main

July 9, 2021

1   Examine genes that are disconcordant across brain regrions

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

```python
[2]: config = {
         'caudate': '../../../../_m/genes/diffExpr_szVctl_full.txt',
         'dlpfc': '/ceph/projects/v4_phase3_paper/inputs/public_data/_m/phase2/
     ↪dlpfc_diffExpr_szVctl_full.txt',
         'hippo': '/ceph/projects/v4_phase3_paper/inputs/public_data/_m/phase2/
     ↪hippo_diffExpr_szVctl_full.txt',
     }
```

```python
[3]: @functools.lru_cache()
     def get_deg(tissue):
         dft = pd.read_csv(config[tissue], sep='\t', index_col=0)
         dft['Feature'] = dft.index
         dft['Dir'] = np.sign(dft['t'])
         if 'gene_id' in dft.columns:
             dft['ensemblID'] = dft.gene_id.str.replace('\\..*', '')
         elif 'ensembl_gene_id' in dft.columns:
             dft.rename(columns={'ensembl_gene_id': 'ensemblID'}, inplace=True)
         return dft[['Feature', 'ensemblID', 'Symbol', 'adj.P.Val', 'logFC', 't',
     ↪'Dir']]


     @functools.lru_cache()
     def get_deg_sig(tissue, fdr):
         dft = get_deg(tissue)
         return dft[(dft['adj.P.Val'] < fdr)]


     @functools.lru_cache()
     def merge_dataframes(tissue1, tissue2):
         return get_deg(tissue1).merge(get_deg(tissue2), on='Feature',
                                       suffixes=['_%s' % tissue1, '_%s' % tissue2])
```

```
@functools.lru_cache()
def merge_dataframes_sig(tissue1, tissue2):
    fdr1 = 0.05 if tissue1 != 'dlpfc' else 0.05
    fdr2 = 0.05 if tissue2 != 'dlpfc' else 0.05
    return get_deg_sig(tissue1, fdr1).merge(get_deg_sig(tissue2, fdr2),
 ↪on='Feature',
                                            suffixes=['_%s' % tissue1, '_%s' %
 ↪tissue2])
```

```
[4]: def extract_disconcordant(tissue1, tissue2):
    df = merge_dataframes_sig(tissue1, tissue2)
    df = df[((df['Dir_%s' % tissue1] == 1) & (df['Dir_%s' % tissue2] == -1)) |
            ((df['Dir_%s' % tissue1] == -1) & (df['Dir_%s' % tissue2] == 1))]
    return df.loc[:, ['Feature', 'Dir_%s' % tissue1, 'Dir_%s' % tissue2]]\
            .merge(get_deg(tissue1), on='Feature').drop('Dir', axis=1)
```

1.1 BrainSeq Tissue Comparison

```
[5]: cd = extract_disconcordant('caudate', 'dlpfc')
cd.to_csv("disconcordant_genes_%s_%s.csv" % ('caudate', 'dlpfc'), index=False)
print(cd.shape)
cd
```

```
(17, 8)
```

[5]:

