloid dendritic cell
Frequency
0 50 150
0 10 20 30 40
Abundance score
Neutrophil
Frequency
0 20 40 60 80
0 200 600 1000
Abundance score
Endothelial cell
Frequency
0 20 60 100
0 5 10 15 20 25
Abundance score

Figure S1

We then tested the effect of sex on each cell count, adjusting by age and BMI.

```
load("./data/gtexBlood.RData")

sex <- pheno$GENDER
cellnames <- colnames(pheno)[13:22][-7]

plist <- list()
for(i in 1:length(cellnames[cellnames%in%colnames(pheno)])){

  xx <- cellnames[cellnames%in%colnames(pheno)][i]

  p <- pheno[[xx]]
  p[p==0] <- NA
  p <- log2(pheno[[xx]]+0.01)

  sex.factor <- factor(sex, labels=c("Male", "Female"))

  x <- sex.factor
  y <- p

  dat <- data.frame(x, y)

  mm <- summary(lm(p~ sex + pheno$AGE + pheno$BMI))
  lg <- paste0(c("Mean diff.: ", "P="),
               round(mm$coeff["sex",c(1,4)],3), collapse="\n")

  plist[[i]] <- ggplot(dat, aes(x=x, y=y, fill=x)) +
  geom_boxplot()+ylab(paste(xx, "\n log2 Cell proportion"))+xlab("")+
  ggtitle(lg)+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(legend.position = "none") +
  scale_fill_manual(values = c("orange", "blue"))
}

grid.arrange(grobs = plist, nrow = 3)
```



1.2 Transcriptome-wide interaction analysis

We performed a differential gene expression analysis for the interaction between sex and cell proportion. We first extracted surrogate variables and defined the models for each cell type.

```
#SVA and model specification for the DE analysis of the interaction between sex and cell proportion, fo

#t:treatment (sex)
#eff:effect (cell abundance)

datgtex <- lapply(cellnames, function(type){

  phenored <- data.frame(eff = pheno[[type]],
                         t = pheno$GENDER, age = pheno$AGE,
```

```
                          bmi = pheno$BMI)

  mod0 <- model.matrix(~ t + eff + age + bmi, data = phenored)
  mod <- model.matrix(~ t:eff + t + eff + age + age + bmi,
                      data = phenored)

  #sva
  ns <- num.sv(counts, mod, method = "be")
  ss <- svaseq(counts, mod, mod0, n.sv = ns)$sv

  colnames(ss) <- paste("cov", 1:ncol(ss), sep="")

  modss <- cbind(mod, ss)

  #estimation
  design <- model.matrix(~ t:eff + t+ eff + age +bmi,
                         data = phenored)
  v <- voom(counts, design = design)

  expr <- v$E
  nmsgenes <- rownames(expr)

  expr <- expr[!is.na(nmsgenes),]
  rownames(expr) <- nmsgenes[!is.na(nmsgenes)]

  list(expr=expr, phenos=modss)
})


names(datgtex) <- cellnames

save(datgtex, file="./data/datgtex.RData")
```

We performed inference for the interaction term, performed volcano plots

```
load("./data/datgtex.RData")

#t:eff is the interaction between sex and cell abundance

inter <- lapply(datgtex, function(x)
{
  exprs <- x$expr
  phenos <- x$phenos
  subsids <- colnames(exprs)

  fit <- lmFit(exprs, phenos)
  eBayes(fit)
})


  i <- 0
  sig <- list()
  par(mfrow=c(3,3))
  for(xx in cellnames[cellnames%in%colnames(pheno)]){
```

```
    i <- i+1
  volcanoplot(inter[[xx]], highlight=5,
              coef="t:eff",
              names=rownames(inter[[xx]]$coefficients), cex=0.1)

  title(main=xx)

  tt <- topTable(inter[[xx]], number=Inf, coef="t:eff",
                 adjust.method="bonferroni")

  sig[[i]] <- tt[tt$adj.P.Val<0.05,c(1,5)]
}
```

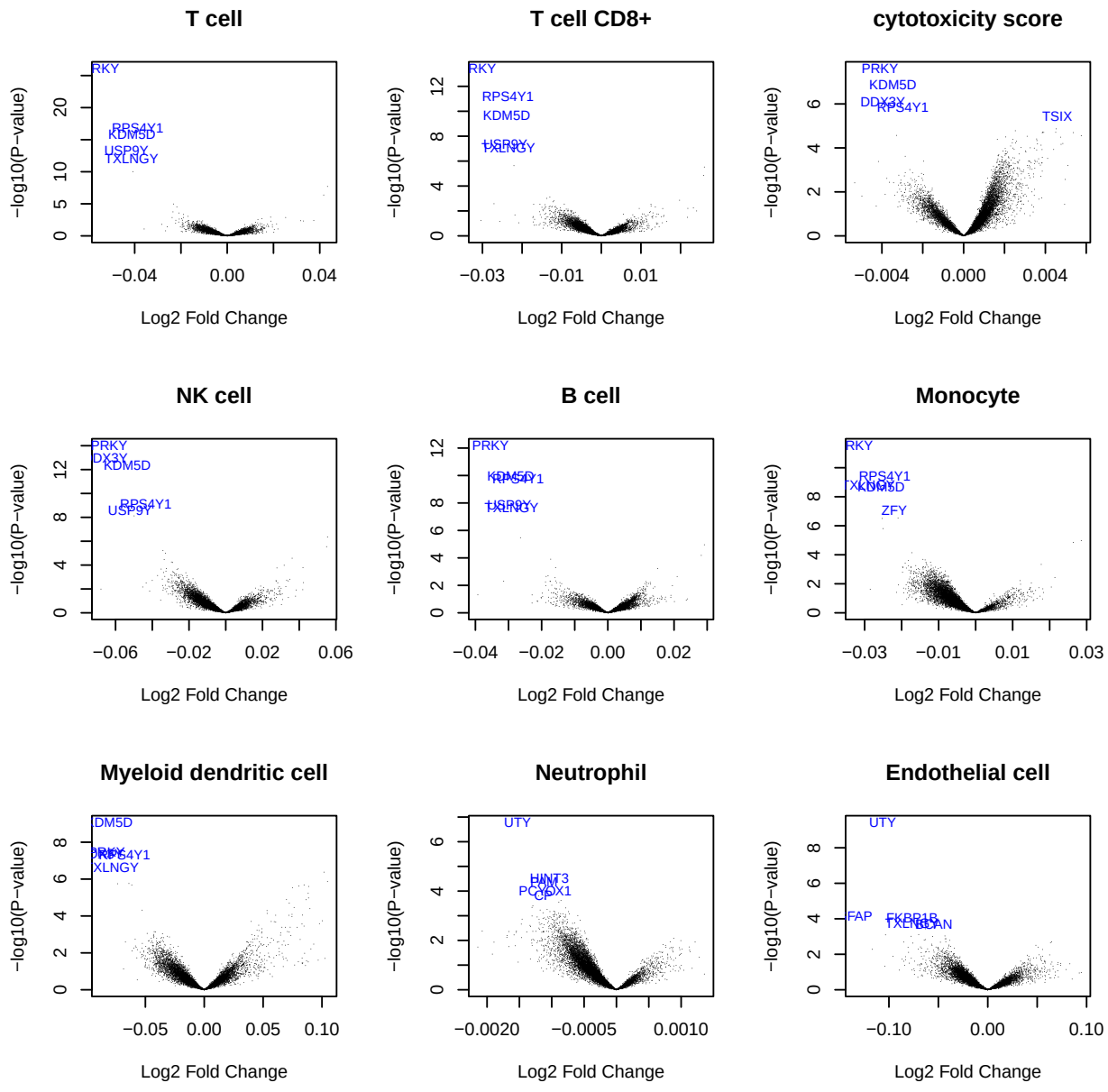


Figure S2

```
wr <- sapply(names(inter), function(x) {
  tb <- format(topTable(inter[[x]], coef="t:eff", adjust.method="bonferroni"), digits=4)

  write.table(tb, file=paste0(c("table/", x,".txt"), collapse=""),
              sep="\t", row=TRUE, col=TRUE, quote=FALSE)
})
```

and wrote significant results in tables

```
getSiggenes <- function(x)
{
exprs <- x$expr
phenos <- x$phenos
subsids <- colnames(exprs)


fit1 <- lmFit(exprs, phenos)
fit1 <- eBayes(fit1)
tt1 <- topTable(fit1, number=Inf, coef="t:eff")

dd1 <- as.matrix(tt1[,2:5])
commonnms <- unique(rownames(tt1))

pval <- dd1[,"P.Value"]
logFC <- dd1[,"AveExpr"]

fdr <- p.adjust(pval, "fdr")
colfdr <-  fdr < 0.05

genesinf <- commonnms[colfdr]
}

sigGenes <- lapply(datgtex[(1:7)[-3]],getSiggenes)

genestab <- sort(table(unlist(sigGenes)))

genestab[genestab>5]

##
##  DDX3Y  KDM5D   PRKY RPS4Y1   TSIX TXLNGY  USP9Y
##      6      6      6      6      6      6      6
```

We observed that a list of seven genes were significantly associated with the sex-cell type abundance, across six cell types.

1.3 Estimate expected sex difference in immune cell abundance

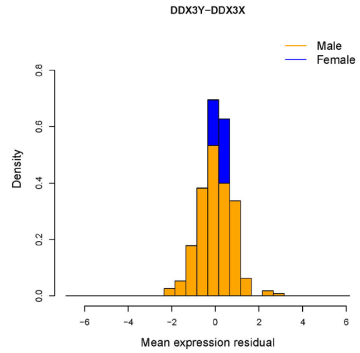
We used the `teff` package to estimate the expected sex difference in immune cell abundance associated to each individual's observed features. The features were defined as the transcription residuals, adjusted by covariates and surrogate variables, and averaged between homologous pairs of the genes previously identified in the interaction analysis.

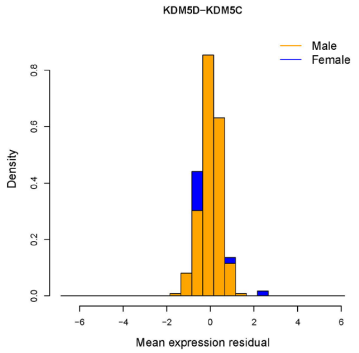
```
#format for teff, as obtained in the interaction analysis
teffgtex <- lapply(datgtex, function(x){
```

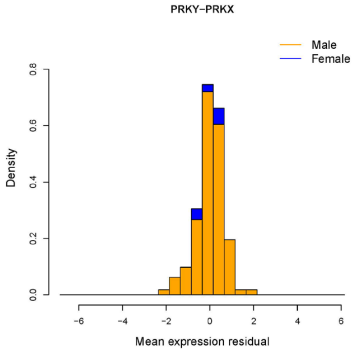
```
          res <- x
          names(res) <- c("features", "teffdata")
          rmvars <- !colnames(res$teffdata)%in%c("(Intercept)",
                                                 "t:eff")
          res$teffdata <- res$teffdata[,rmvars]
          res$features <- t(res$features)
          res
})

names(teffgtex) <- names(datgtex)
```

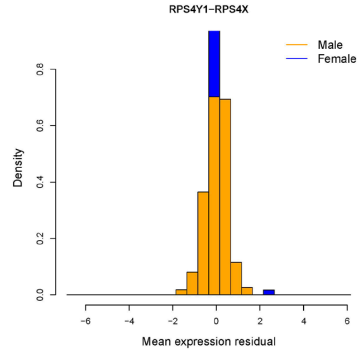
The selected observed features were given by the homologous pairs, XIST and TSIX. Plots of feature overlap and interactions across sexes were performed to check that causal inferences are suitable on the data. `predicteff` was used to plot feature overlap and interactions across sexes, and to perform causal random forest analysis. A CRF was grown internally in the `predicteff` function, for a random set (train-set) of 80% of the individuals. Inference of the effect of sex for the remaining 20% (test-set) of the expression profiles of individuals was estimated with confidence intervals.

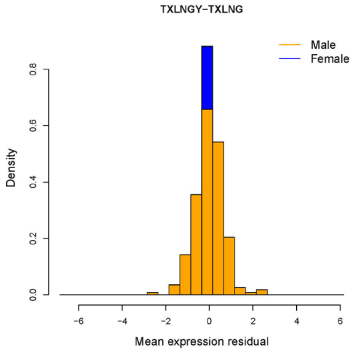
```
XYhomol<- matrix(c("DDX3Y","DDX3X","KDM5D","KDM5C","PRKY","PRKX","RPS4Y1","RPS4X","TXLNGY",
                   "TXLNG", "USP9Y", "USP9X", "XIST", "XIST", "TSIX", "TSIX"), nrow=2)

#plot overlap
tcell <- teffgtex[[1]]

pred <- predicteff(tcell, featuresinf=XYhomol,
                   profile=TRUE,
                   plot.overlap = TRUE)

## [1] "... plots of t:eff interactions on the outcome in interactions.pdf \n"
## [1] "... plots of covariate overlap across tretments in overlap.pdf \n"
```

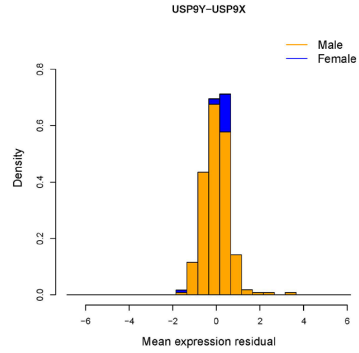

DDX3Y−DDX3X
Male
Female
Density
Mean expression residual

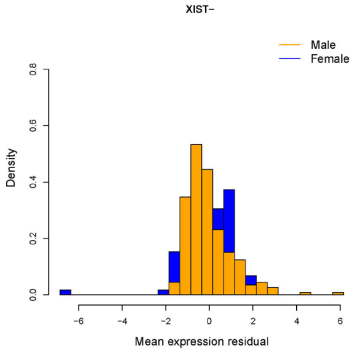

KDM5D−KDM5C
Male
Female
Density
Mean expression residual

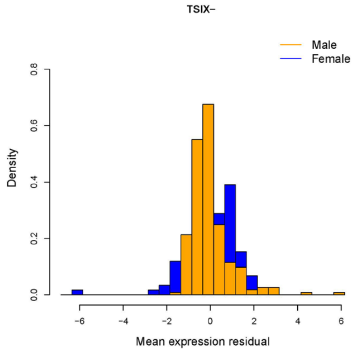

PRKY−PRKX
Male
Female
Density
Mean expression residual


RPS4Y1−RPS4X
Male
Female
Density
Mean expression residual


TXLNGY−TXLNG
Male
Female
Density
Mean expression residual


USP9Y−USP9X
Male
Female
Density
Mean expression residual


XIST−
Male
Female
Density
Mean expression residual


TSIX−
Male
Female
Density
Mean expression residual

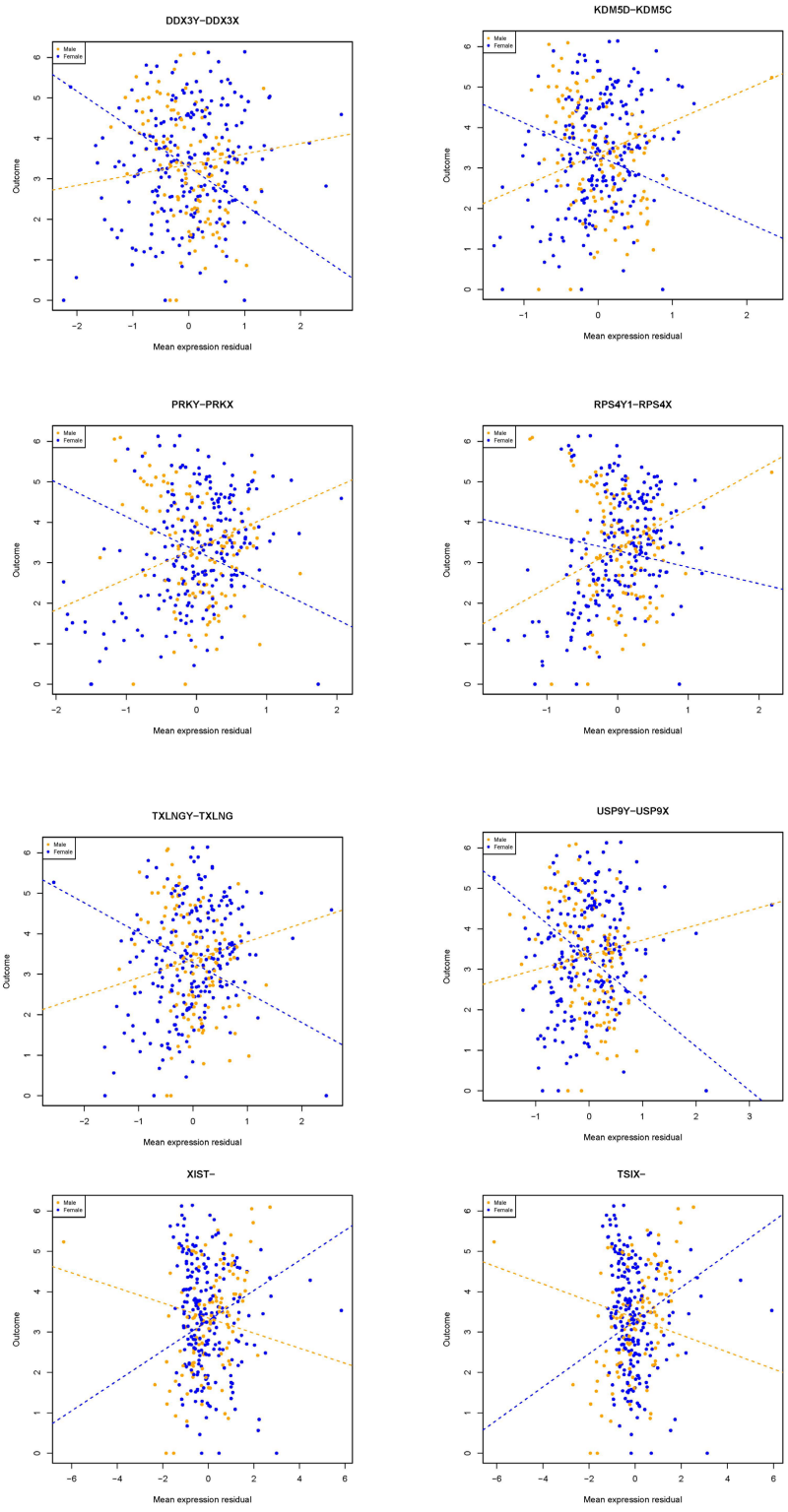
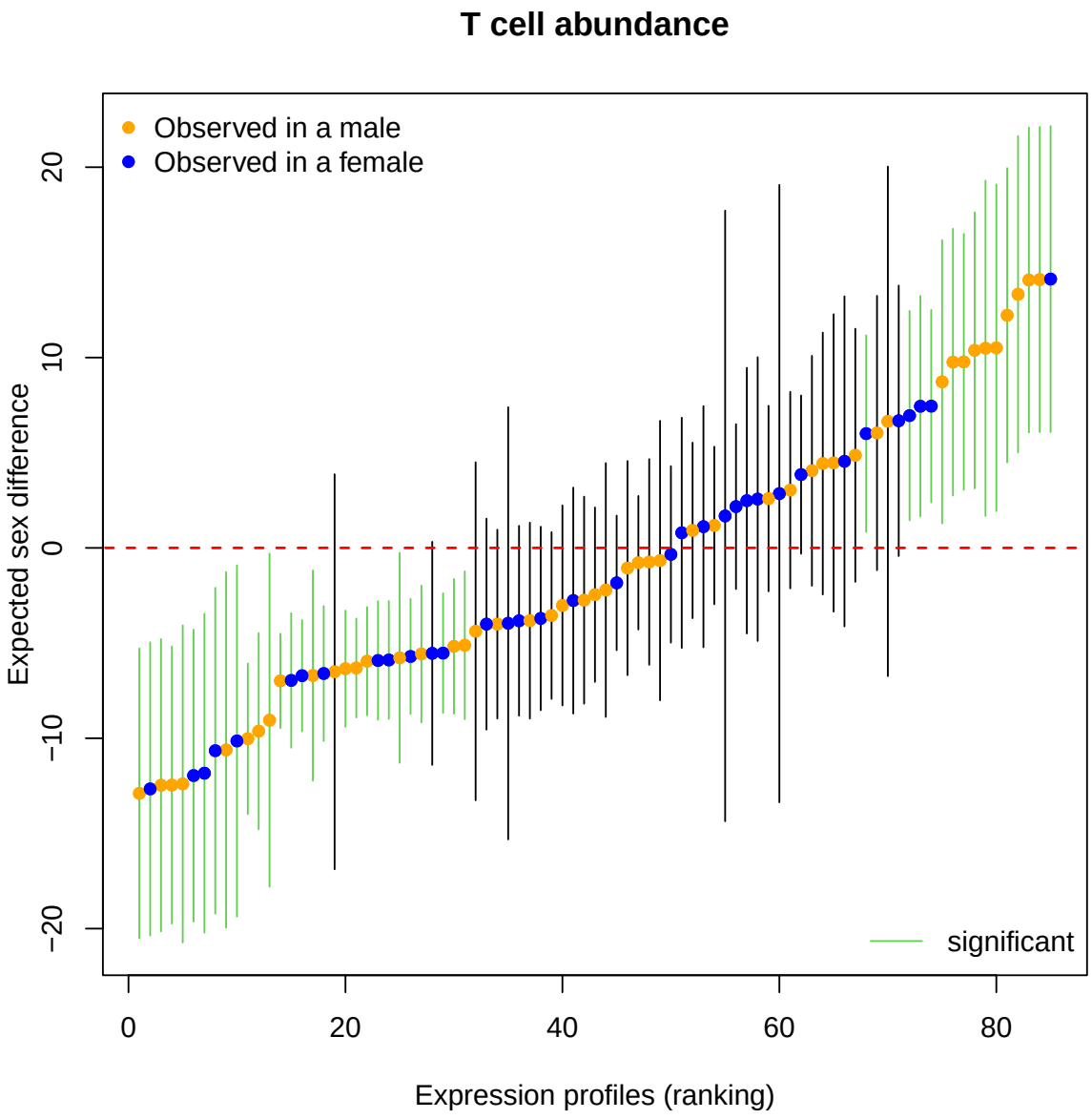

DDX3Y−DDX3X
KDM5D−KDM5C
PRKY−PRKX
RPS4Y1−RPS4X
TXLNGY−TXLNG
USP9Y−USP9X
XIST−
TSIX−
Male
Female
Outcome
Mean expression residual

Figure S3

```r
#prediction of expected sex differences of expression profiles in the test-set

pred
```

```
## object of class: pteff
## Estimated treatment effects in $predictions:
##  [1]   6.6485833  -5.7028466  -3.8289295  -3.8192694  -3.7057803   4.4653225   6.6794771
##  [8]   2.5649402 -12.8958318 -11.9609187   6.0041126  -1.8374791  -6.3450333 -12.4676735
## [15] -10.0255919   2.4823997  -5.5275620  -6.9856559  -5.5720955   4.0512810  14.0750727
## [22]  10.3782952   3.0360949   6.9510375  -5.1137459   6.0374294   2.1693907   9.7728061
## [29]  13.3242438  -2.4570938  -5.9521672 -10.6135121   1.1786464  -6.5074171  -6.7046309
## [36]  -6.3080904   4.8647779   2.8521892 -11.8356216 -12.4006121  -0.7404155  -6.7135344
## [43]  -5.8881988  -4.3798962  -5.7743037  -1.0602535   3.8517047  10.4898261  -2.2121442
## [50]   4.5478292  -5.5397219   8.7262339  -4.0027492  14.0994602  12.2207131  -3.9611612
## [57]  -4.0048134   1.6783265  -3.0231956  -3.5509816  -5.9151559  -6.5992608   9.7602988
## [64] -12.6630520   1.1142965  -0.7789992  -6.9613671  -5.1730401 -10.6536716  -0.6666106
## [71]   0.7942378  14.1245064  -0.3412159   4.4328506 -10.1408222  -9.0526475  10.5106305
## [78]  -2.7657815 -12.4594603   2.5919673  -9.6278145   7.4425594   7.4480063  -2.7459211
## [85]   0.9192135
```

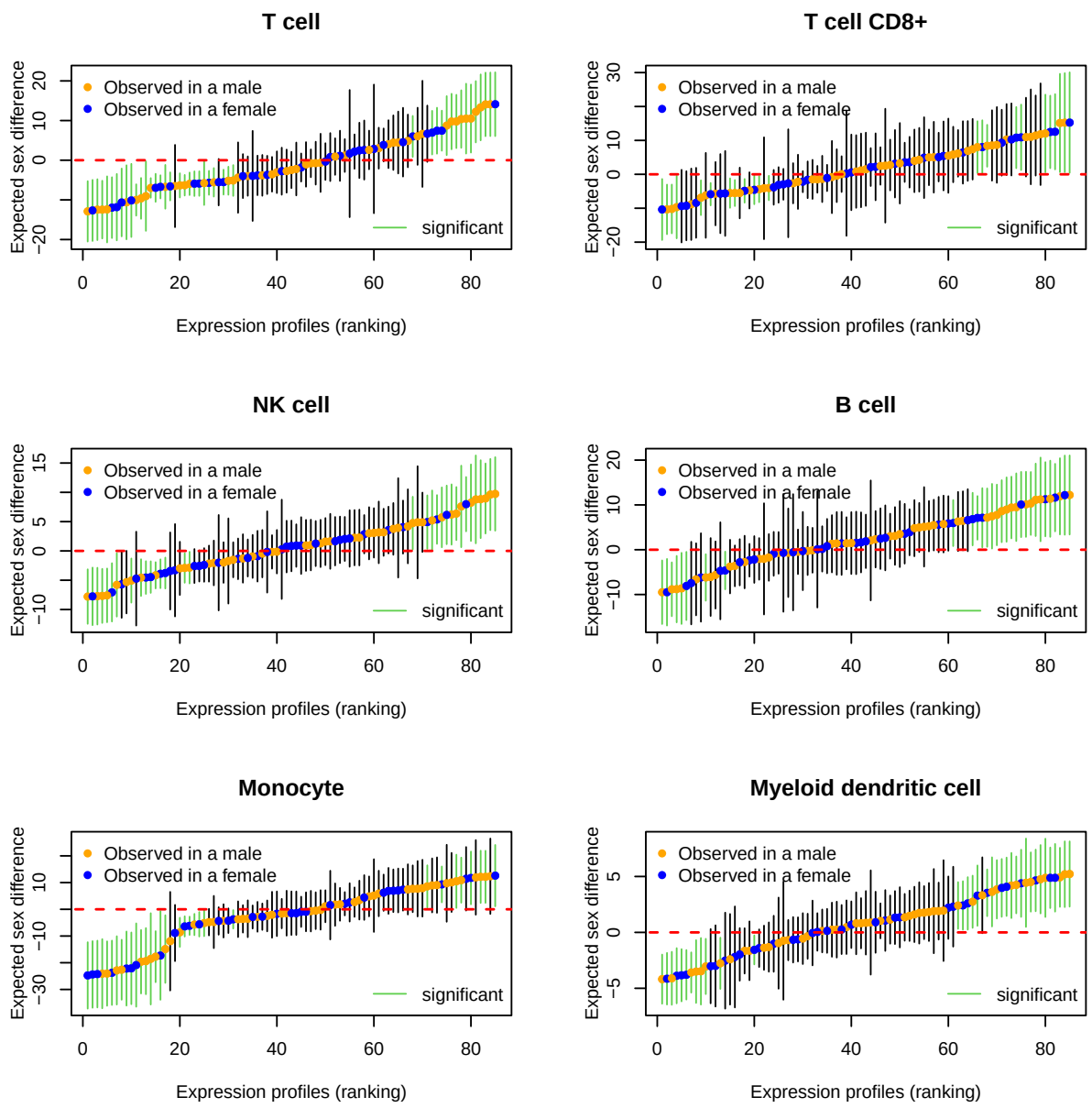
```r
plotPredict(pred, lb = "Expected sex difference",
            ctrl.plot=list(lb=c("Observed in a male",
                                "Observed in a female"),
                           wht="topleft", whs = "bottomright"),
            main="T cell abundance",
            xlab="Expression profiles (ranking)")
```



We performed CRF, to estimate the expected sex differences for all the observed transcription profiles in the test-set in all immune cell types.

```
htrcells <- lapply(teffgtex[1:7], predicteff, featuresinf=XYhomol, profile=TRUE)

names(htrcells) <- names(teffgtex)[1:7]

par(mfrow=c(3,2))
for(nn in names(htrcells)[-3]){
  plotPredict(htrcells[[nn]], lb = "Expected sex difference",
              ctrl.plot=list(lb=c("Observed in a male",
                                  "Observed in a female"),
                             wht="topleft", whs = "bottomright"),
              main=nn,
              xlab="Expression profiles (ranking)")
```

```
}
```



All the expression profiles of individuals with significant sex differences were used to extract binarized patterns associated to positive and negative sexual dimorphisms.

```
# average binarized profile of individuals with
#significanlty positive sexual-dimorphisms across
#all cell types.

genepairs <- paste(paste(XYhomol[1,],XYhomol[2,], sep="-"))

prof <- sapply(htrcells[-3], function(x) x$profile$profpositive[,genepairs])

prof
```

```
##              T cell T cell CD8+ NK cell B cell Monocyte Myeloid dendritic cell
## DDX3Y-DDX3X   FALSE       FALSE   FALSE  FALSE    FALSE                  FALSE
## KDM5D-KDM5C   FALSE       FALSE   FALSE  FALSE    FALSE                  FALSE
## PRKY-PRKX     FALSE       FALSE   FALSE  FALSE    FALSE                  FALSE
## RPS4Y1-RPS4X  FALSE       FALSE   FALSE  FALSE    FALSE                  FALSE
## TXLNGY-TXLNG  FALSE       FALSE   FALSE  FALSE    FALSE                  FALSE
## USP9Y-USP9X   FALSE       FALSE   FALSE  FALSE    FALSE                  FALSE
## XIST-XIST      TRUE        TRUE    TRUE   TRUE     TRUE                   TRUE
## TSIX-TSIX      TRUE        TRUE    TRUE   TRUE     TRUE                   TRUE
```

