

main

July 10, 2021

1 Analyze the effect of antipsychotics on gene expression

```
[1]: library(tidyverse)
library(ggpubr)
```

```
Attaching packages: tidyverse
1.3.1
```

```
ggplot2 3.3.5    purrr  0.3.4
tibble  3.1.2    dplyr  1.0.7
tidyr   1.1.3    stringr 1.4.0
readr   1.4.0    forcats 0.5.1
```

```
Conflicts
tidyverse_conflicts()
dplyr::filter() masks stats::filter()
dplyr::lag()    masks stats::lag()
```

1.1 Functions

```
[2]: save_img <- function(image, fn, w, h){
  for(ext in c(".svg", ".pdf", ".png")){
    ggsave(file=paste0(fn, ext), plot=image, width=w, height=h)
  }
}

get_pheno <- function(){
  phenotypes <- data.table::fread(paste0("/ceph/projects/v4_phase3_paper/
  ↪inputs/",
                                         "phenotypes/_m/merged_phenotypes.
  ↪csv")) %>%
  mutate_if(is.character, as.factor)
  return(phenotypes)
}

memPHENO <- memoise::memoise(get_pheno)
```

```
[3]: pca_select <- function(norm_df){
  ### Dimensional reduction (PCA)
  log2cpm = norm_df %>% column_to_rownames("V1") %>% t
  pca_df = prcomp(log2cpm, center=TRUE)$x
  dt = pca_df %>% as.data.frame %>% rownames_to_column() %>%
    pivot_longer(-rowname, names_to="PC", values_to="PC_values")
  return(dt)
}

memPCA <- memoise::memoise(pca_select)
```

1.2 Normalized expression

```
[4]: memPHENO() %>% filter(Dx %in% c("CTL", "SZ"), Age > 13,
  Region == "Caudate") %>%
  group_by(Dx, antipsychotics) %>% summarize(N=n())
```

`summarise()` has grouped output by 'Dx'. You can override using the `.groups` argument.

	Dx <fct>	antipsychotics <lgl>	N <int>
A grouped_df: 4 × 3	CTL	FALSE	246
	SZ	FALSE	49
	SZ	TRUE	104
	SZ	NA	1

```
[5]: norm_df <- data.table::fread("../..//main_plots/_m/caudate/normalized_expression.
  ↪tsv")
norm_df[1:2, 1:5]
```

	V1 <chr>	R12864 <dbl>	R12865 <dbl>	R12866 <dbl>	R12867 <dbl>
A data.table: 2 × 5	ENSG00000227232.5	1.547013	1.538154	1.6061889	1.565640
	ENSG00000278267.1	-1.405526	-1.866254	-0.7728476	-1.954495

```
[6]: dt = memPCA(norm_df) %>% mutate_if(is.character, as.factor)
dt %>% dim
dt %>% head(2)
```

1. 154449 2. 3

	rowname <fct>	PC <fct>	PC_values <dbl>
A tibble: 2 × 3	R12864	PC1	-35.34019
	R12864	PC2	25.33034

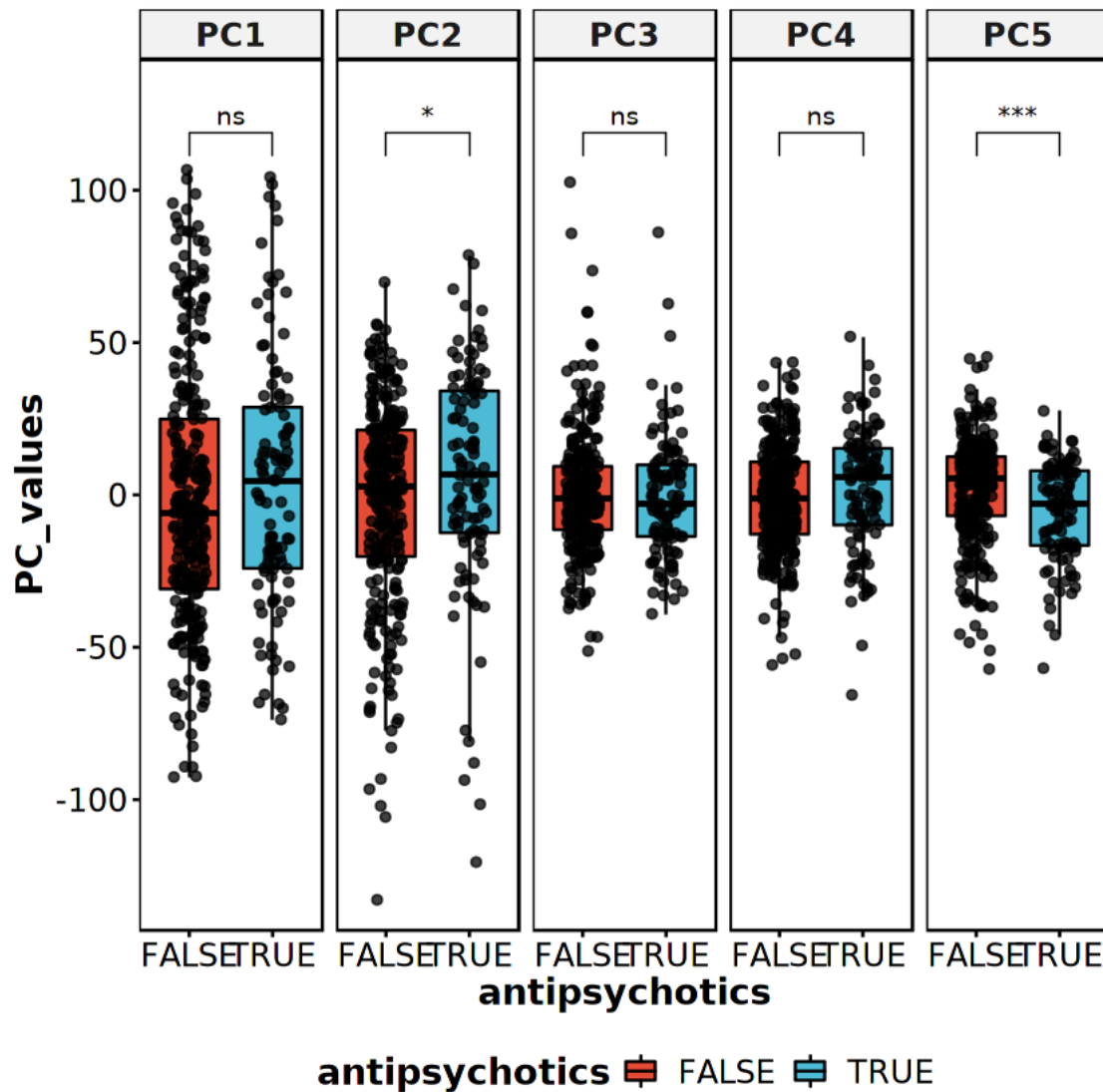
```
[7]: bxp = dt %>% filter(PC %in% c("PC1", "PC2", "PC3", "PC4", "PC5")) %>%
      inner_join(memPHENO(), by=c("rowname"="RNum")) %>% drop_na() %>%
      ggboxplot(x='antipsychotics', y="PC_values", fill='antipsychotics',
      ↪add='jitter',
      palette="npg", facet.by="PC", legend='bottom', add.
      ↪params=list(alpha=0.75),
      panel.labs.font=list(face='bold', size=14), ncol=5, ylim=c(-130,
      ↪130)) +
      stat_compare_means(comparisons=list(c("TRUE", "FALSE")), aes(label=..p.
      ↪signif..),
      method="wilcox.test") +
      font("title", size=18, color="black", face="bold") +
      font("xy.title", size=16, face="bold") + font("xy.text", size=14) +
      font("legend.text", size=14) + font("legend.title", size=16,
      ↪face="bold")
      save_img(bxp, paste0("boxplot_antipsychotics_normalized"), w=12, h=4)
      print(bxp)
```

