main

February 20, 2023

1 Feature summary of differential expression analysis

[1]:
```
import numpy as np
import pandas as pd
```

[2]:
```
def annotate_DE(feature):
    # Annotate DE results
    df = pd.read_csv(f'../../_m/{feature.lower()}s/diffExpr_maleVfemale_full.
 ↪txt',
                     sep='\t', index_col=0)\
           .rename(columns={"gene_id": "gencodeID", "gencodeGeneID":␣
 ↪"gencodeID",
                            "gene_name": "Symbol"})
    df = df[(df['adj.P.Val'] < 0.05)].sort_values('adj.P.Val')
    df['Feature'] = df.index
    df['ensemblID'] = df.gencodeID.str.replace("\\..*", "", regex=True)
    df['Type'] = feature; df["Region"] = "Hippocampus"
    return df[['Feature', 'Symbol', 'ensemblID',
               'logFC', 'SE', 'adj.P.Val', "Type"]]
```

1.1 Summary plots

1.1.1 Genes

[3]:
```
genes = annotate_DE("Gene")
genes.head(2)
```

[3]:
```
                                          Feature Symbol        ensemblID  \
KDM5D|ENSG00000012817.16  KDM5D|ENSG00000012817.16  KDM5D  ENSG00000012817
USP9Y|ENSG00000114374.13  USP9Y|ENSG00000114374.13  USP9Y  ENSG00000114374

                                logFC        SE  adj.P.Val  Type
KDM5D|ENSG00000012817.16  12.534081  0.100323        0.0  Gene
USP9Y|ENSG00000114374.13  13.663095  0.027720        0.0  Gene
```

1.1.2 Transcripts

[4]:
```
trans = annotate_DE("Transcript")
trans.head(2)
```

[4]:
```
                                                Feature Symbol  \
USP9Y-204|ENST00000440408.5  USP9Y-204|ENST00000440408.5  USP9Y
XIST-208|ENST00000602495.1    XIST-208|ENST00000602495.1   XIST

                                  ensemblID     logFC        SE  \
USP9Y-204|ENST00000440408.5  ENSG00000114374  4.688350  0.060887
XIST-208|ENST00000602495.1   ENSG00000229807 -7.629944  0.069951

                                 adj.P.Val        Type
USP9Y-204|ENST00000440408.5  6.653766e-261  Transcript
XIST-208|ENST00000602495.1   3.380416e-242  Transcript
```

1.1.3 Exons

[5]:
```
exons = annotate_DE("Exon")
exons.head(2)
```

[5]:
```
                                        Feature  Symbol        ensemblID  \
chrY:19703865-19706345-  chrY:19703865-19706345-   KDM5D  ENSG00000012817
chrY:14622009-14622591+  chrY:14622009-14622591+  NLGN4Y  ENSG00000165246

                             logFC        SE      adj.P.Val  Type
chrY:19703865-19706345-  10.269275  0.110655  7.952783e-291  Exon
chrY:14622009-14622591+   9.976809  0.125443  4.682665e-281  Exon
```

1.1.4 Junctions

[6]:
```
juncs = annotate_DE("Junction")
juncs.head(2)
```

[6]:
```
                                          Feature   Symbol        ensemblID  \
chrX:73833375-73837439:-  chrX:73833375-73837439:-  DDX11L1  ENSG00000223972
chrX:73829232-73831065:-  chrX:73829232-73831065:-  DDX11L1  ENSG00000223972

                             logFC        SE      adj.P.Val      Type
chrX:73833375-73837439:- -8.127682  0.109739  6.624010e-218  Junction
chrX:73829232-73831065:- -8.695041  0.140100  2.219527e-216  Junction
```

1.2 DE summary

1.2.1 DE (feature)

[7]:
```
gg = len(set(genes['Feature']))
tt = len(set(trans['Feature']))
ee = len(set(exons['Feature']))
jj = len(set(juncs['Feature']))

print(f"\nGene:\t\t{gg}\nTranscript:\t{tt}\nExon:\t\t{ee}\nJunction:\t{jj}")
```

```
Gene:           147
Transcript:     434
Exon:           2030
Junction:       887
```

DE (EnsemblID)

[8]:
```
gg = len(set(genes['ensemblID']))
tt = len(set(trans['ensemblID']))
ee = len(set(exons['ensemblID']))
jj = len(set(juncs['ensemblID']))

print(f"\nGene:\t\t{gg}\nTranscript:\t{tt}\nExon:\t\t{ee}\nJunction:\t{jj}")
```

```
Gene:           147
Transcript:     212
Exon:           341
Junction:       9
```

DE (Gene Symbol)

[9]:
```
gg = len(set(genes['Symbol']))
tt = len(set(trans['Symbol']))
ee = len(set(exons['Symbol']))
jj = len(set(juncs['Symbol']))

print(f"\nGene:\t\t{gg}\nTranscript:\t{tt}\nExon:\t\t{ee}\nJunction:\t{jj}")
```

```
Gene:           147
Transcript:     212
Exon:           345
Junction:       11
```