We observed that the positive and negative patterns of were oposite.

```
# average binarized profile of individuals with
#significanlty negative sexual-dimorphisms
#across all cell types.

genepairs <- paste(paste(XYhomol[1,],XYhomol[2,], sep="-"))

prof <- sapply(htrcells[-3], function(x) x$profile$profnegative[,genepairs])

prof
```

```
##              T cell T cell CD8+ NK cell B cell Monocyte Myeloid dendritic cell
## DDX3Y-DDX3X    TRUE        TRUE    TRUE   TRUE    FALSE                   TRUE
## KDM5D-KDM5C    TRUE        TRUE    TRUE   TRUE     TRUE                   TRUE
## PRKY-PRKX      TRUE        TRUE    TRUE   TRUE    FALSE                  FALSE
## RPS4Y1-RPS4X   TRUE        TRUE    TRUE   TRUE     TRUE                   TRUE
## TXLNGY-TXLNG   TRUE        TRUE    TRUE   TRUE     TRUE                   TRUE
## USP9Y-USP9X    TRUE        TRUE    TRUE   TRUE     TRUE                   TRUE
## XIST-XIST     FALSE       FALSE   FALSE  FALSE    FALSE                  FALSE
## TSIX-TSIX     FALSE       FALSE   FALSE  FALSE    FALSE                  FALSE
```

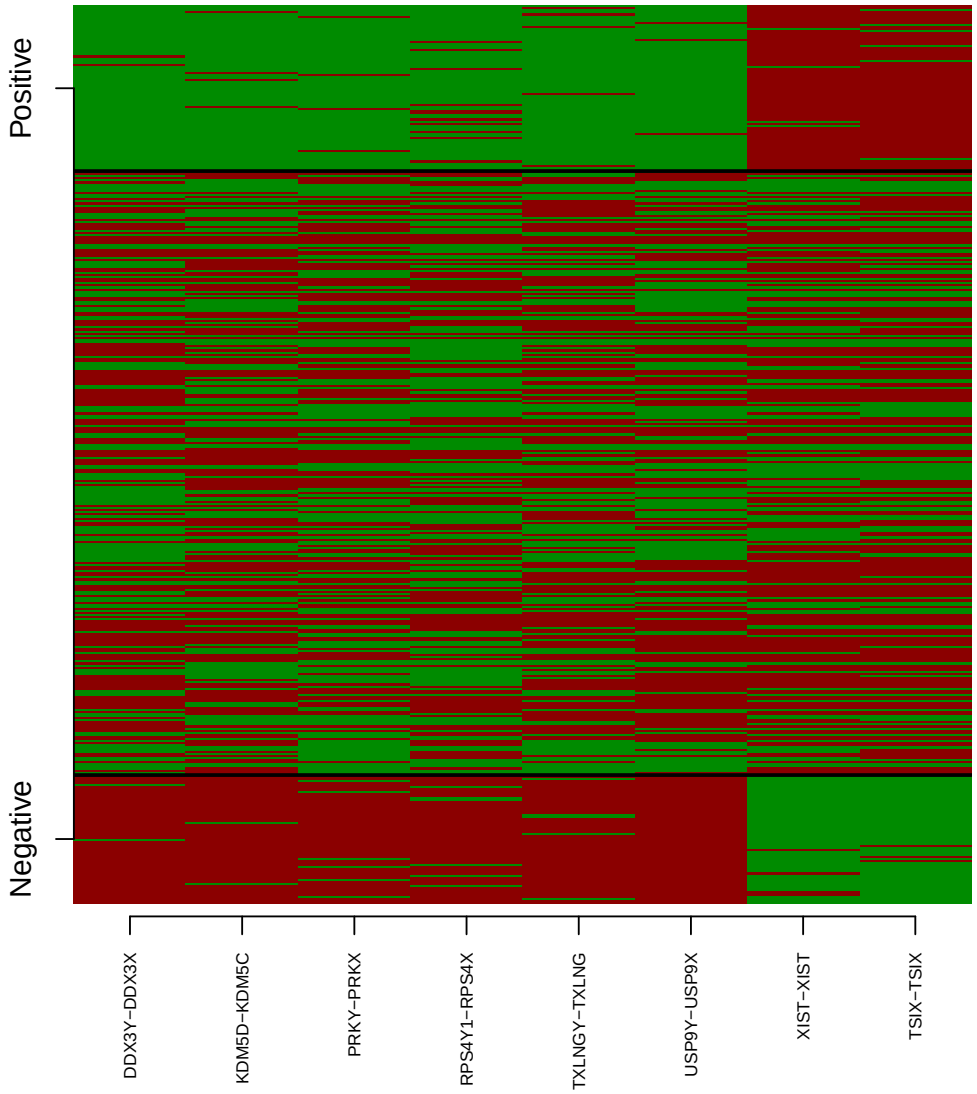
1.4 Targeting

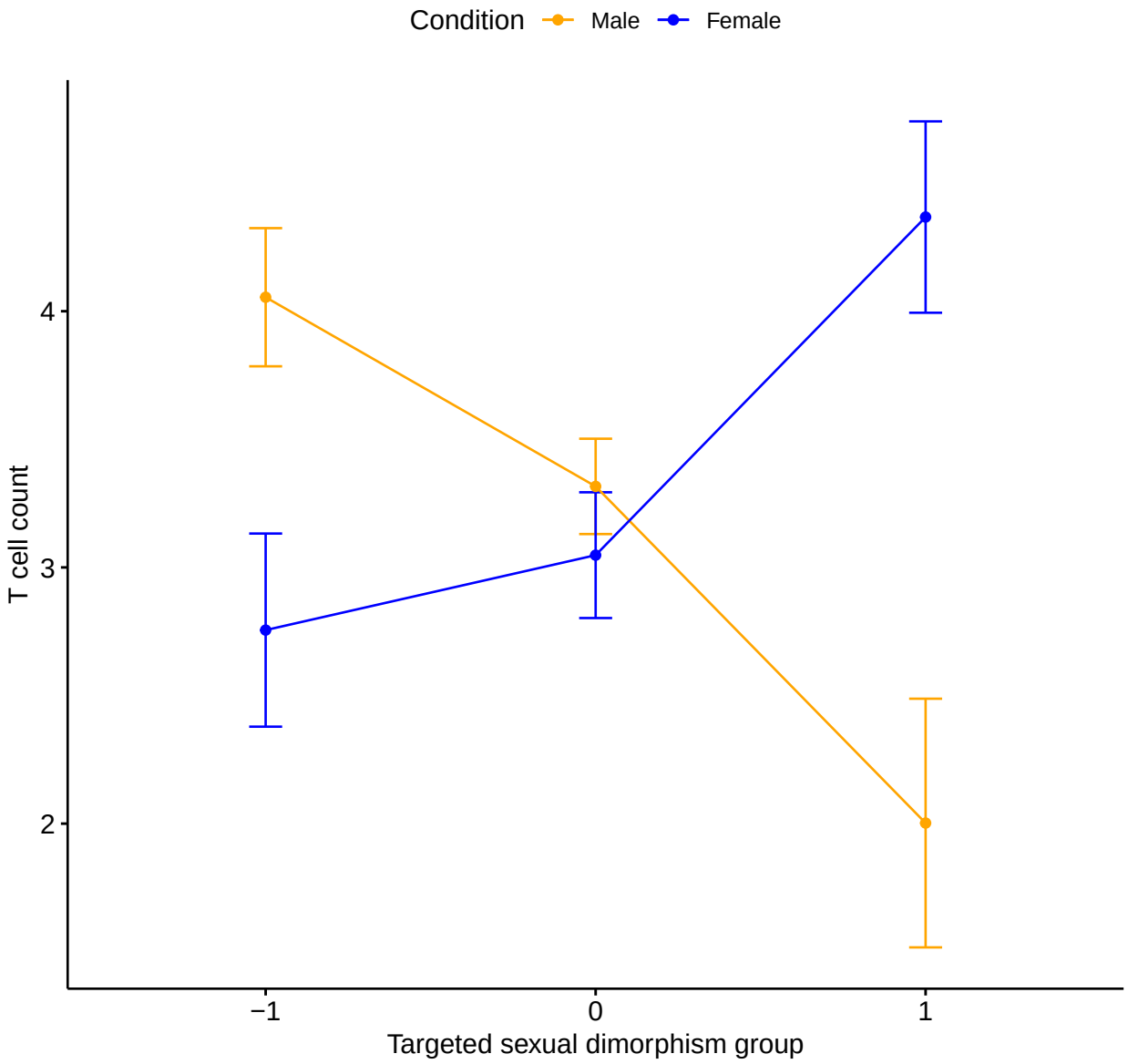
We used the binary profiles to target individuals in the whole data set and test whether the classification indeed modulated the association between immune cell abundance and sex (interaction analysis). We used the function `target` with the the pattern given by T-cell abundance to test the targeting and the interaction.

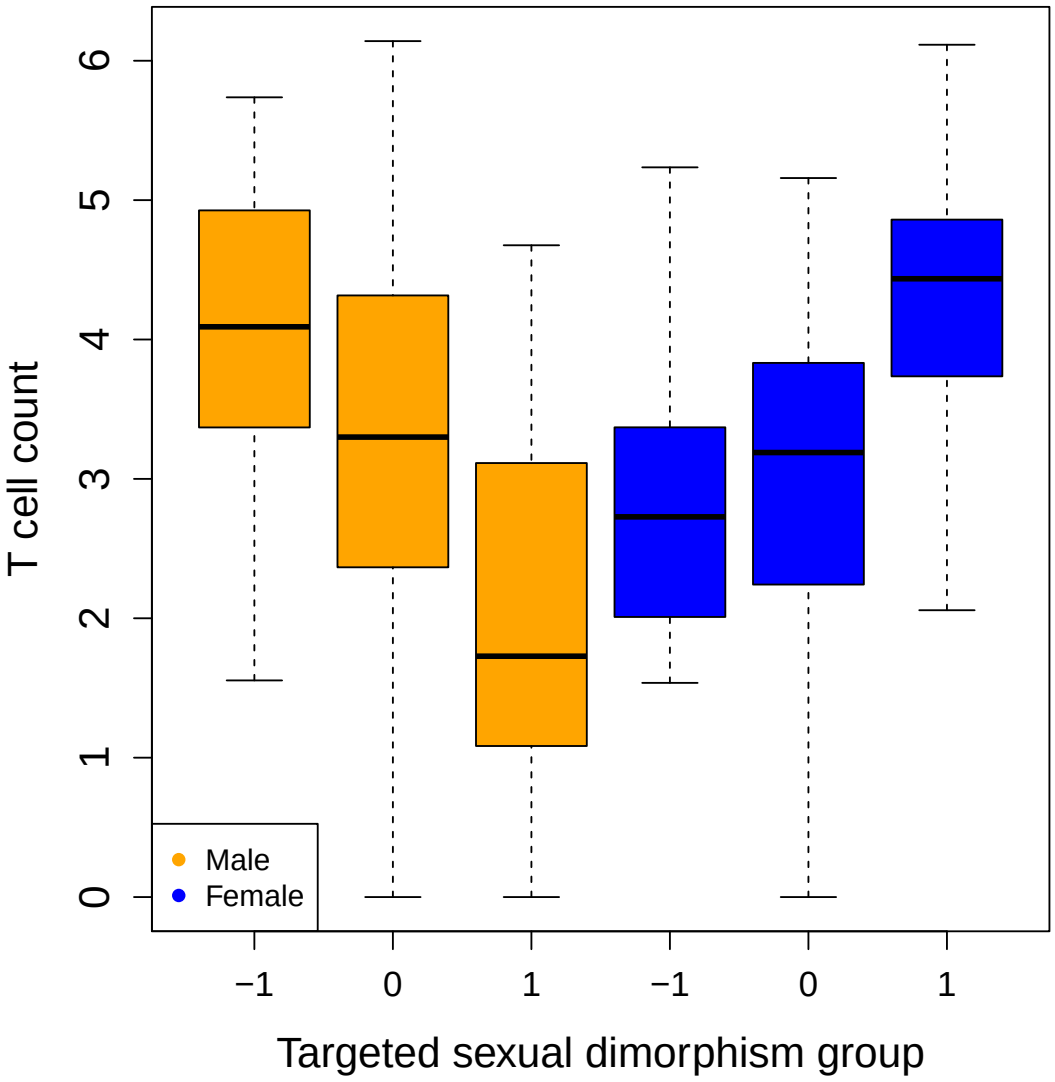
```
res <- target(teffgtex$"T cell", htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              nmcov="age", model="log2")

res
```

```
## object of class: tarteff
##
## classification into
##   negative treatment effect: -1
##   neutral: 0
##   positive treatment: 1
##
##  -1   0   1
##  79 288  61
##
```

```
## interaction fitted model: log2
##     Estimate   Std. Error      t value     Pr(>|t|)
## 1.630906e+01 2.178184e+00 7.487459e+00 4.121995e-13

plotTarget(res,  labs=c("Targeted sexual dimorphism group",
                        "T cell count", "Condition",
                        "Male", "Female"))

boxPlot(res, labs=c("Targeted sexual dimorphism group",
                    "T cell count", "Condition",
                    "Male", "Female"), lg="bottomleft")
```




Condition
Male
Female
T cell count
4
3
2
−1
0
1
Targeted sexual dimorphism group



1.5 Translation to other tissues

We aimed to determine the extent to which the classification into the immune dimorphisms could be assessed in other tissues. We targeted samples of individuals across 15 different tissues from the GTEx project. In each tissue we inferred the T cell abundance and performed the classification associated to it. The following code is similar to that for whole blood but loops over the different tissues to extract the transcriptomic and clinical data, in `teff` format.

```
load("./data/rse_gene.RData")

rse_gene <- scale_counts(rse_gene)

#count data
recountCounts <- assays(rse_gene)$counts
```

```
#count info
recountMap <-rowRanges(rse_gene)

#gene ids
geneids <- unlist(recountMap$symbol)
gtexPd <- colData(rse_gene)

#sample ids
sampid <- gtexPd$sampid

#gene names
geneIncounts <- sapply(strsplit(rownames(recountCounts), "\\."),
                       function(x) x[[1]])
gensymbolscounts <- geneids[geneIncounts]

#counts with mapped gene ID
rmna <- !is.na(gensymbolscounts)
recountCounts<- recountCounts[rmna,]
gensymbolscounts <- gensymbolscounts[rmna]

#identify genes with multiple count data
dup <- names(table(gensymbolscounts))[table(gensymbolscounts)>1]
recountdup <- recountCounts[which(gensymbolscounts%in%dup),]
symbolsdup <- gensymbolscounts[which(gensymbolscounts%in%dup)]

#slect maximum count data per gene
recountmaxdup <- lapply(unique(symbolsdup), function(x)
{
  sapply(1:ncol(recountdup), function(ss) max(recountdup[symbolsdup%in%x,ss]))
})


recountmaxdup <- do.call(rbind, recountmaxdup)

#identify genes with single (not duplicated) count data
recountCountsnodup <- recountCounts[which(!gensymbolscounts%in%dup),]

#re-join both data
recount2 <- rbind(recountCountsnodup, recountmaxdup)
rownames(recount2) <- c(gensymbolscounts[which(!gensymbolscounts%in%dup)],
                        unique(symbolsdup))
colnames(recount2) <- sampid

#load covariates
cov <- read.table("./data/covGTEx.txt",
                  as.is=TRUE, header=TRUE, sep="\t", fill=TRUE)

idscov <- sapply(strsplit(cov[,2],"GTEX-"), function(x) x[[2]])
rownames(cov) <- idscov


#select tissues
tbtissues <- table(gtexPd$smtsd)
```

```
selind <- c(1, 3, 6, 13, 28, 29, 34, 36, 37, 39, 42, 46, 48, 49, 51, 54)

tissues <- names(tbtissues)[selind]

targettissues <- lapply(tissues, function(tt){

  tt <- tissues[i]
  print(tt)
  mask1 <- which(gtexPd$smtsd==tt)
  countstissue <- recount2[,mask1]
  filt <- rowSums(countstissue)

  #select genes with at least 100 count across individuals
  countstissue <-  countstissue[filt > 100,]
  countstissue <- countstissue[!is.na(rownames(countstissue)),]

  #save gene info
  geneids <- data.frame(recountMap)

  whichpheno <- sapply(idscov, function(x) grep(x,colnames(countstissue))[1])

  matchphenoexp<-whichpheno[complete.cases(whichpheno)]

  #match pheno and count data for the same individuals
  countstissue <- countstissue[,matchphenoexp]
  covtissue <- cov[names(matchphenoexp),]

  exprTPM <- lapply(1:ncol(countstissue),
                    function(ss) countstissue[,ss]/sum(countstissue[,ss])*1e6)
  exprTPM <- do.call(cbind,exprTPM)
  exprTPM <- as.matrix(exprTPM)
  colnames(exprTPM) <- colnames(countstissue)
  rownames(exprTPM) <- rownames(countstissue)

  oo <- order(rownames(exprTPM))
  countstissue <- countstissue[oo,]
  exprTPM <- exprTPM[oo,]
  exprTPM <- exprTPM[!is.na(rownames(exprTPM)),]

  cellcomp1 <- deconvolute(exprTPM, "mcp_counter")
  cellnames <- cellcomp1$cell_type
  cm <- matrix(as.numeric(t(cellcomp1)[-1,]),
               ncol=length(cellnames))
  colnames(cm) <- cellnames
  rownames(cm) <- colnames(cellcomp1)[-1]

  pheno <- data.frame(covtissue, data.frame(cm,check.names = FALSE),
                      check.names = FALSE)

  phenored <- data.frame(eff = pheno$"T cell",
                             t = pheno$GENDER, age = pheno$AGE,
                             bmi = pheno$BMI)
```

```
  mod0 <- model.matrix(~ t + eff + age + bmi, data = phenored)
  mod <- model.matrix(~ t:eff + t + eff + age + age + bmi,
                      data = phenored)

  #sva
  ns <- num.sv(countstissue, mod, method = "be")
  ss <- svaseq(countstissue, mod, mod0, n.sv = ns)$sv

  colnames(ss) <- paste("cov", 1:ncol(ss), sep="")

  modss <- cbind(mod, ss)

  #estimation
  design <- model.matrix(~ t:eff + t+ eff + age +bmi,
                         data = phenored)
  v <- voom(countstissue, design = design)

  expr <- v$E
  nmsgenes <- rownames(expr)

  expr <- expr[!is.na(nmsgenes),]
  rownames(expr) <- nmsgenes[!is.na(nmsgenes)]

  ddat <- list(features=t(expr), teffdata=modss)

  rmvars <- !colnames(ddat$teffdata)%in%
      c("(Intercept)","t:eff")

  ddat$teffdata <- ddat$teffdata[,rmvars]

  #target individuals
  res <- target(ddat, htrcells$"T cell",
                effect="positiveandnegative", featuresinf=XYhomol,
                nmcov="age", model="log2")
})

names(targettissues) <- tissues
save(targettissues, file="./data/targettissues.RData")
```

For each tissue we estimated the association between the levels of T cell abundance and the interaction between sex and the targeting of individuals into in the dimorphic groups. The estimated were meta-analized across tissues.

```
load("./data/targettissues.RData")

celltissues<-lapply(targettissues[-c(16)],
               function(AA)
               {
                 out <- AA$summary.model$coefficients
                 out <- out["WTRUE:pf",c(1,2)]
                 out
               })

celltissues <- do.call(rbind,celltissues)
```

```r
datmet <- data.frame(TE = celltissues[,1], SE = celltissues[,2])

tnames <- sapply(strsplit(row.names(datmet), " - ") , function(x) x[[1]])

#Perform meta-analysis
metaresTissues <- meta::metagen(TE, SE, data=datmet,
                                studlab= tnames,
                                level.ci = 0.95)

forest(metaresTissues, layout="JAMA",
       leftlabs=c("Meta-analysis", "Beta (95% CI)"), xlab="Interaction Beta",
       title="", xlim=c(-5,5))
```

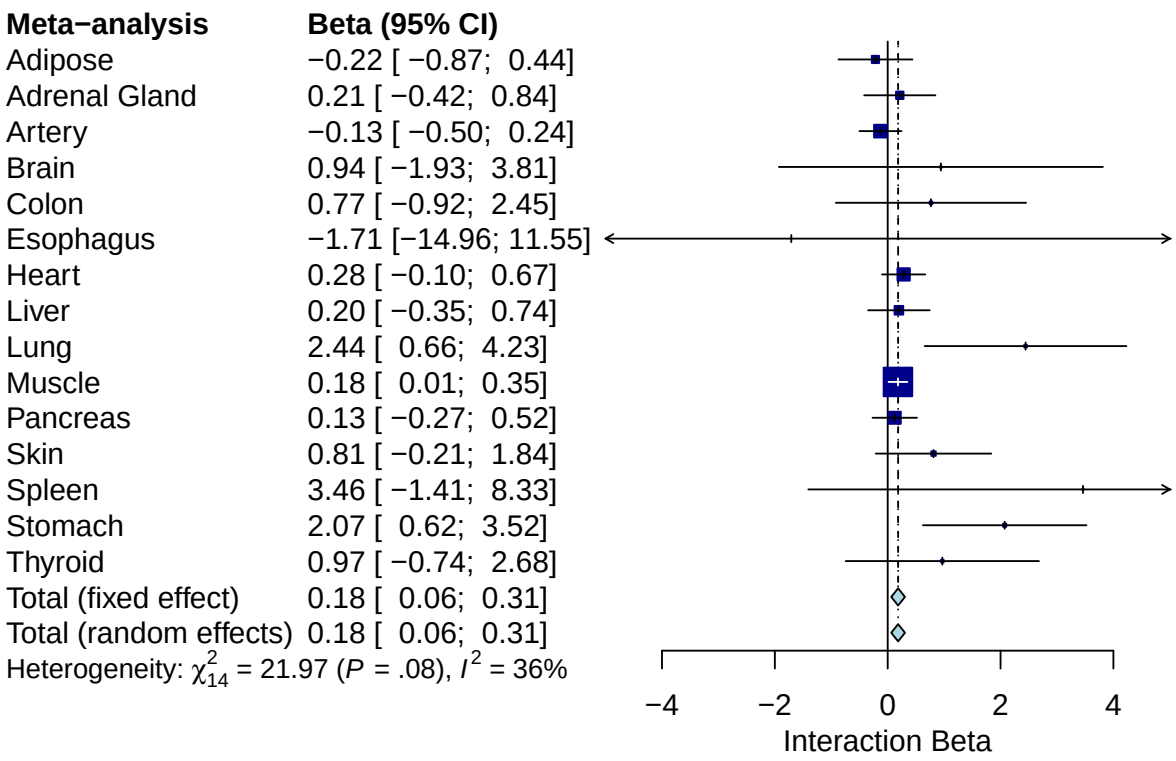
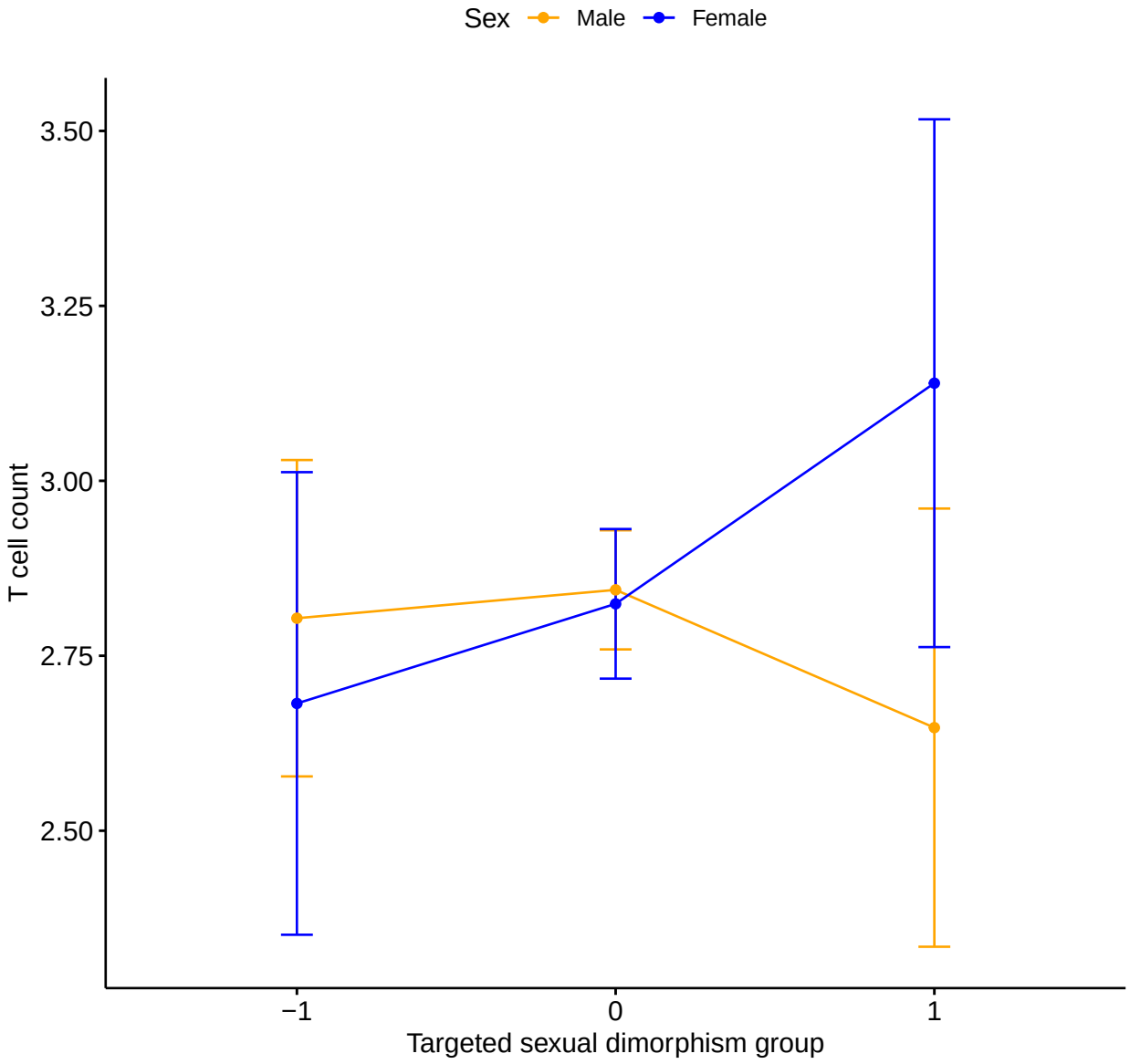


Figure S4

We observed a significant association for lung

```
plotTarget(targettissues$"Lung",
           labs=c("Targeted sexual dimorphism group",
                  "T cell count", "Sex", "Male", "Female"))
```



1.6 Validation

GSE48348

We targeted individuals in two transcriptomic studies by the profiles of positive and negative immune sexual dimorphisms. GSE48348 is a general population study of Estonia. Data was downloaded and sex was inferred from the transcription data on sex chromosomes. We calculated the relative expression of chromosome Y (*Ry*) and cluster males as those with the highest values. T cell abundance in blood

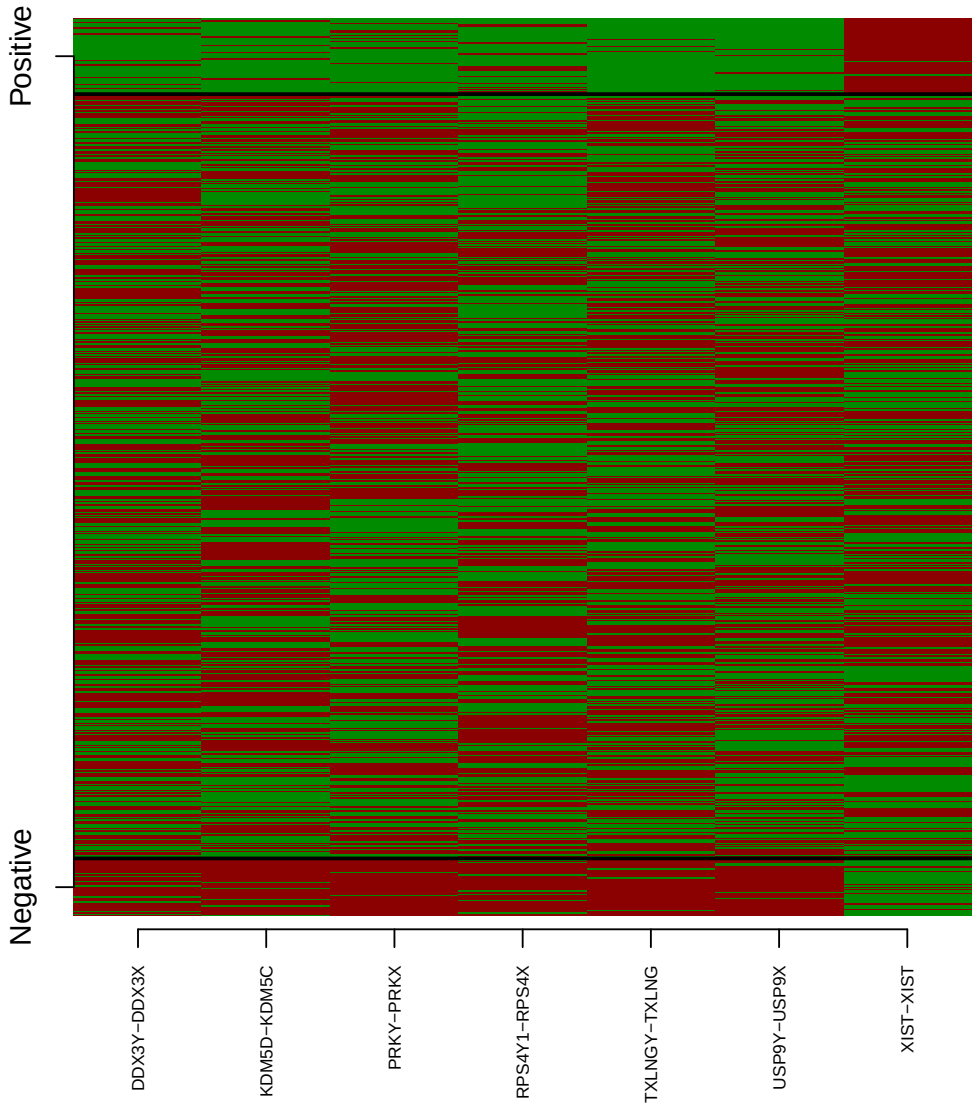
was inferred from transcriptomic data with `mcp_counter`. Residual expression data was obtained from adjusting by surrogate variables analysis that protected the interaction between sex and T cell abundance (`t:eff`).

