Report about conserved transcriptional signatures and evolutionary relationships of germ/stem cells in the PIWI-piRNA pathway

Constantinos Yeles (Konstantinos Geles)

Last update : Mon Nov 15 2021

Contents

Conserved transcriptional signatures and evolutionary relationships of germ/stem cells

The PIWI-piRNA pathway expression in germ/stem cells

Using the Expression Atlas Bioconductor Package we will search about the expression profiles of genes that are involved in the piRNA biogenesis in various organisms and conditions.

Materials and Methods

The workflow has been primarily carried out on a Linux server, but it can be used easily on a Windows or Mac OS machine with plenty of RAM.

The workflow utilizes *R* scripting for various operations. For the application of the workflow, the following tools/ workflows have been used:

- *Rstudio* for R scripting,

- ***Expression Atlas*** for getting the datasets,
- *Creating a network of human gene homology with R and D3* for the identification of gene IDs and homologous genes between species with the help of **biomaRt** and **AnnotationDbi**

Workflow

Install and load libraries

```
if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")

BiocManager::install(c("ExpressionAtlas", "biomaRt", "AnnotationDbi",
                       "EnsDb.Hsapiens.v86", "tidybulk", "tidySummarizedExperiment"))

install.packages(c("tidyverse", "magrittr"))

suppressPackageStartupMessages({
  library('ExpressionAtlas')
  library('biomaRt')
  library('AnnotationDbi')
  library('EnsDb.Hsapiens.v86')
  library('tidybulk')
  library('tidySummarizedExperiment')
  library('tidyverse')
  library('magrittr')
})
```

Search which datasets in Expression Atlas have “stem” and “germ” terms

```
atlasRes_stem <- searchAtlasExperiments( properties = "stem")
atlasRes_germ <- searchAtlasExperiments( properties = "germ")
```

Search if there are identical datasets in both sets

```
atlasRes_germ %>% as_tibble %>% filter(is_in(Accession, atlasRes_stem$Accession))
```

join the two sets

```
atlasRes <- as_tibble(atlasRes_stem) %>% full_join(as_tibble(atlasRes_germ))

rm(atlasRes_stem, atlasRes_germ)
```

Search which species are in the dataset

```
atlasRes %>% dplyr::count(Species, sort = TRUE)
```

We will remove plant species from the dataset and microarray experiments

```
atlasRes <- atlasRes %>% dplyr::filter(is_in(Species,
                                            c("Mus musculus",
                                              "Drosophila melanogaster",
                                              "Rattus norvegicus",
                                              "Caenorhabditis elegans")),
                                       str_detect(Type, "RNA-seq"))
```

Get the AtlasData summaries

```
rnaseqExps <- getAtlasData(atlasRes$Accession)
```

scale the counts using tidybulk

```
datasets_atlass_scaled <- map(
    names(rnaseqExps)[1:2] %>%
         purrr::set_names(),
    ~ rnaseqExps %>%
        extract2(.x) %>%
        extract2("rnaseq") %>%
        identify_abundant(factor_of_interest = AtlasAssayGroup) %>%
        scale_abundance()
)

rm(atlasRes, rnaseqExps)
```

Following *Shirin's playgRound* instructions on getting gene identifiers

*with small modifications #### Use biomaRt to access the datasets of species we prefere

```
ensembl = useMart("ensembl")

datasets <- listDatasets(ensembl) %>%
    dplyr::filter(str_detect(dataset,
                  c("mmusculus|rnorvegicus|melanogaster|celegans")))


human <- useMart("ensembl", dataset = "hsapiens_gene_ensembl")
specieslist  <-  datasets$dataset
```

```
for (i in seq_along(specieslist)) {
    print(specieslist[i])
    ensembl <- datasets[i, 1]
    assign(paste0(ensembl), useMart("ensembl", dataset = paste0(ensembl)))
}
```

Make an object for each species of interest from the Ensembl database

```
# get all Ensembl gene IDs
human_EnsDb <- keys(EnsDb.Hsapiens.v86, keytype = "GENEID")

# IDs of piRNA biogenesis pathway genes
PIWI_pirna_biogenesis <- c("P24928", "P62487", "P52435", "P36954", "O15514",
                           "P61218", "P52434", "P62875", "P19387", 'P53803',
                           "P19388", "P30876", 'Q5T8I9', "Q7Z3Z4", "Q96JY0",
                           "Q9Y2W6", "Q8NDG6", "Q8TC59", "Q587J7", "Q8WWH4",
                           "Q9NQI0", "Q9BXT6", "Q9BXT4", "Q96J94", "O60522",
                           "Q8N2A8", "P07900", "O75344", "P10243")

# select on the above identifiers
gene_symbols <- AnnotationDbi::select(EnsDb.Hsapiens.v86,
                        keys = human_EnsDb,
                        columns = c("SYMBOL", "UNIPROTID", "GENEBIOTYPE")) %>%
    dplyr::filter(UNIPROTID %in% PIWI_pirna_biogenesis)
```

Identify human gene homologs for the Protein coding genes in piRNA biogenesis pathway
Get dataframe of human genes and their homologs

```
for (species in specieslist) {
    print(species)
    assign(paste0("homologs_human_", species), getLDS(attributes = c("ensembl_gene_id", "chromosome_nam
                                                      filters = "ensembl_gene_id",
                                                      values = gene_symbols$GENEID,
                                                      mart = human,
                                                      attributesL = c("ensembl_gene_id",
                                                                      "chromosome_name",
                                                                      "external_gene_name"),
                                                      martL = get(species)))
}

homologs_human_celegans_gene_ensembl <-  homologs_human_celegans_gene_ensembl %>%
    mutate(organism_mart = "celegans") %>%
    as_tibble() %>%
    mutate(Chromosome.scaffold.name.1 = as.character(Chromosome.scaffold.name.1))

homologs_human_dmelanogaster_gene_ensembl <-
    homologs_human_dmelanogaster_gene_ensembl %>%
    mutate(organism_mart = "dmelanogaster") %>%
    as_tibble() %>%
    mutate(Chromosome.scaffold.name.1 = as.character(Chromosome.scaffold.name.1))

homologs_human_mmusculus_gene_ensembl <-
    homologs_human_mmusculus_gene_ensembl %>%
    mutate(organism_mart = "mmusculus") %>%
    as_tibble()%>%
    mutate(Chromosome.scaffold.name.1 = as.character(Chromosome.scaffold.name.1))

homologs_human_rnorvegicus_gene_ensembl <-
    homologs_human_rnorvegicus_gene_ensembl %>%
    mutate(organism_mart = "rnorvegicus") %>%
```

```
    as_tibble()%>%
    mutate(Chromosome.scaffold.name.1 = as.character(Chromosome.scaffold.name.1))

piwi_genes <- dplyr::bind_rows(homologs_human_celegans_gene_ensembl,
                               homologs_human_dmelanogaster_gene_ensembl,
                               homologs_human_mmusculus_gene_ensembl,
                               homologs_human_rnorvegicus_gene_ensembl)
```

Keep only the piRNA biogenesis genes from the datasets

```
 piwi_genes_dat <- map(datasets_atlass_scaled,
    ~filter( .x, is_in(.feature, piwi_genes$Gene.stable.ID.1)) %>%
    left_join(distinct(piwi_genes, Gene.stable.ID.1, .keep_all = "TRUE"),
              by = c(.feature = "Gene.stable.ID.1" ))
    )

########################

atlasRes %>%
    names %>%
    set_names() %>%
    map( ~dplyr::count(as_tibble(atlasRes), .data[[.x]], sort = TRUE))
```