1.2.2 Feature effect size summary

[10]:
```
feature_list = ['Genes', 'Transcript', 'Exons', 'Junctions']
feature_df = [genes, trans, exons, juncs]
for ii in range(4):
    ff = feature_df[ii]
    half = len(set(ff[(np.abs(ff['logFC']) >= 0.5)].Feature))
    one = len(set(ff[(np.abs(ff['logFC']) >= 1)].Feature))
    print(f"\nThere are {half} unique {feature_list[ii]} with abs(log2FC) >= 0.
 ↪5")
    print(f"There are {one} unique {feature_list[ii]} with abs(log2FC) >= 1")
```

```
There are 87 unique Genes with abs(log2FC) >= 0.5
There are 51 unique Genes with abs(log2FC) >= 1

There are 297 unique Transcript with abs(log2FC) >= 0.5
There are 202 unique Transcript with abs(log2FC) >= 1

There are 1042 unique Exons with abs(log2FC) >= 0.5
There are 622 unique Exons with abs(log2FC) >= 1

There are 409 unique Junctions with abs(log2FC) >= 0.5
There are 276 unique Junctions with abs(log2FC) >= 1
```

[11]:
```
feature_list = ['Genes', 'Transcripts', 'Exons', 'Junctions']
feature_df = [genes, trans, exons, juncs]
for ii in range(4):
    ff = feature_df[ii]
    half = len(set(ff[(np.abs(ff['logFC']) >= 0.5)].ensemblID))
    one = len(set(ff[(np.abs(ff['logFC']) >= 1)].ensemblID))
    print(f"\nThere are {half} unique {feature_list[ii]} with abs(log2FC) >= 0.
 ↪5")
    print(f"There are {one} unique {feature_list[ii]} with abs(log2FC) >= 1")
```

```
There are 87 unique Genes with abs(log2FC) >= 0.5
There are 51 unique Genes with abs(log2FC) >= 1

There are 125 unique Transcripts with abs(log2FC) >= 0.5
There are 76 unique Transcripts with abs(log2FC) >= 1

There are 124 unique Exons with abs(log2FC) >= 0.5
There are 62 unique Exons with abs(log2FC) >= 1

There are 7 unique Junctions with abs(log2FC) >= 0.5
There are 6 unique Junctions with abs(log2FC) >= 1
```

1.3 Autosomal only

[12]:
```
from pyhere import here
from functools import lru_cache
```

[13]:
```
@lru_cache()
def get_annotation(feature):
    feat_lt = {"gene": "gene", "transcript": "tx",
               "exon": "exon", "junction": "jxn"}
    new_feature = feat_lt[feature]
    fn = here(f"input/counts/text_files_counts/_m/hippocampus/
 ↪{new_feature}_annotation.txt")
    return pd.read_csv(fn, sep='\t')
```

[14]:
```
def annotate_autosomes(feature):
    # Get annotation
    annot = get_annotation(feature.lower())
    # Annotate DE results
    df = pd.read_csv(f'../../_m/{feature.lower()}s/diffExpr_maleVfemale_full.
 ↪txt',
                     sep='\t', index_col=0)\
           .rename(columns={"gene_id": "gencodeID", "gencodeGeneID":␣
 ↪"gencodeID",
                            "gene_name": "Symbol"})
    df = df[(df['adj.P.Val'] < 0.05)].sort_values('adj.P.Val')
    df['name'] = df.index
    df['ensemblID'] = df.gencodeID.str.replace("\\..*", "", regex=True)
    df = annot.merge(df, on='name').rename(columns={"name": "Feature"})
    df = df[["Feature", "seqnames", "start", "end", "width", "gencodeID",␣
 ↪"ensemblID",
             "Symbol", "logFC", "AveExpr", "t", "P.Value", "adj.P.Val", "B",␣
 ↪"SE"]]
    df['Type'] = feature; df["Region"] = "Hippocampus"
    # Save annotated file
    df.sort_values('adj.P.Val').to_csv(f'chrom_annotation_{feature.lower()}.
 ↪txt',
                                       sep='\t', index=False)
    df = df[(df.seqnames.str.contains('chr\d+'))].copy()
    # Save autosomal DE features
    df.to_csv(f'{feature.lower()}_autosomal_DE.csv', index=False, header=True)
    return df[['Feature', 'seqnames', 'Symbol', 'ensemblID', 'logFC', 'SE',␣
 ↪'adj.P.Val', "Type"]]
```

1.3.1 Genes

[15]:
```
feature = "Gene"
genes = annotate_autosomes(feature)
genes.head(2)
```

[15]:
```
                     Feature seqnames Symbol        ensemblID     logFC  \
4   NLRP2|ENSG00000022556.16    chr19  NLRP2  ENSG00000022556 -0.840398
10   DDX43|ENSG00000080007.8     chr6  DDX43  ENSG00000080007  0.759825

           SE  adj.P.Val  Type
4   0.038033   0.000024  Gene
10  0.035350   0.000017  Gene
```