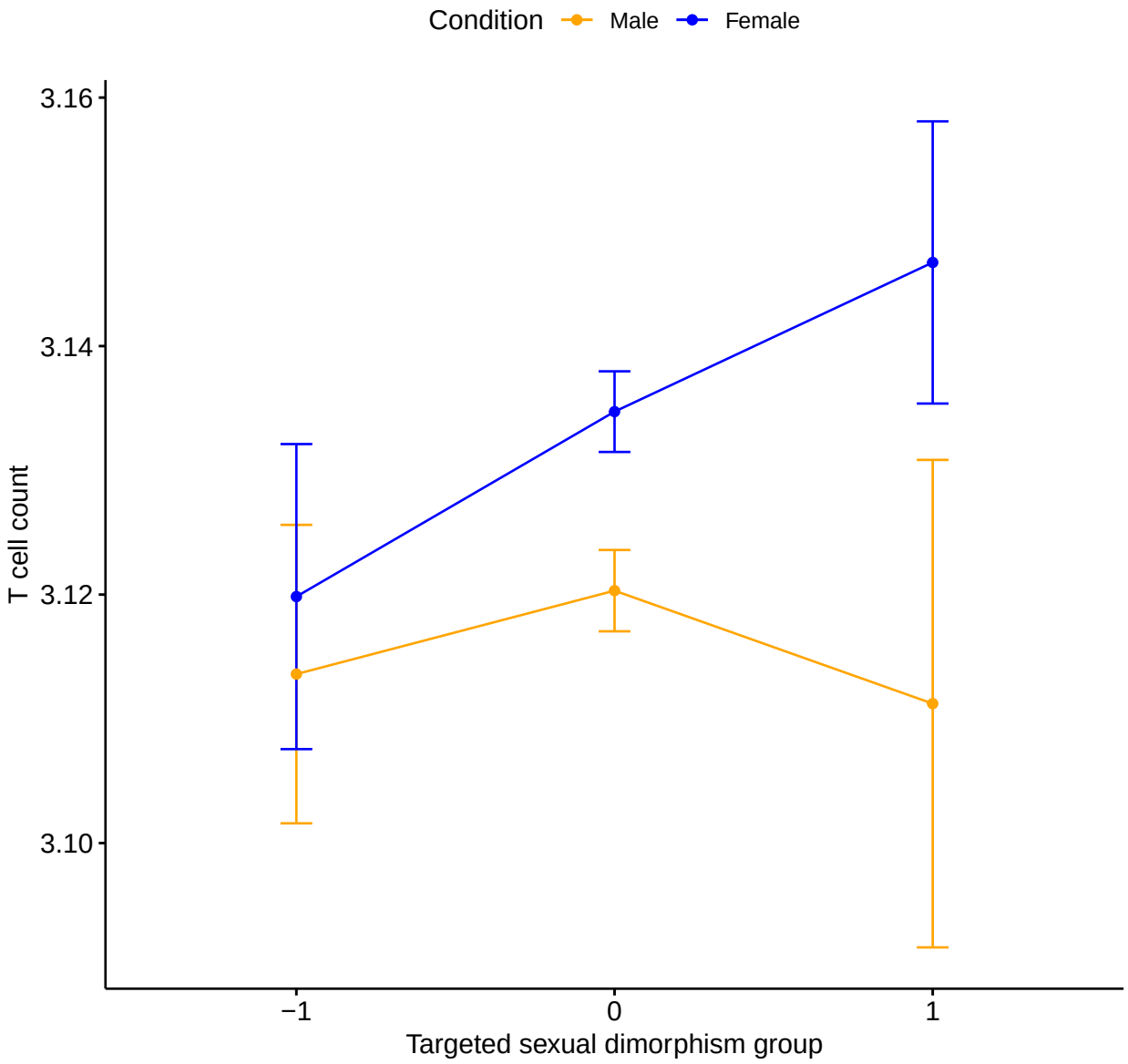
```
gsm<-getGEO("GSE48348",destdir ="./data", AnnotGPL =TRUE)

expr <- exprs(gsm[[1]])
genesIDs <- fData(gsm[[1]])
rownames(expr) <- genesIDs$"Gene symbol"

cellcomp2 <- deconvolute(gsm[[1]], "mcp_counter", arrays=TRUE,column ="Gene symbol")

cellnames <- cellcomp2$cell_type
cm<- matrix(as.numeric(t(cellcomp2)[-1,]), ncol=length(cellnames))
colnames(cm) <- cellnames
rownames(cm) <- colnames(cellcomp2)[-1]
eff <- cm[,"T cell"]

###infer sex from chromose Y data
chrdata <- sapply(strsplit(genesIDs$"Chromosome annotation", ","), function(x) x[1])

chrdata <- sapply(strsplit(chrdata, "Chromosome "), function(x) x[2])

selChr <- which(chrdata%in%"Y")
selnoChr<-which(chrdata%in%as.character(1:22))

expr<-data.frame(log2(exprs(gsm[[1]])-min(exprs(gsm[[1]]),na.rm=TRUE )+1))

subdataChr<-expr[selChr,]
selin<-rowMeans(subdataChr!=0)!=0
subdataChr<-subdataChr[selin,]

subdatanoChr<-expr[selnoChr,]
selin<-rowMeans(subdatanoChr!=0)!=0
subdatanoChr<-subdatanoChr[selin,]

datChrMean<-colMeans(subdataChr,na.rm=TRUE)
datnoChrMean<-colMeans(subdatanoChr,na.rm = TRUE)

chrYdata<-datChrMean-datnoChrMean

gender <- princomp(t(subdataChr))$scores[,1]>0

if(mean(chrYdata[gender])<mean(chrYdata[!gender]))
{
   gender <- as.numeric(gender) + 1
}else{
   gender <- -as.numeric(gender) + 2
}

plot(chrYdata, col=gender)
dev.off()
#################
```

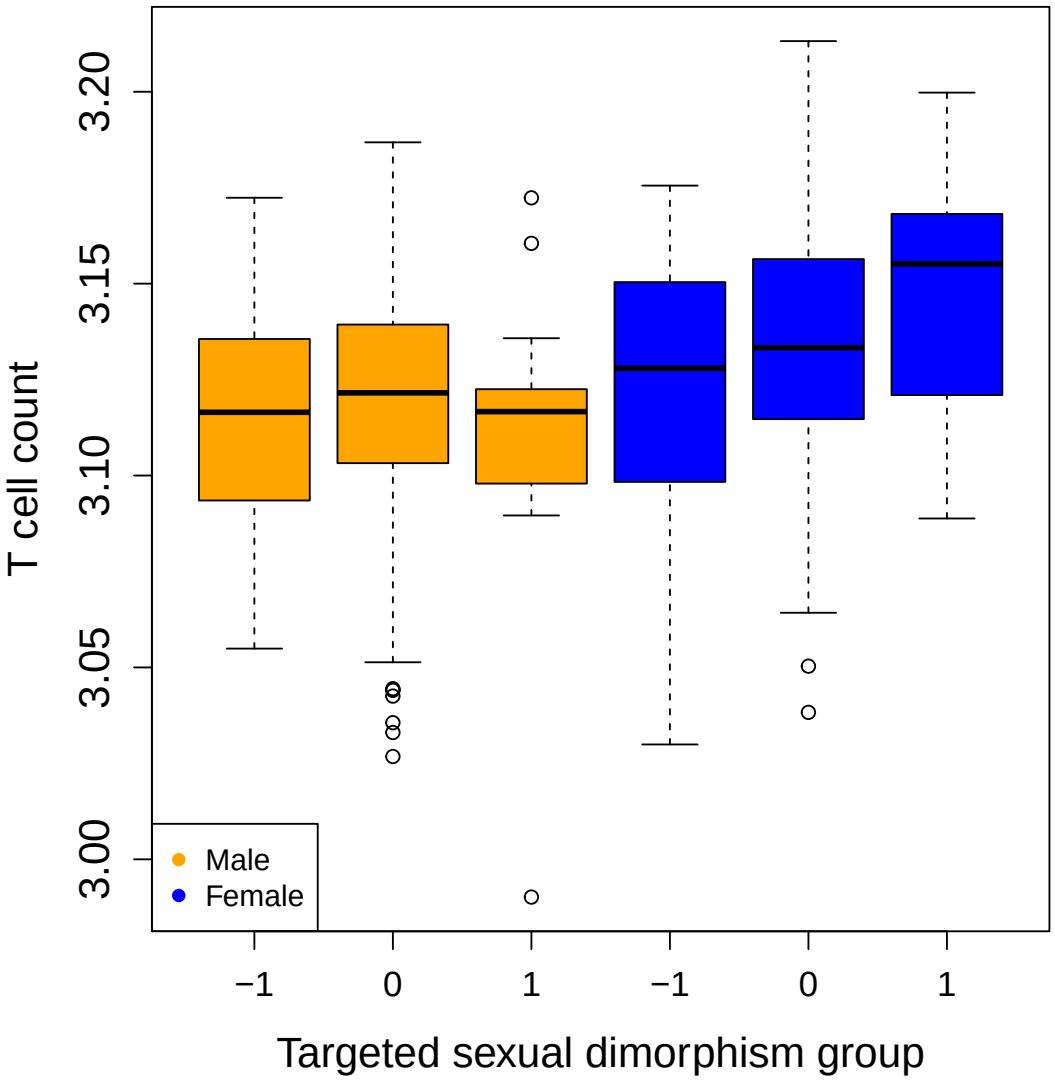
```
#Obtain phenotype and expression data in teff format
phenodat <- data.frame(eff=eff, t=gender)


#get data for complete.cases only
ss <- sapply(1:nrow(expr), function(x) sum(is.na(expr[x,])))
expr <- expr[ss==0, ]

selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]


mod0 <- model.matrix( ~ t + eff , data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff, data = phenodat)
ns <- num.sv(expr, mod, method="leek")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv
colnames(ss) <- paste("cov", 1:ncol(ss), sep="")

modss <- cbind(mod, ss)

Estonia <- list(features=t(expr), teffdata=modss)

rmvars <- !colnames(Estonia$teffdata)%in%
            c("(Intercept)","t:eff")

Estonia$teffdata <- Estonia$teffdata[,rmvars]

save(Estonia, file="data/Estonia.RData")
```

We targeted individuals with the immune cell profiles and confirmed that the classification modulated the association between T-cell abundance and sex.

```
load(file="data/Estonia.RData")

tarEstonia <- target(Estonia, htrcells$"T cell",
                     effect="positiveandnegative",
                     featuresinf=XYhomol,
                     model="log2")


plotTarget(tarEstonia,  labs=c("Targeted sexual dimorphism group", "T cell count",
                               "Condition", "Male", "Female"))

boxPlot(tarEstonia, labs=c("Targeted sexual dimorphism group", "T cell count",
                           "Condition", "Male", "Female"), lg="bottomleft")
```


Positive
Negative
DDX3Y−DDX3X
KDM5D−KDM5C
PRKY−PRKX
RPS4Y1−RPS4X
TXLNGY−TXLNG
USP9Y−USP9X
XIST−XIST


Condition
Male
Female
3.16
3.14
3.12
3.10
T cell count
−1
0
1
Targeted sexual dimorphism group



E-MTAB-3732

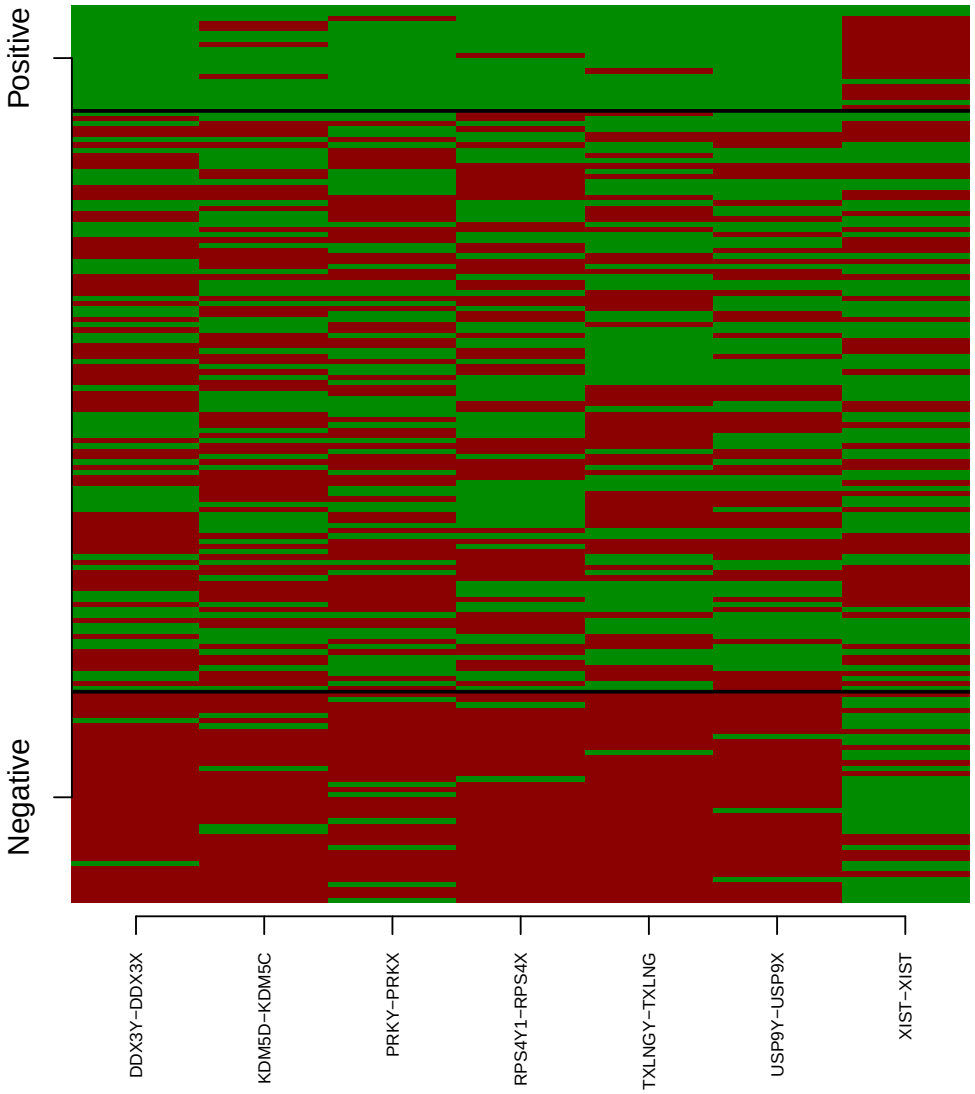
We targeted individuals from a large collection of microarrays E-MTAB-3732 (ArrayExpress), as previously downloaded and analyzed in Caceres et al. 2020 (JNCI;112:913-920). We selected data from peripheral blood and estimated immune cell abundance in all cell types. Residual expression data was obtained from adjusting by surrogate variables analysis that protected the interaction between sex and T cell abundance (`t:eff`).

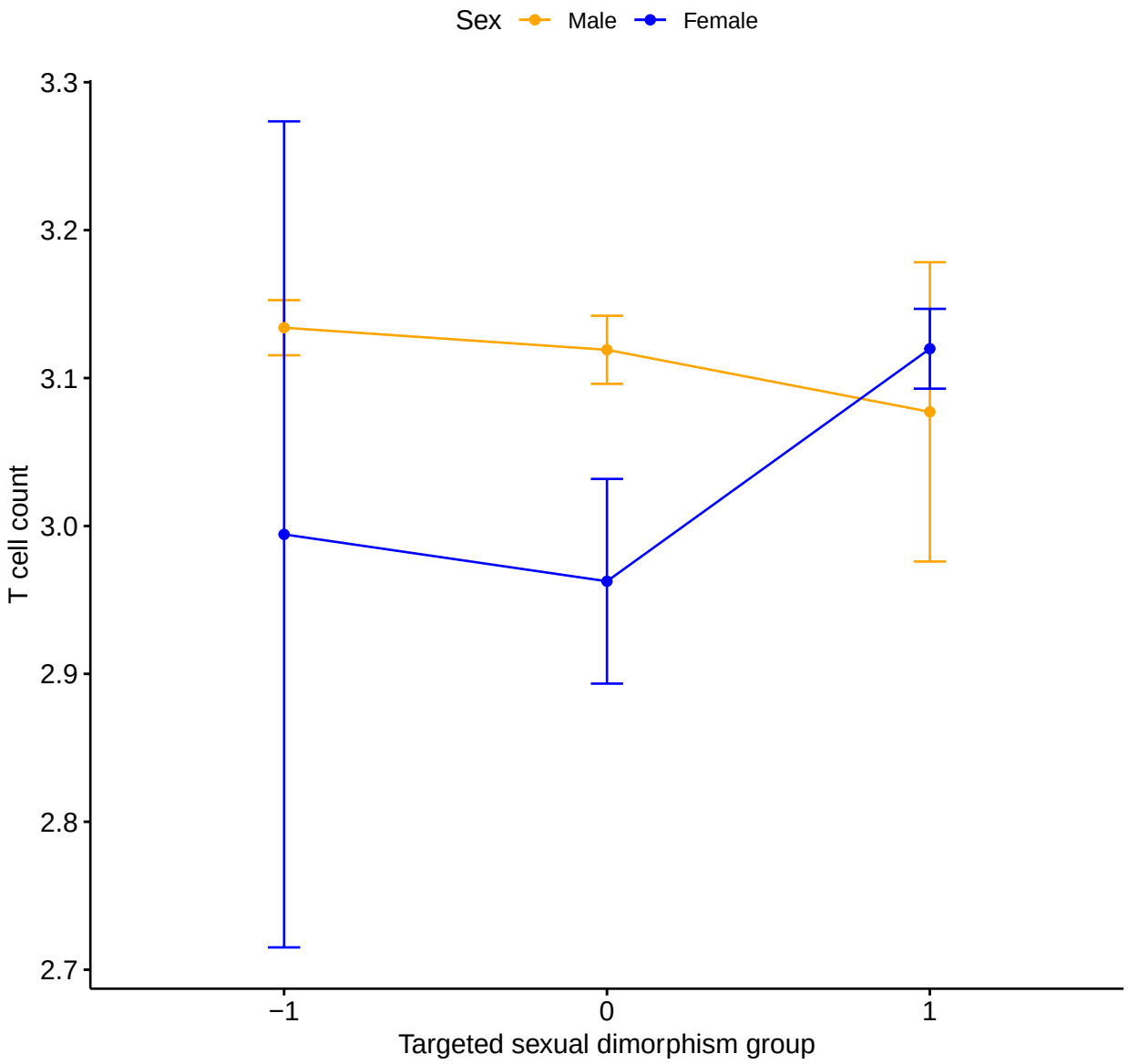
```
load("./data/micro.RData")

cellcomp2 <- deconvolute(expr, "mcp_counter", arrays=TRUE)

##
## >> Running mcp_counter
```

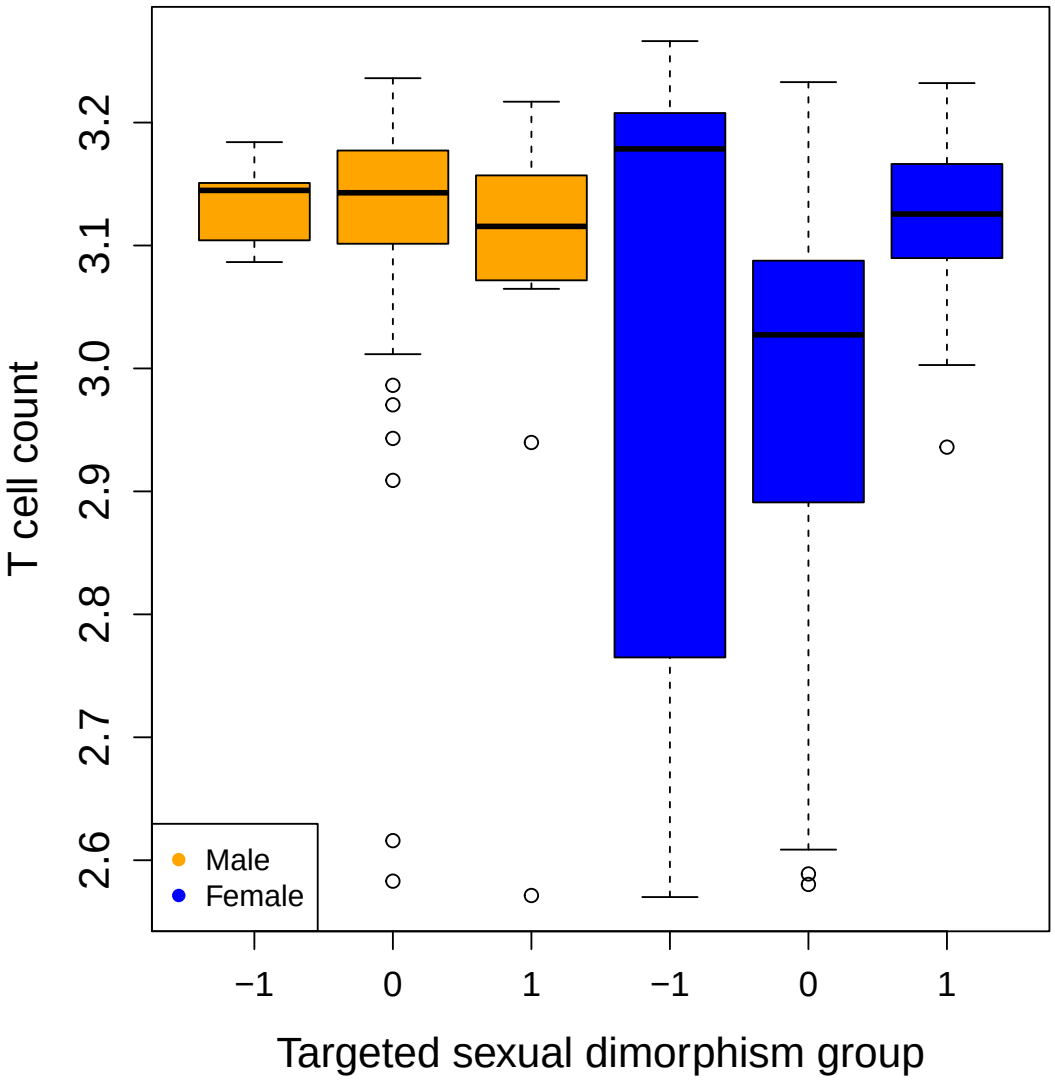
```
cellnames <- cellcomp2$cell_type
cm<- matrix(as.numeric(t(cellcomp2)[-1,]), ncol=length(cellnames))
colnames(cm) <- cellnames
rownames(cm) <- colnames(cellcomp2)[-1]
eff <- cm[,"T cell"]


phenodat <- data.frame(eff=eff, t=sex)

#get data for complete.cases only
ss <- sapply(1:nrow(expr), function(x) sum(is.na(expr[x,])))
expr <- expr[ss==0, ]

selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]
expr <- as.matrix(expr)

mod0 <- model.matrix( ~ t + eff, data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff, data = phenodat)
ns <- num.sv(expr, mod, method="leek")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  2
## Iteration (out of 5 ):1  2  3  4  5

colnames(ss) <- paste("cov", 1:ncol(ss), sep="")

modss <- cbind(mod, ss)

Micro <- list(features=t(expr), teffdata=modss)

rmvars <- !colnames(Micro$teffdata)%in%
            c("(Intercept)","t:eff")

Micro$teffdata <- Micro$teffdata[,rmvars]
```

We targeted individuals with the immune cell profiles and confirmed that the classification modulated the association between T-cell abundance and sex.

```
tarMicro <- target(Micro, htrcells$"T cell",
                   effect="positiveandnegative",
                   featuresinf=XYhomol,
                   model="log2")


plotTarget(tarMicro,  labs=c("Targeted sexual dimorphism group", "T cell count",
                             "Sex", "Male", "Female"))

boxPlot(tarMicro, labs=c("Targeted sexual dimorphism group", "T cell count",
                         "Sex", "Male", "Female"), lg="bottomleft")
```


Positive
Negative
DDX3Y−DDX3X
KDM5D−KDM5C
PRKY−PRKX
RPS4Y1−RPS4X
TXLNGY−TXLNG
USP9Y−USP9X
XIST−XIST


Sex
Male
Female
3.3
3.2
3.1
3.0
2.9
2.8
2.7
T cell count
−1
0
1
Targeted sexual dimorphism group



1.7 Gonosome regualtion

We aimed to test the functional correlations of the inferred groups of differential immune dimorphism with the overall transcription output of sex chromosomes. We retrieved phenotype data and classifications for all cell types, toghether with their infferered sex effects in the test-data.

```
taupf<-lapply(names(htrcells)[-3], function(nm)
{
  eff <- log2(datgtex[[nm]]$phenos[,"eff"]+1)
  names(eff) <- colnames(datgtex[[nm]]$expr)

  age <- datgtex[[nm]]$phenos[,"age"]
  names(age) <- colnames(datgtex[[nm]]$expr)
```

```
  t <-  datgtex[[nm]]$phenos[,"t"]
  names(t) <- colnames(datgtex[[nm]]$expr)


  pf <- target(teffgtex$"T cell", htrcells$"T cell",
               effect="positiveandnegative", featuresinf=XYhomol,
               nmcov="age", model="log2", plot = FALSE)$classification

  #sex effect
  tau <- htrcells[[nm]]$predictions
  names(tau) <-  htrcells[[nm]]$subsids

  list(phenos=data.frame(eff=eff, t=t, age=age, pf=pf), tau=tau)
})

names(taupf) <- names(htrcells)[-3]
```

Chromosome Y

We defined the relative transcription output Ry of chromosome Y in relation to the autosomes. For each individual, we measured the relative expression of the entire chromosome with respect to the autosomes. Having N transcriptps in chromosome Y, with x_e read count for the e-th transcript, we computed

$$y = \sum_{e=1,..N} \frac{\log_2(x_e + 1)}{N}$$

as a measure of the average expression of Y. Likewise, we obtained the mean expression in autosomes

$$a = \sum_{e=1,..M} \frac{\log_2(x_e + 1)}{N}$$

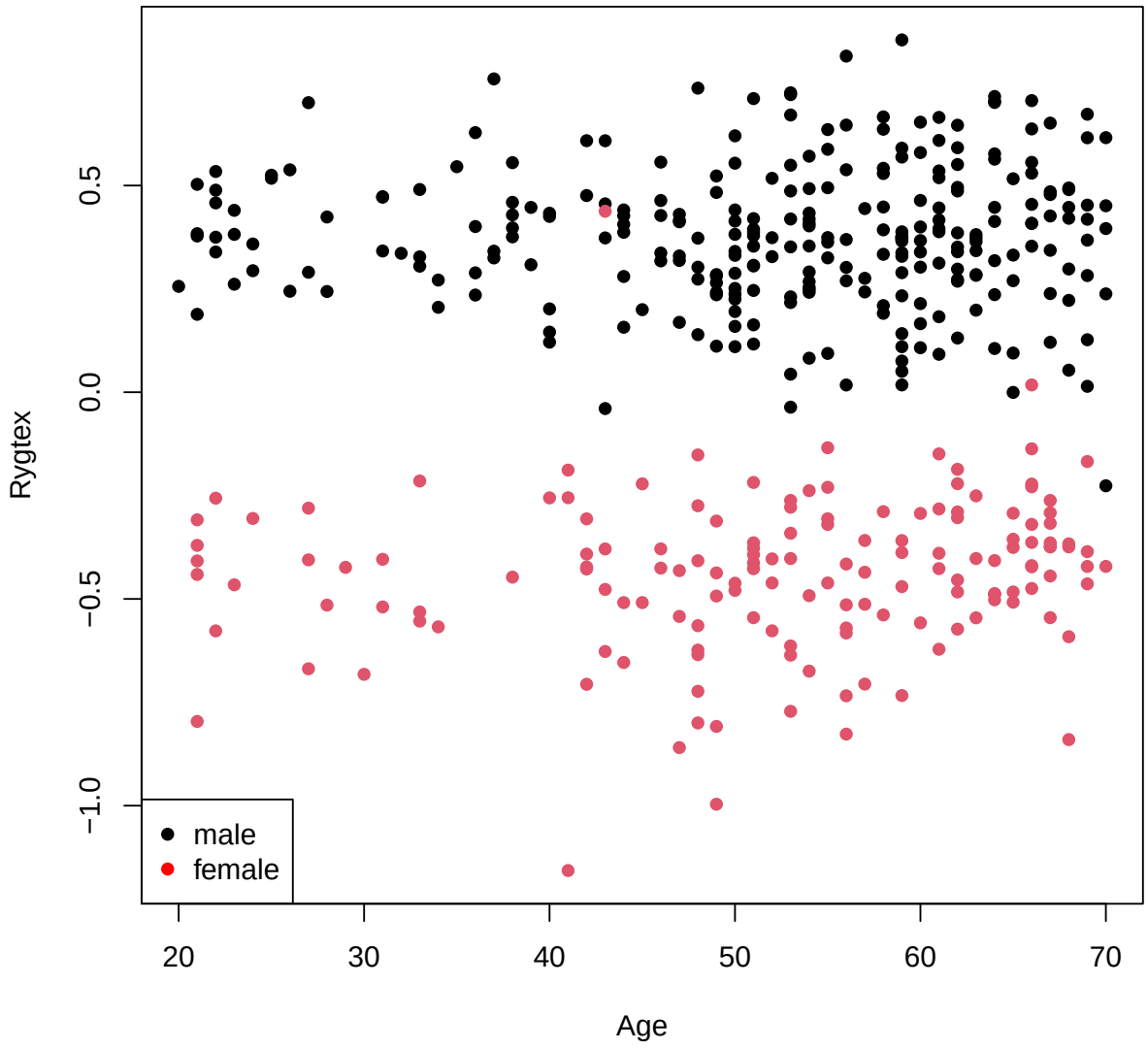
where M is the number transcripts with count data in the autosomes. The relative amount of an individual's Y expression with respect to the individual's autosomes was then defined as

$$Ry = y - a.$$

We confirmed the lower, noise values of Ry, expected for women.

```
load("./data/gtexBlood.RData")
load("./data/geneids.RData")


Ry <- function(expr, ygenes)
{
  selChr <- rownames(expr)%in%ygenes
  selnoChr <- !rownames(expr)%in%ygenes

  #remove transcipts with zero counts across all individuals
  subdataChr <- expr[selChr,]
  selin <-rowMeans(subdataChr!=0)!=0
  subdataChr <- subdataChr[selin,]

  subdatanoChr <- expr[selnoChr,]
  selin <- rowMeans(subdatanoChr!=0)!=0
```

```
  subdatanoChr <- subdatanoChr[selin,]

  #compute mean of log2 counts
  datChrMean <- colMeans(log2(subdataChr+1), na.rm=TRUE)
  datnoChrMean <- colMeans(log2(subdatanoChr+1), na.rm = TRUE)

  #estimate the relative expression of Y with respect to genome
  datChrMean-datnoChrMean
}

expr <- counts

#select the transcripts that are in chrY
iny <- unlist(geneids$symbol[geneids$seqnames=="chrY"])
iny <- iny[complete.cases(iny)]

Rygtex <- Ry(expr, iny)

plot(taupf$`T cell`$phenos[,"age"], Rygtex,
     col=taupf$`T cell`$phenos[,"t"], pch=16,
     xlab="Age")

legend("bottomleft", legend=c("male", "female"),
       col=c("black", "red"), pch=16 )
```



