# main

August 23, 2021

## 1 Tissue comparison for differential expression analysis

[1]:

```
import functools
import numpy as np
import pandas as pd
from plotnine import *
from scipy.stats import binom_test, fisher_exact, linregress

from warnings import filterwarnings
from matplotlib.cbook import mplDeprecation
filterwarnings('ignore', category=mplDeprecation)
filterwarnings('ignore', category=UserWarning, module='plotnine.*')
filterwarnings('ignore', category=DeprecationWarning, module='plotnine.*')
```

[2]:

```
config = {
    'caudate': '../../../caudate/_m/genes/diffExpr_EAvsAA_full.txt',
    'dlpfc': '../../../dlpfc/_m/genes/diffExpr_EAvsAA_full.txt',
    'hippo': '../../../hippocampus/_m/genes/diffExpr_EAvsAA_full.txt',
    'gyrus': '../../../dentateGyrus/_m/genes/diffExpr_EAvsAA_full.txt',
}
```

[3]:

```
@functools.lru_cache()
def get_deg(filename):
    dft = pd.read_csv(filename, 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('\\..*', '', regex=True)
    elif 'ensembl_gene_id' in dft.columns:
        dft.rename(columns={'ensembl_gene_id': 'ensemblID'}, inplace=True)
    return dft[['Feature', 'ensemblID', 'adj.P.Val', 'logFC', 't', 'Dir']]

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

```
@functools.lru_cache()
def merge_dataframes(tissue1, tissue2):
    return get_deg(config[tissue1]).merge(get_deg(config[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(config[tissue1], fdr1).
 ↪merge(get_deg_sig(config[tissue2], fdr2),
                                                    on='Feature',
                                                    suffixes=['_%s' % tissue1,␣
 ↪'_%s' % tissue2])
```

[4]:

```
def enrichment_binom(tissue1, tissue2, merge_fnc):
    df = merge_fnc(tissue1, tissue2)
    df['agree'] = df['Dir_%s' % tissue1] * df['Dir_%s' % tissue2]
    dft = df.groupby('agree').size().reset_index()
    print(dft)
    return binom_test(dft[0].iloc[1], dft[0].sum()) if dft.shape[0] != 1 else␣
 ↪print("All directions agree!")


def cal_fishers(tissue1, tissue2):
    df = merge_dataframes(tissue1, tissue2)
    fdr1 = 0.05 if tissue1 != 'dlpfc' else 0.05
    fdr2 = 0.05 if tissue2 != 'dlpfc' else 0.05
    table = [[np.sum((df['adj.P.Val_%s' % tissue1]<fdr1) &
                     ((df['adj.P.Val_%s' % tissue2]<fdr2))),
              np.sum((df['adj.P.Val_%s' % tissue1]<fdr1) &
                     ((df['adj.P.Val_%s' % tissue2]>=fdr2)))],
             [np.sum((df['adj.P.Val_%s' % tissue1]>=fdr1) &
                     ((df['adj.P.Val_%s' % tissue2]<fdr2))),
              np.sum((df['adj.P.Val_%s' % tissue1]>=fdr1) &
                     ((df['adj.P.Val_%s' % tissue2]>=fdr2)))]]
    print(table)
    return fisher_exact(table, alternative='greater')


def calculate_corr(xx, yy):
    '''This calculates R^2 correlation via linear regression:
         - used to calculate relationship between 2 arrays
         - the arrays are principal components 1 or 2 (PC1, PC2) AND gender
         - calculated on a scale of 0 to 1 (with 0 being no correlation)
        Inputs:
          x: array of Gender (converted to binary output)
          y: array of PC
        Outputs:
          1. r2
          2. p-value, two-sided test
            - whose null hypothesis is that two sets of data are uncorrelated
          3. slope (beta): directory of correlations
    '''
    slope, intercept, r_value, p_value, std_err = linregress(xx, yy)
    return r_value, p_value


def corr_annotation(tissue1, tissue2, merge_fnc):
    dft = merge_fnc(tissue1, tissue2)
    xx = dft['t_%s' % tissue1]
    yy = dft['t_%s' % tissue2]
    r_value1, p_value1 = calculate_corr(xx, yy)
    return 'R2: %.2f\nP-value: %.2e' % (r_value1**2, p_value1)


def tissue_annotation(tissue):
    return {'dlpfc': 'DLPFC', 'hippo': 'Hippocampus',
            'caudate': 'Caudate', 'gyrus': 'Dentate Gyrus'}[tissue]
```

[5]:

```
def plot_corr_impl(tissue1, tissue2, merge_fnc):
    dft = merge_fnc(tissue1, tissue2)
    title = '\n'.join([corr_annotation(tissue1, tissue2, merge_fnc)])
    xlab = 'T-statistic (%s)' % tissue_annotation(tissue1)
    ylab = 'T-statistic (%s)' % tissue_annotation(tissue2)
    pp = ggplot(dft, aes(x='t_%s'%tissue1, y='t_%s' % tissue2))\
    + geom_point(alpha=0.75, size=3)\
    + theme_matplotlib()\
    + theme(axis_text=element_text(size=18),
            axis_title=element_text(size=20, face='bold'),
            plot_title=element_text(size=22))
    pp += labs(x=xlab, y=ylab, title=title)
    return pp


def plot_corr(tissue1, tissue2, merge_fnc):
    return plot_corr_impl(tissue1, tissue2, merge_fnc)


def save_plot(p, fn, width=7, height=7):
    '''Save plot as svg, png, and pdf with specific label and dimension.'''
    for ext in ['.svg', '.png', '.pdf']:
        p.save(fn+ext, width=width, height=height)
```

## 1.1 BrainSeq Tissue Comparison

```
[6]: caudate = get_deg(config['caudate'])
     caudate.groupby('Dir').size()
```