Warning message:

"Using `as.character()` on a quosure is deprecated as of rlang 0.3.0.

Please use `as_label()` or `as_name()` instead.

This warning is displayed once per session."

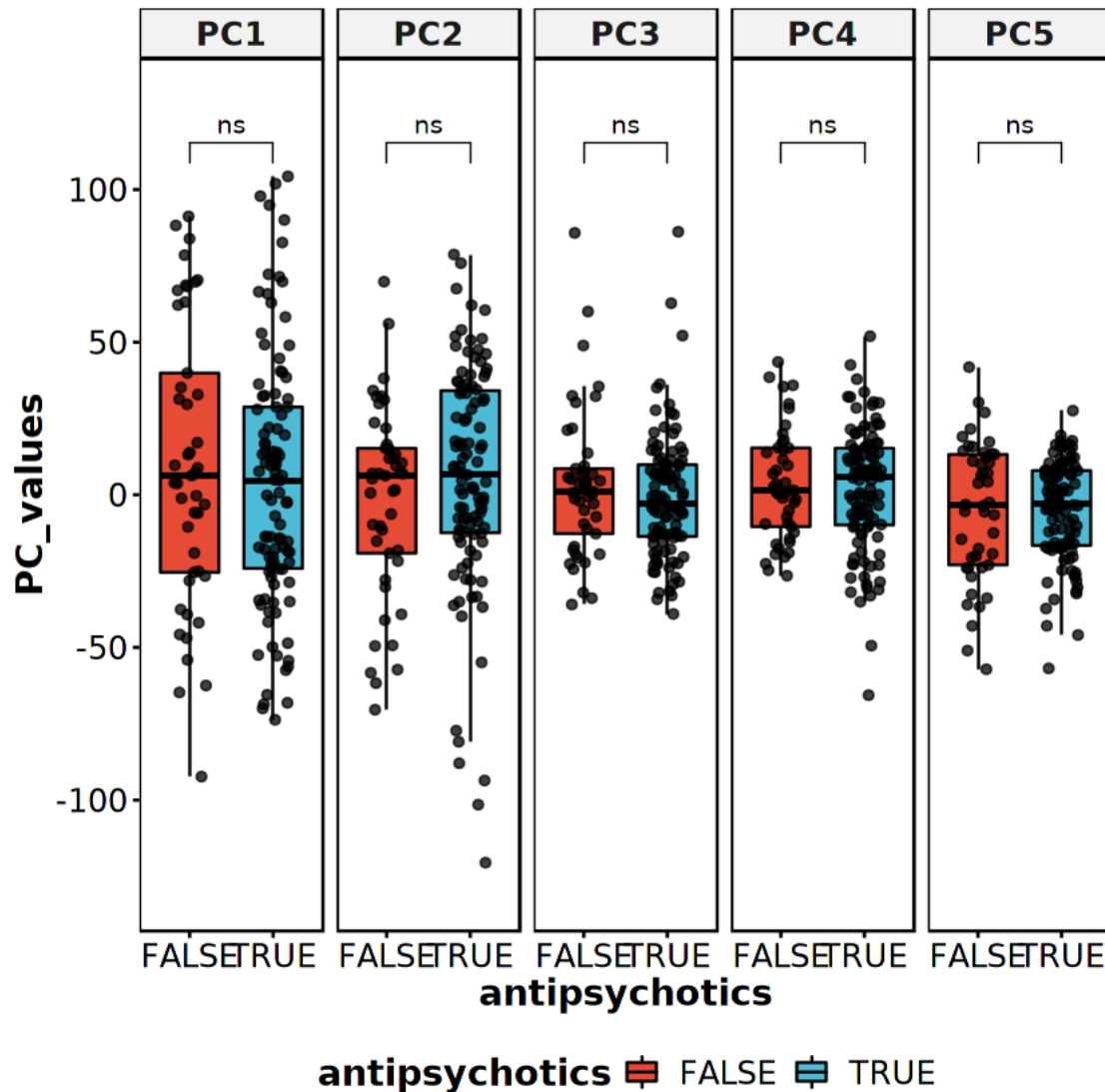


```
[8]: bxp = dt %>% filter(PC %in% c("PC1", "PC2", "PC3", "PC4", "PC5")) %>%
  inner_join(memPHENO(), by=c("rowname"="RNum")) %>% drop_na() %>%
  filter(Dx == "SZ") %>%
  ggboxplot(x='antipsychotics', y="PC_values", fill='antipsychotics',
  ↪add='jitter',
  palette="npg", facet.by="PC", legend='bottom', add.
  ↪params=list(alpha=0.75),
  panel.labs.font=list(face='bold', size=14), ncol=5, ylim=c(-130,
  ↪130)) +
  stat_compare_means(comparisons=list(c("TRUE", "FALSE")), aes(label=..p.
  ↪signif..),
  method="wilcox.test") +
```

```

font("title", size=18, color="black", face="bold") +
font("xy.title", size=16, face="bold") + font("xy.text", size=14) +
font("legend.text", size=14) + font("legend.title", size=16,
  ↳face="bold")
save_img(bxp, paste0("boxplot_antipsychotics_normalized_szOnly"), w=12, h=4)
print(bxp)