We tested in men the association between immune cell abundance and *Ry* across all cell types

```
Ryteff <- sapply(taupf, function(x){

  dat <- data.frame(x$phenos[names(Rygtex),], Rygtex)

  out <- summary(lm(eff ~  Rygtex + age,
                 dat[dat$t==1,]))$coeff["Rygtex",c(1,4)]

  out
})

t(Ryteff)

##                        Estimate     Pr(>|t|)
```

```
## T cell                  5.314760 1.904747e-37
## T cell CD8+             5.244304 3.831441e-23
## NK cell                 4.807446 5.751183e-25
## B cell                  3.334939 5.010871e-17
## Monocyte                1.933737 1.710414e-08
## Myeloid dendritic cell 2.746367 1.695075e-09
```

We confirmed that *Ry* strongly associated with the classification of males into the different subpopulations of immune dimorphisms.

```
dat <- data.frame(taupf[[1]]$phenos[names(Rygtex),], Rygtex)

rg<- summary(lm(Rygtex ~ pf + age,
                  dat[dat$t==1,]))$coeff["pf",c(1,4)]
rg

##      Estimate      Pr(>|t|)
## -1.331874e-01  2.214415e-13
```

Chromosome X

For females, we defined the relative transcription output of genes that escape X-inactivation *Resc* in relation to those that do not escape. We obtained a list of escapees from Tukiainen, T. et al. 2017 (Nature 550, 244). Having N transcriptps from escapee genes, with x_e read count for the e-th transcript, we computed

$$esc = \sum_{e=1,..N} \frac{\log_2(x_e+1)}{N}$$

as a measure of the average expression from escapees. Likewise, we obtained the mean expression of inactive genes

$$i = \sum_{e=1,..M} \frac{\log_2(x_e+1)}{N}$$

where M is the number transcipts with count data in inactive genes. The relative of escapees in relation to inactive genes is

$$Resc = esc - i.$$

We tested in men the association between immune T cell abundance and *Resc*

```
load("./data/gtexBlood.RData")
load("./data/geneids.RData")

esc <- read.delim("./data/Suppl.Table.1.csv", as.is=TRUE, sep=";", header=TRUE, skip=1)

#genes that always escape
always <- esc$Gene.name[esc$Combined.XCI.status=="escape"]

#genes that are inactive
inactive <- esc$Gene.name[esc$Combined.XCI.status=="inactive"]

#genes in X
inx <- unlist(geneids$symbol[geneids$seqnames=="chrX"])
inx <- inx[complete.cases(inx)]
```

```r
#genes not in X
ninx <- unlist(geneids$symbol[geneids$seqnames!="chrX"])
ninx <- ninx[complete.cases(ninx)]

Raxe <- function(expr, always, inactive)
{
  #remove transcipts with zero counts across all individuals
  selChr<-rownames(expr)%in%inactive
  subdataChr<-expr[selChr,]
  selin<-rowMeans(subdataChr!=0)!=0
  subdataChrx<-subdataChr[selin,]

  selnoChr<-rownames(expr)%in%always
  subdatanoChr <- expr[selnoChr,]
  selin <- rowMeans(subdatanoChr!=0)!=0
  subdatanoChr <- subdatanoChr[selin,]

  #compute mean of log2 counts
  mn <- min(expr,na.rm=TRUE )

   datChrMeanIn <- colMeans(log2(subdataChrx-mn+1), na.rm=TRUE)

   datChrMeanAc <- colMeans(log2(subdatanoChr-mn+1), na.rm = TRUE)

  #estimate the relative expression escapees Vs inactive
  datChrMeanAc- datChrMeanIn
}

#Resc defined as Ralways
Ralways <- Raxe(expr, always, inx[!inx%in%always])

#Relative expression of the genes in the immune
#profile in relation to those in chrX
Rdimor <- Raxe(expr,  as.vector(XYhomol[2,1:6]), inx[!inx%in%XYhomol[2,1:6]])

#Relative expression of the genes in X
#in relation to autosomes
Rx <- Ry(expr, inx)

dat <- data.frame(dat[names(Rdimor),], Ralways=Ralways, Rdimor=Rdimor, Rx=Rx)

summary(lm(Ralways ~ pf + age, dat[dat$t==2,]))
```

```
##
## Call:
## lm(formula = Ralways ~ pf + age, data = dat[dat$t == 2, ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38579 -0.06190  0.01529  0.08822  0.27307
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.5142852  0.0392406  13.106  < 2e-16 ***
```

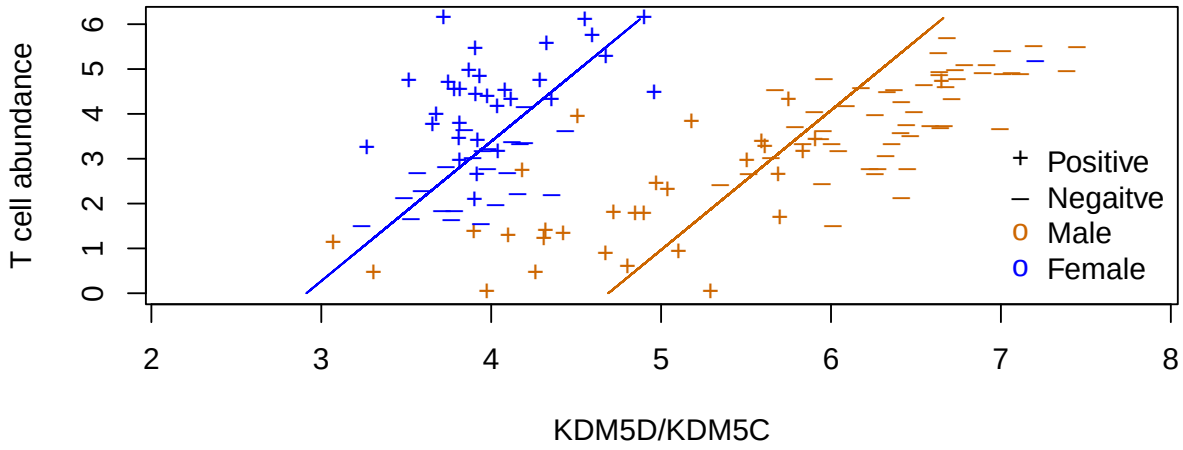
```
## pf           0.0650555  0.0157269   4.137 5.85e-05 ***
## age         -0.0003008  0.0007308  -0.412    0.681
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1168 on 150 degrees of freedom
## Multiple R-squared:  0.1062,Adjusted R-squared:  0.09426
## F-statistic: 8.909 on 2 and 150 DF,  p-value: 0.0002208
```

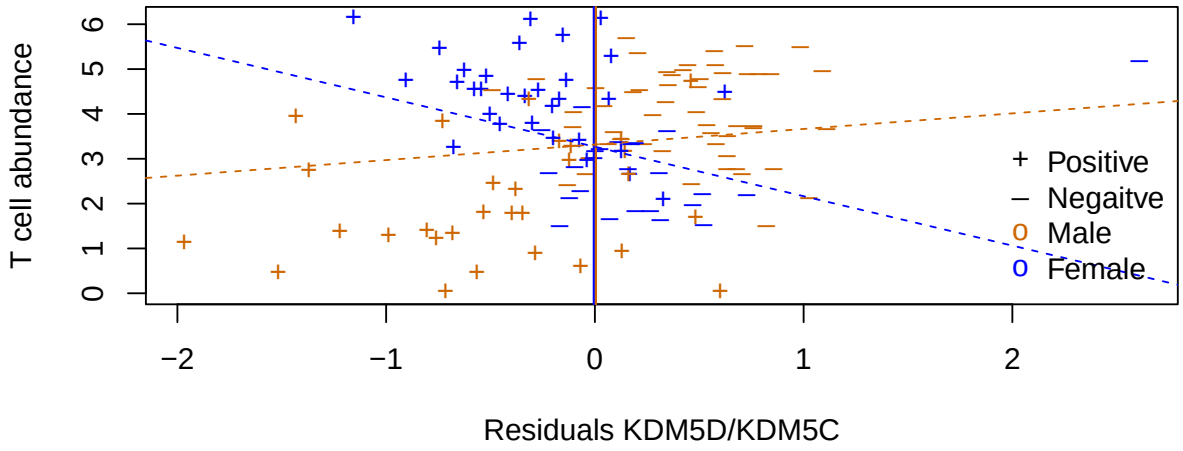
We confirmed that *Resc* strongly associated with the classification of females into the different subpopulations of immune dimorphisms.

```
summary(lm(eff ~ Ralways + age, dat[dat$t==2,]))

##
## Call:
## lm(formula = eff ~ Ralways + age, data = dat[dat$t == 2, ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.82091 -0.83603  0.04907  0.74142  3.11405
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.856064   0.567578   1.508    0.134
## Ralways     4.008930   0.773830   5.181 7.02e-07 ***
## age         0.007730   0.007287   1.061    0.290
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.169 on 150 degrees of freedom
## Multiple R-squared:  0.1543,Adjusted R-squared:  0.143
## F-statistic: 13.68 on 2 and 150 DF,  p-value: 3.479e-06
```

```
KDM5 <- rowMeans(teffgtex$"T cell"$features[,XYhomol[,2]])

dat <- data.frame(taupf[[1]]$phenos[names(Rygtex),], Rygtex)

dat$KDM5 <- KDM5


cc <- as.character(factor(dat$t, labels=c("darkorange3", "blue")))
pch <- factor(dat$pf+2, labels = c("_", NA, "+" ))

#plot this model with residuals
mod <- lm(KDM5 ~ t + eff, data=dat)
rs <- mod$residuals
dat$rs <- rs

cf <- mod$coefficients

xintm <- dat$eff[dat$t==1]
xintf <- dat$eff[dat$t==2]
```

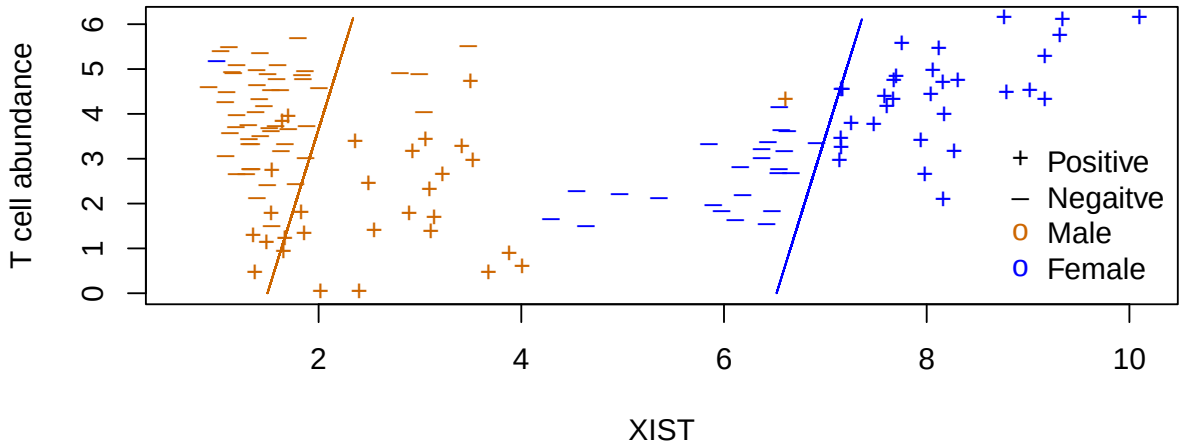
```
ymodm <- cf[1] + 1*cf[2] + cf[3]*xintm
ymodf <- cf[1] + 2*cf[2] + cf[3]*xintf


par(mfrow=c(2,1))

plot(eff ~ KDM5, data=dat, col=cc, pch=as.character(pch),
     xlab="KDM5D/KDM5C", ylab="T cell abundance")
lines(ymodm, xintm, col="darkorange3")
lines(ymodf, xintf, col="blue")


legend("bottomright",
       legend = c("Positive",
                  "Negaitve",
                  "Male", "Female"),
       pch=c("+", "_", "o", "o"),
       col=c("black", "black", "darkorange3", "blue"),
       bty = "n")

plot(eff ~ rs, data=dat, col=cc, pch=as.character(pch),
     xlab="Residuals KDM5D/KDM5C", ylab="T cell abundance")
abline(v=-0.005,col="blue", lwd=1.2)
abline(v=0.005,col="darkorange3", lwd=1.2)

abline(lm(eff ~ rs, data=dat[dat$t==1,]), col="darkorange3", lty=2)
abline(lm(eff ~ rs, data=dat[dat$t==2,]), col="blue", lty=2)

legend("bottomright",
       legend = c("Positive",
                  "Negaitve",
                  "Male", "Female"),
       pch=c("+", "_", "o", "o"),
       col=c("black", "black", "darkorange3", "blue"),
       bty = "n")
```

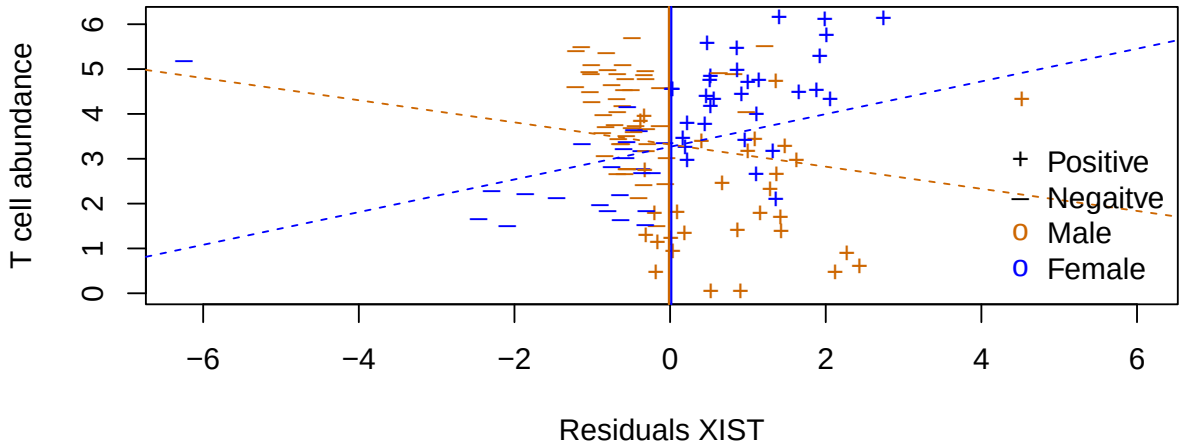





```
XIST <- rowMeans(teffgtex$"T cell"$features[,XYhomol[,7]])

dat$XIST <- XIST


cc <- as.character(factor(dat$t, labels=c("darkorange3", "blue")))
pch <- factor(dat$pf+2, labels = c("_", NA, "+" ))

#plot this model with residuals
mod <- lm(XIST ~ t + eff, data=dat)
rs <- mod$residuals
dat$rs <- rs

cf <- mod$coefficients
```

```
xintm <- dat$eff[dat$t==1]
xintf <- dat$eff[dat$t==2]

ymodm <- cf[1] + 1*cf[2] + cf[3]*xintm
ymodf <- cf[1] + 2*cf[2] + cf[3]*xintf

par(mfrow=c(2,1))

plot(eff ~ XIST, data=dat, col=cc, pch=as.character(pch),
     xlab="XIST", ylab="T cell abundance")
lines(ymodm, xintm, col="darkorange3")
lines(ymodf, xintf, col="blue")


legend("bottomright",
       legend = c("Positive",
                  "Negaitve",
                  "Male", "Female"),
       pch=c("+", "_", "o", "o"),
       col=c("black", "black", "darkorange3", "blue"),
       bty = "n")

plot(eff ~ rs, data=dat, col=cc, pch=as.character(pch),
     xlab="Residuals XIST", ylab="T cell abundance")
abline(v=-0.015,col="darkorange3", lwd=1.2)
abline(v=0.015,col="blue", lwd=1.2)

abline(lm(eff ~ rs, data=dat[dat$t==1,]), col="darkorange3", lty=2)
abline(lm(eff ~ rs, data=dat[dat$t==2,]), col="blue", lty=2)

legend("bottomright",
       legend = c("Positive",
                  "Negaitve",
                  "Male", "Female"),
       pch=c("+", "_", "o", "o"),
       col=c("black", "black", "darkorange3", "blue"),
       bty = "n")
```





Turner syndrome

We downloaded transcriptomic data from GSE46687 study comprising Tuner syndrome patients and females. We compared the *Resc* values for both cases. In the Turner syndrome we inferred the abundance of T cells and correlated it with *Resc*

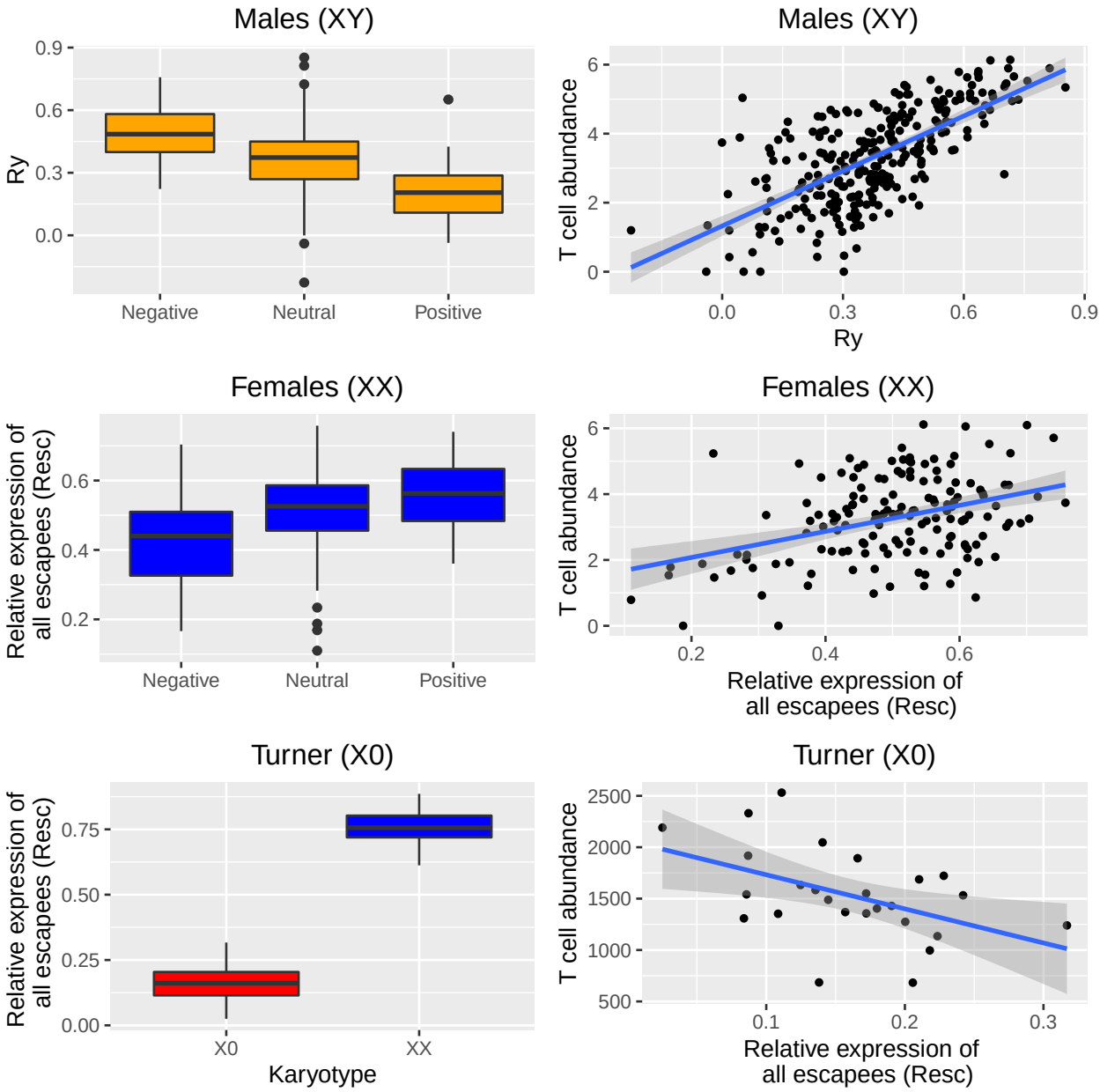
```
gsm <- getGEO("GSE46687",destdir ="./data", AnnotGPL =TRUE)[[1]]

expr <- exprs(gsm)
genesIDs <- fData(gsm)
rownames(expr) <- genesIDs$'Gene symbol'

phenobb <- pData(phenoData(gsm))
```

```r
cellcomp2 <- deconvolute(gsm, "mcp_counter", arrays=TRUE,column ="Gene symbol")
cellnames <- cellcomp2$cell_type
cm<- matrix(as.numeric(t(cellcomp2)[-1,]), ncol=length(cellnames))
colnames(cm) <- cellnames
rownames(cm) <- colnames(cellcomp2)[-1]
eff <- cm[,"T cell"]

t <- as.numeric(factor(phenobb$"karyotype:ch1", labels=c("X0","X0", "XX")))
phenodat <- data.frame(eff, t)

selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

Ralways <- Raxe(expr, always, inx[!inx%in%always])
Rdimor <- Raxe(expr,  as.vector(XYhomol[2,1:6]), inx[!inx%in%XYhomol[2,1:6]])

phenodat$Ralways <- Ralways[rownames(phenodat)]
phenodat$Rdimor <- Rdimor[rownames(phenodat)]

summary(glm(log2(eff) ~ Ralways, data=phenodat[phenodat$t==1, ]))

##
## Call:
## glm(formula = log2(eff) ~ Ralways, data = phenodat[phenodat$t ==
##     1, ])
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.16433  -0.13991   0.04306   0.28577   0.64099
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   10.995      0.228  48.219   <2e-16 ***
## Ralways       -2.977      1.331  -2.237   0.0349 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1760703)
##
##     Null deviance: 5.1065  on 25  degrees of freedom
## Residual deviance: 4.2257  on 24  degrees of freedom
## AIC: 32.545
##
## Number of Fisher Scoring iterations: 2
```

We plotted all previous results in a single panel plot

```
x <- factor(dat$pf[dat$t==1], labels=c("Negative", "Neutral", "Positive"))
y <- dat$Rygtex[dat$t==1]
datplot <- data.frame(x, y)

p1 <- ggplot(datplot, aes(x=x, y=y, fill=x)) +
  geom_boxplot()+ylab("Ry")+xlab("")+
  ggtitle("Males (XY)")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(legend.position="none") +
  scale_fill_manual(values=c("orange", "orange", "orange"))

x <- dat$Rygtex[dat$t==1]
y <- dat$eff[dat$t==1]
datplot <- data.frame(x, y)


p2 <- ggplot(datplot, aes(x=x, y=y)) +
  geom_point(shape=16) +
  geom_smooth(method=lm) +
  ylab("T cell abundance")+xlab("Ry")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(legend.position="none") +
  ggtitle("Males (XY)")


x <- factor(dat$pf[dat$t==2], labels=c("Negative", "Neutral", "Positive"))
y <- dat$Ralways[dat$t==2]
datplot <- data.frame(x, y)

p3 <- ggplot(datplot, aes(x=x, y=y, fill=x)) +
  geom_boxplot()+ylab("Relative expression of \n all escapees (Resc)")+xlab("")+
  ggtitle("Females (XX)")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(legend.position="none") +
  scale_fill_manual(values=c("blue", "blue", "blue"))

x <- dat$Ralways[dat$t==2]
y <- dat$eff[dat$t==2]
datplot <- data.frame(x, y)

p4 <- ggplot(datplot, aes(x=x, y=y)) +
  geom_point(shape=16) +
  geom_smooth(method=lm) +
  ylab("T cell abundance")+xlab("Relative expression of \n all escapees (Resc)")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(legend.position="none") +
  ggtitle("Females (XX)")



x <- factor(phenodat$t, labels=c("XO", "XX"))
y <- phenodat$Ralways
```

```r
datplot <- data.frame(x, y)

p5 <- ggplot(datplot, aes(x=x, y=y, fill=x)) +
  geom_boxplot()+ylab("Relative expression of \n all escapees (Resc)")+xlab("Karyotype")+
  ggtitle("Turner (X0)")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(legend.position="none") +
  scale_fill_manual(values=c("red", "blue"))


x <- phenodat$Ralways[phenodat$t==1]
y <- phenodat$eff[phenodat$t==1]
datplot <- data.frame(x, y)

p6 <- ggplot(datplot, aes(x=x, y=y)) +
  geom_point(shape=16) +
  geom_smooth(method=lm) +
  ylab("T cell abundance")+xlab("Relative expression of \n all escapees (Resc)")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(legend.position="none") +
  ggtitle("Turner (X0)")

grid.arrange(p1,p2,p3,p4,p5,p6, nrow = 3)

## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



2 Cancer risk

We downloaded normalized pre-processed RNAseq data and clinical data from 10 studies from the TCGA using the R package `RTCGA`. The studies comprised tumor and healthy tissue from cancer donors.
We obtained transcription residuals adjusting by surrogate variables and age. Surrogate variables were computed protecting the interaction between age (`t`) and tumor status of the samples (`eff`). Surrogate variables were computed from genes in the genome with more than 15 counts in 25% of the individuals. Formatted data for `teff` was obtained for each cancer study and saved in a single list data-structure.

```
#phenotype data
survInfo <- survivalTCGA(BLCA.clinical, COAD.clinical,
                         KICH.clinical, KIRC.clinical, KIRP.clinical,
                         LIHC.clinical, LUAD.clinical, LUSC.clinical,
```

```
                          READ.clinical, THCA.clinical,
  extract.cols = c("admin.disease_code",
                   "patient.age_at_initial_pathologic_diagnosis",
                   "patient.gender",
                   "patient.race",
                   "patient.primary_therapy_outcome_success"))


survInfo<-survInfo[!duplicated(survInfo$bcr_patient_barcode), ]
rownames(survInfo) <- survInfo$bcr_patient_barcode

cancertypes <- unique(survInfo$admin.disease_code)



#obtain gene expression data for each cancer study,
#compute SVA and transcription residuals

dattcga <- list()

for(type in cancertypes){
  print(type)
  counts <- get(paste0(toupper(type),".rnaseq"))
  counts <-  counts[c(grep("-01A-", counts[,1]),grep("-11A-", counts[,1])), ]
  rownames(counts) <- counts[,1]
  counts <- t(ceiling(counts[,-1]))
  nmcounts <- substr(colnames(counts), 1, 12)


  filter <- apply(counts, 1, function(x) mean(x>15)>0.25)
  counts <- counts[filter,]

  selcancer <- survInfo$admin.disease_code==type
  cancerdat <- survInfo[selcancer, ]
  cancerdat <- cancerdat[nmcounts,]

  genesIDs <- sapply(strsplit(rownames(counts), "\\|") , function(x)  x[[1]] )

  nmsgenes <- sapply(strsplit(genesIDs, "\\|"), function(x) x[1])
  rownames(counts) <- nmsgenes

  commonsubs <- rownames(cancerdat)

  cancercounts<-counts[,!(commonsubs=="NA")]
  cancerdat <-  cancerdat[!(commonsubs=="NA"), ]
  commonsubs <- rownames(cancerdat)


  cancerdat$eff <- rep(1, length(commonsubs))
  cancerdat$eff[grep("\\.1", commonsubs)] <- 0

  cancerdat$t <- as.numeric(as.factor(cancerdat$patient.gender))
  cancerdat$t <- 3 - cancerdat$t
  cancerdat$age <- as.numeric(cancerdat$patient.age_at_initial_pathologic_diagnosis)
```

```
  cc <- complete.cases(cancerdat[c("eff", "t", "age")])
  cancerdat <- cancerdat[cc, ]
  cancercounts<-cancercounts[,cc]

  mod0 <- model.matrix(~ t + eff + age, data=cancerdat)
  mod <- model.matrix(~ t:eff + t + eff + age, data=cancerdat)

  ns <- num.sv(cancercounts, mod, method="be")
  ss <- svaseq(cancercounts, mod, mod0, n.sv=ns)$sv
  colnames(ss) <- paste("cov", 1:ncol(ss), sep="")


  modss <- cbind(mod, ss)

  design <- model.matrix(~ t:eff+ t + eff + age, data=cancerdat)
  v <- voom(cancercounts, design=design)

  expr <- v$E
  nmsgenes <- sapply(strsplit(rownames(expr), "\\|"), function(x) x[1])
  nmsgenes[nmsgenes=="?"] <- NA

  expr <- expr[!is.na(nmsgenes),]
  rownames(expr) <- nmsgenes[!is.na(nmsgenes)]

  event <- cancerdat$patient.vital_status
  time <- as.numeric(cancerdat$times)

  dattcga[[type]] <- list(features=t(expr),
                          teffdata=data.frame(time=time,
                                              event=event,
                                              modss,check.names = FALSE))


  rmvars <- !colnames(dattcga[[type]]$teffdata) %in% c("(Intercept)","t:eff")

  dattcga[[type]]$teffdata <- dattcga[[type]]$teffdata[,rmvars]

  save(dattcga, file="./data/dattcgarisk.RData")
}
```

We targeted individuals in each cancer study by the profiles associated with the differential immune dimorphism.