```
[6]: Dir
     -1.0    10767
      1.0    11607
     dtype: int64
```

```
[7]: caudate[(caudate['adj.P.Val'] < 0.05)].shape
```

```
[7]: (2970, 6)
```

```
[8]: dlpfc = get_deg(config['dlpfc'])
     dlpfc.groupby('Dir').size()
```

```
[8]: Dir
     -1.0    11691
      1.0    10707
     dtype: int64
```

```
[9]: dlpfc[(dlpfc['adj.P.Val'] < 0.05)].shape
```

```
[9]: (2760, 6)
```

```
[10]: hippo = get_deg(config['hippo'])
      hippo.groupby('Dir').size()
```

```
[10]: Dir
      -1.0    11213
       1.0    11056
      dtype: int64
```

```
[11]: hippo[(hippo['adj.P.Val'] < 0.05)].shape
```

```
[11]: (2956, 6)
```

```
[12]: gyrus = get_deg(config['gyrus'])
      gyrus.groupby('Dir').size()
```

```
[12]: Dir
      -1.0    10855
       1.0    10285
```

```
dtype: int64
```

```
[13]: gyrus[(gyrus['adj.P.Val'] < 0.05)].shape
```

```
[13]: (786, 6)
```

### 1.1.1 Enrichment of DEG

```
[14]: cal_fishers('caudate', 'dlpfc')
```

```
[[1115, 1692], [1507, 16814]]
```

```
[14]: (7.352453718737303, 0.0)
```

```
[15]: cal_fishers('caudate', 'hippo')
```

```
[[1142, 1681], [1726, 16648]]
```

```
[15]: (6.552690661010558, 0.0)
```

```
[16]: cal_fishers('dlpfc', 'hippo')
```

```
[[1251, 1437], [1610, 17300]]
```

```
[16]: (9.354504078113045, 0.0)
```

```
[17]: cal_fishers('caudate', 'gyrus')
```

```
[[311, 2231], [415, 16472]]
```

```
[17]: (5.532979430046497, 1.069007184730363e-91)
```

```
[18]: cal_fishers('dlpfc', 'gyrus')
```

```
[[342, 2117], [386, 16989]]
```

```
[18]: (7.110264549746562, 1.068424000195717e-122)
```

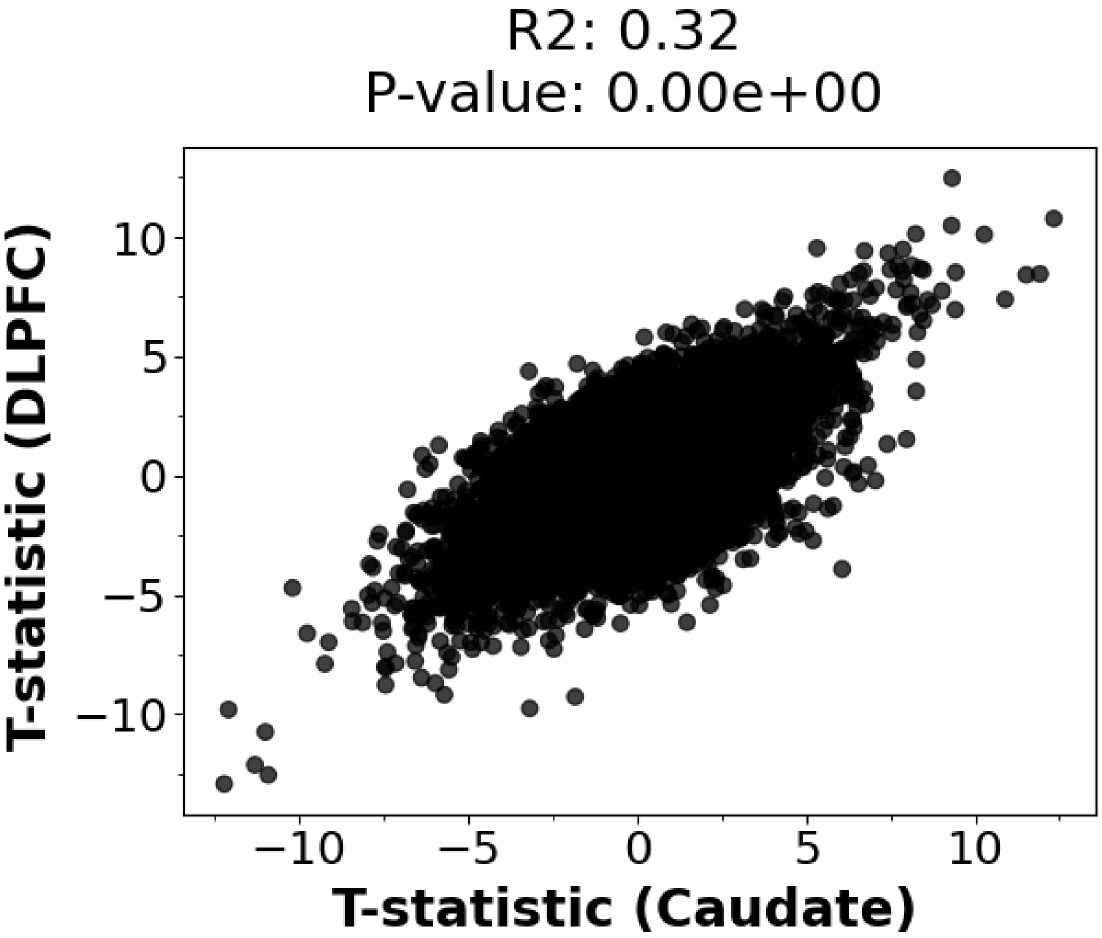
```
[19]: cal_fishers('hippo', 'gyrus')
```

```
[[361, 2267], [382, 16834]]
```

```
[19]: (7.0174550862939, 3.432494766723408e-126)
```

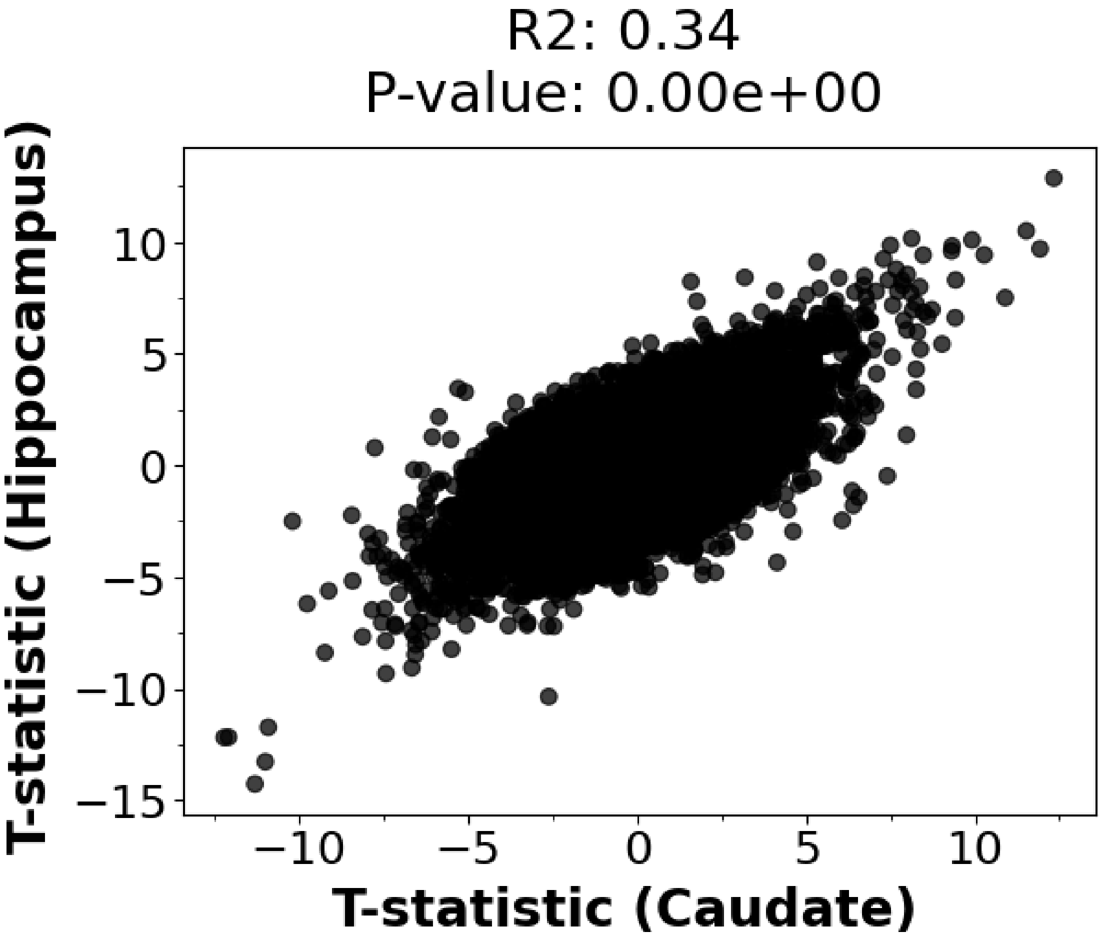
### 1.1.2 Correlation