```



```

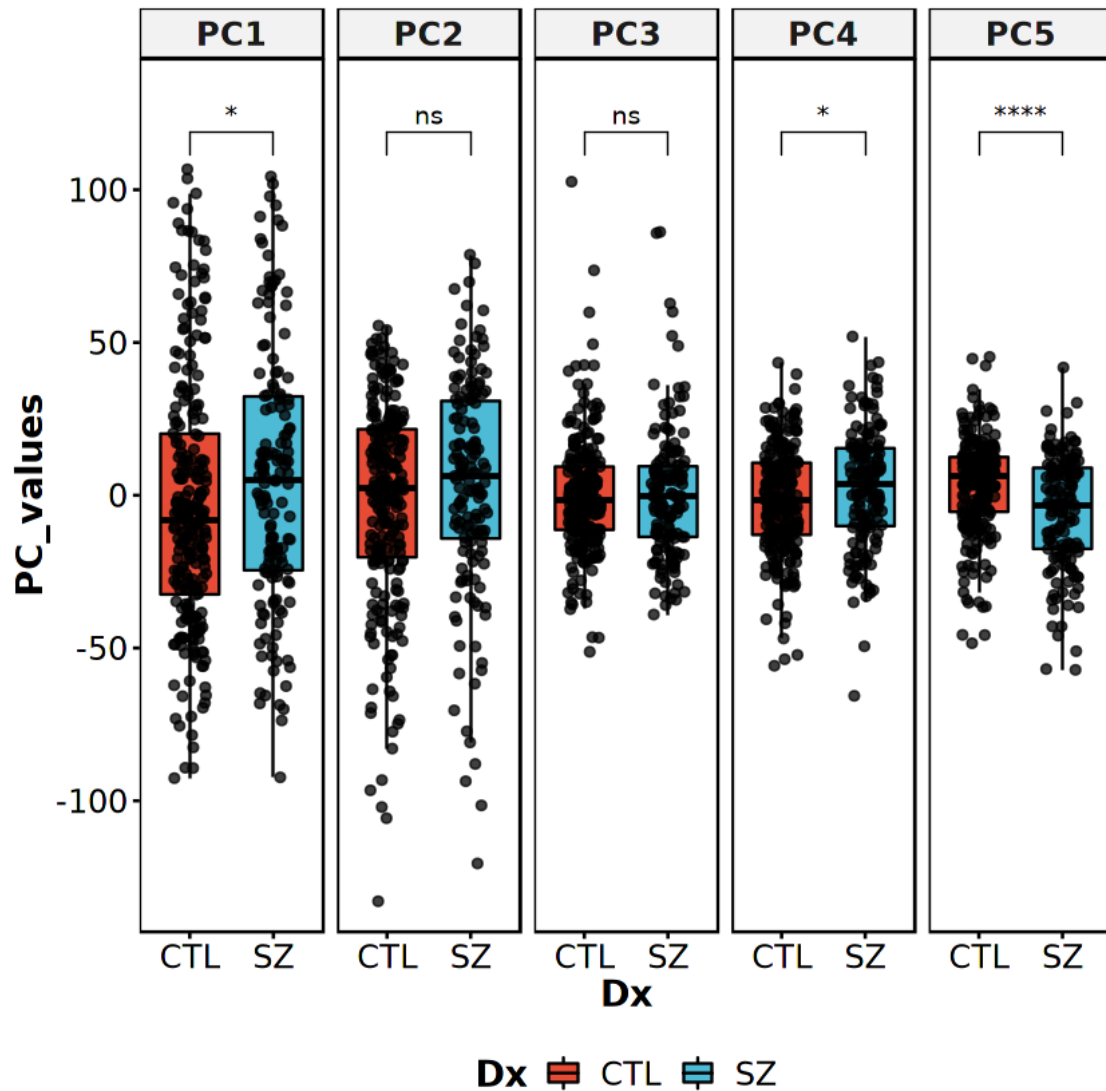
[9]: bxp = dt %>% filter(PC %in% c("PC1", "PC2", "PC3", "PC4", "PC5")) %>%
inner_join(memPHENO(), by=c("rowname"="RNum")) %>% drop_na() %>%
ggboxplot(x='Dx', y="PC_values", fill='Dx', add='jitter',
  palette="npg", facet.by="PC", legend='bottom', add.
  ↳params=list(alpha=0.75),