```
load(file="./data/dattcgarisk.RData")
pdf("./figure/tcgatargetRisk.pdf")
TT <- lapply(dattcga,function(x){
  target(x, htrcells$"T cell",
         effect="positiveandnegative", featuresinf=XYhomol,
         model="binomial", match=0.7, nmcov = "age")
})
dev.off()

## pdf
##   2
```

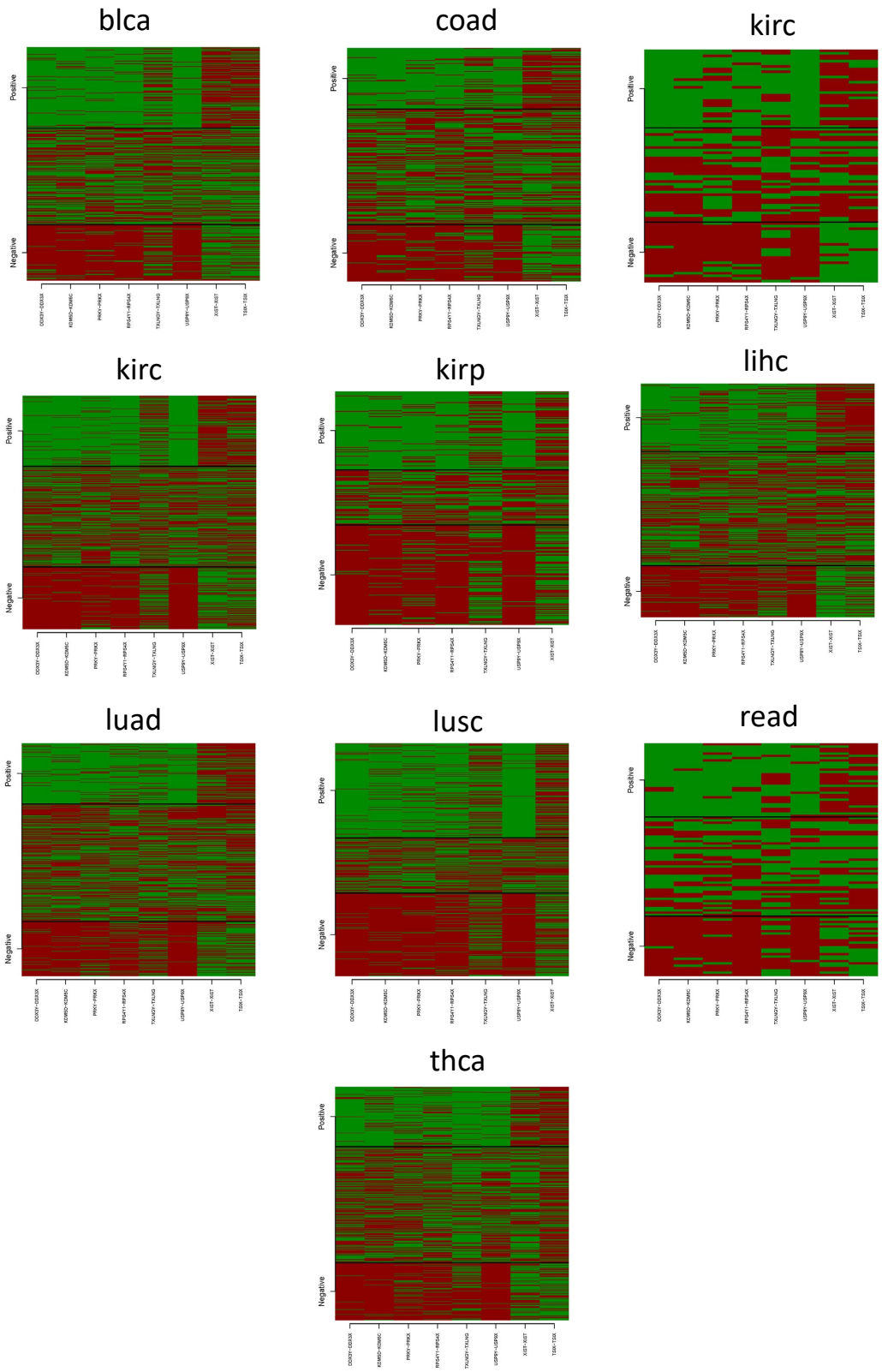

blca
coad
kirc
kirc
kirp
lihc
luad
lusc
read
thca
Positive
Negative

Figure S5

We extracted the classification into positive, negative or neutral dimorphisms of each individual in each cancer study and tested the association of cancer status of the samples with the interaction between sex (`t`) and the classification (`pf`). We performed a pooled analysis and adjusted by age and study

```
#pool-analysis
pf <- sapply(TT, function(x) x$classification)
names(pf) <- NULL
pf <- unlist(pf)

clindat <- lapply(dattcga, function(x) x$teffdata[,c("eff","age", "t")] )
names(clindat) <- NULL
clindat <- do.call(rbind, clindat)

clindat <- data.frame(clindat, pf=pf)

clindat$sex <- factor(ifelse(clindat$t == 1, "male", "female"))
clindat$dimorphism <- factor(clindat$pf, labels = c("negative", "neutral", "positive"))
clindat$study <- unlist(lapply(names(dattcga), function(ii) {
  rep(ii, nrow(dattcga[[ii]][[2]]))
  }))

summary(glm(eff ~ t*pf+age+study, data = clindat, family="binomial"))
```

```
##
## Call:
## glm(formula = eff ~ t * pf + age + study, family = "binomial",
##     data = clindat)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6025   0.3396   0.4247   0.5097   0.9869
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.970652   0.416774   9.527  < 2e-16 ***
## t           -0.133053   0.123354  -1.079 0.280756
## pf           1.420094   0.229046   6.200 5.64e-10 ***
## age         -0.008989   0.004363  -2.060 0.039379 *
## studycoad   -0.663766   0.313262  -2.119 0.034100 *
## studykich   -2.227133   0.345714  -6.442 1.18e-10 ***
## studykirc   -1.125630   0.269494  -4.177 2.96e-05 ***
## studykirp   -0.998432   0.302826  -3.297 0.000977 ***
## studylihc   -1.129937   0.282212  -4.004 6.23e-05 ***
## studyluad   -0.916505   0.276094  -3.320 0.000902 ***
## studylusc   -0.820743   0.278841  -2.943 0.003246 **
## studyread   -0.354192   0.484482  -0.731 0.464734
## studythca   -1.027471   0.291746  -3.522 0.000429 ***
## t:pf        -0.941497   0.155653  -6.049 1.46e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2546.8  on 3910  degrees of freedom
## Residual deviance: 2455.8  on 3897  degrees of freedom
```

```
## AIC: 2483.8
##
## Number of Fisher Scoring iterations: 5
```

We inspected the association wihtin each study and performed a meta-analysis across studies that confirmed the consitency of the association.

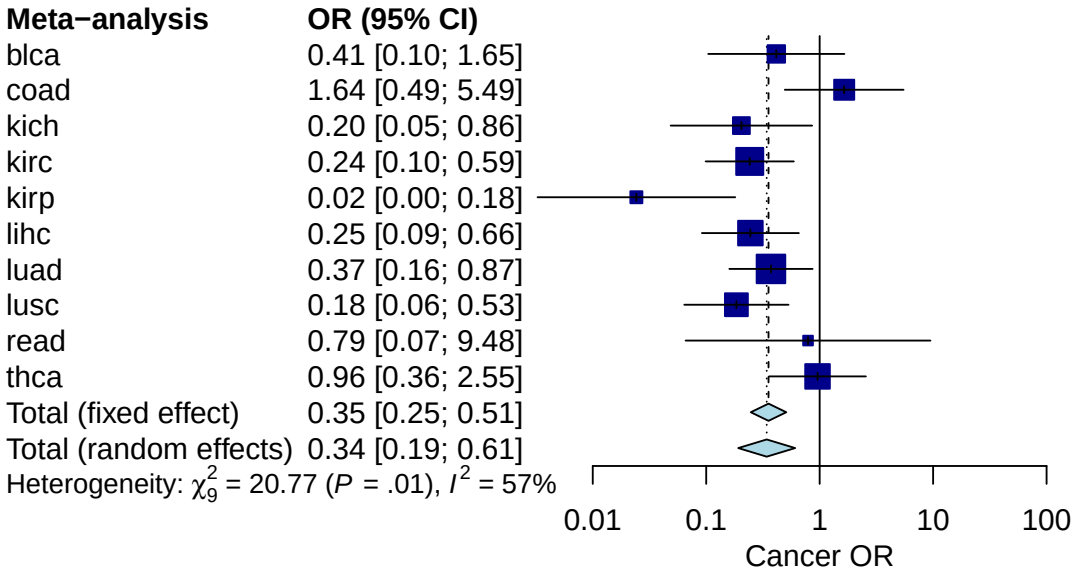
```
#plot proportions of cancer cell by sex and classification
load(file="./data/dattcgarisk.RData")
lbs <- c("Negative", "Neutral", "Positive")

for(type in names(dattcga)){
  png(paste("./figure/", "plot.png", sep=type))

  pls <- list()
  for(pf in c(-1,0,1)){
  sel <- which(clindat$study==type & clindat$pf==pf)
  dt <- clindat[sel,c("t", "eff")]
  dt$t <- factor(dt$t, labels=c("male", "female"))
  dt$eff <- factor(dt$eff, labels=c("N", "C"))

  pls[[pf+2]] <- ggplot(dt, aes(eff, fill = t)) +
    geom_bar(position = "fill") +
    scale_y_continuous(labels = scales::percent)+
    labs(x = lbs[match(pf, c(-1,0,1))], y="", fill = "Sex")+
    scale_fill_manual(values = c("male" = "orange", "female" = "blue"))+
    theme(text = element_text(size=rel(4))) +
    theme(legend.text = element_text(size=10))+
    theme(legend.position="none")

  }

  grid.arrange(pls[[1]], pls[[2]], pls[[3]], nrow = 1)

  dev.off()

}
```

The meta-analysis was performed with `metagen`

```
#meta-analysis
cellAl<-lapply(TT,
               function(AA)
               {
                 out <- AA$summary.model$coefficients
                 out <- out["WTRUE:pf",c(1,2)]
                 out
               })

cellAl <- do.call(rbind,cellAl)

datmet <- data.frame(TE = cellAl[,1], SE = cellAl[,2])

#Perform meta-analysis
```

```
metaresTCGA <- meta::metagen(TE, SE, data=datmet,
                             studlab= row.names(datmet),
                             level.ci = 0.95, sm="OR")
```

```
forest(metaresTCGA, layout="JAMA",
       leftlabs=c("Meta-analysis", "OR (95% CI)"), xlab="Cancer OR", title="")
```



3 Cancer survival

We downloaded RNAseq data from tumor tissues of cancer patients in 13 studies of the TCGA with phenotype data on survival. Targeting of this data into the positive, negative and neutral profiles of immune sexual dimorphism was performed with transcriptomic residuals that were adjusted by age and suroogate variables. The surrogate variables were computed protecting the interaction between sex (`t`) and time to event (`eff`)

```
load("./data/dattcga.RData")
pdf("./figure/tcgatarget.pdf")
TT <- lapply(dattcga,function(x){
  target(x, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="hazard", match=0.7, nmcov = "age")
})
dev.off()

## pdf
##   2
```

We first fitted a Cox proportional-hazard models for the pooled data across all studies adjusting for age and study. In this studies, we did not observe an association between survival and sex. We then tested the association between survival and the interaction between sex and the classification of the individuals into their dimorphism group (`pf`)

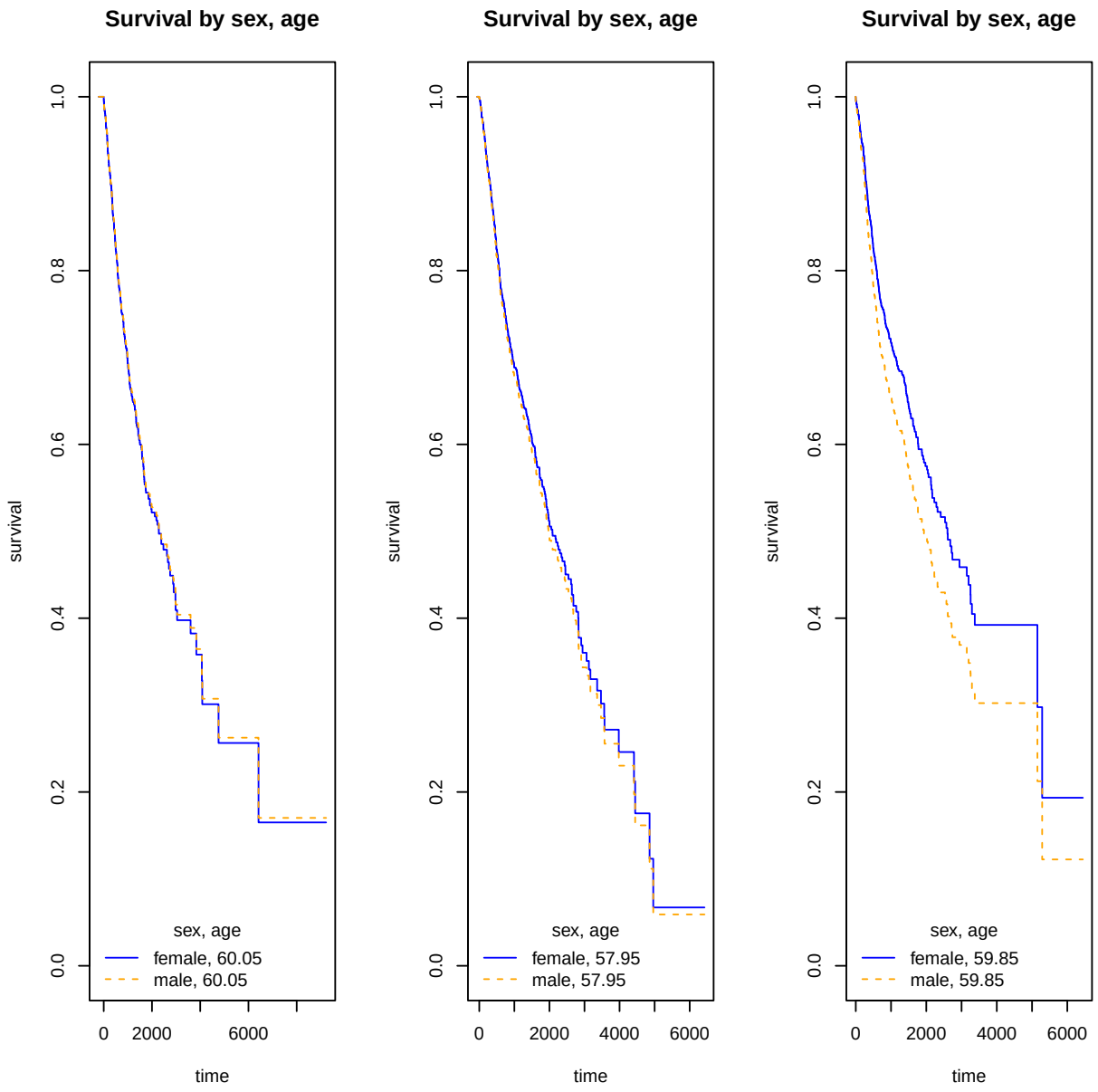
```
#pool-analysis
pf <- sapply(TT, function(x) x$classification)
names(pf) <- NULL
pf <- unlist(pf)

clindat <- lapply(dattcga, function(x) x$teffdata[,c("time", "event", "age", "t")] )

clindat <- do.call(rbind, clindat)

ids <- strsplit(rownames(clindat), "\\.")
ids <- data.frame(do.call(rbind, ids))
names(ids) <- c("study", "ID")
rownames(clindat) <- ids$ID

clindat <- data.frame(ids, clindat, pf=pf[rownames(clindat)])


clindat$sex <- factor(ifelse(clindat$t == 1, "male", "female"))
clindat$dimorphism <- factor(clindat$pf, labels = c("negative", "neutral", "positive"))
clindat$study <- factor(clindat$study)


#no association with sex
coxph(Surv(time, event) ~ t+age, data = clindat)

## Call:
## coxph(formula = Surv(time, event) ~ t + age, data = clindat)
##
##          coef exp(coef)  se(coef)      z      p
## t   -0.063007  0.938937  0.058598 -1.075  0.282
```

```
## age  0.029438  1.029876  0.002142 13.744 <2e-16
##
## Likelihood ratio test=209.5  on 2 df, p=< 2.2e-16
## n= 5129, number of events= 1238

#correlation between sex and dimorphic group
chisq.test(table(clindat$t, clindat$pf))

##
##  Pearson's Chi-squared test
##
## data:  table(clindat$t, clindat$pf)
## X-squared = 194.02, df = 2, p-value < 2.2e-16

#association with the interaction between sex and dimorphic group
coxph(Surv(time, event) ~ t*pf+age+study, data = clindat)

## Call:
## coxph(formula = Surv(time, event) ~ t * pf + age + study, data = clindat)
##
##                coef exp(coef)  se(coef)      z        p
## t          0.015358  1.015476  0.060019  0.256  0.79804
## pf         0.285856  1.330901  0.112566  2.539  0.01110
## age        0.031568  1.032072  0.002584 12.217  < 2e-16
## studycoad -0.522604  0.592975  0.185770 -2.813  0.00491
## studygbm   1.757832  5.799851  0.147817 11.892  < 2e-16
## studyhnsc  0.258860  1.295453  0.125452  2.063  0.03907
## studykirc -0.256599  0.773679  0.126349 -2.031  0.04227
## studykirp -0.960418  0.382733  0.202573 -4.741 2.13e-06
## studylaml  1.416328  4.121957  0.145718  9.720  < 2e-16
## studylgg   0.254804  1.290208  0.157441  1.618  0.10557
## studylihc  0.098643  1.103672  0.145190  0.679  0.49688
## studyluad  0.101701  1.107052  0.133945  0.759  0.44769
## studylusc  0.082387  1.085876  0.126445  0.652  0.51468
## studypaad  0.741850  2.099817  0.163969  4.524 6.06e-06
## studypcpg -1.660886  0.189971  0.423535 -3.921 8.80e-05
## studyread -0.683613  0.504790  0.390721 -1.750  0.08018
## studystad  0.224780  1.252047  0.390711  0.575  0.56508
## studythca -1.707043  0.181401  0.288789 -5.911 3.40e-09
## t:pf      -0.184974  0.831126  0.080651 -2.294  0.02182
##
## Likelihood ratio test=714.4  on 19 df, p=< 2.2e-16
## n= 5129, number of events= 1238
```

We plotted the interaction with survival

```
mod1 <- coxph(Surv(time, event) ~ sex+age, data = clindat, subset = which(clindat$pf==-1))

mod2 <- coxph(Surv(time, event) ~ sex+age, data = clindat, subset = which(clindat$pf==0))

mod3 <- coxph(Surv(time, event) ~ sex+age, data = clindat, subset = which(clindat$pf==1))

par(mfrow=c(1,3))
  RcmdrPlugin.survival::plot.coxph(mod1, byfactors=TRUE,
                                   col=c("blue", "orange"))
```

```
  RcmdrPlugin.survival::plot.coxph(mod2, byfactors=TRUE,
                                   col=c("blue", "orange"))
  RcmdrPlugin.survival::plot.coxph(mod3, byfactors=TRUE,
                                   col=c("blue", "orange"))
```



and tested the hazard models on the dimorphism classification stratifying by sex

```
#associations with dimorphic group stratified by sex
#females
coxph(Surv(time, event) ~ pf + age + study , data = clindat,
      subset = which(clindat$sex=="female"))

## Call:
## coxph(formula = Surv(time, event) ~ pf + age + study, data = clindat,
##     subset = which(clindat$sex == "female"))
```

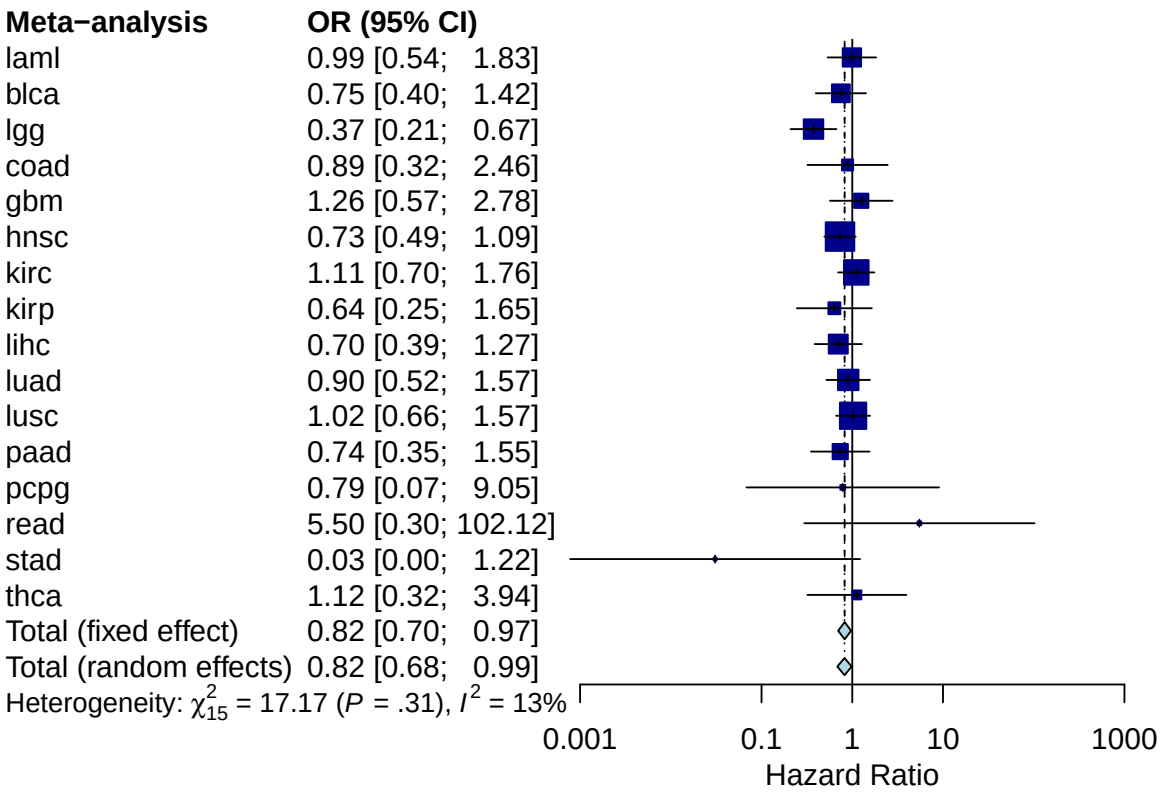
```
##
##                coef exp(coef)  se(coef)      z        p
## pf         -0.088909  0.914928  0.067562 -1.316  0.18819
## age         0.031705  1.032213  0.004218  7.517 5.61e-14
## studycoad -0.858014  0.424003  0.319606 -2.685  0.00726
## studygbm    1.937905  6.944188  0.253949  7.631 2.33e-14
## studyhnsc   0.145275  1.156357  0.224486  0.647  0.51754
## studykirc -0.399329  0.670770  0.218363 -1.829  0.06744
## studykirp -0.739058  0.477564  0.395032 -1.871  0.06136
## studylaml   1.381848  3.982253  0.240985  5.734 9.80e-09
## studylgg    0.232419  1.261648  0.256128  0.907  0.36418
## studylihc   0.069475  1.071945  0.240077  0.289  0.77229
## studyluad   0.074635  1.077491  0.213557  0.349  0.72672
## studylusc -0.042430  0.958458  0.237143 -0.179  0.85800
## studypaad   0.554160  1.740478  0.274866  2.016  0.04379
## studypcpg -2.379061  0.092638  0.732648 -3.247  0.00117
## studyread -0.760322  0.467516  0.529947 -1.435  0.15137
## studystad   0.083439  1.087019  0.728801  0.114  0.90885
## studythca -1.930354  0.145097  0.383715 -5.031 4.89e-07
##
## Likelihood ratio test=364.1  on 17 df, p=< 2.2e-16
## n= 2075, number of events= 471

#males
coxph(Surv(time, event) ~ pf + age + study , data = clindat,
      subset = which(clindat$sex=="male"))

## Call:
## coxph(formula = Surv(time, event) ~ pf + age + study, data = clindat,
##     subset = which(clindat$sex == "male"))
##
##                coef exp(coef)  se(coef)      z        p
## pf          0.099959  1.105126  0.045285  2.207  0.02729
## age         0.031469  1.031969  0.003341  9.420  < 2e-16
## studycoad -0.314789  0.729943  0.228325 -1.379  0.16799
## studygbm    1.656603  5.241474  0.182561  9.074  < 2e-16
## studyhnsc   0.301931  1.352467  0.152344  1.982  0.04749
## studykirc -0.181946  0.833646  0.155507 -1.170  0.24199
## studykirp -0.999614  0.368022  0.236250 -4.231 2.32e-05
## studylaml   1.428624  4.172954  0.185166  7.715 1.21e-14
## studylgg    0.235310  1.265301  0.203883  1.154  0.24844
## studylihc   0.101612  1.106953  0.184470  0.551  0.58175
## studyluad   0.088757  1.092815  0.179680  0.494  0.62133
## studylusc   0.136913  1.146728  0.149914  0.913  0.36110
## studypaad   0.865107  2.375261  0.204721  4.226 2.38e-05
## studypcpg -1.023636  0.359286  0.519451 -1.971  0.04877
## studyread -0.666664  0.513418  0.589419 -1.131  0.25803
## studystad   0.280651  1.323991  0.463318  0.606  0.54469
## studythca -1.324878  0.265835  0.463397 -2.859  0.00425
##
## Likelihood ratio test=353  on 17 df, p=< 2.2e-16
## n= 3054, number of events= 767
```

We then performed a meta-analysis across studies and confirmed the consistency of the association and

lack of heterogeneity of the estimates.

```
#meta-analysis
cellA1<-lapply(TT,
  function(AA)
  {
    out <- AA$summary.model$coefficients
    out <- out["WTRUE:pf",c(1,3)]
    out
  })

cellA1 <- do.call(rbind,cellA1)

datmet <- data.frame(TE = cellA1[,1], SE = cellA1[,2])

#Perform meta-analysis
metaresTCGA <- meta::metagen(TE, SE, data=datmet,
                 studlab= names(dattcga),
                 level.ci = 0.95, sm="OR")
```

```
forest(metaresTCGA, layout="JAMA",
       leftlabs=c("Meta-analysis", "OR (95% CI)"), xlab="Hazard Ratio", title="")
```



3.1 Methlylation on cancer samples

We downloaded methylation data using the `curatedTCGAData` R-package. We retrieved data from the gonosomes in 13 cancer studies.

```
library(curatedTCGAData)

nmscan <- c("LAML","BLCA","LGG","COAD","GBM","HNSC",
            "KIRC","KIRP","LIHC","LUAD","LUSC","PAAD",
            "PCPG")


met0 <- list()
for(i in nmscan){
```

```
  print(i)
  lggmae <- curatedTCGAData(i, "Methylation", FALSE)
  cpgid <- rowData(lggmae[[1]])
  selchrs <- cpgid$Chromosome%in%c("X", "Y")
  met0[[i]] <- lggmae[[1]][selchrs,]
  save(met0, file="./data/met0.RData")
}


save(cpgid, file="./data/cpgid.RData")
```

We merged clinical data with methylation data from gonosomes. Methylation data for chrX was available for all studies but for chrY was available for 6. We selected CpGs with less than 20% of missing values across subjects.

```
load("./data/met0.RData")
load("./data/cpgid.RData")

#probes in Y and gene subset
cpgidy <- rownames(cpgid)[cpgid$Gene_Symbol%in%iny]

#probes of genes in Y that define the dimorphism profiles
cpgidhomy <- rownames(cpgid)[cpgid$Gene_Symbol%in%as.vector(XYhomol[1,1:6])]

#probes in X
cpgidinx <- rownames(cpgid)[cpgid$Gene_Symbol%in%inx]

#probes in escapee genes
cpgidalways <- rownames(cpgid)[cpgid$Gene_Symbol%in%always]

#probes of inactive genes
cpgidinactive <- rownames(cpgid)[cpgid$Gene_Symbol%in%inactive]

#probes of genes in X that define the dimorphism profiles
cpgidhom <- rownames(cpgid)[cpgid$Gene_Symbol%in%as.vector(XYhomol[2,1:6])]


#obtain methylation data across studies
cpgsall <- lapply(met0, function(lggmet){
  lgg <- matrix(as.numeric(assay(lggmet)), ncol=ncol(lggmet))
  colnames(lgg) <- colnames(assay(lggmet))
  rownames(lgg) <- rownames(assay(lggmet))
  lgg<-t(lgg)

  lgg <- lgg[c(grep("01A", rownames(lgg)),grep("03A", rownames(lgg))),]
  rownames(lgg) <- substr(rownames(lgg), 1, 12)


  cpgsstudy <- lgg
  cgnames<-grep("cg", colnames(lgg))
  cpgsstudy[,cgnames]
})

names(cpgsall) <- NULL
```

```
cpgnames <- sapply(cpgsall, colnames)

#select cpgs that are common in all 13 studies
tb <- table(unlist(cpgnames))
cpgnames <- names(tb)[tb ==13]

cpgsallsel <- lapply(cpgsall, function(x) x[,cpgnames])
cpgsallsel <- do.call(rbind, cpgsallsel)

#select cpgs with less than 20% of missings across subjects
selmis<-colMeans(is.na(cpgsallsel))<0.2
cpgsallsel <- cpgsallsel[,selmis]

#names of subjects with clinical and mathlylation data
cmnames <- intersect(rownames(cpgsallsel), rownames(clindat))

#merge phenotype-methylation data with study variable,
#and selecting those studies with more than 100 samples.
ph <- cbind(clindat[cmnames, ], cpgsallsel[cmnames, ])
selstudies <- names(table(ph$study))[table(ph$study)>100]
ph <- ph[ph$study%in%selstudies,]
ph$study <- factor(ph$study)

#There is fewer studies with methylation data in chrY
selstudy <- sapply(cpgsall, function(x) sum(cpgidhomy%in%colnames(x)))>1
cpgsall <- cpgsall[selstudy]

names(met0)[selstudy]

## [1] "BLCA" "LGG"  "HNSC" "LIHC" "PAAD" "PCPG"

#Add the cpgs in chrY for these studies
cpgnames <- sapply(cpgsall, colnames)
tb <- table(unlist(cpgnames))
cpgnames <- names(tb)[names(tb)%in%cpgidy]


cpgsallsel <- lapply(cpgsall, function(x) x[,cpgnames])
cpgsallsel <- do.call(rbind, cpgsallsel)

#select cpgs in Y with less than 20% of missings across subjects
selmis<-colMeans(is.na(cpgsallsel))<0.2
cpgsallsel <- cpgsallsel[,selmis]

selcpg <- colnames(cpgsallsel)[!colnames(cpgsallsel) %in% colnames(ph)]
selmat <- data.frame(cpgsallsel)[rownames(ph), selcpg]

#merge data of cpgs in chrY in previous merged data
ph <- data.frame(ph, selmat)
```

We then selected probes with high values of methylation (> 0.8). For each individual, the values of the hypermethylated probes were averaged across all the genes inactivated in chromosome X, genes that escape chromosome X inactivation, and genes that define the sexual dimorphism profile. We also computed the average hypermethylation in chromosome Y and in the genes that define the sexual dimorphism profile.

We then tested the association between the hypermethylation of the genes in the sexual dimorphism profile with the classification of the individuals in the dimorphism profiles, stratifying by sex.