```
[20]: pp = plot_corr('caudate', 'dlpfc', merge_dataframes)
      pp
```



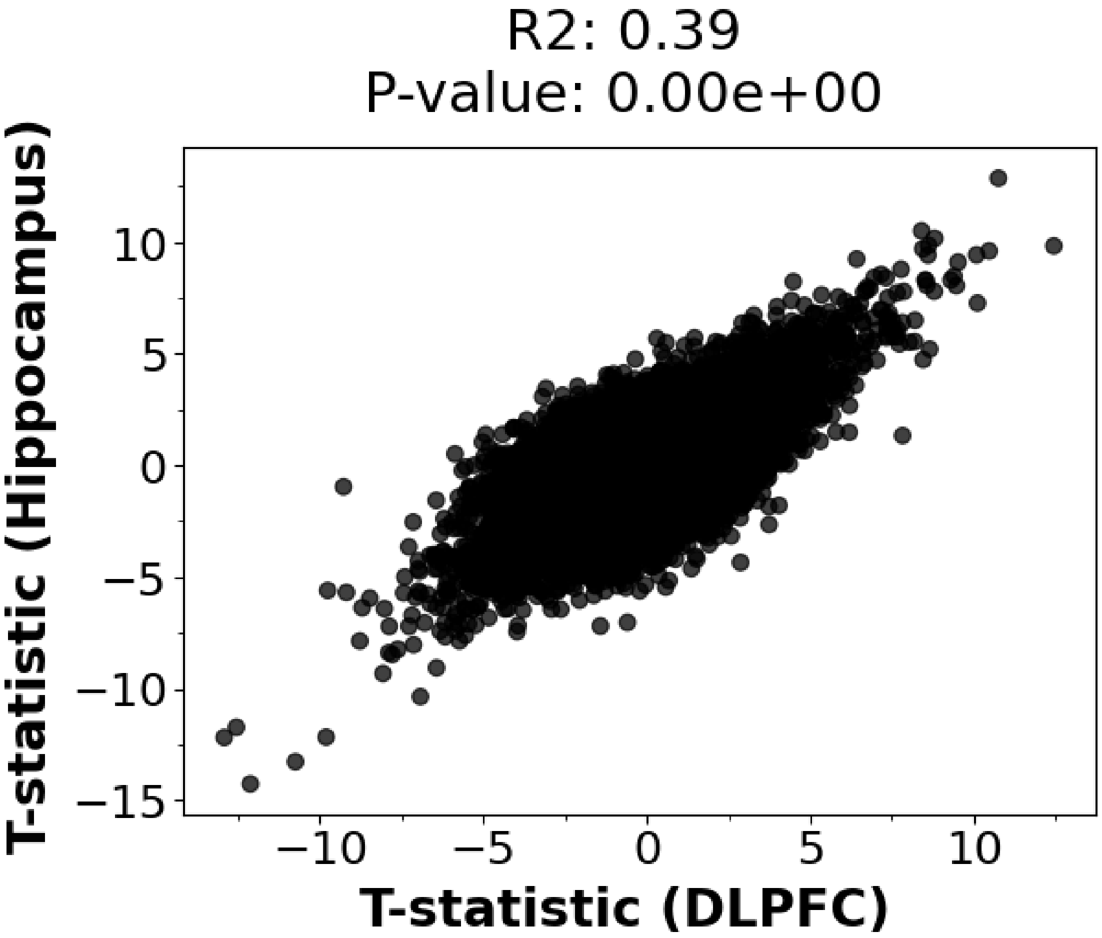
```
[20]: <ggplot: (8759208470616)>
```