[16]:
```
genes.shape
```

[16]:
```
(68, 8)
```

[17]:
```
genes.groupby('ensemblID').first().reset_index().shape
```

[17]:
```
(68, 8)
```

1.3.2 Transcripts

[18]:
```
trans = annotate_autosomes("Transcript")
trans.head(2)
trans.shape
```

[18]:
```
(144, 8)
```

[19]:
```
trans.groupby('ensemblID').first().reset_index().shape
```

[19]:
```
(133, 8)
```

1.3.3 Exons

[20]:
```
exons = annotate_autosomes("Exon")
exons.head(2)
exons.shape
```

[20]:
```
(669, 8)
```

[21]:
```
exons.groupby('ensemblID').first().reset_index().shape
```

[21]:
```
(242, 8)
```

1.3.4 Junctions

[22]:
```
juncs = annotate_autosomes("Junction")
juncs.head(2)
juncs.shape
```

[22]:
```
(370, 8)
```

[23]:
```
juncs.groupby('ensemblID').first().reset_index().shape
```

[23]:
```
(3, 8)
```

1.4 DE summary

1.4.1 DE (feature)

[24]:
```
gg = len(set(genes['Feature']))
tt = len(set(trans['Feature']))
ee = len(set(exons['Feature']))
jj = len(set(juncs['Feature']))

print(f"\nGene:\t\t{gg}\nTranscript:\t{tt}\nExon:\t\t{ee}\nJunction:\t{jj}")
```

```
Gene:           68
Transcript:     144
Exon:           669
Junction:       370
```

DE (EnsemblID)

[25]:
```
gg = len(set(genes.groupby('ensemblID').first().reset_index()['ensemblID']))
tt = len(set(trans.groupby('ensemblID').first().reset_index()['ensemblID']))
ee = len(set(exons.groupby('ensemblID').first().reset_index()['ensemblID']))
jj = len(set(juncs.groupby('ensemblID').first().reset_index()['ensemblID']))

print(f"\nGene:\t\t{gg}\nTranscript:\t{tt}\nExon:\t\t{ee}\nJunction:\t{jj}")
```

```
Gene:           68
Transcript:     133
Exon:           242
Junction:       3
```

DE (Gene Symbol)

[26]:
```
gg = len(set(genes.groupby('Symbol').first().reset_index()['Symbol']))
tt = len(set(trans.groupby('Symbol').first().reset_index()['Symbol']))
ee = len(set(exons.groupby('Symbol').first().reset_index()['Symbol']))
```

```
jj = len(set(juncs.groupby('Symbol').first().reset_index()['Symbol']))

print(f"\nGene:\t\t{gg}\nTranscript:\t{tt}\nExon:\t\t{ee}\nJunction:\t{jj}")
```

```
Gene:           68
Transcript:     133
Exon:           244
Junction:       3
```

1.4.2 Feature effect size summary

[27]:
```
feature_list = ['Genes', 'Transcript', 'Exons', 'Junctions']
feature_df = [genes, trans, exons, juncs]
for ii in range(4):
    ff = feature_df[ii]
    half = len(set(ff[(np.abs(ff['logFC']) >= 0.5)].Feature))
    one = len(set(ff[(np.abs(ff['logFC']) >= 1)].Feature))
    print(f"\nThere are {half} unique {feature_list[ii]} with abs(log2FC) >= 0.
 ↪5")
    print(f"There are {one} unique {feature_list[ii]} with abs(log2FC) >= 1")
```

```
There are 39 unique Genes with abs(log2FC) >= 0.5
There are 13 unique Genes with abs(log2FC) >= 1

There are 74 unique Transcript with abs(log2FC) >= 0.5
There are 36 unique Transcript with abs(log2FC) >= 1

There are 210 unique Exons with abs(log2FC) >= 0.5
There are 30 unique Exons with abs(log2FC) >= 1

There are 150 unique Junctions with abs(log2FC) >= 0.5
There are 69 unique Junctions with abs(log2FC) >= 1
```

[28]:
```
feature_list = ['Genes', 'Transcripts', 'Exons', 'Junctions']
feature_df = [genes, trans, exons, juncs]
for ii in range(4):
    ff = feature_df[ii]
    half = len(set(ff[(np.abs(ff['logFC']) >= 0.5)].ensemblID))
    one = len(set(ff[(np.abs(ff['logFC']) >= 1)].ensemblID))
    print(f"\nThere are {half} unique {feature_list[ii]} with abs(log2FC) >= 0.
 ↪5")
    print(f"There are {one} unique {feature_list[ii]} with abs(log2FC) >= 1")
```

```
There are 39 unique Genes with abs(log2FC) >= 0.5
There are 13 unique Genes with abs(log2FC) >= 1
```

```
There are 72 unique Transcripts with abs(log2FC) >= 0.5
There are 36 unique Transcripts with abs(log2FC) >= 1

There are 59 unique Exons with abs(log2FC) >= 0.5
There are 17 unique Exons with abs(log2FC) >= 1

There are 3 unique Junctions with abs(log2FC) >= 0.5
There are 2 unique Junctions with abs(log2FC) >= 1
```

1.5 Session information

[29]:
```
import session_info
session_info.show()
```

[29]:
```
<IPython.core.display.HTML object>
```