```

```

panel.labs.font=list(face='bold', size=14), ncol=5, ylim=c(-130,
↪130)) +
  stat_compare_means(comparisons=list(c("CTL", "SZ")), aes(label=..p.signif..
↪),
                    method="wilcox.test") +
  font("title", size=18, color="black", face="bold") +
  font("xy.title", size=16, face="bold") + font("xy.text", size=14) +
  font("legend.text", size=14) + font("legend.title", size=16,
↪face="bold")
save_img(bxp, paste0("boxplot_dx_normalized"), w=12, h=4)
print(bxp)

```



1.2.1 Statistical test for all significant PCs for antipsychotics

```
[10]: antipsychotics = dt %>% inner_join(memPHENO(), by=c("rowname"="RNum")) %>%
  drop_na()
pcs = c()
pvals = c()
for(pc in unique(antipsychotics$PC)){
  flush.console()
  dx = antipsychotics %>% filter(PC == pc)
  res = wilcox.test(filter(dx, antipsychotics=="TRUE")$PC_values,
                    filter(dx, antipsychotics=="FALSE")$PC_values)
  if(res$p.value < 0.05){
    pcs = c(pcs, pc)
    pvals = c(pvals, res$p.value)
  }
}
```

```
[11]: pcs
```

1. 'PC2' 2. 'PC5' 3. 'PC11' 4. 'PC13' 5. 'PC15' 6. 'PC16' 7. 'PC17' 8. 'PC19' 9. 'PC21' 10. 'PC22'
11. 'PC23' 12. 'PC24' 13. 'PC28' 14. 'PC32' 15. 'PC38' 16. 'PC40' 17. 'PC46' 18. 'PC60' 19. 'PC81'
20. 'PC386'

1.3 Residualized

```
[12]: res_df <- data.table::fread(".././main_plots/_m/caudate/
  residualized_expression.tsv")
res_df[1:2, 1:5]
```

	V1	R12864	R12865	R12866	R12867
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
A data.table: 2 × 5	ENSG00000227232.5	-0.8066279	-0.90143820	-0.4152018	-0.8407762
	ENSG00000278267.1	0.9416640	-0.06697099	1.3918914	0.5206415

```
[13]: dt = memPCA(res_df) %>% mutate_if(is.character, as.factor)
dt %>% dim
dt %>% head(2)
```

1. 154449 2. 3

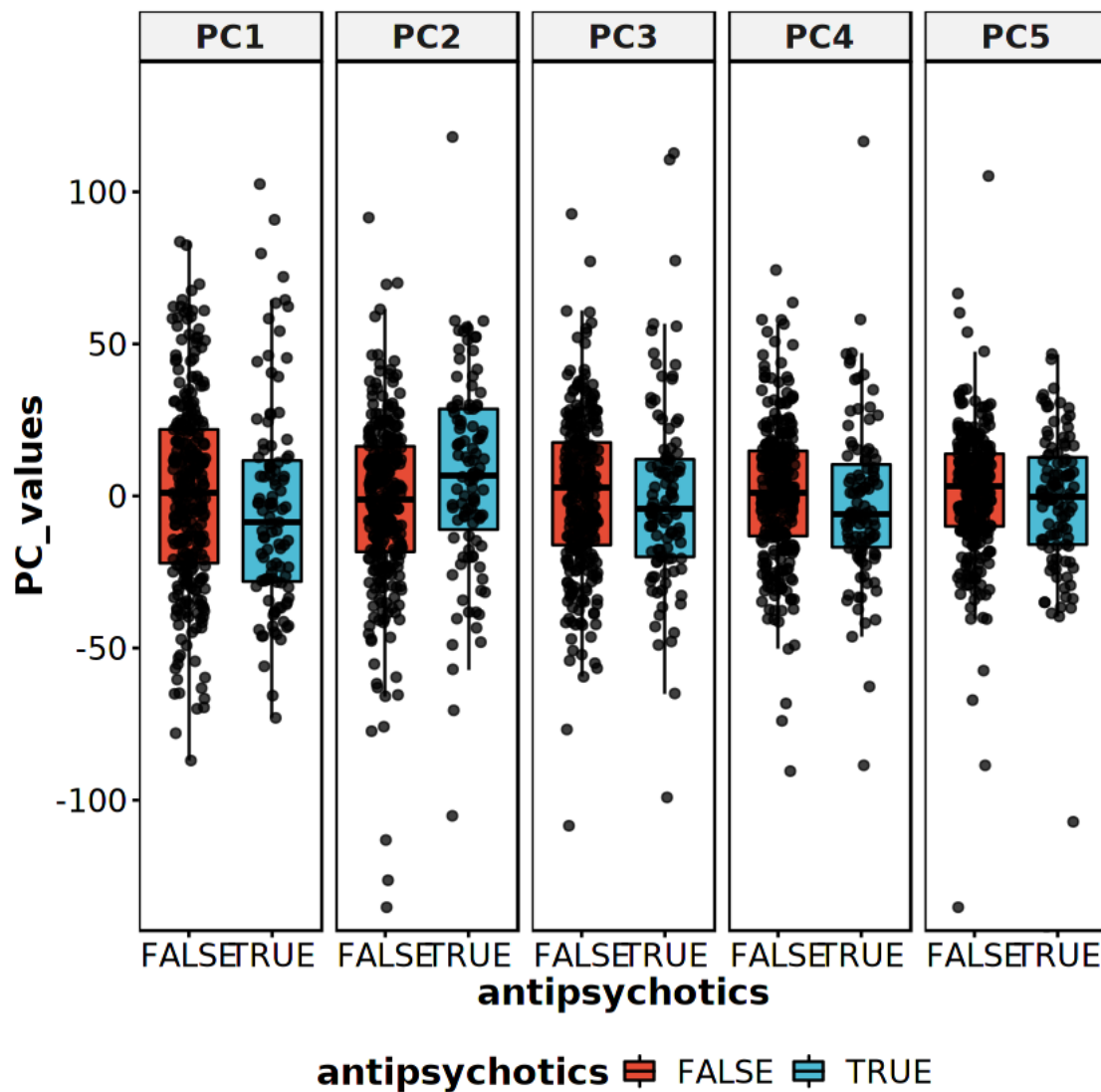
	rowname	PC	PC_values
	<fct>	<fct>	<dbl>
A tibble: 2 × 3	R12864	PC1	-0.1760693
	R12864	PC2	57.6305446

```
[14]: bxp = dt %>% filter(PC %in% c("PC1", "PC2", "PC3", "PC4", "PC5")) %>%
  inner_join(memPHENO(), by=c("rowname"="RNum")) %>% drop_na() %>%
  ggboxplot(x='antipsychotics', y="PC_values", fill='antipsychotics',
  add='jitter',
```

```

palette="npg", facet.by="PC", legend='bottom', add.
→params=list(alpha=0.75),
panel.labs.font=list(face='bold', size=14), ncol=5, ylim=c(-130,
→130)) +
stat_compare_means(comparisons=list(c("TRUE", "FALSE")), aes(label=..p.
→signif..),
method="wilcox.test") +
font("title", size=18, color="black", face="bold") +
font("xy.title", size=16, face="bold") + font("xy.text", size=14) +
font("legend.text", size=14) + font("legend.title", size=16,
→face="bold")
save_img(bxp, paste0("boxplot_antipsychotics_residualized"), w=12, h=4)
print(bxp)

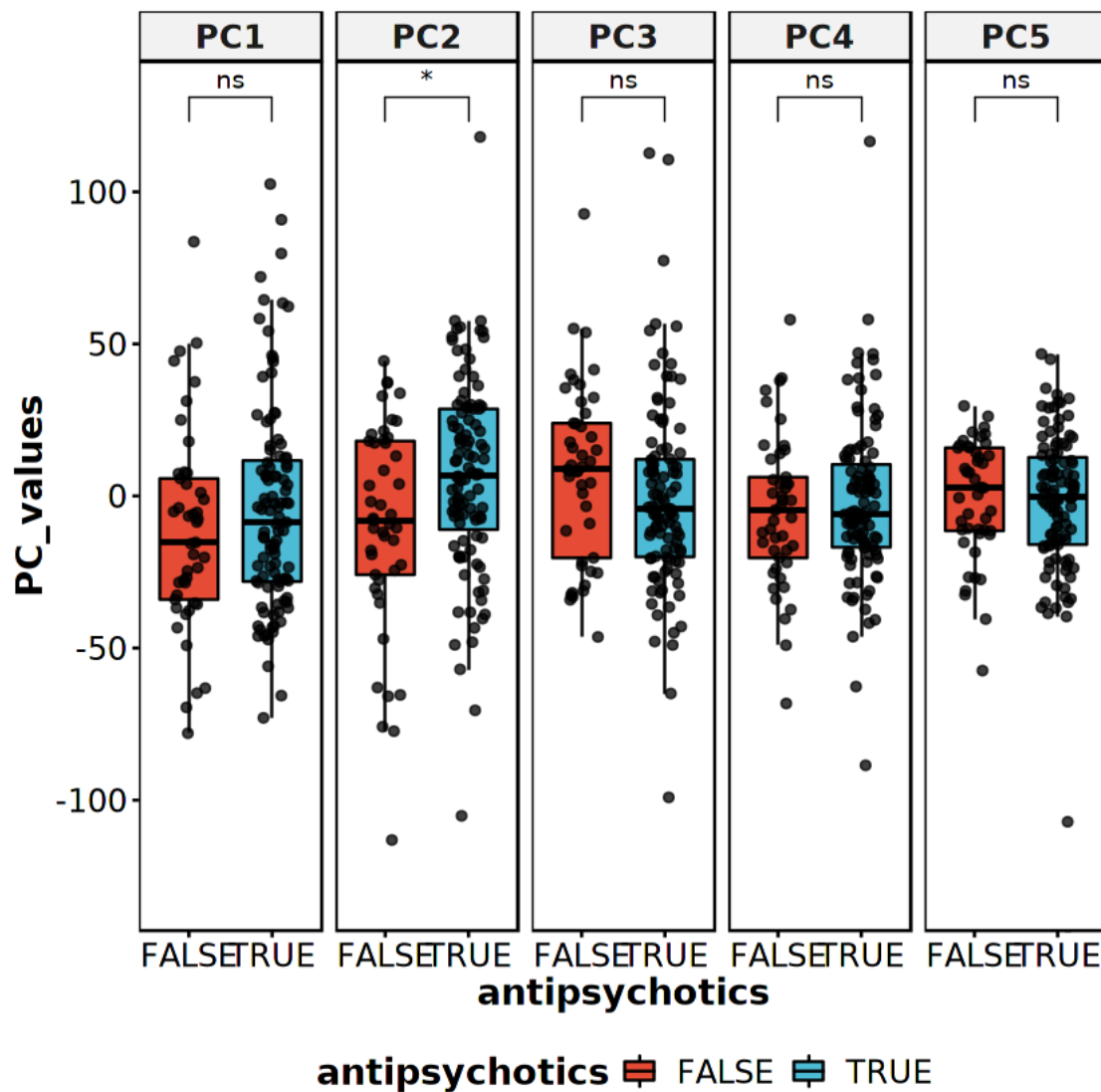
```




```

[15]: bxp = dt %>% filter(PC %in% c("PC1", "PC2", "PC3", "PC4", "PC5")) %>%
      inner_join(memPHENO(), by=c("rowname"="RNum")) %>% drop_na() %>%
      filter(Dx == "SZ") %>%
      ggboxplot(x='antipsychotics', y="PC_values", fill='antipsychotics',
      ↪add='jitter',
      palette="npg", facet.by="PC", legend='bottom', add.
      ↪params=list(alpha=0.75),
      panel.labs.font=list(face='bold', size=14), ncol=5, ylim=c(-130,
      ↪130)) +
      stat_compare_means(comparisons=list(c("TRUE", "FALSE")), aes(label=..p.
      ↪signif..),
      method="wilcox.test") +
      font("title", size=18, color="black", face="bold") +
      font("xy.title", size=16, face="bold") + font("xy.text", size=14) +
      font("legend.text", size=14) + font("legend.title", size=16,
      ↪face="bold")
save_img(bxp, paste0("boxplot_antipsychotics_residualized_szOnly"), w=12, h=4)
print(bxp)

```

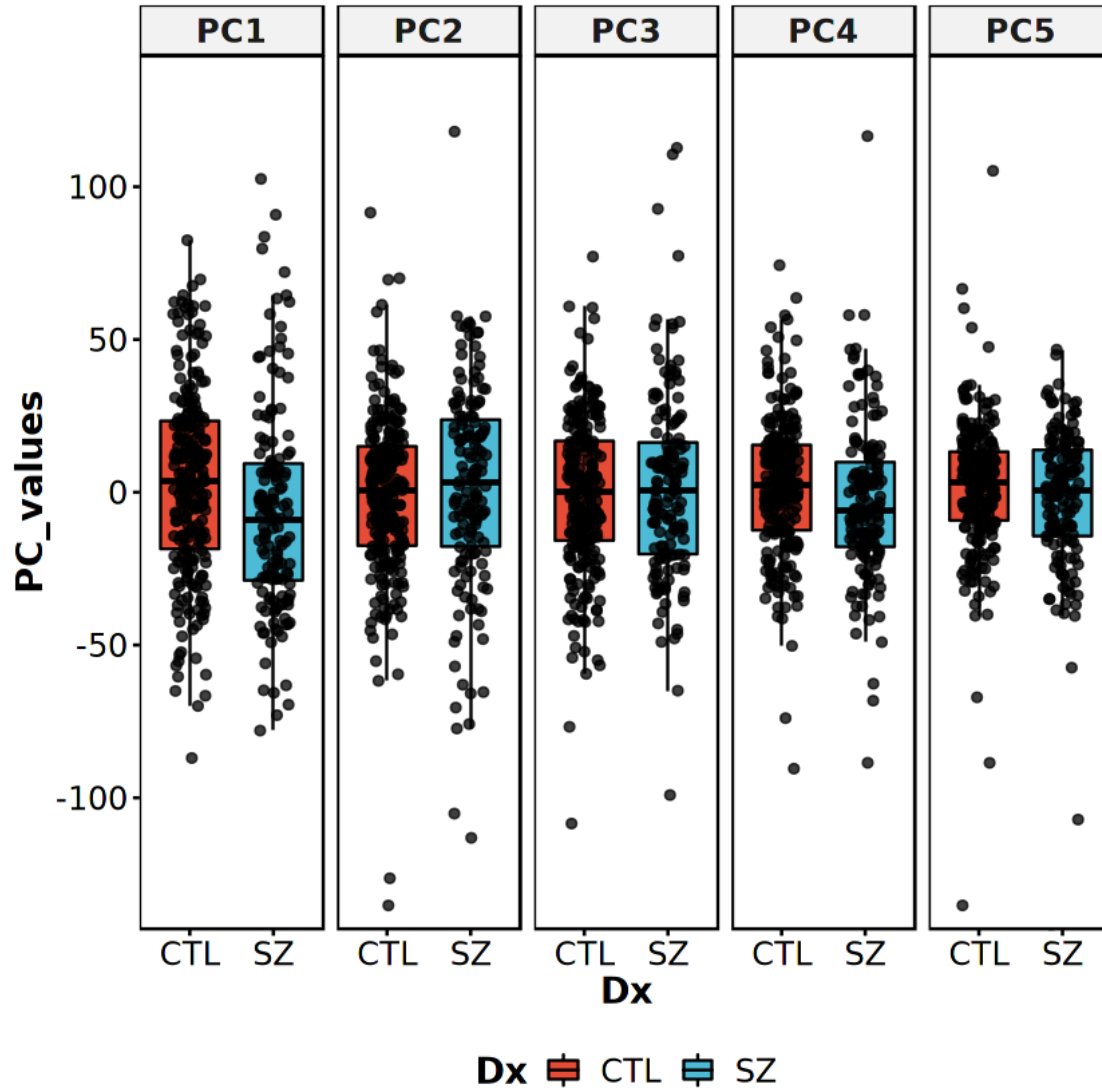


```
[16]: bxp = dt %>% filter(PC %in% c("PC1", "PC2", "PC3", "PC4", "PC5")) %>%
  inner_join(memPHENO(), by=c("rowname"="RNum")) %>% drop_na() %>%
  ggboxplot(x='Dx', y="PC_values", fill='Dx', add='jitter',
    palette="npg", facet.by="PC", legend='bottom', add.
  ↪params=list(alpha=0.75),
    panel.labs.font=list(face='bold', size=14), ncol=5, ylim=c(-130,
  ↪130)) +
  stat_compare_means(comparisons=list(c("CTL", "SZ")), aes(label=..p.signif..
  ↪),
    method="wilcox.test") +
  font("title", size=18, color="black", face="bold") +
  font("xy.title", size=16, face="bold") + font("xy.text", size=14) +
```

```

font("legend.text", size=14) + font("legend.title", size=16,
↪face="bold")
save_img(bxp, paste0("boxplot_dx_residualized"), w=12, h=4)
print(bxp)

```



1.3.1 Statistical test for all significant PCs for antipsychotics

```

[17]: antipsychotics = dt %>% inner_join(memPHENO(), by=c("rowname"="RNum")) %>%
↪drop_na()
pcs2 = c()
pvals2 = c()
for(pc in unique(antipsychotics$PC)){

```

```

flush.console()
dx = antipsychotics %>% filter(PC == pc)
res = wilcox.test(filter(dx, antipsychotics=="TRUE")$PC_values,
                  filter(dx, antipsychotics=="FALSE")$PC_values)
if(res$p.value < 0.05){
  pcs2 = c(pcs2, pc)
  pvals2 = c(pvals2, res$p.value)
}
}

pcs2

```

1. 'PC1' 2. 'PC2' 3. 'PC6' 4. 'PC7' 5. 'PC9' 6. 'PC13' 7. 'PC17' 8. 'PC18' 9. 'PC19' 10. 'PC24'
 11. 'PC30' 12. 'PC33' 13. 'PC297' 14. 'PC371' 15. 'PC375' 16. 'PC379' 17. 'PC382' 18. 'PC393'

1.4 Reproducibility Information

```

[18]: Sys.time()
proc.time()
options(width = 120)
sessioninfo::session_info()

```

```
[1] "2021-07-10 12:05:52 EDT"
```

```

      user system elapsed
567.832 357.648  88.237

```

```

Session info
setting  value
version  R version 4.0.3 (2020-10-10)
os       Arch Linux
system   x86_64, linux-gnu
ui       X11
language (EN)
collate  en_US.UTF-8
ctype    en_US.UTF-8
tz       America/New_York
date     2021-07-10

```

```

Packages
package      * version  date      lib source
abind         1.4-5    2016-07-21 [1] CRAN (R 4.0.2)
assertthat    0.2.1    2019-03-21 [1] CRAN (R 4.0.2)
backports     1.2.1    2020-12-09 [1] CRAN (R 4.0.2)
base64enc     0.1-3    2015-07-28 [1] CRAN (R 4.0.2)
broom         0.7.8    2021-06-24 [1] CRAN (R 4.0.3)
cachem        1.0.5    2021-05-15 [1] CRAN (R 4.0.3)
Cairo         1.5-12.2 2020-07-07 [1] CRAN (R 4.0.2)
car           3.0-11   2021-06-27 [1] CRAN (R 4.0.3)

```

carData	3.0-4	2020-05-22	[1]	CRAN	(R 4.0.2)
cellranger	1.1.0	2016-07-27	[1]	CRAN	(R 4.0.2)
cli	3.0.0	2021-06-30	[1]	CRAN	(R 4.0.3)
colorspace	2.0-2	2021-06-24	[1]	CRAN	(R 4.0.3)
crayon	1.4.1	2021-02-08	[1]	CRAN	(R 4.0.3)
curl	4.3.2	2021-06-23	[1]	CRAN	(R 4.0.3)
data.table	1.14.0	2021-02-21	[1]	CRAN	(R 4.0.3)
DBI	1.1.1	2021-01-15	[1]	CRAN	(R 4.0.2)
dbplyr	2.1.1	2021-04-06	[1]	CRAN	(R 4.0.3)
digest	0.6.27	2020-10-24	[1]	CRAN	(R 4.0.2)
dplyr	* 1.0.7	2021-06-18	[1]	CRAN	(R 4.0.3)
ellipsis	0.3.2	2021-04-29	[1]	CRAN	(R 4.0.3)
evaluate	0.14	2019-05-28	[1]	CRAN	(R 4.0.2)
fansi	0.5.0	2021-05-25	[1]	CRAN	(R 4.0.3)
farver	2.1.0	2021-02-28	[1]	CRAN	(R 4.0.3)
fastmap	1.1.0	2021-01-25	[1]	CRAN	(R 4.0.2)
forcats	* 0.5.1	2021-01-27	[1]	CRAN	(R 4.0.2)
foreign	0.8-80	2020-05-24	[2]	CRAN	(R 4.0.3)
fs	1.5.0	2020-07-31	[1]	CRAN	(R 4.0.2)
generics	0.1.0	2020-10-31	[1]	CRAN	(R 4.0.2)
ggplot2	* 3.3.5	2021-06-25	[1]	CRAN	(R 4.0.3)
ggpubr	* 0.4.0	2020-06-27	[1]	CRAN	(R 4.0.2)
ggsci	2.9	2018-05-14	[1]	CRAN	(R 4.0.2)
ggsignif	0.6.2	2021-06-14	[1]	CRAN	(R 4.0.3)
glue	1.4.2	2020-08-27	[1]	CRAN	(R 4.0.2)
gtable	0.3.0	2019-03-25	[1]	CRAN	(R 4.0.2)
haven	2.4.1	2021-04-23	[1]	CRAN	(R 4.0.3)
hms	1.1.0	2021-05-17	[1]	CRAN	(R 4.0.3)
htmltools	0.5.1.1	2021-01-22	[1]	CRAN	(R 4.0.2)
httr	1.4.2	2020-07-20	[1]	CRAN	(R 4.0.2)
IRdisplay	1.0	2021-01-20	[1]	CRAN	(R 4.0.2)
IRkernel	1.2	2021-05-11	[1]	CRAN	(R 4.0.3)
jsonlite	1.7.2	2020-12-09	[1]	CRAN	(R 4.0.2)
labeling	0.4.2	2020-10-20	[1]	CRAN	(R 4.0.2)
lifecycle	1.0.0	2021-02-15	[1]	CRAN	(R 4.0.3)
lubridate	1.7.10	2021-02-26	[1]	CRAN	(R 4.0.3)
magrittr	2.0.1	2020-11-17	[1]	CRAN	(R 4.0.2)
memoise	2.0.0	2021-01-26	[1]	CRAN	(R 4.0.2)
modelr	0.1.8	2020-05-19	[1]	CRAN	(R 4.0.2)
munsell	0.5.0	2018-06-12	[1]	CRAN	(R 4.0.2)
openxlsx	4.2.4	2021-06-16	[1]	CRAN	(R 4.0.3)
pbdZMQ	0.3-5	2021-02-10	[1]	CRAN	(R 4.0.3)
pillar	1.6.1	2021-05-16	[1]	CRAN	(R 4.0.3)
pkgconfig	2.0.3	2019-09-22	[1]	CRAN	(R 4.0.2)
purrr	* 0.3.4	2020-04-17	[1]	CRAN	(R 4.0.2)
R6	2.5.0	2020-10-28	[1]	CRAN	(R 4.0.2)
Rcpp	1.0.7	2021-07-07	[1]	CRAN	(R 4.0.3)
readr	* 1.4.0	2020-10-05	[1]	CRAN	(R 4.0.2)

readxl	1.3.1	2019-03-13	[1]	CRAN	(R 4.0.2)
repr	1.1.3	2021-01-21	[1]	CRAN	(R 4.0.2)
reprex	2.0.0	2021-04-02	[1]	CRAN	(R 4.0.3)
rio	0.5.27	2021-06-21	[1]	CRAN	(R 4.0.3)
rlang	0.4.11	2021-04-30	[1]	CRAN	(R 4.0.3)
rstatix	0.7.0	2021-02-13	[1]	CRAN	(R 4.0.3)
rstudioapi	0.13	2020-11-12	[1]	CRAN	(R 4.0.2)
rvest	1.0.0	2021-03-09	[1]	CRAN	(R 4.0.3)
scales	1.1.1	2020-05-11	[1]	CRAN	(R 4.0.2)
sessioninfo	1.1.1	2018-11-05	[1]	CRAN	(R 4.0.2)
stringi	1.6.2	2021-05-17	[1]	CRAN	(R 4.0.3)
stringr	* 1.4.0	2019-02-10	[1]	CRAN	(R 4.0.2)
svglite	2.0.0	2021-02-20	[1]	CRAN	(R 4.0.3)
systemfonts	1.0.2	2021-05-11	[1]	CRAN	(R 4.0.3)
tibble	* 3.1.2	2021-05-16	[1]	CRAN	(R 4.0.3)
tidyr	* 1.1.3	2021-03-03	[1]	CRAN	(R 4.0.3)
tidyselect	1.1.1	2021-04-30	[1]	CRAN	(R 4.0.3)
tidyverse	* 1.3.1	2021-04-15	[1]	CRAN	(R 4.0.3)
utf8	1.2.1	2021-03-12	[1]	CRAN	(R 4.0.3)
uuid	0.1-4	2020-02-26	[1]	CRAN	(R 4.0.2)
vctrs	0.3.8	2021-04-29	[1]	CRAN	(R 4.0.3)
withr	2.4.2	2021-04-18	[1]	CRAN	(R 4.0.3)
xml2	1.3.2	2020-04-23	[1]	CRAN	(R 4.0.2)
zip	2.2.0	2021-05-31	[1]	CRAN	(R 4.0.3)

[1] /home/jbenja13/R/x86_64-pc-linux-gnu-library/4.0

[2] /usr/lib/R/library