main

December 10, 2020

1 Extract unique female specific SZ-associated genes

[1]:

```
import functools
import numpy as np
import pandas as pd
from scipy.stats import mannwhitneyu
from statsmodels.stats.multitest import fdrcorrection
```

[2]:

```
@functools.lru_cache()
def get_res_df():
    return pd.read_csv('../../../../../interaction_sex_sz/cmc_dlpfc/_m/genes/
 ↪residualized_expression.tsv', sep='\t').T


@functools.lru_cache()
def get_pheno_df():
    return pd.read_csv('/ceph/users/jbenja13/projects/sex_sz_ria/input/
 ↪commonMind/phenotypes/combine_files/_m/CMC_phenotypes_all.csv').
 ↪set_index("RNAseq:Sample_RNA_ID")


@functools.lru_cache()
def get_res_pheno_df():
    return pd.merge(get_pheno_df(), get_res_df(), left_index=True,␣
 ↪right_index=True)
```

[3]:

```
def get_de(feature):
    f = pd.read_csv('../../../../female_analysis/_m/%s/diffExpr_szVctl_full.txt' %␣
 ↪feature, sep='\t')\
           .rename(columns={'gene_id': 'gencodeID'})
    f['ensemblID'] = f.gencodeID.str.replace("\\..*", "")
    f.set_index('ensemblID', inplace=True)
    m = pd.read_csv('../../../../male_analysis/_m/%s/diffExpr_szVctl_full.txt' %␣
 ↪feature, sep='\t')\
           .rename(columns={'gene_id': 'gencodeID'})
    m['ensemblID'] = m.gencodeID.str.replace("\\..*", "")
    m.set_index('ensemblID', inplace=True)
    a = pd.read_csv('/ceph/projects/v3_phase3_paper/inputs/cmc/_m/
 ↪CMC_MSSM-Penn-Pitt_DLPFC_mRNA_IlluminaHiSeq2500_gene-adjustedSVA-differentialExpression-inc
 ↪tsv', sep='\t')\
            .rename(columns={"MAPPED_genes": 'gene_name'}).set_index('genes')
    return f, m, a


def get_unique(x, y, thres=0.05):
    return x.merge(pd.DataFrame(index = list(set(x[(x['adj.P.Val'] <= thres)].
 ↪index) -
                                             set(y[(y['adj.P.Val'] <= thres)].
 ↪index))),
                         left_index=True, right_index=True)

def subset_sz_male():
    df = get_res_pheno_df()
    ctl = df[(df['Dx'] == 'Control') & (df['Sex'] == 'XY')].copy()
    sz = df[(df['Dx'] == 'SCZ') & (df['Sex'] == 'XY')].copy()
    return ctl, sz


def add_pvals_adjustPval(df):
    ctl, sz = subset_sz_male()
    m_pval = []
    for gene_id in df.Feature:
        stat, pval = mannwhitneyu(ctl[gene_id], sz[gene_id])
        m_pval.append(pval)
    fdr_m = fdrcorrection(m_pval)
    return pd.concat([df.set_index('Feature'),
                      pd.DataFrame({'Male_Pval': m_pval,
                                    'Male_FDR': fdr_m[1]},
                                   index=df.Feature)], axis=1)
```

1.1 Genes

[4]:

```
f, m, a = get_de('genes')
f['Feature'] = f.gencodeID
f['ensemblID'] = f.index
#genes = get_unique(get_unique(f, m), a)
genes = get_unique(f, m).rename(columns={'chromosome_name': 'Chrom',␣
 ↪'hgnc_symbol': 'Symbol'})
genes = genes[['Feature', 'gencodeID', 'Symbol', 'ensemblID',
               'Chrom', 'logFC', 't', 'adj.P.Val']].sort_values('adj.P.Val')
genes.Chrom = 'chr'+genes.Chrom
genes = add_pvals_adjustPval(genes)
genes = genes[~(genes['Male_Pval'] <= 0.05)] ## Stringents
```

```
genes['Type'] = 'gene'
genes.shape
```

[4]:

```
(583, 10)
```

[5]:

```
genes[(genes['t']>0)].head(5)
```

[5]:

```
                                  gencodeID     Symbol        ensemblID  Chrom  \
Feature
ENSG00000153132.12  ENSG00000153132.12       CLGN  ENSG00000153132   chr4
ENSG00000179083.6    ENSG00000179083.6     FAM133A  ENSG00000179083   chrX
ENSG00000165733.7    ENSG00000165733.7        BMS1  ENSG00000165733  chr10
ENSG00000183023.18  ENSG00000183023.18      SLC8A1  ENSG00000183023   chr2
ENSG00000236268.5    ENSG00000236268.5   LINC01361  ENSG00000236268   chr1

                       logFC         t  adj.P.Val  Male_Pval  Male_FDR  Type
Feature
ENSG00000153132.12  0.389937  5.559139   0.000123   0.283020  0.313711  gene
ENSG00000179083.6   0.261268  5.004488   0.000535   0.272019  0.302973  gene
ENSG00000165733.7   0.150918  4.986552   0.000535   0.169787  0.205428  gene
ENSG00000183023.18  0.245819  4.925477   0.000632   0.082759  0.108963  gene
ENSG00000236268.5   0.404532  4.865744   0.000700   0.110632  0.140404  gene
```

1.2 DE summary

1.2.1 DE (feature)

[6]:

```
gg = len(set(genes['gencodeID']))

print("\nGene:\t\t%d" % (gg))
```

```

Gene:           583
```

[7]:

```
genes.to_csv('female_specific_DE_genes.txt', sep='\t', index=True, header=True)
```

1.3 Number of DEGs on allosomes

[8]:

```
genes[(genes['Chrom'].isin(['chrX', 'chrY']))].groupby(['Type', 'Chrom']).size()
```

[8]:

```
Type  Chrom
gene  chrX     21
dtype: int64
```

[ ]:

```
```