```
[21]: qq = plot_corr('caudate', 'hippo', merge_dataframes)
      qq
```



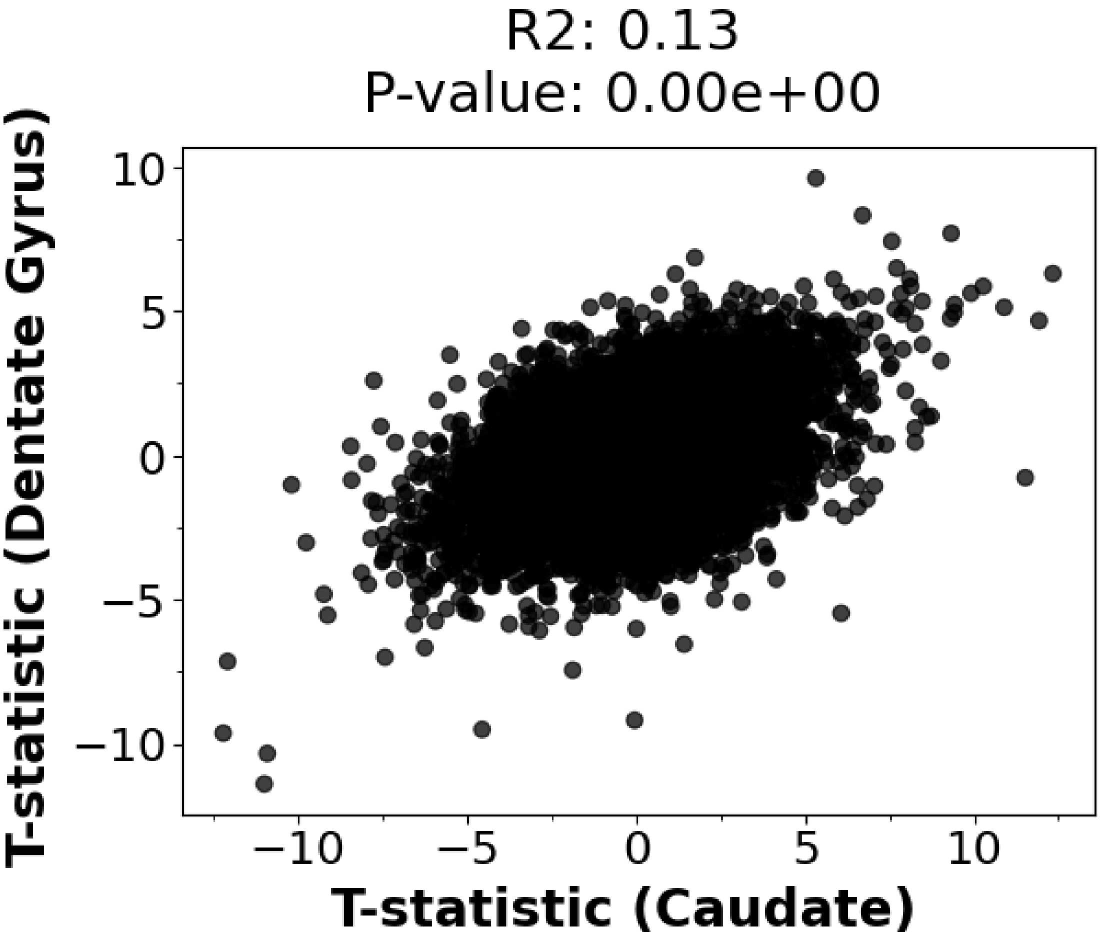
```
[21]: <ggplot: (8759213660871)>
```

```
[22]: ww = plot_corr('dlpfc', 'hippo', merge_dataframes)
      ww
```



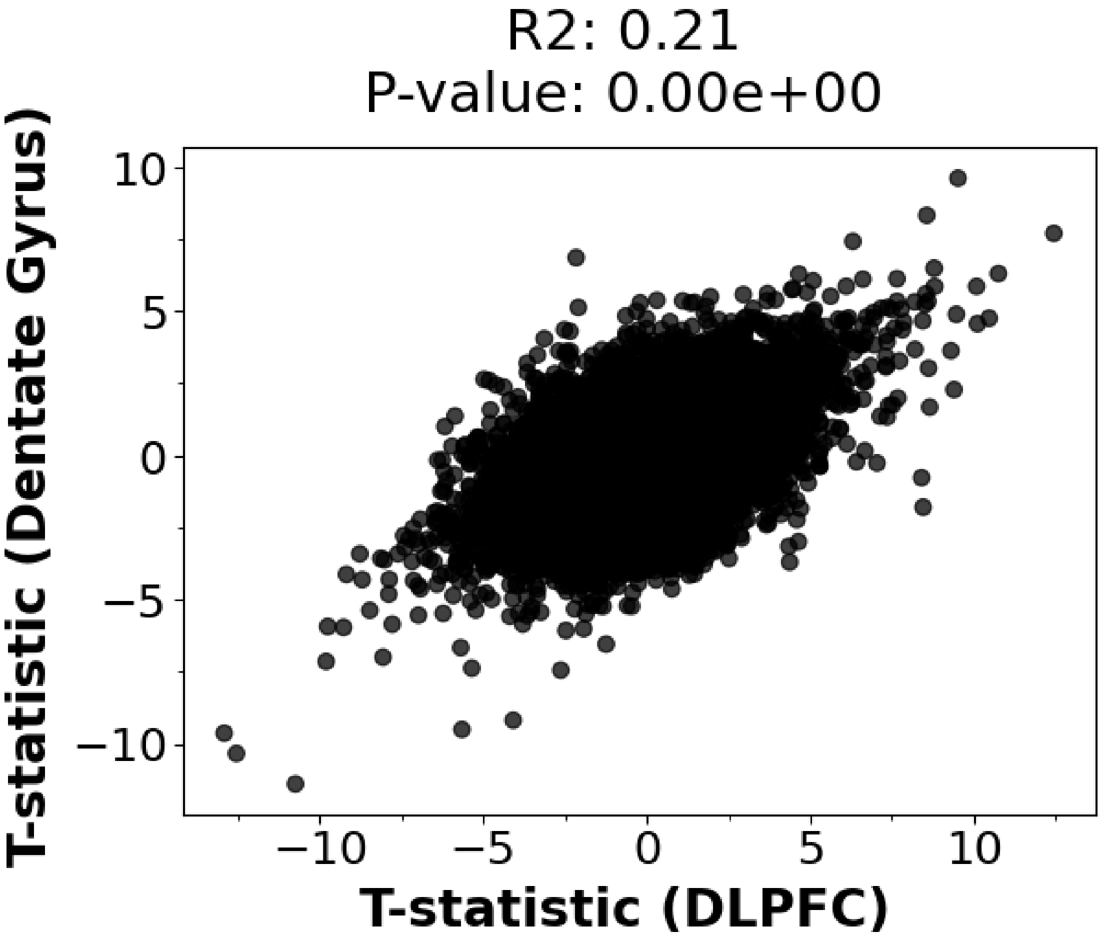
```
[22]: <ggplot: (8759208557599)>
```

