# main

July 14, 2021

## 1 Visualize GO analysis

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

[2]:
```
def get_top_GO(tissue, fn, label):
    df = pd.read_csv(fn, sep='\t').sort_values('p_value').head(10)
    df['Log10'] = -np.log10(df['p_value'])
    df['Tissue'] = tissue
    df['Direction'] = label
    return df
```

[3]:
```
tissue = 'caudate'
config = {
    'All': '../../_m/DEGs_functional_enrichment.txt',
    'Up': '../../_m/upreg_DEGs_functional_enrichment.txt',
    'Down': '../../_m/downreg_DEGs_functional_enrichment.txt',
}

df = pd.DataFrame()
for bias in ['All', 'Up', 'Down']:
    df = pd.concat([df, get_top_GO(tissue, config[bias], bias)], axis=0)
```

[4]:
```
df.to_csv("%s_functional_analysis.txt" % tissue, sep='\t', index=False)
```

[5]:
```
df.shape
```

[5]:
```
(30, 17)
```

### 1.1 Plot

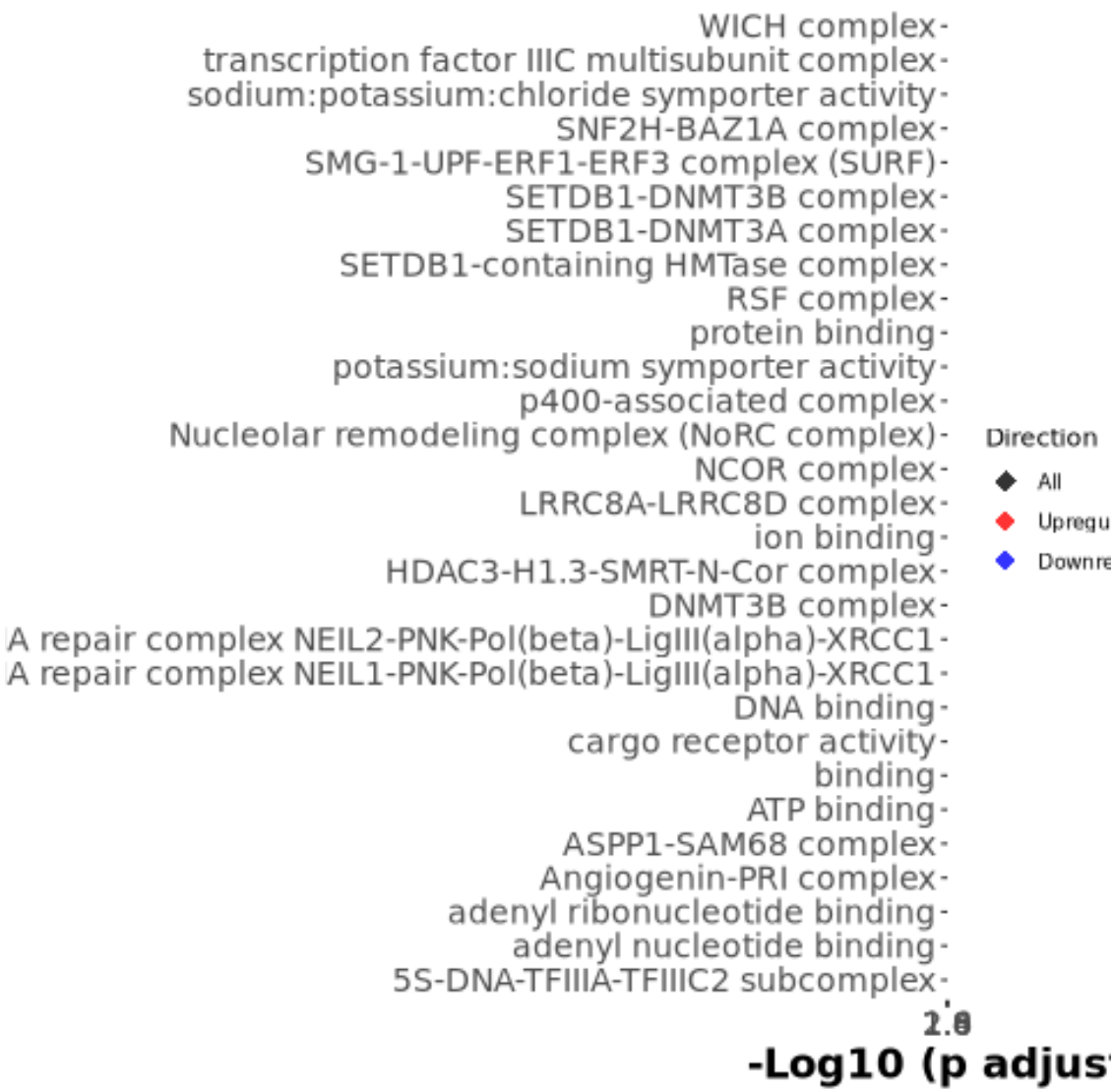
[6]:
```
%load_ext rpy2.ipython
```

[7]:
```
%%R -i df
library(ggplot2)
library(tidyverse)

save_plot <- function(p, fn, w, h){
```

```
    for(ext in c('.svg', '.png', '.pdf')){
        ggsave(file=paste0(fn,ext), plot=p, width=w, height=h)
    }
}

plot_GO <- function(){
    cbPalette <- c("#000000", "Red", "Blue")
    gg1 = df %>%
        ggplot(aes(x=Log10, y=term_name, color=Direction)) +
        geom_point(shape=18, alpha=0.8, size=4) + labs(y='', x='-Log10 (p␣
 ↪adjust)') +
        theme_bw() +
        scale_colour_manual(name="Direction", values=cbPalette,
                            labels=c("All", "Upregulated", "Downregulated")) +
        geom_vline(xintercept = -log10(0.05), linetype = "dotted") +
        theme(axis.text=element_text(size=14),
              axis.title=element_text(size=18, face='bold'),
              strip.text=element_text(size=18, face='bold'))
    return(gg1)
}
```

```
R[write to console]:   Attaching packages
                     tidyverse 1.3.1

R[write to console]:  tibble  3.1.2      dplyr   1.0.7
 tidyr   1.1.3      stringr 1.4.0
 readr   1.4.0      forcats 0.5.1
 purrr   0.3.4

R[write to console]:   Conflicts
tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()
```

[8]:
```
%%R
gg1 = plot_GO()
print(gg1)
save_plot(gg1, "caudate_GO_top10_stacked", 10, 6)
```


WICH complex
transcription factor IIIC multisubunit complex
sodium:potassium:chloride symporter activity
SNF2H-BAZ1A complex
SMG-1-UPF-ERF1-ERF3 complex (SURF)
SETDB1-DNMT3B complex
SETDB1-DNMT3A complex
SETDB1-containing HMTase complex
RSF complex
protein binding
potassium:sodium symporter activity
p400-associated complex
Nucleolar remodeling complex (NoRC complex)
NCOR complex
LRRC8A-LRRC8D complex
ion binding
HDAC3-H1.3-SMRT-N-Cor complex
DNMT3B complex
A repair complex NEIL2-PNK-Pol(beta)-LigIII(alpha)-XRCC1
A repair complex NEIL1-PNK-Pol(beta)-LigIII(alpha)-XRCC1
DNA binding
cargo receptor activity
binding
ATP binding
ASPP1-SAM68 complex
Angiogenin-PRI complex
adenyl ribonucleotide binding
adenyl nucleotide binding
5S-DNA-TFIIIA-TFIIIC2 subcomplex
1.0
-Log10 (p adjus
Direction
All
Upregu
Downre

[ ]: