Introduction to XgeneR for homozygous crosses in multiple conditions.

Maria T. Carilli

```
devtools::load_all('../../XgeneR/')
```

i Loading XgeneR

```
# library(XgeneR)
```

Load in RNAseq count data and metadata describing the samples. This example uses samples from the liver and brown adipose tissue from male mice reared in cold and warm conditions from Ballinger et al. (2023).

The counts must be a matrix in the format genes by sample, with row names gene names and column names sample names. The metadata must include a column with header "Allele" that for every sample in the count columns indicates if the sample is from parent strain 1 (P1), parent strain 2 (P2), allele specific expression in hybrids of parental allele 1 (H1), or allele specific expression in hybrids of parental allele 2 (H2). The other fields in metadata must indicate condition, in this case "Tissue" and "Temperature".

```
count_path <- "/home/maria/XgeneR/inst/extdata/ballinger_counts.csv"
metadata_path <- "/home/maria/XgeneR/inst/extdata/ballinger_metadata.csv"


counts <- read.csv(count_path, row.names = 1)
counts <- as.matrix(counts)
metadata <- read.csv(metadata_path, row.names = 1)
```

Now, create an XgeneR fitObject, which requires at a minimum counts and metadata. If testing in multiple conditions, the character vector `fields_to_test` must include fields that match metadata column names..

```
# create fitObject
fit_obj <- new("fitObject",counts = counts, metadata = metadata, fields_to_test = c("Tissue","Temperatu
```

Run edgeR to produce raw p-values and Benjamini-Hochberg false discovery rates (corrected by the number of genes per test) per condition for the null hypotheses of "Condition null: no cis" regulation and no trans regulation ("Condition null: no trans"). The format of condition will be taken from the metadata. For example, for results for the liver of mice reared in warm conditions, the condition will be "Liver*Warm"

```
# run edgeR tests
fit_obj <- fit_edgeR(fit_obj)
```

Using classic mode.

[1] 8

```
head(fit_obj@raw_pvals[["Liver*Warm null: no cis"]])
```

```
## [1] 0.7664577 0.8824884 0.3506785 0.9992774 0.3131767 0.7188849
```

```
head(fit_obj@raw_pvals[["Liver*Warm null: no trans"]])
```

```
## [1] 0.74247482 0.82547143 0.33444342 0.92182058 0.26385794 0.08823016
```

```
head(fit_obj@BH_FDRs[["Liver*Warm null: no cis"]])
```

```
## [1] 0.51701645 0.73556166 0.09409198 0.99787639 0.07519314 0.44234226
```

```
head(fit_obj@BH_FDRs[["Liver*Warm null: no trans"]])
```

```
## [1] 0.410794431 0.559138992 0.048941510 0.786600841 0.027930283 0.002767823
```

Create diagnostic plots.

The following creates a histogram of raw p-values and Benjamini Hochberg corrected FDRs. MUST give the argument "combo" which sets the condition to show results for. The format is "cond1*cond2...", where each field is from a different column in the metadata that was tested in `fields_to_test`.

```
fig_dir <- "./figures"
if (!dir.exists(fig_dir)) {
  dir.create(fig_dir, recursive = TRUE)
}
png("./figures/pvalue_histogram_multiple-conditions.png", width=400, height = 400)
pval_plot <- plotPvalHistograms(fit_obj,combo="Liver*Warm")
ggplot2::ggsave("./figures/pvalue_histogram_multiple-conditions.png",plot=pval_plot,width=4,height=4)
```

The following function 'getAssignmentsandPlot' returns a data frame with regulatory assignments based on an FDR threshold of 'alpha' as well as a plot that produces a visualization of the log2 of parental ratios and hybrid ratios colored by assignment in untransformed and transformed coordinate systems and the proportion cis. Note again the argument "combo" indicating condition must be input. `results$df` is a dataframe regulatory assignments for the given `combo`, while `results$plot` is the plot.

```
png("./figures/tri_plot_multiple-conditions.png", width=400, height = 1000)
results <- getAssignmentsAndPlot(fit_obj,combo="Liver*Warm",alpha=0.05)
ggplot2::ggsave("./figures/tri_plot_multiple-conditions.png",plot=results$plot,width=4,height=10)
```

Finally, plot histogram of genes assigned to each category.

```
png("./figures/tri_plot.png", width=400, height = 400)
p <- plotRegulatoryHistogram (results$df,title="Liver*Warm")
```

```
## [1] "conserved" "trans"     "cisxtrans" "cis"       "cis+trans"
```

```r
ggplot2::ggsave("./figures/reg_histogram_multiple-conditions.png",plot=p,width=4,height=4)
```

```
```