```
[23]: rr = plot_corr('caudate', 'gyrus', merge_dataframes)
      rr
```



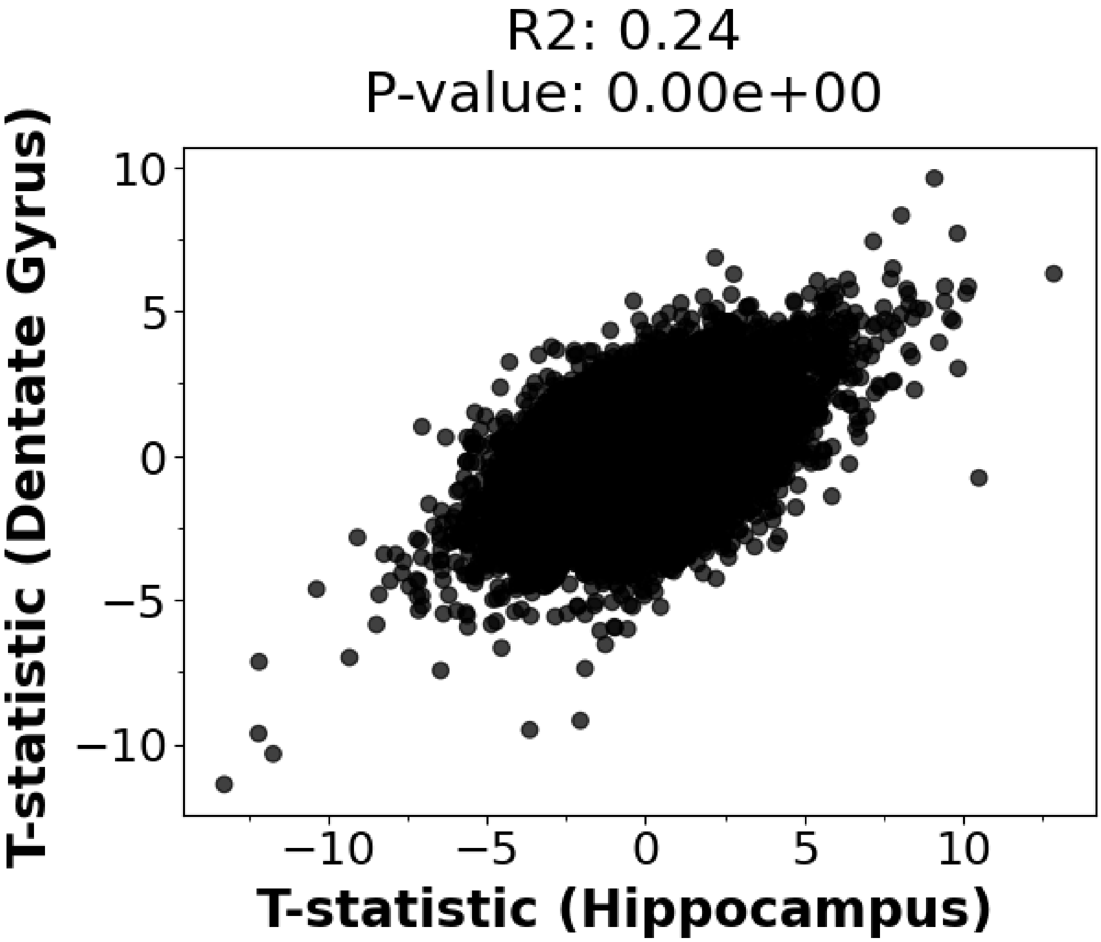
```
[23]: <ggplot: (8759208561704)>
```

```
[24]: ss = plot_corr('dlpfc', 'gyrus', merge_dataframes)
      ss
```



```
[24]: <ggplot: (8759208557698)>
```

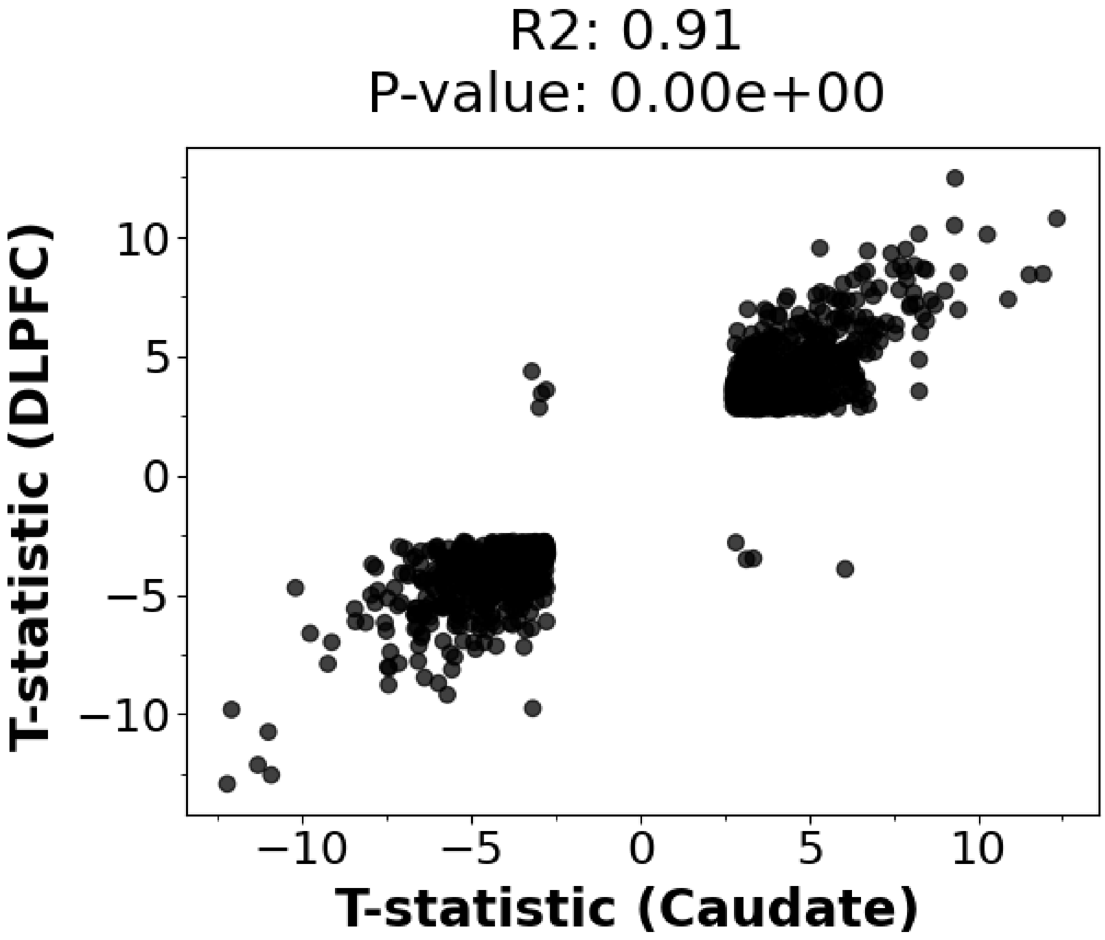
```
[25]: tt = plot_corr('hippo', 'gyrus', merge_dataframes)
      tt
```