```
####select hypermethylated probes
phpos <- ph
phpos[ph<0.8] <- NA


#cpgs in X genes within the dimorphism profiles
ph$metdimhiper <- rowMeans(phpos[,colnames(ph)%in%cpgidhom],
                           na.rm=TRUE)
#cpgs in X escapees
ph$metalwayshiper <- rowMeans(phpos[,colnames(ph)%in%cpgidalways],
                              na.rm=TRUE)
#cpgs X inactive genes
ph$metinactivehiper <- rowMeans(phpos[,colnames(ph)%in%cpgidinactive],
                                na.rm=TRUE)

#cpgs in Y
ph$metyhiper <- rowMeans(phpos[,colnames(ph)%in%cpgidy],
                         na.rm=TRUE)

#cpgs in Y genes within the dimorphism profiles
ph$metdimyhiper <- rowMeans(phpos[,colnames(ph)%in%cpgidhomy],
                            na.rm=TRUE)

summary(lm(metdimhiper ~ pf + study +age,
           data=ph, subset=which(ph$sex=="female")))
```

```
##
## Call:
## lm(formula = metdimhiper ~ pf + study + age, data = ph, subset = which(ph$sex ==
##     "female"))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.125584 -0.007873  0.005384  0.016459  0.055856
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.315e-01  6.081e-03 153.183  < 2e-16 ***
## pf           2.571e-03  1.415e-03   1.817  0.06953 .
## studyhnsc    2.572e-03  3.796e-03   0.678  0.49827
## studykirc   -1.426e-02  4.368e-03  -3.264  0.00114 **
## studylaml   -6.887e-03  4.826e-03  -1.427  0.15399
## studylgg     2.225e-03  4.008e-03   0.555  0.57901
## studylihc    1.646e-03  3.961e-03   0.416  0.67785
## studylusc    4.114e-03  5.728e-03   0.718  0.47289
## studypaad    1.168e-03  4.301e-03   0.272  0.78604
## studypcpg   -3.593e-02  6.742e-03  -5.329 1.27e-07 ***
## age         -9.159e-05  7.649e-05  -1.197  0.23147
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02845 on 827 degrees of freedom
```

```
##   (102 observations deleted due to missingness)
## Multiple R-squared:  0.07351,Adjusted R-squared:  0.0623
## F-statistic: 6.561 on 10 and 827 DF,  p-value: 8.036e-10

summary(lm(metdimyhiper ~ pf + study +age,
           data=ph, subset=which(ph$sex=="male")))

##
## Call:
## lm(formula = metdimyhiper ~ pf + study + age, data = ph, subset = which(ph$sex ==
##     "male"))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.099538 -0.020296  0.005548  0.023823  0.065836
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.932e-01  5.608e-03 159.288  < 2e-16 ***
## pf          -1.041e-02  1.212e-03  -8.587  < 2e-16 ***
## studyhnsc    5.494e-03  2.833e-03   1.939   0.0527 .
## studylgg    -6.913e-04  3.467e-03  -0.199   0.8420
## studylihc   -4.203e-03  3.129e-03  -1.343   0.1794
## studypaad   -1.616e-02  4.065e-03  -3.974 7.47e-05 ***
## studypcpg    2.715e-02  4.716e-03   5.757 1.08e-08 ***
## age          8.453e-05  7.674e-05   1.101   0.2709
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0326 on 1218 degrees of freedom
##   (469 observations deleted due to missingness)
## Multiple R-squared:  0.1101,Adjusted R-squared:  0.105
## F-statistic: 21.53 on 7 and 1218 DF,  p-value: < 2.2e-16

#hymermethilation comparisons between gene groups
y <- c(ph$metinactivehiper, ph$metalwayshiper, ph$metdimhiper)
x <- c(rep("XI", length(ph$metinactivehiper)),
       rep("Xesc", length(ph$metalwayshiper)),
       rep("Dimor. genes", length(ph$metdimhiper)))

ttcomp <- t.test(y[x=="XI"], y[x=="Xesc"])
ttcomp <- t.test(y[x=="Xesc"], y[x=="Dimor. genes"])
```

We performed the same analysis for the hypomethylated probes (< 0.20). We tested the association between the average hypomethylation in the genes that defined the immune dimorphism and and the classification of individuals in the dimorphisms.

```
#select hypomethylated probes
phpos <- ph
phpos[ph>0.2] <- NA
ph$metdimhipo <- rowMeans(phpos[,colnames(ph)%in%cpgidhom], na.rm=TRUE)
ph$metalwayshipo <- rowMeans(phpos[,colnames(ph)%in%cpgidalways], na.rm=TRUE)
ph$metinactivehipo <- rowMeans(phpos[,colnames(ph)%in%cpgidinactive], na.rm=TRUE)
```

```
ph$metyhipo <- rowMeans(phpos[,colnames(ph)%in%cpgidy],
                        na.rm=TRUE)
ph$metdimyhipo <- rowMeans(phpos[,colnames(ph)%in%cpgidhomy],
                           na.rm=TRUE)


summary(lm(metdimhipo ~ pf + study +age,
           data=ph, subset=which(ph$sex=="female")))

##
## Call:
## lm(formula = metdimhipo ~ pf + study + age, data = ph, subset = which(ph$sex ==
##     "female"))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.027524 -0.007535 -0.002820  0.003385  0.118868
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.004e-02  2.518e-03  15.899  < 2e-16 ***
## pf           1.684e-03  5.910e-04   2.850 0.004475 **
## studyhnsc   -2.077e-03  1.647e-03  -1.261 0.207661
## studykirc   -1.382e-02  1.908e-03  -7.245 9.10e-13 ***
## studylaml   -2.261e-02  2.027e-03 -11.150  < 2e-16 ***
## studylgg     2.714e-03  1.707e-03   1.590 0.112148
## studylihc   -5.261e-03  1.706e-03  -3.083 0.002107 **
## studylusc   -2.017e-02  2.519e-03  -8.008 3.47e-15 ***
## studypaad    2.109e-03  1.878e-03   1.123 0.261628
## studypcpg   -6.346e-03  1.878e-03  -3.379 0.000758 ***
## age         -3.167e-06  3.130e-05  -0.101 0.919429
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01262 on 929 degrees of freedom
## Multiple R-squared:  0.2705,Adjusted R-squared:  0.2627
## F-statistic: 34.45 on 10 and 929 DF,  p-value: < 2.2e-16

summary(lm(metdimyhipo ~ pf + study +age,
           data=ph, subset=which(ph$sex=="male")))

##
## Call:
## lm(formula = metdimyhipo ~ pf + study + age, data = ph, subset = which(ph$sex ==
##     "male"))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.037586 -0.010517 -0.001645  0.008166  0.083982
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.970e-02  2.607e-03  22.902  < 2e-16 ***
## pf          5.912e-03  5.445e-04  10.857  < 2e-16 ***
## studyhnsc   1.772e-03  1.275e-03   1.390  0.16462
```

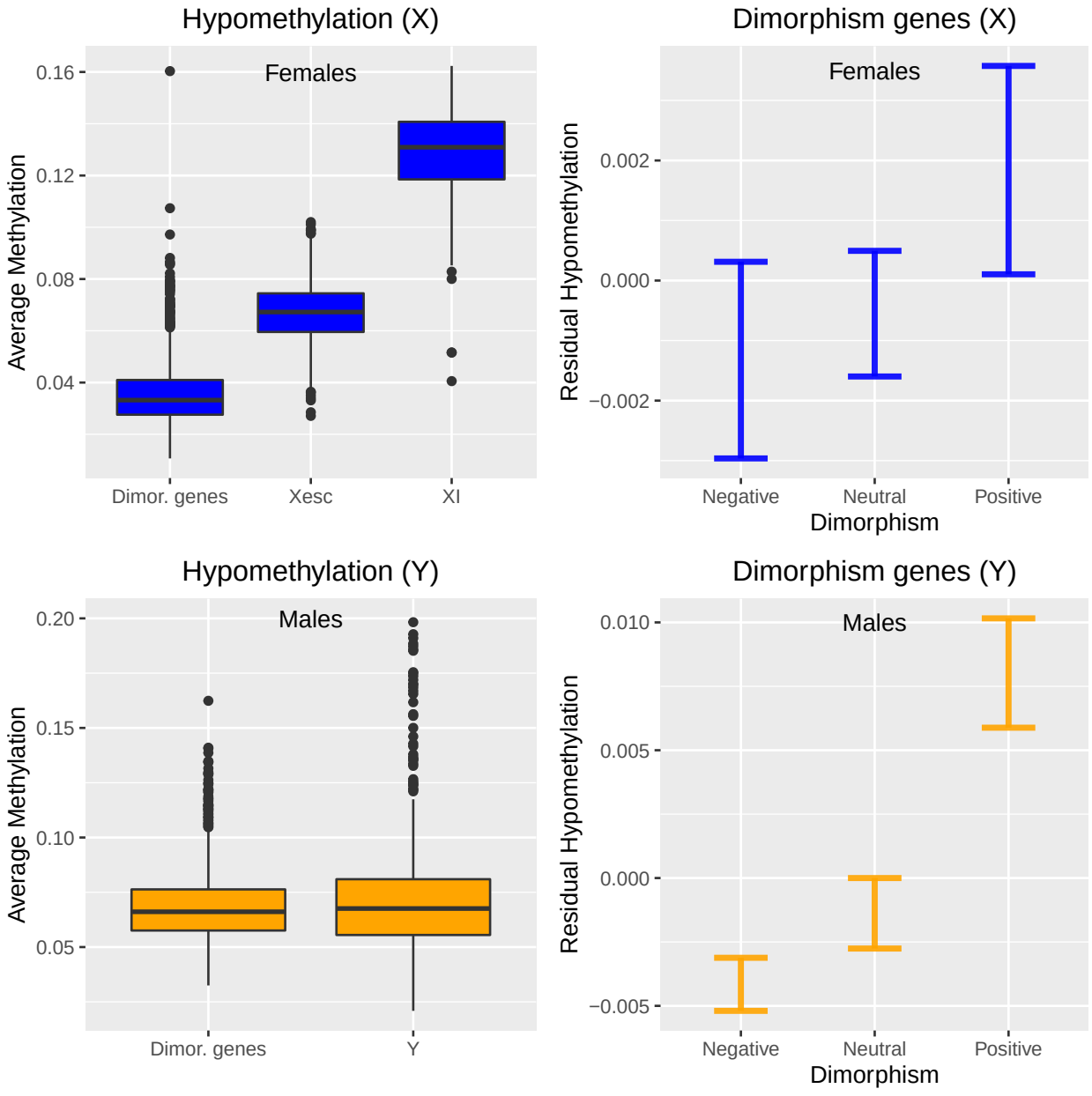
```
## studylgg    5.013e-03  1.609e-03   3.117  0.00187 **
## studylihc   3.944e-03  1.412e-03   2.793  0.00530 **
## studypaad   3.673e-04  1.865e-03   0.197  0.84396
## studypcpg   9.296e-03  2.204e-03   4.217 2.64e-05 ***
## age         1.178e-04  3.603e-05   3.270  0.00110 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01598 on 1367 degrees of freedom
##   (320 observations deleted due to missingness)
## Multiple R-squared:  0.09694,Adjusted R-squared:  0.09232
## F-statistic: 20.96 on 7 and 1367 DF,  p-value: < 2.2e-16
```

We summarized the results in a panel plot

```
res <- summary(lm(metdimhipo ~ study +age,
                  data=ph, subset=which(ph$sex=="female")))$residual

ph$reshipox <- NA
ph[names(res),]$reshipox <- res

res <- summary(lm(metdimyhipo ~ study +age,
                  data=ph, subset=which(ph$sex=="male")))$residual

ph$reshipoy <- NA
ph[names(res),]$reshipoy <- res

###plot1
y <- c(ph$metinactivehipo, ph$metalwayshipo, ph$metdimhipo)
x <- c(rep("XI", length(ph$metinactivehipo)),
       rep("Xesc", length(ph$metalwayshipo)),
       rep("Dimor. genes", length(ph$metdimhipo)))

ttcomp <- t.test(y[x=="XI"], y[x=="Xesc"])
ttcomp <- t.test(y[x=="Xesc"], y[x=="Dimor. genes"])

dat <- data.frame(x, y)[ph$sex=="female", ]

p1 <- ggplot(dat, aes(x=x, y=y, fill=x)) +
  geom_boxplot()+ylab("Average Methylation")+xlab("")+
  ggtitle("Hypomethylation (X)")+
  theme(plot.title = element_text(hjust = 0.5))+
  annotate(geom="text", x=2, y=0.16, label="Females",
              color="black")+
  theme(legend.position="none") +
  scale_fill_manual(values=c("blue", "blue", "blue"))


###plot2
tt1 <- t.test(ph$reshipox[ph$sex=="female" & ph$pf==1])
tt2 <- t.test(ph$reshipox[ph$sex=="female" & ph$pf==0])
tt3 <- t.test(ph$reshipox[ph$sex=="female" & ph$pf==-1])
```

```
data <- data.frame(
  Dimorphism=c("Negative", "Neutral", "Positive"),
  mean=c(tt3$statistic, tt2$statistic, tt1$statistic),
  cl=c(tt3$conf.int[1], tt2$conf.int[1], tt1$conf.int[1]),
  cu=c(tt3$conf.int[2], tt2$conf.int[2], tt1$conf.int[2])
)

p2 <- ggplot(data) +
    geom_errorbar( aes(x=Dimorphism, ymin=cl, ymax=cu),
                   width=0.4, colour="blue", alpha=0.9, size=1.3)+
  ylab("Residual Hypomethylation")+
  ggtitle("Dimorphism genes (X)")+
  theme(plot.title = element_text(hjust = 0.5))+
  annotate(geom="text", x=2, y=0.0035, label="Females",
              color="black")



###plot3
y <- c(ph$metyhipo, ph$metdimyhipo)
x <- c(rep("Y", length(ph$metyhipo)),
       rep("Dimor. genes", length(ph$metdimyhipo)))

ttcomp <- t.test(y[x=="Y"], y[x=="Dimor. genes"])

dat <- data.frame(x, y)[ph$sex=="male", ]

p3 <- ggplot(dat, aes(x=x, y=y, fill=x)) +
  geom_boxplot()+ylab("Average Methylation")+xlab("")+
  ggtitle("Hypomethylation (Y)")+
  theme(plot.title = element_text(hjust = 0.5))+
  annotate(geom="text", x=1.5, y=0.2, label="Males",
              color="black")+
  theme(legend.position="none") +
  scale_fill_manual(values=c("orange", "orange", "orange"))


###plot4
tt1 <- t.test(ph$reshipoy[ph$sex=="male" & ph$pf==1])
tt2 <- t.test(ph$reshipoy[ph$sex=="male" & ph$pf==0])
tt3 <- t.test(ph$reshipoy[ph$sex=="male" & ph$pf==-1])


data <- data.frame(
  Dimorphism=c("Negative", "Neutral", "Positive"),
  mean=c(tt3$statistic, tt2$statistic, tt1$statistic),
  cl=c(tt3$conf.int[1], tt2$conf.int[1], tt1$conf.int[1]),
  cu=c(tt3$conf.int[2], tt2$conf.int[2], tt1$conf.int[2])
)

p4 <- ggplot(data) +
    geom_errorbar( aes(x=Dimorphism, ymin=cl, ymax=cu),
                   width=0.4, colour="orange", alpha=0.9, size=1.3)+
```

```
  ylab("Residual Hypomethylation")+
  ggtitle("Dimorphism genes (Y)")+
  theme(plot.title = element_text(hjust = 0.5))+
  annotate(geom="text", x=2, y=0.01, label="Males",
              color="black")


grid.arrange(p1, p2, p3, p4, nrow = 2)
```



3.2 Validation of associations for cancer survival

We downloaded microarray transcriptomic data of five cancer studies with survival data, with the aim of validating the significan associations observed in TCGA. We targeted individuals with the positive, negative and neutral immune sexual dimorphisms. We then tested the association between cancer survival and the interaction betwen sex and the dimorphism classification. We fitted Cox proportional hazard models.

3.3 GSE41271

```
gsm <- getGEO("GSE41271", destdir ="./data", AnnotGPL =TRUE)

phenobb <- pData(phenoData(gsm[[1]]))[,44:54]

time <- as.numeric(as.Date(phenobb$"last follow-up survival:ch1")-
                     as.Date(phenobb$"date of surgery:ch1"))

event <- as.numeric(factor(phenobb$"vital statistics:ch1"))-1
event[event==2]<-NA

age <- as.numeric(as.Date(phenobb$"date of surgery:ch1")-
                    as.Date(phenobb$"date of birth:ch1"))

gender <- factor(phenobb$"gender:ch1")
gender <- -as.numeric(gender)+2


smk <- phenobb$"tobacco history:ch1"
smk[smk=="Missing"] <- NA

phenodat <- data.frame(t=gender, eff=time, event, smk, age)

genesIDs <- fData(gsm[[1]])

expr<-exprs(gsm[[1]])
rownames(expr) <- genesIDs$'Gene symbol'


#get data for complete.cases only
selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

mod0 <- model.matrix( ~ t + eff + age + smk , data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff + age + smk, data = phenodat)
ns <- num.sv(expr, mod, method="be")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  37
## Iteration (out of 5 ):1  2  3  4  5

modss <- cbind(mod, ss)

datteff <- list(features=t(expr),
```

```
                teffdata=data.frame(time=phenodat$eff,
                                    event=phenodat$event,
                                    modss,check.names = FALSE))

rmvars <- !colnames(datteff$teffdata)%in%
             c("(Intercept)","t:eff")

datteff$teffdata <- datteff$teffdata[,rmvars]

datLung0 <- datteff


Lung0surv <- target(datLung0, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="hazard", match=0.7, nmcov = "age")
```

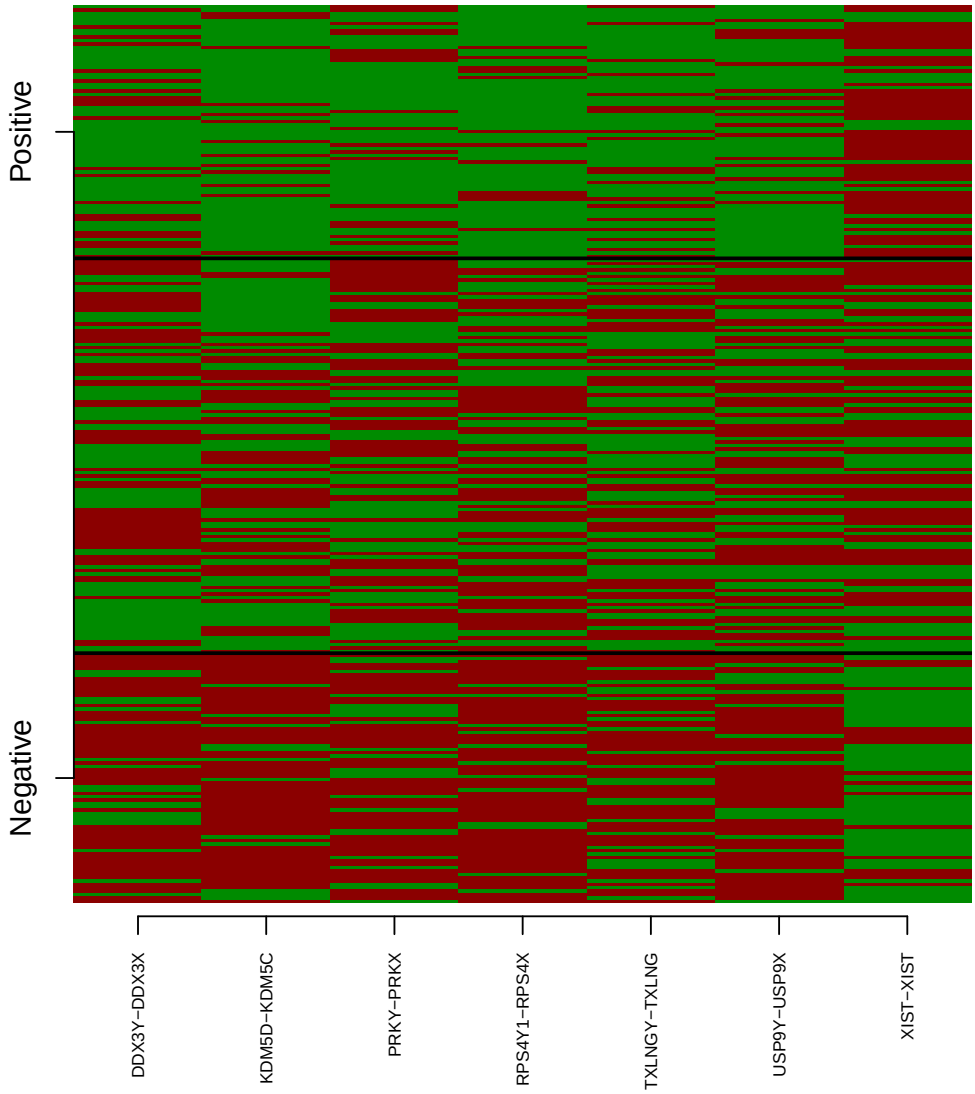

Positive
Negative
DDX3Y−DDX3X
KDM5D−KDM5C
PRKY−PRKX
RPS4Y1−RPS4X
TXLNGY−TXLNG
USP9Y−USP9X
XIST−XIST

Figure S6

3.4 GSE72094

```
gsm <- getGEO("GSE72094", destdir ="./data", getGPL=FALSE)
gp <- getGEO("GPL15048", destdir ="./data")

genesIDs <-(Table(gp)$GeneSymbol)
names(genesIDs) <- Table(gp)$ID
genesIDs <- genesIDs[rownames(gsm[[1]])]

expr<-exprs(gsm[[1]])
rownames(expr) <- genesIDs

phenobb <- pData(phenoData(gsm[[1]]))

event <- as.numeric(factor(phenobb$"vital_status:ch1"))-1
event[event==2]<-NA

age <- as.numeric(phenobb$'age_at_diagnosis:ch1')

gender <- factor(phenobb$"gender:ch1")
gender <- -as.numeric(gender)+2

time <- as.numeric(phenobb$'survival_time_in_days:ch1')

race <- phenobb$"race:ch1"
race[race!="WHITE"] <- NA
smk <- phenobb$"smoking_status:ch1"
smk[smk=="Missing"] <- NA

phenodat <- data.frame(t=gender, eff=time, event, smk, race, age)


#get data for complete.cases only
selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

mod0 <- model.matrix( ~ t + eff + age + smk , data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff + age + smk, data = phenodat)
ns <- num.sv(expr, mod, method="be")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  47
## Iteration (out of 5 ):1  2  3  4  5

modss <- cbind(mod, ss)


datteff <- list(features=t(expr),
                teffdata=data.frame(time=phenodat$eff,
                                    event=phenodat$event,
                                    modss,check.names = FALSE))

rmvars <- !colnames(datteff$teffdata)%in%
```

```
              c("(Intercept)","t:eff")

datteff$teffdata <- datteff$teffdata[,rmvars]

datLung1 <- datteff


Lung1surv <- target(datLung1, htrcells$"T cell",
               effect="positiveandnegative", featuresinf=XYhomol,
               model="hazard", match=0.7, nmcov = "age")
```

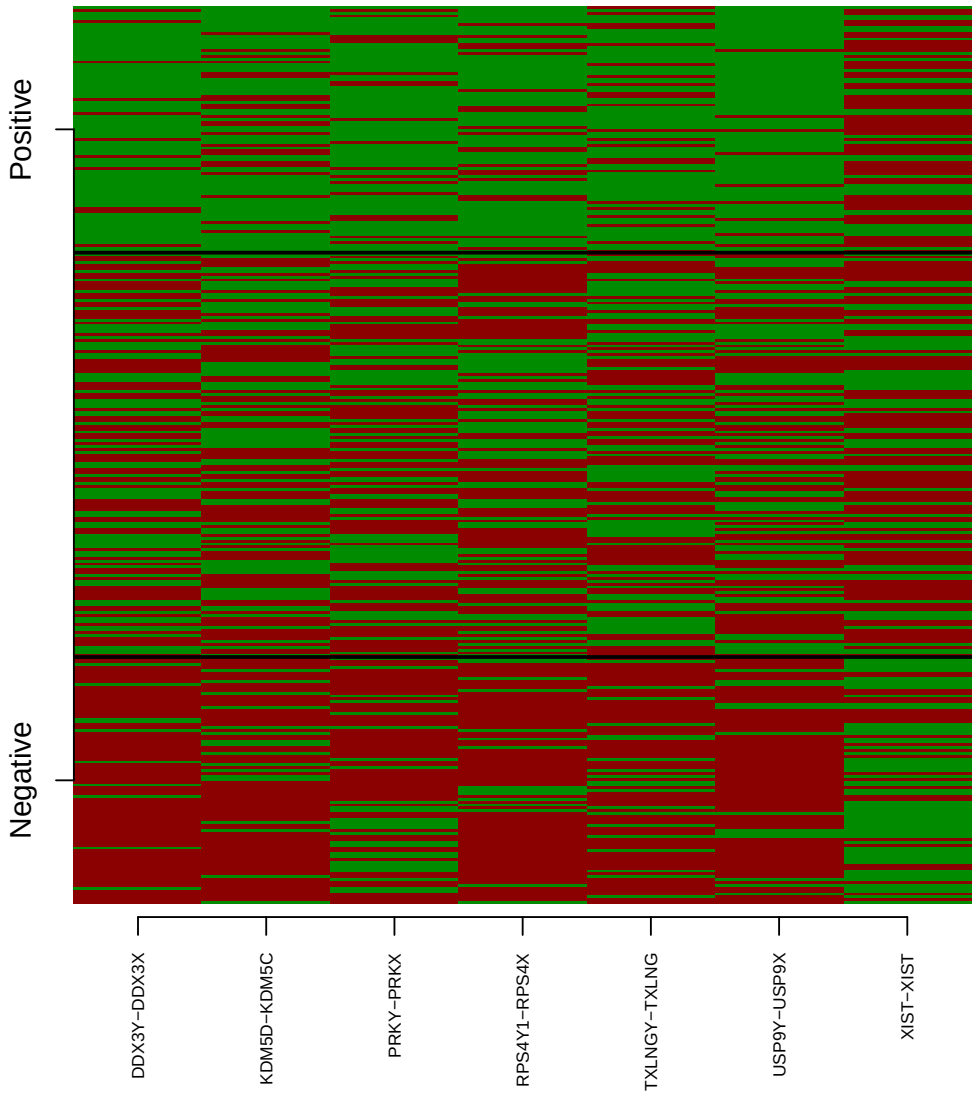


Figure S7

3.5 GSE42127

```
gsm <- getGEO("GSE42127", destdir ="./data", AnnotGPL =TRUE)

phenobb <- pData(phenoData(gsm[[1]]))[,41:47]

time <- as.numeric(phenobb$"overall survival months:ch1")*365/12

event <- as.numeric(factor(phenobb$"survival status:ch1"))-1
event[event==2]<-NA

age <- as.numeric(phenobb$"age at surgery:ch1")

gender <- factor(phenobb$"gender:ch1")
gender <- -as.numeric(gender)+2


phenodat <- data.frame(t=gender, eff=time, time, event, age)

genesIDs <- fData(gsm[[1]])

expr<-exprs(gsm[[1]])
rownames(expr) <- genesIDs$'Gene symbol'