	Feature	Dir_caudate	Dir_dlpfc	ensemblID	Symbol	\
0	ENSG00000188730.4	1.0	-1.0	ENSG00000188730	VWC2	
1	ENSG00000132639.12	1.0	-1.0	ENSG00000132639	SNAP25	
2	ENSG00000116679.15	1.0	-1.0	ENSG00000116679	IVNS1ABP	
3	ENSG00000100285.9	1.0	-1.0	ENSG00000100285	NEFH	
4	ENSG00000136040.8	-1.0	1.0	ENSG00000136040	PLXNC1	
5	ENSG00000204856.11	1.0	-1.0	ENSG00000204856	FAM216A	
6	ENSG00000036530.8	-1.0	1.0	ENSG00000036530	CYP46A1	
7	ENSG00000111262.4	1.0	-1.0	ENSG00000111262	KCNA1	
8	ENSG00000151917.17	1.0	-1.0	ENSG00000151917	BEND6	
9	ENSG00000143858.11	1.0	-1.0	ENSG00000143858	SYT2	
10	ENSG00000177181.14	1.0	-1.0	ENSG00000177181	RIMKLA	
11	ENSG00000006747.14	1.0	-1.0	ENSG00000006747	SCIN	
12	ENSG00000082497.11	1.0	-1.0	ENSG00000082497	SERTAD4	
13	ENSG00000118242.15	1.0	-1.0	ENSG00000118242	MREG	
14	ENSG00000101079.20	1.0	-1.0	ENSG00000101079	NDRG3	
15	ENSG00000105711.10	1.0	-1.0	ENSG00000105711	SCN1B	
16	ENSG00000232973.11	-1.0	1.0	ENSG00000232973	CYP1B1-AS1	

```
        adj.P.Val      logFC           t
```

```
0    1.288636e-13   0.372958   8.788995
1    3.108746e-07   0.200929   6.221280
2    1.107091e-06   0.204549   5.961901
3    2.902409e-05   0.262753   5.200391
4    1.714953e-04  -0.141934  -4.751764
5    6.751079e-04   0.078137   4.367314
6    1.860460e-03  -0.088329  -4.050691
7    2.291543e-03   0.126388   3.979680
8    1.075646e-02   0.075881   3.431775
9    1.247544e-02   0.133316   3.370904
10   1.699589e-02   0.070601   3.243639
11   1.866443e-02   0.164288   3.208440
12   2.527903e-02   0.079187   3.077812
13   2.527903e-02   0.112288   3.077800
14   3.459039e-02   0.038372   2.938792
15   4.230935e-02   0.065344   2.849365
16   4.774394e-02  -0.154245  -2.792397
```

[6]:
```python
ch = extract_disconcordant('caudate', 'hippo')
ch.to_csv("disconcordant_genes_%s_%s.csv" % ('caudate', 'hippo'), index=False)
print(ch.shape)
ch
```

```
(2, 8)
```

[6]:
```
            Feature  Dir_caudate  Dir_hippo         ensemblID  Symbol  \
0   ENSG00000128203.6          1.0       -1.0  ENSG00000128203   ASPHD2
1  ENSG00000132872.11          1.0       -1.0  ENSG00000132872    SYT4

    adj.P.Val      logFC         t
0    0.000008   0.174283  5.518281
1    0.020210   0.112905  3.173980
```

[7]:
```python
print("There are %d genes where caudate is different from both DLPFC and␣
 ↪hippocampus!\n" %
 len(set(ch.ensemblID ) & set(cd.ensemblID)))

cd[(cd['ensemblID'].isin(list(set(ch.ensemblID ) & set(cd.ensemblID))))]
```

```
There are 0 genes where caudate is different from both DLPFC and hippocampus!
```

[7]:
```
Empty DataFrame
Columns: [Feature, Dir_caudate, Dir_dlpfc, ensemblID, Symbol, adj.P.Val, logFC,
t]
Index: []
```

```
[8]: dh = extract_disconcordant('dlpfc', 'hippo')
     #dh.to_csv("disconcordant_genes_%s_%s.csv" % ('dlpfc', 'hippo'), index=False)
     print(dh.shape)
     dh
```

```
(0, 8)
```

```
[8]: Empty DataFrame
     Columns: [Dir_dlpfc, Dir_hippo, Feature, ensemblID, Symbol, adj.P.Val, logFC, t]
     Index: []
```

```
[9]: print("There are %d genes where hippocampus is different from both caudate and␣
      ↪DLPFC!\n" %
      len(set(dh.ensemblID ) & set(ch.ensemblID)))

     dh[(dh['ensemblID'].isin(list(set(dh.ensemblID ) & set(ch.ensemblID))))]
```

```
There are 0 genes where hippocampus is different from both caudate and DLPFC!
```

```
[9]: Empty DataFrame
     Columns: [Dir_dlpfc, Dir_hippo, Feature, ensemblID, Symbol, adj.P.Val, logFC, t]
     Index: []
```

```
[10]: print("There are %d genes where DLPFC is different from both caudate and␣
       ↪hippocampus!\n" %
       len(set(dh.ensemblID ) & set(cd.ensemblID)))
```

```
There are 0 genes where DLPFC is different from both caudate and hippocampus!
```

```
[ ]:
```