```
[25]: <ggplot: (8759195672890)>
```

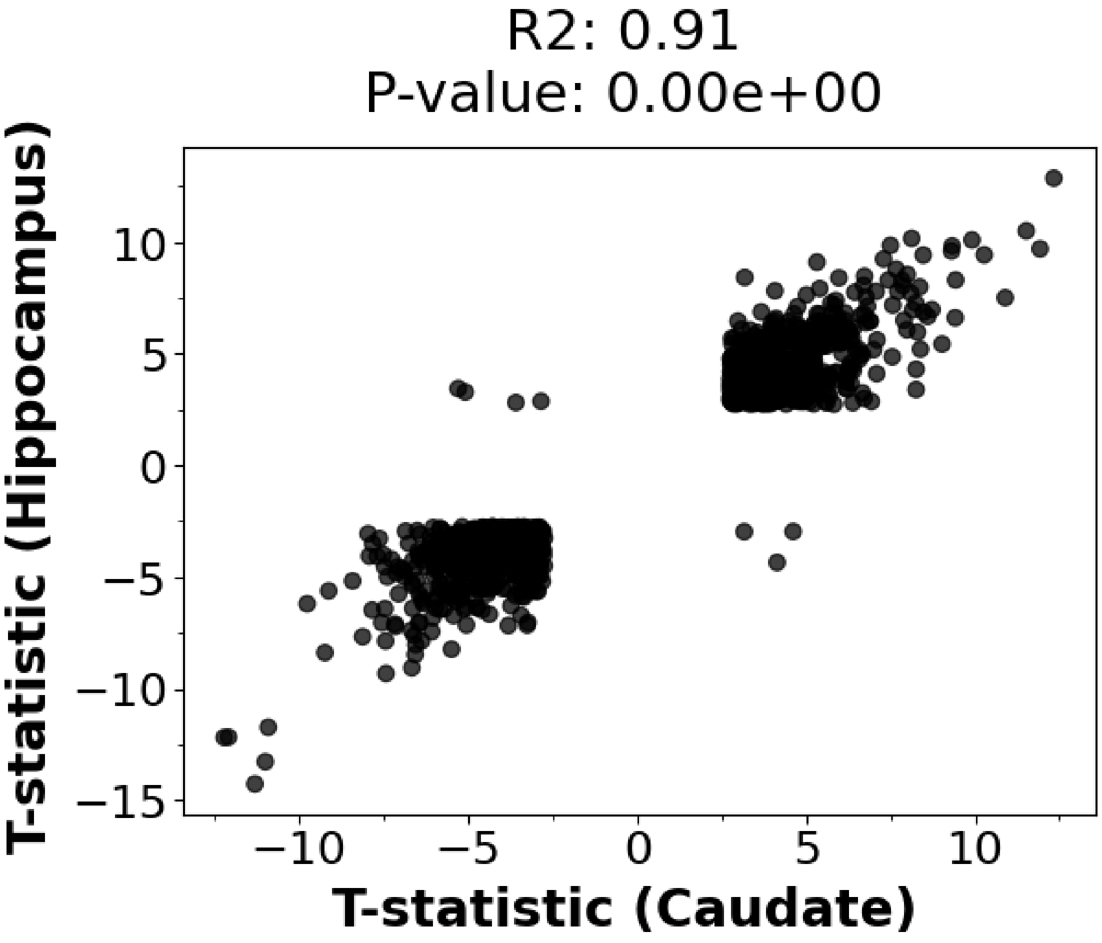
### 1.1.3 Significant correlation, FDR < 0.05

```
[26]: pp = plot_corr('caudate', 'dlpfc', merge_dataframes_sig)
      pp
```



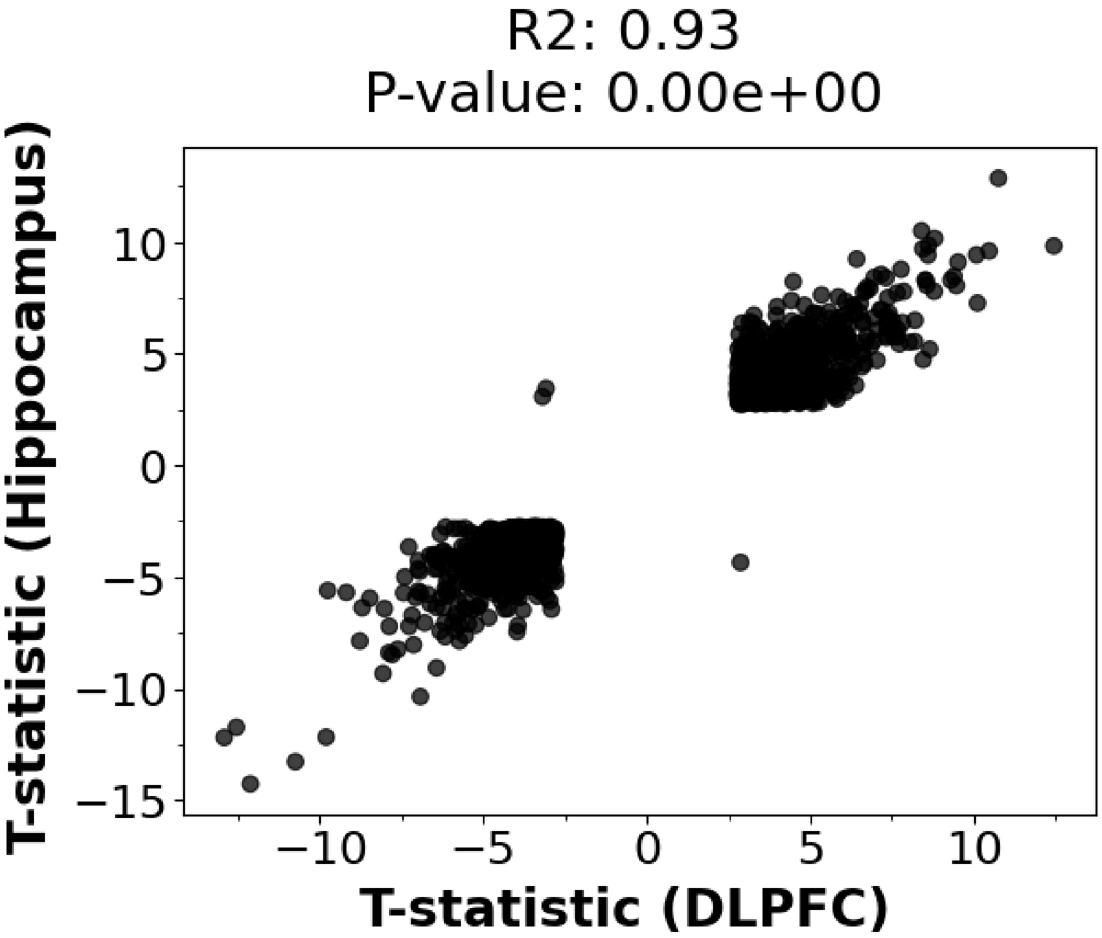
```
[26]: <ggplot: (8759192644148)>
```

```
[27]: qq = plot_corr('caudate', 'hippo', merge_dataframes_sig)
      qq
```



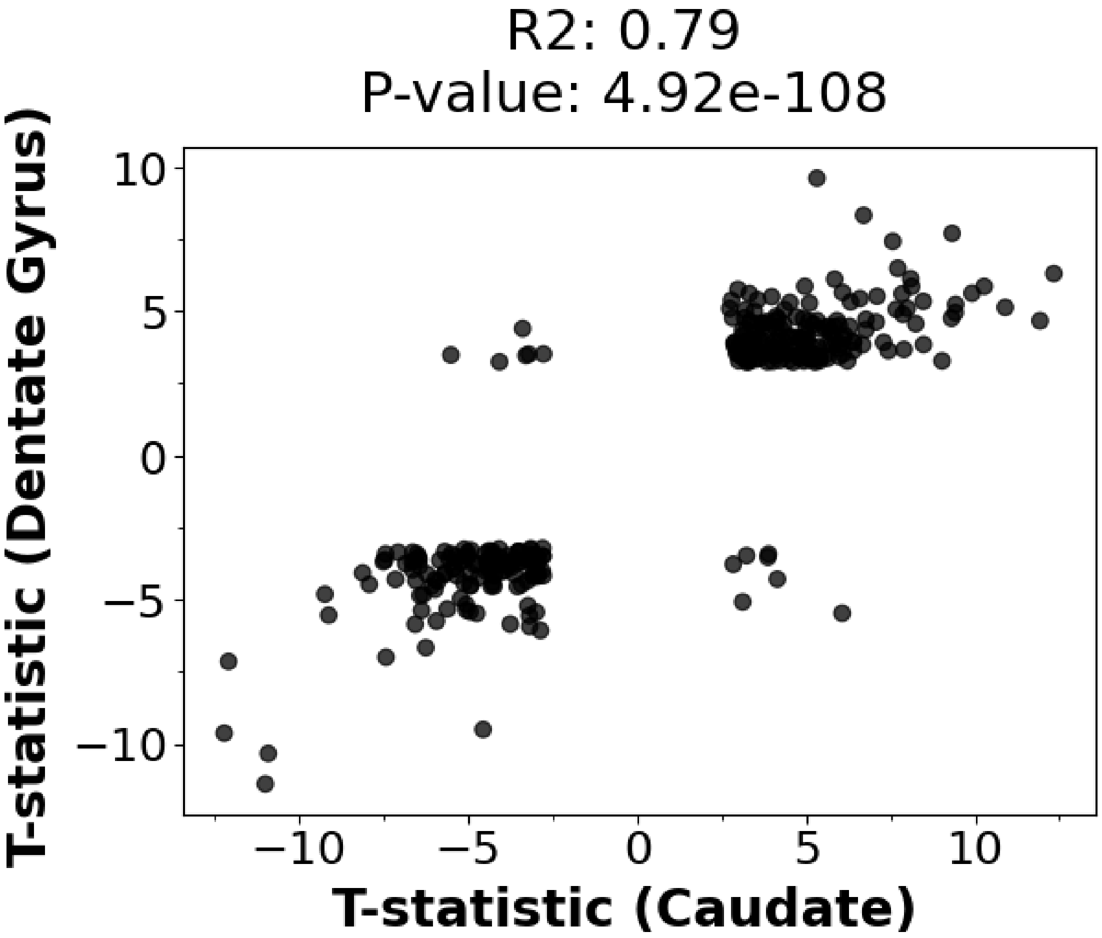
```
[27]: <ggplot: (8759208467351)>
```

```
[28]: ww = plot_corr('dlpfc', 'hippo', merge_dataframes_sig)
      ww
```



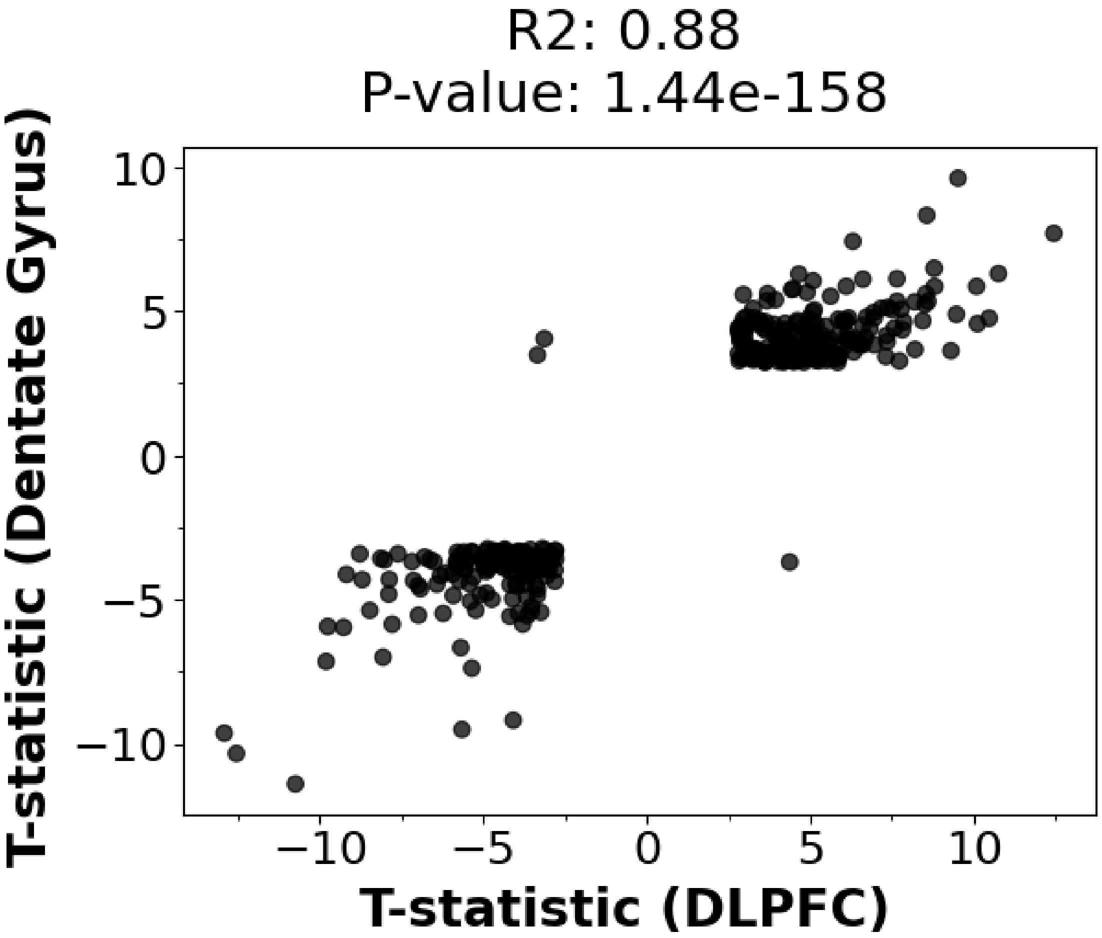
```
[28]: <ggplot: (8759192268457)>
```

```
[29]: rr = plot_corr('caudate', 'gyrus', merge_dataframes_sig)
      rr
```



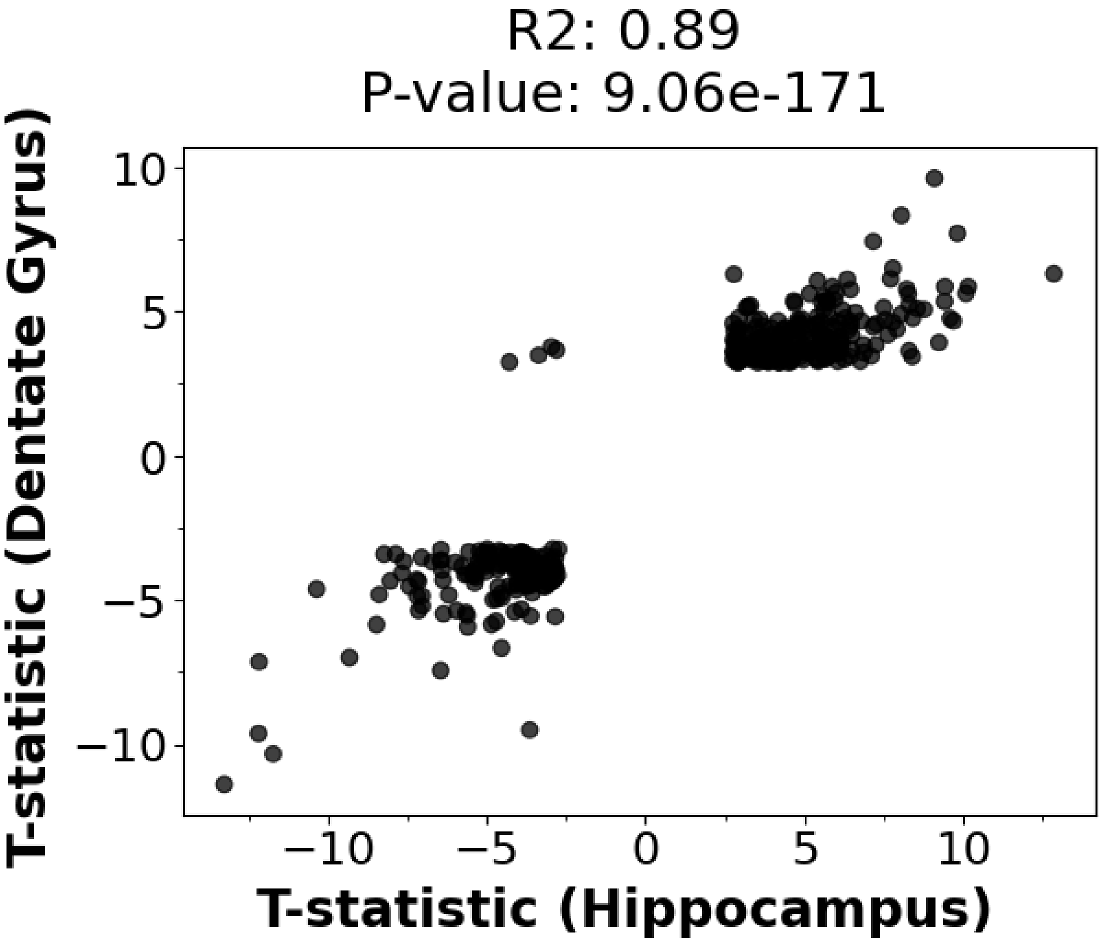
```
[29]: <ggplot: (8759192817833)>
```

```
[30]: ss = plot_corr('dlpfc', 'gyrus', merge_dataframes_sig)
      ss
```



```
[30]: <ggplot: (8759213670325)>
```

```
[31]: tt = plot_corr('hippo', 'gyrus', merge_dataframes_sig)
      tt
```



```
[31]: <ggplot: (8759192817803)>
```

```
[32]: #save_plot(pp, 'dlpfc_caudate_tstatistic_corr_sig')
      #save_plot(qq, 'hippo_caudate_tstatistic_corr_sig')
      #save_plot(ww, 'hippo_dlpfc_tstatistic_corr_sig')
```

### 1.1.4 Directionality test

**All genes**

```
[33]: enrichment_binom('caudate', 'dlpfc', merge_dataframes)
```

```
   agree      0
0   -1.0   7508
1    1.0  13620
```

```
[33]: 5e-324
```

```
[34]: enrichment_binom('caudate', 'hippo', merge_dataframes)
```

```
   agree      0
0   -1.0   7219
1    1.0  13978
```

```
[34]: 5e-324
```

```
[35]: enrichment_binom('dlpfc', 'hippo', merge_dataframes)
```

```
   agree      0
0   -1.0   7251
1    1.0  14347
```

```
[35]: 5e-324
```

```
[36]: enrichment_binom('caudate', 'gyrus', merge_dataframes)
```

```
   agree      0
0   -1.0   8063
1    1.0  11366
```

```
[36]: 1.1857793882825218e-124
```

```
[37]: enrichment_binom('dlpfc', 'gyrus', merge_dataframes)
```

```
   agree      0
0   -1.0   7509
1    1.0  12325
```

```
[37]: 9.716255782985859e-259
```

```
[38]: enrichment_binom('hippo', 'gyrus', merge_dataframes)
```

```
   agree      0
0   -1.0   7108
1    1.0  12736
```

```
[38]: 5e-324
```

### Significant DEG (FDR < 0.05)

```
[39]: enrichment_binom('caudate', 'dlpfc', merge_dataframes_sig)
```

```
   agree     0
0   -1.0     8
1    1.0  1107
```

```
[39]: 2.61503106e-316
```

```
[40]: enrichment_binom('caudate', 'hippo', merge_dataframes_sig)
```

```
   agree     0
0   -1.0     7
1    1.0  1135
```

```
[40]: 0.0
```

```
[41]: enrichment_binom('dlpfc', 'hippo', merge_dataframes_sig)
```

```
   agree     0
0   -1.0     3
1    1.0  1248
```

```
[41]: 0.0
```

```
[42]: enrichment_binom('caudate', 'gyrus', merge_dataframes_sig)
```

```
   agree    0
0   -1.0   13
1    1.0  298
```

```
[42]: 1.589014927874492e-71
```

```
[43]: enrichment_binom('dlpfc', 'gyrus', merge_dataframes_sig)
```

```
   agree    0
0   -1.0    3
1    1.0  339
```

```
[43]: 1.488391483735955e-96
```

```
[44]: enrichment_binom('hippo', 'gyrus', merge_dataframes_sig)
```

```
   agree    0
0   -1.0    4
1    1.0  357
```

```
[44]: 2.996700665181341e-100
```

```
[ ]:
```