#get data for complete.cases only
selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

mod0 <- model.matrix( ~ t + eff + age, data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff + age, data = phenodat)
ns <- num.sv(expr, mod, method="be")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  27
## Iteration (out of 5 ):1  2  3  4  5

modss <- cbind(mod, ss)

datteff <- list(features=t(expr),
                teffdata=data.frame(time=phenodat$eff,
                                    event=phenodat$event,
                                    modss,check.names = FALSE))

rmvars <- !colnames(datteff$teffdata)%in%
             c("(Intercept)","t:eff")

datteff$teffdata <- datteff$teffdata[,rmvars]

datLung2 <- datteff


Lung2surv <- target(datLung2, htrcells$"T cell",
```

```
            effect="positiveandnegative", featuresinf=XYhomol,
            model="hazard", match=0.7, nmcov = "age")
```

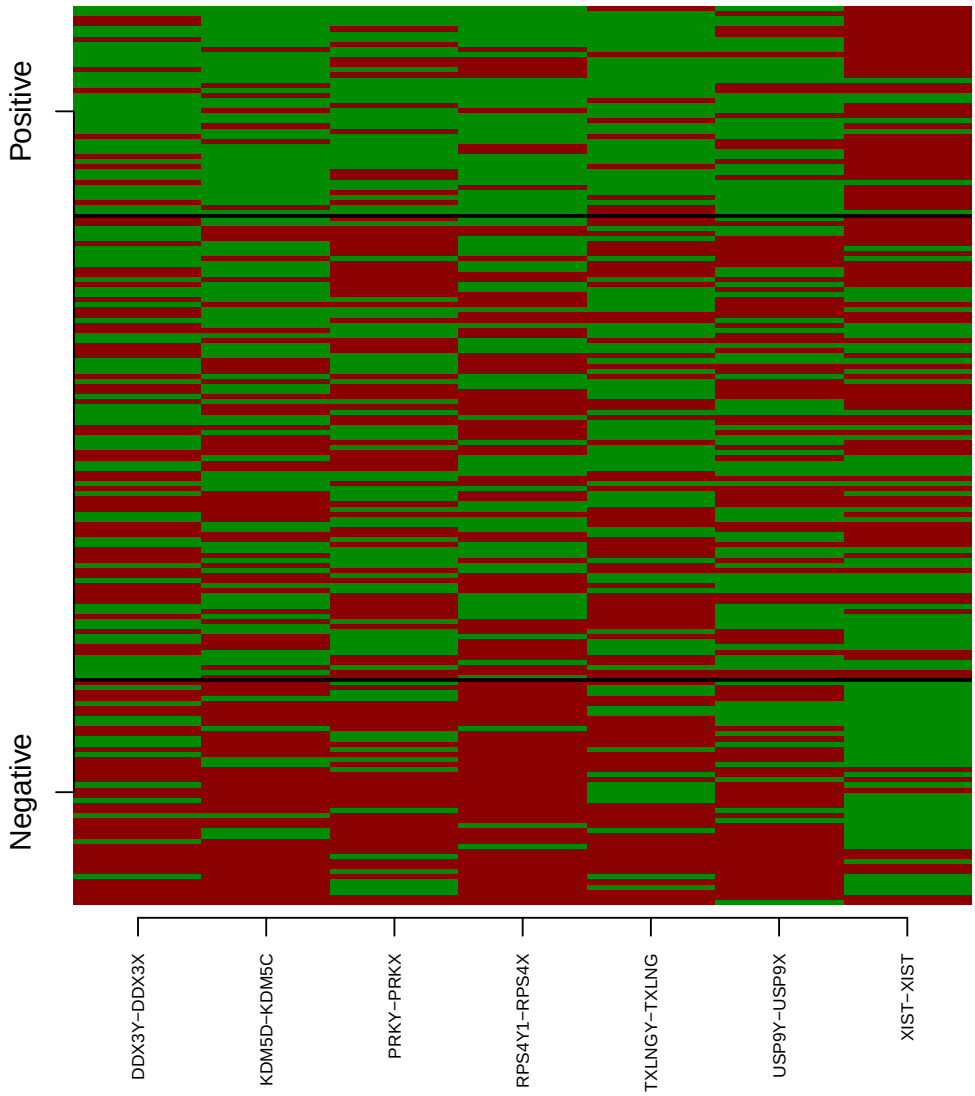


Figure S8

3.6 GSE68465

```
gsm <- getGEO("GSE68465", destdir ="./data", getGPL = FALSE)

gp <- getGEO("GPL96", destdir ="./data")
genesIDs <-(Table(gp)$"Gene Symbol")
names(genesIDs) <- Table(gp)$ID
genesIDs <- genesIDs[rownames(gsm[[1]])]

expr<-exprs(gsm[[1]])
rownames(expr) <- genesIDs


phenobb <- pData(phenoData(gsm[[1]]))[,48:63]

time <- as.numeric(as.character(phenobb$"months_to_last_contact_or_death:ch1"))*365/12

event <- as.numeric(factor(phenobb$"vital_status:ch1"))-1
event[event==2]<-NA

age <- as.numeric(phenobb$"age:ch1")

gender <- factor(phenobb$"Sex:ch1")
gender <- -as.numeric(gender)+2


race <- phenobb$"race:ch1"

smk <-  smk <- factor( phenobb$"smoking_history:ch1", labels=c(NA,"Yes", "No", "Yes", NA))

phenodat <- data.frame(t=gender, eff=time, time, event, age, smk, race)


#get data for complete.cases only
selcom <- complete.cases(phenodat) & complete.cases(t(expr)) &  phenodat$race=="White"

phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

mod0 <- model.matrix( ~ t + eff + age + smk, data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff + age + smk, data = phenodat)
ns <- num.sv(expr, mod, method="be")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  17
## Iteration (out of 5 ):1  2  3  4  5

modss <- cbind(mod, ss)


datteff <- list(features=t(expr),
                teffdata=data.frame(time=phenodat$eff,
                                    event=phenodat$event,
                                    modss,check.names = FALSE))
```

```
rmvars <- !colnames(datteff$teffdata)%in%
             c("(Intercept)","t:eff")

datteff$teffdata <- datteff$teffdata[,rmvars]

datLung3 <- datteff


Lung3surv <- target(datLung3, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="hazard", match=0.7, nmcov = "age")
```

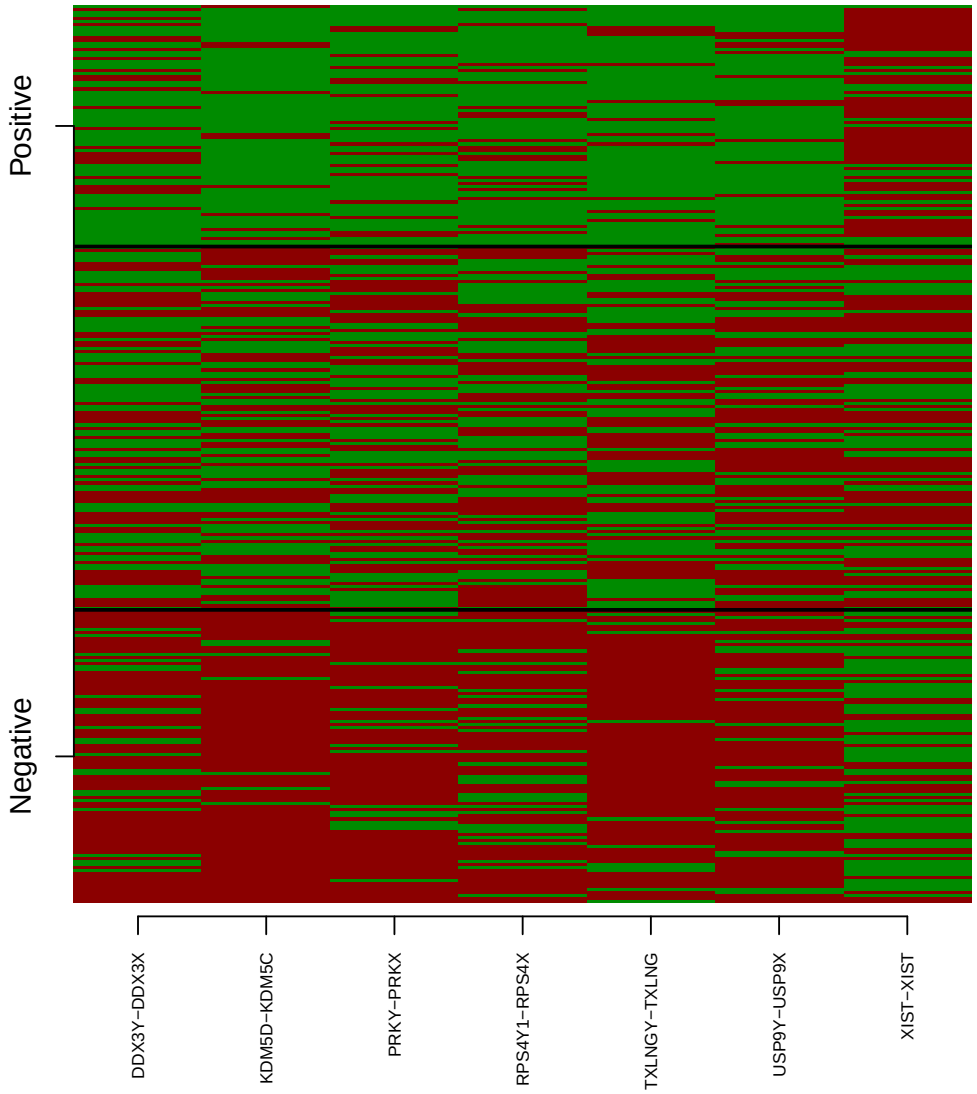


Figure S9

3.7 GSE13041

```
gsm <- getGEO("GSE13041", destdir ="./data", AnnotGPL =TRUE)

probes <- intersect(rownames(exprs(gsm[[1]])),
                    rownames(exprs(gsm[[3]])))

expr <- normalizeBetweenArrays(cbind(exprs(gsm[[1]])[probes,],
                                     exprs(gsm[[3]])[probes,]),
                               method="quantile")
expr <- log2(expr)

genesIDs <- fData(gsm[[3]])[probes,]

rownames(expr) <- genesIDs$'Gene symbol'


phenodat<- lapply(c(1,3), function(i)
{
phenobb <- pData(phenoData(gsm[[i]]))

event <- as.numeric(factor(phenobb$"Vital Status:ch1"))-1

age <- as.numeric(phenobb$'Age(years):ch1')

gender <- factor(phenobb$"Gender:ch1")
gender <- -as.numeric(gender)+2

time <- as.numeric(phenobb$'TTS(days):ch1')
pn <- phenobb$'HC:ch1'

data.frame(t=gender, eff=time, age, time, event, pn)
})

phenodat <- do.call(rbind,phenodat)
array<-c(rep(1, ncol(gsm[[1]])), rep(2, ncol(gsm[[3]])))

phenodat <- data.table(phenodat, array=array)

#get data for complete.cases only
selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

mod0 <- model.matrix( ~ t + eff + age + array , data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff + age + array, data = phenodat)
ns <- num.sv(expr, mod, method="be")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  24
## Iteration (out of 5 ):1  2  3  4  5

modss <- cbind(mod, ss)
```

```
datteff <- list(features=t(expr),
                teffdata=data.frame(time=phenodat$eff,
                                    event=phenodat$event,
                                    modss,check.names = FALSE))

rmvars <- !colnames(datteff$teffdata)%in%
             c("(Intercept)","t:eff")

datteff$teffdata <- datteff$teffdata[,rmvars]

datGliom <- datteff


Gliomsurv <- target(datGliom, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="hazard", match=0.7, nmcov = "age")
```

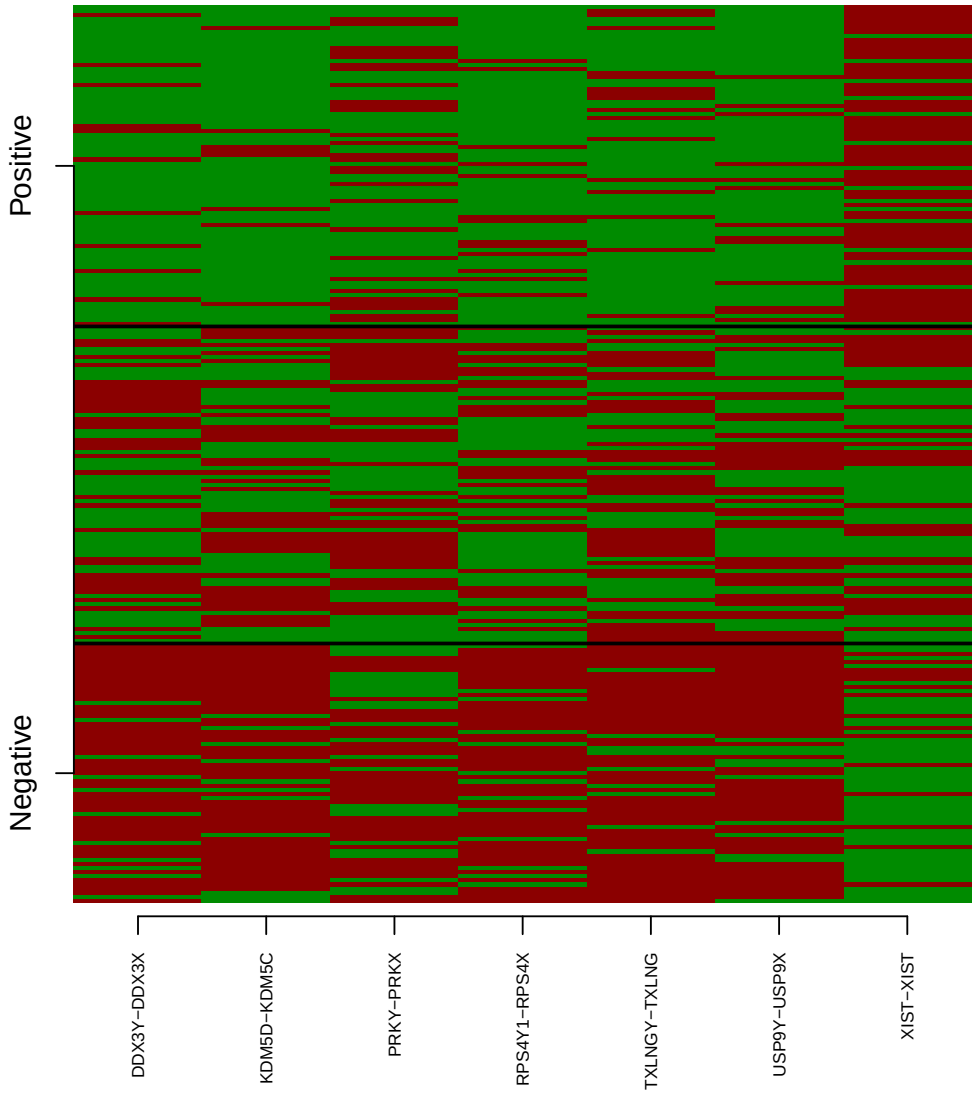


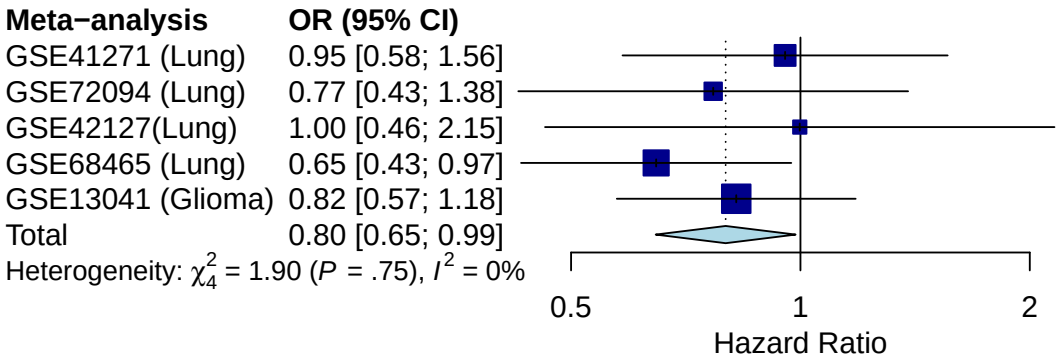
Figure S10

Meta-analysis

We meta-analyzed the associations between survival and the interaction between sex (`t`) and the dimorphism classification of individuals (`pf`).

```
Survres <- list(Lung0surv, Lung1surv, Lung2surv, Lung3surv, Gliomsurv)


cellSurv<-lapply(Survres,
  function(AA)
  {
    out <- AA$summary.model$coefficients
```

```
    out <- out["WTRUE:pf",c(1,3)]
    out
  })

cellSurv <- do.call(rbind,cellSurv)

datmet <- data.frame(TE = cellSurv[,1], SE = cellSurv[,2])

#Perform meta-analysis
metaresSurv <- metagen(TE, SE, data=datmet,
                studlab= c("GSE41271 (Lung)",
                           "GSE72094 (Lung)",
                           "GSE42127(Lung)",
                           "GSE68465 (Lung)",
                           "GSE13041 (Glioma)"),
                level.ci = 0.95, sm="OR",fixed = FALSE)

forest(metaresSurv, layout="JAMA",
       leftlabs=c("Meta-analysis", "OR (95% CI)"),
       xlab="Hazard Ratio", title="")
```



We also performed a pooled analysis of the associations.

```
#pool-analysis
pf <- sapply(Survres, function(x) x$classification)
names(pf) <- NULL
pf <- unlist(pf)


datlist <- list(datLung0, datLung1, datLung2, datLung3, datGliom)

names(datlist) <-  c("GSE41271",
                     "GSE72094",
                     "GSE42127",
                     "GSE68465",
                     "GSE13041")
```

```r
clindat <- lapply(datlist, function(x) x$teffdata[,c("time", "event", "age", "t")] )

clindat <- do.call(rbind, clindat)
ids <- strsplit(rownames(clindat), "\\.")
ids <- data.frame(do.call(rbind, ids))
names(ids) <- c("study", "ID")


ID <- unlist(lapply(datlist, function(x) rownames(x$features)))
ids$ID <- ID


clindat <- data.frame(ids, clindat, pf=pf[ID])


clindat$sex <- factor(ifelse(clindat$t == 0, "male", "female"))
clindat$dimorphism <- factor(clindat$pf, labels = c("negative", "neutral", "positive"))
clindat$study <- factor(clindat$study)


#association with sex
coxph(Surv(time, event) ~ t+age, data = clindat)
```

```
## Call:
## coxph(formula = Surv(time, event) ~ t + age, data = clindat)
##
##           coef  exp(coef)   se(coef)      z        p
## t   -4.204e-01  6.568e-01  8.138e-02 -5.167 2.38e-07
## age -1.154e-05  1.000e+00  4.258e-06 -2.709  0.00675
##
## Likelihood ratio test=33.59  on 2 df, p=5.094e-08
## n= 1267, number of events= 625
```

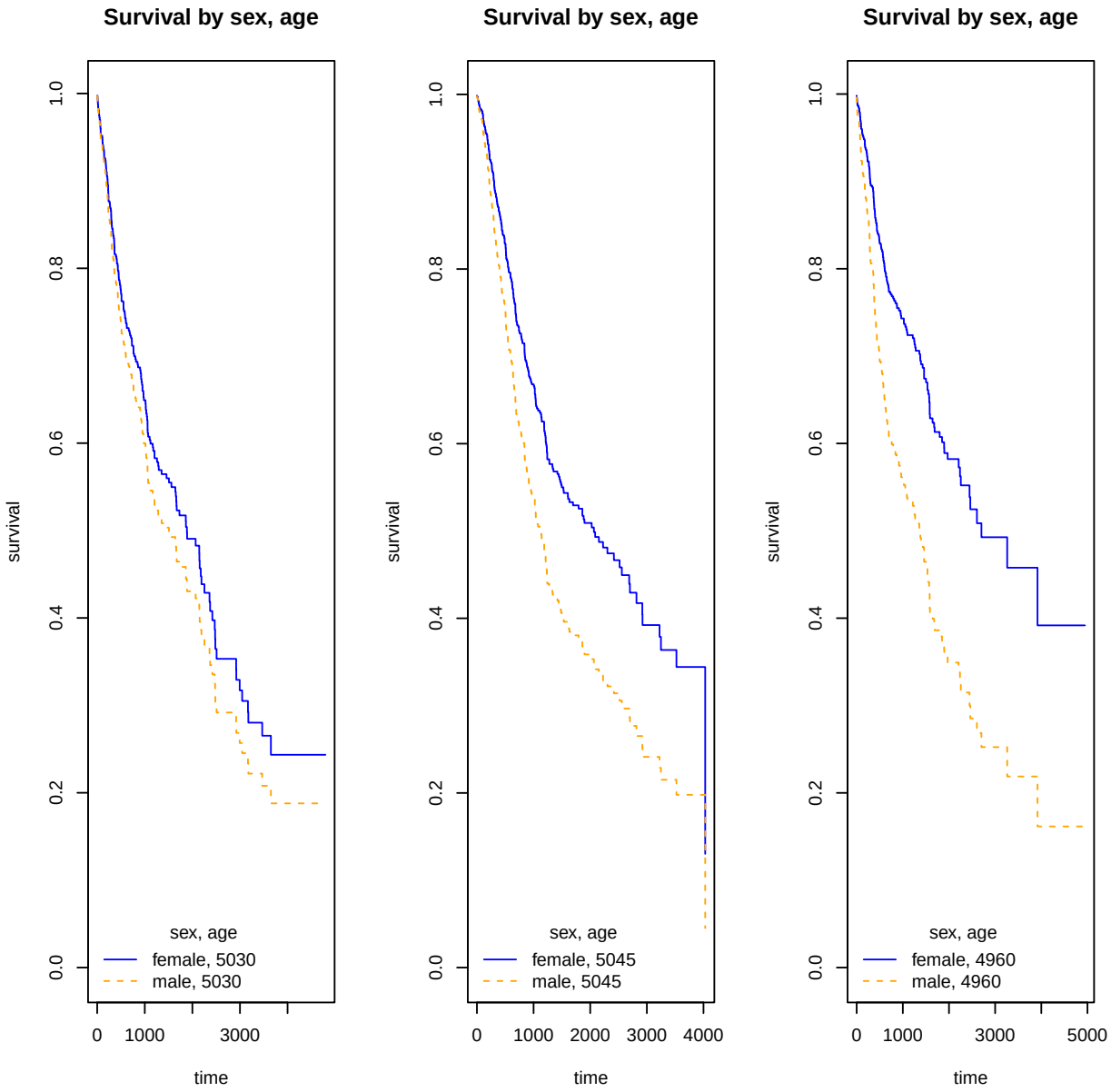
```r
#correlation between sex and dimorphic group
chisq.test(table(clindat$t, clindat$pf))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(clindat$t, clindat$pf)
## X-squared = 8.7834, df = 2, p-value = 0.01238
```

```r
#association with the interaction between sex and dimorphic group
coxph(Surv(time, event) ~ t*pf+age+study, data = clindat)
```

```
## Call:
## coxph(formula = Surv(time, event) ~ t * pf + age + study, data = clindat)
##
##                      coef  exp(coef)   se(coef)       z        p
## t              -3.036e-01  7.382e-01  8.228e-02  -3.690 0.000224
## pf              1.325e-01  1.142e+00  6.745e-02   1.964 0.049492
## age             4.812e-05  1.000e+00  2.681e-05   1.795 0.072689
## studyGSE41271 -2.710e+00  6.653e-02  6.580e-01  -4.118 3.81e-05
## studyGSE42127 -1.914e+00  1.475e-01  1.467e-01 -13.045  < 2e-16
## studyGSE68465 -1.515e+00  2.198e-01  1.126e-01 -13.458  < 2e-16
```

```
## studyGSE72094 -1.615e+00  1.989e-01  1.305e-01 -12.372  < 2e-16
## t:pf          -2.778e-01  7.575e-01  1.057e-01  -2.628 0.008586
##
## Likelihood ratio test=308.1  on 8 df, p=< 2.2e-16
## n= 1267, number of events= 625

mod1 <- coxph(Surv(time, event) ~ sex+age, data = clindat, subset = which(clindat$pf==-1))

mod2 <- coxph(Surv(time, event) ~ sex+age, data = clindat, subset = which(clindat$pf==0))

mod3 <- coxph(Surv(time, event) ~ sex+age, data = clindat, subset = which(clindat$pf==1))

par(mfrow=c(1,3))
  RcmdrPlugin.survival::plot.coxph(mod1, byfactors=TRUE,
                                   col=c("blue", "orange"))
  RcmdrPlugin.survival::plot.coxph(mod2, byfactors=TRUE,
                                   col=c("blue", "orange"))
  RcmdrPlugin.survival::plot.coxph(mod3, byfactors=TRUE,
                                   col=c("blue", "orange"))
```



4 Rheumatoid Arthritis

We downloaded three transcriptomic studies in rheumatoid arthritis (RA) to test whether the association of the female-risk of RA was modulated by the classification of individuals into the profiles of high immune sexual dimorphisms.

4.1 GSE74143

```
gsm <- getGEO("GSE74143", destdir ="./data", AnnotGPL =TRUE)

gsm <- gsm[[1]]
```

```
datArt1 <- feateff(gsm, tname="gender:ch1", reft=c("M", "F"),
                   effname="rheumatoid factor positivity:ch1",
                   refeff=c("Negative","Positive"),
                   covnames="age:ch1", covtype="n",
                   sva=TRUE, UsegeneSymbol=TRUE)

## Number of significant surrogate variables is:  32
## Iteration (out of 5 ):1  2  3  4  5

#change reference to males
A1 <- target(datArt1, htrcells$"T cell",
             effect="positiveandnegative", featuresinf=XYhomol,
             model="binomial", match=0.7, nmcov = "age.ch1")
```

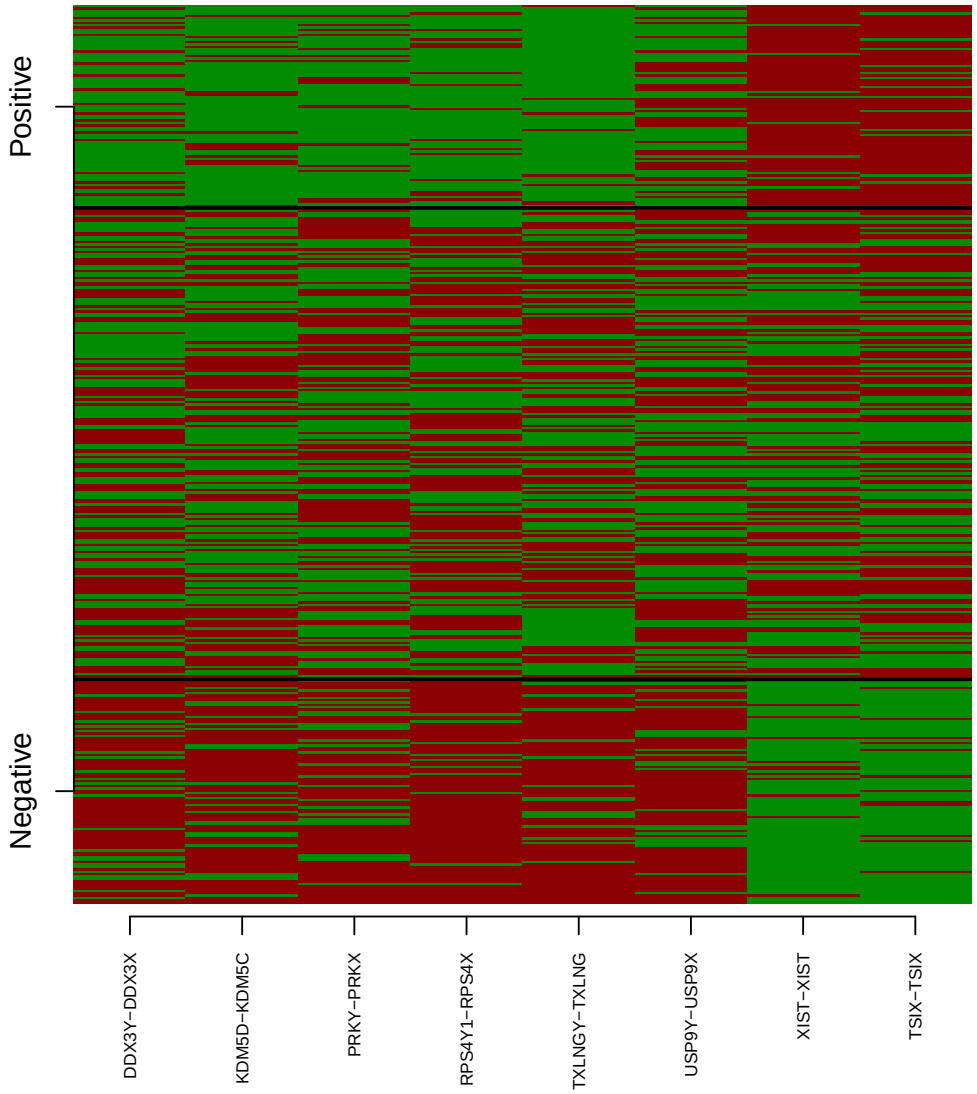


Figure S11

4.2 GSE93777

```
# arthritis
gsm <- getGEO("GSE93777", destdir ="./data", AnnotGPL =TRUE)

gsm <- gsm[[1]]

expr <- exprs(gsm)
genesIDs <- fData(gsm)

rownames(expr) <- genesIDs$'Gene symbol'

phenobb <- pData(phenoData(gsm))

eff <- factor(phenobb$"disease state")
eff <- as.numeric(eff)-1

age <- as.numeric(phenobb$'age:ch1')

t <- factor(phenobb$"gender:ch1")
t <- -as.numeric(t)+2

phenodat <- data.frame(eff, t, age)

#get data for complete.cases only
selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

mod0 <- model.matrix( ~ t + eff + age, data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff + age, data = phenodat)
ns <- num.sv(expr, mod, method="be")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  31
## Iteration (out of 5 ):1  2  3  4  5

colnames(ss) <- paste("cov",1:ncol(ss),sep="")

modss <- cbind(mod, ss)

datArt2<-list(features=t(expr), teffdata=modss)

rmvars <- !colnames(datArt2$teffdata)%in%
            c("(Intercept)","t:eff")

datArt2$teffdata <- datArt2$teffdata[,rmvars]

A2 <- target(datArt2, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="binomial", match=0.7)
```

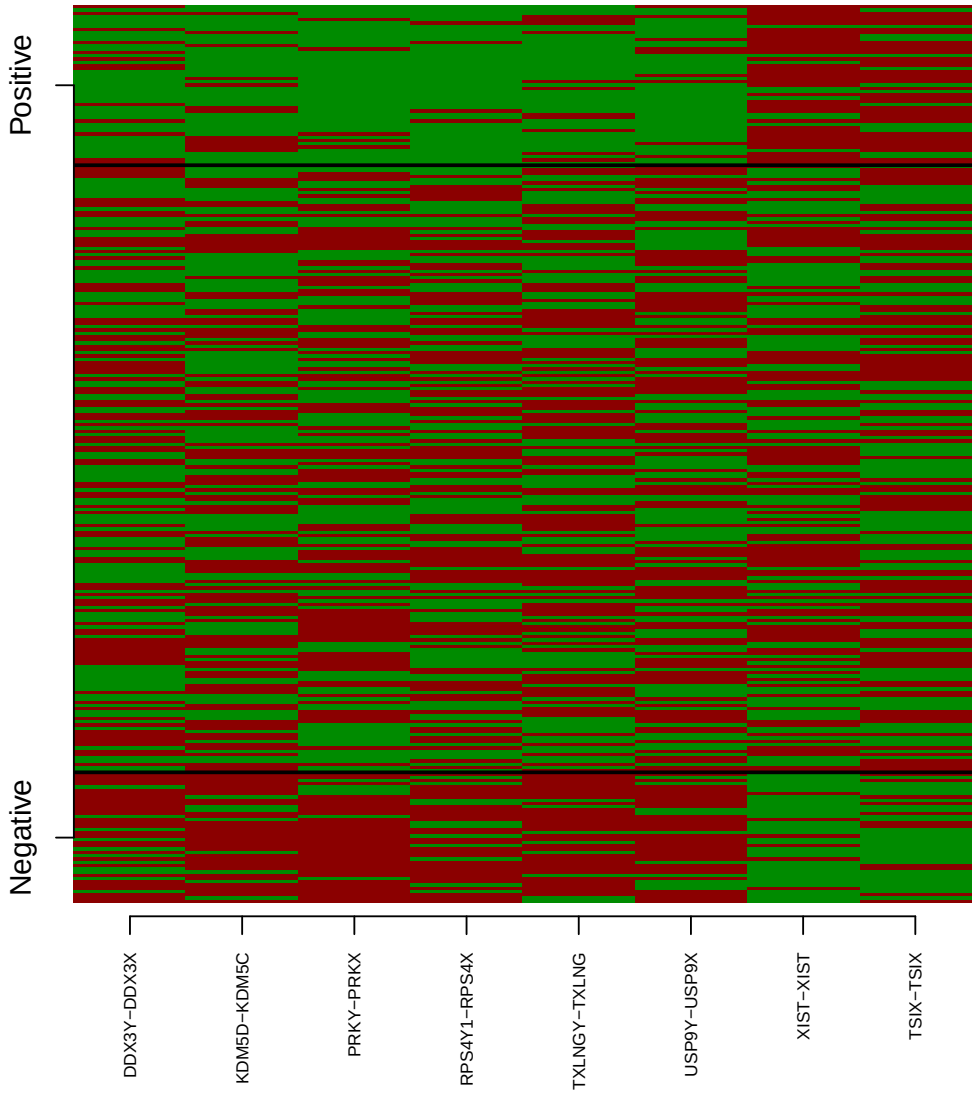


Figure S12

4.3 GSE17755

```
# arthritis
gsm <- getGEO("GSE17755", destdir ="./data", AnnotGPL =TRUE)

gsm <- gsm[[1]]

expr <- exprs(gsm)
genesIDs <- fData(gsm)

rownames(expr) <- genesIDs$"Gene symbol"
```

```
phenobb <- pData(phenoData(gsm))

eff <-rep(NA, nrow(phenobb))

eff[grep("healthy individual", phenobb$"disease:ch1")] <- 0
eff[grep("rheumatoid arthritis", phenobb$"disease:ch1")] <- 1

age <- as.numeric(gsub("age:","", phenobb$"age:ch1"))

gender <- factor(phenobb$"gender:ch1")
gender <- -as.numeric(gender)+2

phenodat <- data.frame(t=gender, eff, age)

#get data for complete.cases only
ss <- sapply(1:nrow(expr), function(x) sum(is.na(expr[x,])))
expr <- expr[ss==0, ]

selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

mod0 <- model.matrix( ~ t + eff + age, data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff + age, data = phenodat)
ns <- num.sv(expr, mod, method="be")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  20
## Iteration (out of 5 ):1  2  3  4  5

modss <- cbind(mod, ss)

datArt3 <- list(features=t(expr), teffdata=data.frame(modss,check.names = FALSE)
)

rmvars <- !colnames(datArt3$teffdata)%in%
             c("(Intercept)","t:eff")

datArt3$teffdata <- datArt3$teffdata[,rmvars]


A3 <- target(datArt3, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="binomial", match=0.7)



A3

## object of class: tarteff
##
## classification into
##   negative treatment effect: -1
##   neutral: 0
```

```
##   positive treatment: 1
##
##  -1   0   1
##  29 110  18
##
## interaction fitted model: binomial
##   Estimate Std. Error    z value   Pr(>|z|)
##  0.8339920  0.9095086  0.9169699  0.3591584
```

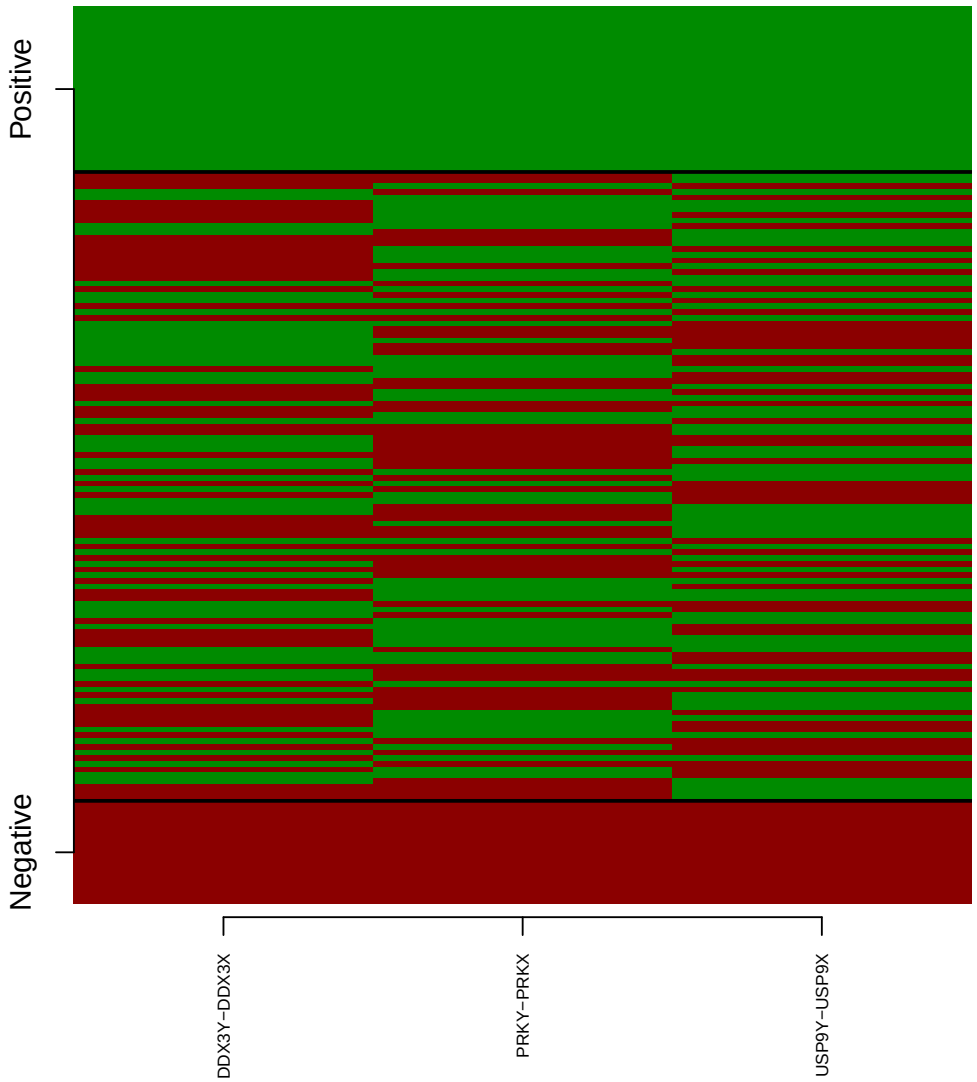
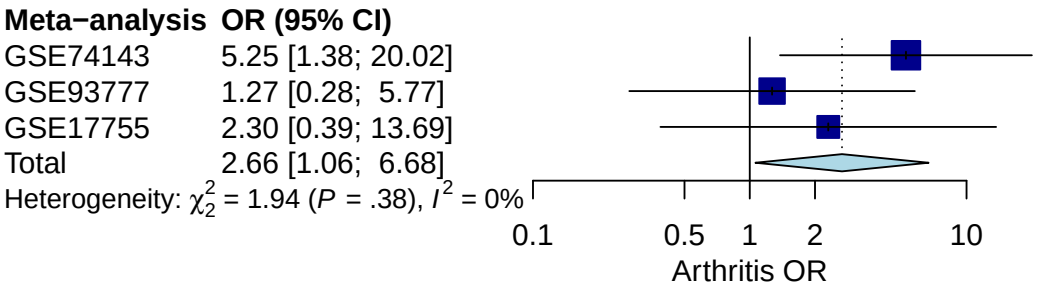


Figure S13

4.4 Meta-analysis

We meta-analyzed the associations of AR risk with the interaction between sex and the classification of individuals into the dimorphism groups.

```
Lres <- list(A1,A2,A3)

cellAl<-lapply(Lres,
  function(AA)
  {
    out <- AA$summary.model$coefficients
    out <- out["WTRUE:pf",c(1,2)]
    out
  })

cellAl <- do.call(rbind,cellAl)

datmet <- data.frame(TE = cellAl[,1], SE = cellAl[,2])

#Perform meta-analysis
metaAr <- metagen(TE, SE, data=datmet,
                studlab= c("GSE74143", "GSE93777", "GSE17755"),
                level.ci = 0.95, sm="OR",fixed = FALSE)


forest(metaAr, layout="JAMA",
       leftlabs=c("Meta-analysis", "OR (95% CI)"),
       xlab="Arthritis OR", title="Asthritis OR")
```

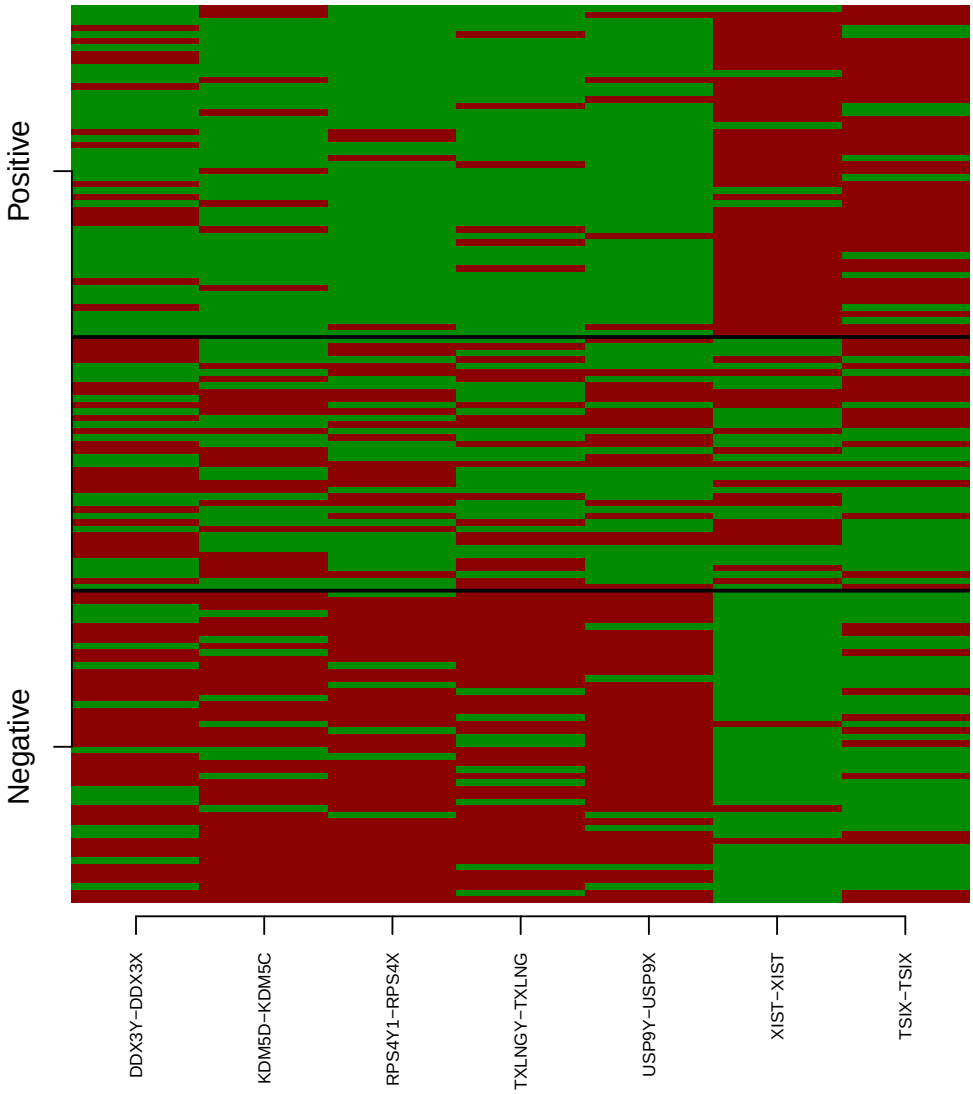

Meta−analysis OR (95% CI)
GSE74143 5.25 [1.38; 20.02]
GSE93777 1.27 [0.28; 5.77]
GSE17755 2.30 [0.39; 13.69]
Total 2.66 [1.06; 6.68]
Heterogeneity: χ^2_2 = 1.94 (P = .38), I^2 = 0%
0.1
0.5
1
2
10
Arthritis OR

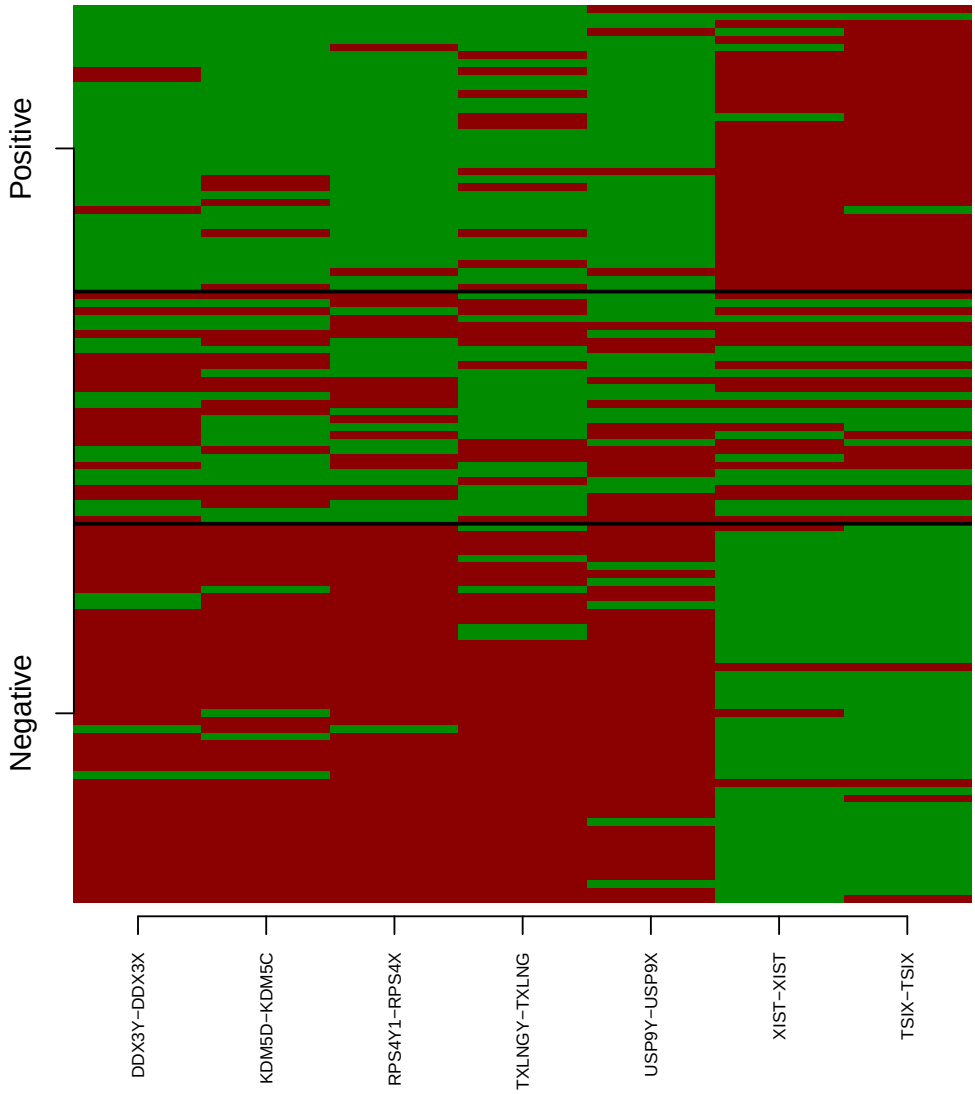
5 Asthma

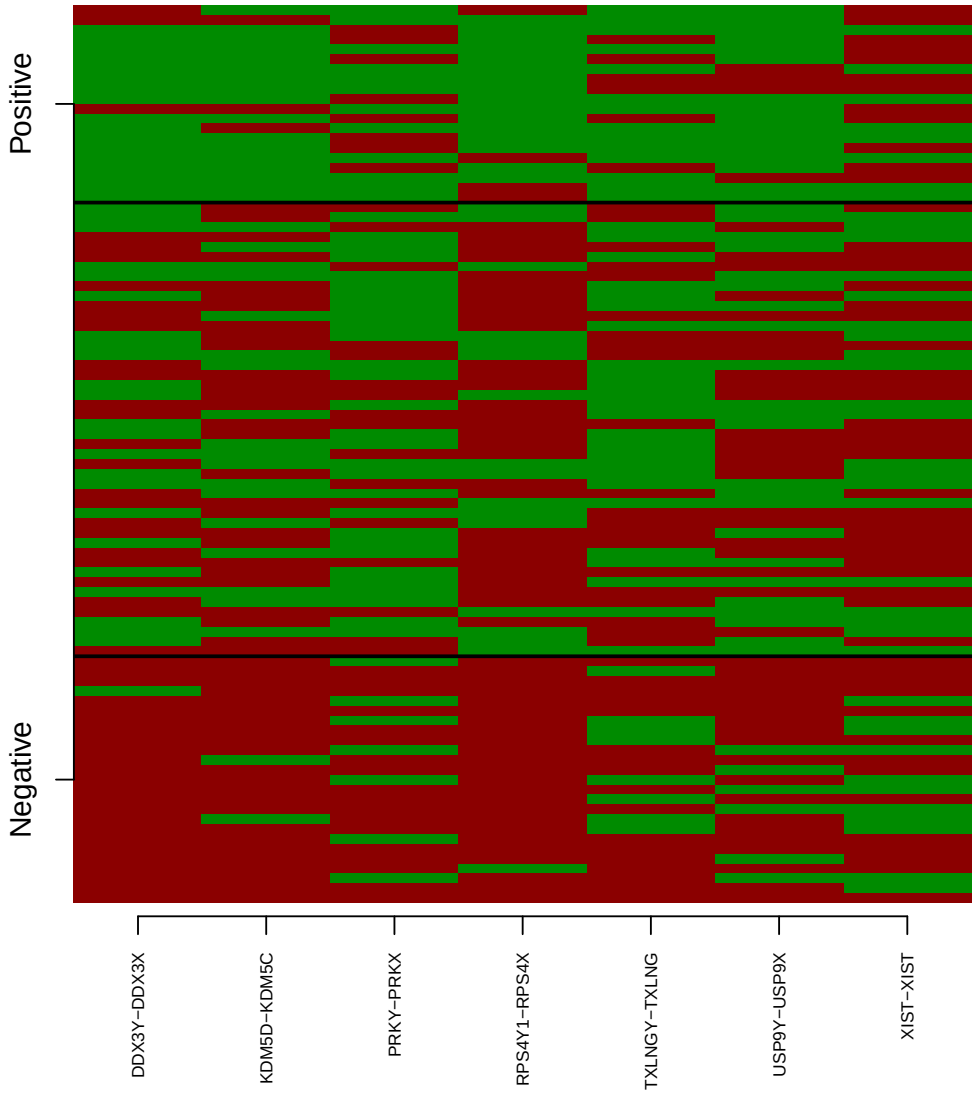
We downloaded three transcriptomic studies on asthma to test whether the association of the risk given by sex was modulated by the classification of individuals into the profiles of high immune sexual dimorphisms. We targeted all the individuals of the three studies with the profiles of immune sexual dimorphisms and then meta-analyzed the associations of asthma statis with the interaction between sex and the dimorphism groups.

```
# asthma

gsm <- getGEO("GSE41861", destdir ="./data", AnnotGPL =TRUE)
gsm <- gsm[[1]]

pp<- pData(phenoData(gsm))

datAsthma1 <- feateff(gsm, tname="Sex:ch1", reft=c("M", "F"),
                      effname="disease:ch1",
                      refeff=c("Control", "Asthma"),
                      covnames="age:ch1", covtype="n",
                      sva=TRUE, UsegeneSymbol=TRUE)

## Number of significant surrogate variables is:  15
## Iteration (out of 5 ):1  2  3  4  5

As1 <- target(datAsthma1, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="binomial", match=0.7)


#Asthma2
gsm <- getGEO("GSE41862", destdir ="./data", AnnotGPL =TRUE)
gsm <- gsm[[1]]

datAsthma2 <- feateff(gsm, tname="Sex:ch1", reft=c("M", "F"),
                      effname="disease:ch1",
                      refeff=c("Control", "Asthma"),
                      covnames="age:ch1", covtype="n",
                      sva=TRUE, UsegeneSymbol=TRUE)

## Number of significant surrogate variables is:  16
## Iteration (out of 5 ):1  2  3  4  5

As2 <- target(datAsthma2, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="binomial", match=0.7)


#Asthma3
gsm <- getGEO("GSE46171", destdir ="./data", getGPL = FALSE)

gp <- getGEO("GPL16981", destdir ="./data")
genesIDs <-(Table(gp)$"GENE_SYMBOL")
names(genesIDs) <- Table(gp)$ID
genesIDs <- genesIDs[rownames(gsm[[1]])]
```

```
expr <- cbind(exprs(gsm[[1]]), exprs(gsm[[2]]))
rownames(expr) <- genesIDs

phenobb <- rbind(pData(phenoData(gsm[[1]])),pData(phenoData(gsm[[2]])) )


eff <- phenobb$"group:ch1"
eff[grep("healthy", eff)] <- 0
eff[eff!=0] <- 1
eff <- as.numeric(eff)

age <- as.numeric(gsub(" yrs","", phenobb$"age:ch1"))

gender <- factor(phenobb$"gender:ch1")
gender <- -as.numeric(gender)+2

phenodat <- data.frame(t=gender, eff, age)

#get data for complete.cases only
ss <- sapply(1:nrow(expr), function(x) sum(is.na(expr[x,])))
expr <- expr[ss==0, ]

selcom <- complete.cases(phenodat) & complete.cases(t(expr))
phenodat <- phenodat[selcom, ]
expr <- expr[, selcom]

mod0 <- model.matrix( ~ t + eff + age, data = phenodat)
mod <- model.matrix( ~ t:eff + t + eff + age, data = phenodat)
ns <- num.sv(expr, mod, method="be")
ss <- sva(expr, mod, mod0, n.sv=ns)$sv

## Number of significant surrogate variables is:  12
## Iteration (out of 5 ):1  2  3  4  5

modss <- cbind(mod, ss)

datAsthma3 <- list(features=t(expr), teffdata=data.frame(modss,check.names = FALSE)
)

rmvars <- !colnames(datAsthma3$teffdata)%in%
            c("(Intercept)","t:eff")

datAsthma3$teffdata <- datAsthma3$teffdata[,rmvars]


As3 <- target(datAsthma3, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="binomial", match=0.7)
```

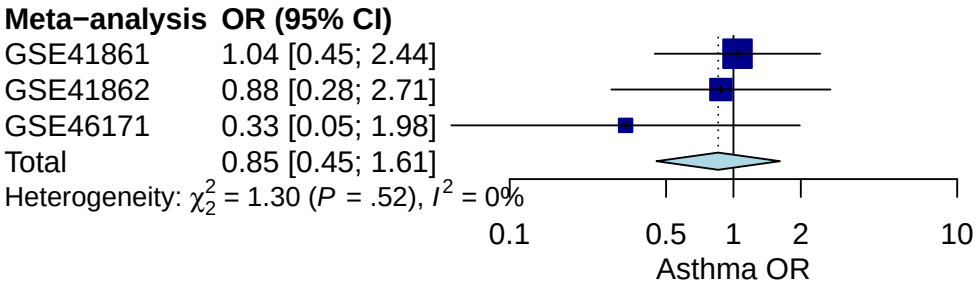

Positive
Negative
DDX3Y−DDX3X
KDM5D−KDM5C
RPS4Y1−RPS4X
TXLNGY−TXLNG
USP9Y−USP9X
XIST−XIST
TSIX−TSIX


Positive
Negative
DDX3Y−DDX3X
KDM5D−KDM5C
RPS4Y1−RPS4X
TXLNGY−TXLNG
USP9Y−USP9X
XIST−XIST
TSIX−TSIX




```
Asres <- list(As1, As2, As3)

cellAl<-lapply(Asres,
  function(AA)
  {
    out <- AA$summary.model$coefficients
    out <- out["WTRUE:pf",c(1,2)]
    out
  })

cellAl <- do.call(rbind,cellAl)

datmet <- data.frame(TE = cellAl[,1], SE = cellAl[,2])
```

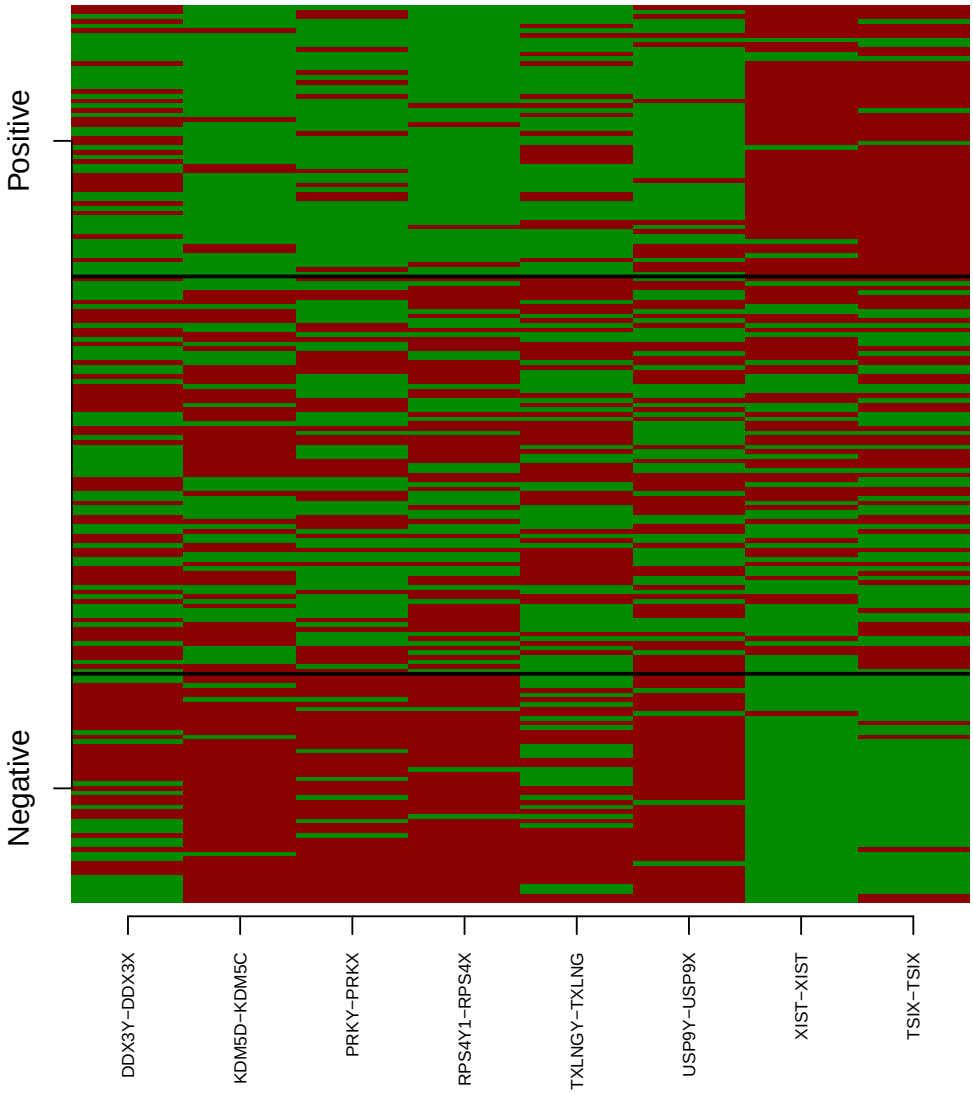
```r
#Perform meta-analysis
metaresAsthma <- metagen(TE, SE, data=datmet,
                 studlab= c("GSE41861", "GSE41862", "GSE46171"),
                 level.ci = 0.95, sm="OR",fixed = FALSE)


forest(metaresAsthma, layout="JAMA",
       leftlabs=c("Meta-analysis", "OR (95% CI)"),
       xlab="Asthma OR", title="Asthritis OR")
```



6 Anxiety

We finally downloaded a transcriptomic study of depression with data on anxiety. As anxiety is a sexually dimorphic trait, we tested whether the classification of individuals into the sexually dimorphic groups modulated the association between axiety and sex.

```
gsm <- getGEO("GSE98793", destdir ="./data", AnnotGPL =TRUE)
gsm <- gsm[[1]]

datAnxietey <- feateff(gsm, tname="gender:ch1", reft=c("M", "F"),
                       effname="anxiety:ch1",
                       refeff=c("no", "yes"),
                       covnames="age:ch1", covtype="n",
                       sva=TRUE, UsegeneSymbol=TRUE)

## Number of significant surrogate variables is:  16
## Iteration (out of 5 ):1  2  3  4  5

Anx <- target(datAnxietey, htrcells$"T cell",
              effect="positiveandnegative", featuresinf=XYhomol,
              model="binomial", match=0.7)


metaanx <- metagen(Anx$summary.model$coeff["WTRUE:pf",1],
                   Anx$summary.model$coeff["WTRUE:pf",2],
                   studlab= "GSE98793", level.ci = 0.95, sm="OR")


forest(metaanx, layout="JAMA",
       leftlabs=c("Estimated", "OR (95% CI)"), xlab="Anxiety OR", title=" OR", overall=FALSE)
```


Positive
Negative
DDX3Y−DDX3X
KDM5D−KDM5C
PRKY−PRKX
RPS4Y1−RPS4X
TXLNGY−TXLNG
USP9Y−USP9X
XIST−XIST
TSIX−TSIX

Estimated OR (95% CI)
GSE98793 1.52 [0.58; 4]