R project BIOF339: Hippocampal gene expression (Cembrowski et al., eLife 2016)

Su Hyun Lee

12/12/2018

Background: Cembrowski et al. have used a technique called next-generation RNA sequencing (RNA-seq) to determine which genes are expressed in groups of neurons that represent the main cell types found in a part of the brain called the hippocampus. This brain region is important for memory, and was chosen because the location and appearance of the main cell types in the hippocampus were already well understood.

Author Cembrowski et al. used next-generation RNA sequencing (RNA-seq) to produce a quantitative, whole genome characterization of gene expression for the major excitatory neuronal classes of the hippocampus; namely, granule cells and mossy cells of the dentate gyrus, and pyramidal cells of areas CA3, CA2, and CA1. Moreover, for the canonical cell classes of the trisynaptic loop, and profiled transcriptomes at both dorsal and ventral poles, producing a cell-class- and region-specific transcriptional description for these populations.

The approach revealed that the main types of neurons in the mouse hippocampus are all very different from each other in terms of gene expression, and that even neurons of the same type can exhibit large differences across the hippocampus. Cembrowski et al. created a website that will allow other researchers to easily navigate, analyze, and visualize gene expression data in these populations of neurons.

The data set is availiable on “Hipposeq”, (http://hipposeq.janelia.org (http://hipposeq.janelia.org)).

Here, we used the data set from Hipposeq and compared the gene expression between CA2 region vs dorsal and ventral CA1 region. We have filtered out gene expression for CA2 region and ranked by p-value and the expression level.

ggplot an dplyr package used. Working directory set and file read.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.4
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.4
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
getwd() #get working directory
```

```
## [1] "/Users/lees44/R_project"
```

```
setwd("/Users/lees44/R_project/") #set working directory
data<-read.table("Spruston Hippo_gene_exp.txt", header = T, sep = "\t") #read table
```

Here, we filtered data for ca2 region only. We ordered gene expression level from highest to lowest.

```
##sort by the descending value
newdata<- filter(data, sample_1 =="ca2") #filter by 'sample 1' 'ca2' only
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
data1<-newdata[order(newdata$value_1, decreasing=T),] #order by gene expression level
from highest to lowest
data2<- data1[c(1,3, 5,6,8,9,12)] #select only relevant columns eg. gene name, gene e
xpression value and p-value
data3 <-data2[1:100,]
names(data3)[names(data3) == 'value_1'] <- 'CA2_gene_expression' #column name altered
to CA2_gene_expression
names(data3)[names(data3) == 'value_2'] <- 'CA1_gene_expression' #column name altered
to CA1_gene_expression
#write and export table
write.table(data3, "/Users/lees44/R_project/data2", sep="\t")
data3[1:10,] #list table rows from one to ten only
```

```
##                  test_id   gene sample_1 sample_2 CA2_gene_expression
## 10394 ENSMUSG00000036438  Calm2      ca2    ca1_d             5721.57
## 48093 ENSMUSG00000036438  Calm2      ca2    ca1_v             5721.57
## 6928  ENSMUSG00000028785   Hpca      ca2    ca1_d             3111.89
## 44627 ENSMUSG00000028785   Hpca      ca2    ca1_v             3111.89
## 5043  ENSMUSG00000025393  Atp5b      ca2    ca1_d             2989.13
## 42742 ENSMUSG00000025393  Atp5b      ca2    ca1_v             2989.13
## 9996  ENSMUSG00000035202  Lars2      ca2    ca1_d             2696.22
## 47695 ENSMUSG00000035202  Lars2      ca2    ca1_v             2696.22
## 36189 ENSMUSG00000092341 Malat1      ca2    ca1_d             2333.65
## 73888 ENSMUSG00000092341 Malat1      ca2    ca1_v             2333.65
##       CA1_gene_expression p_value
## 10394            5914.220 0.67885
## 48093            4383.310 0.00120
## 6928             1828.600 0.00005
## 44627             538.222 0.00005
## 5043             1791.610 0.00005
## 42742            2283.170 0.00795
## 9996             2952.900 0.12510
## 47695            2354.350 0.04595
## 36189            1900.240 0.17990
## 73888            1550.560 0.00935
```

Here, we order the data by p-value. Gene with highest significance value of difference between CA2 and CA1 region.

```
##sorted by p-value
head(newdata)
```

```
##              test_id            gene_id  gene                 locus
## 1 ENSMUSG00000000001 ENSMUSG00000000001 Gnai3  3:107910197-107949064
## 2 ENSMUSG00000000003 ENSMUSG00000000003  Pbsn    X:75083239-75098962
## 3 ENSMUSG00000000028 ENSMUSG00000000028 Cdc45   16:18780539-18835354
## 4 ENSMUSG00000000031 ENSMUSG00000000031   H19  7:149761433-149764048
## 5 ENSMUSG00000000037 ENSMUSG00000000037 Scml2  X:157555124-157696145
## 6 ENSMUSG00000000049 ENSMUSG00000000049  Apoh 11:107794700-108275710
##   sample_1 sample_2 status  value_1   value_2 log2.fold_change. test_stat
## 1      ca2    ca1_d     OK 10.71920 7.8593100          -0.44773 -0.828798
## 2      ca2    ca1_d NOTEST  0.00000 0.0000000           0.00000  0.000000
## 3      ca2    ca1_d     OK  2.20752 0.2019320          -3.45049 -0.762827
## 4      ca2    ca1_d NOTEST  0.00000 0.0000000           0.00000  0.000000
## 5      ca2    ca1_d     OK  0.69732 0.0112204          -5.95763 -0.288255
## 6      ca2    ca1_d NOTEST  0.45634 0.2240910          -1.02602  0.000000
##   p_value  q_value significant
## 1 0.14765 0.332102          no
## 2 1.00000 1.000000          no
## 3 0.08220 0.225052          no
## 4 1.00000 1.000000          no
## 5 0.04960 0.157213          no
## 6 1.00000 1.000000          no
```

```
newdata4<- filter(data1,sample_1 =="ca2") #filter by ca2 sample only
newdata5<- newdata4[order(newdata4$p_value),] #order data by p-value
newdata6<-newdata5[c(1,3,5,6,8,9,10,12)] #filter columns
names(newdata6)[names(newdata6) == 'value_1'] <- 'CA2_gene_expression' #column name a
ltered
names(newdata6)[names(newdata6) == 'value_2'] <- 'CA1_gene_expression' #column name a
ltered
newdata7 <-newdata6[1:100,] #newdata7 only includes 100 rows of newdata6
names(newdata7)[names(newdata7) == 'value_1'] <- 'CA2_gene_expression' #column name a
ltered
names(newdata7)[names(newdata7) == 'value_2'] <- 'CA1_gene_expression' #column name a
ltered
write.table(newdata7, "/Users/lees44/R_project/data5", sep="\t") #write new table
newdata7[1:10,] #display 10 rows of newdata7
```

```
##                test_id   gene sample_1 sample_2 CA2_gene_expression
## 3  ENSMUSG00000028785   Hpca      ca2    ca1_d             3111.89
## 4  ENSMUSG00000028785   Hpca      ca2    ca1_v             3111.89
## 5  ENSMUSG00000025393  Atp5b      ca2    ca1_d             2989.13
## 11 ENSMUSG00000026576 Atp1b1      ca2    ca1_d             2212.32
## 12 ENSMUSG00000026576 Atp1b1      ca2    ca1_v             2212.32
## 15 ENSMUSG00000021087   Rtn1      ca2    ca1_d             2057.43
## 18 ENSMUSG00000032532    Cck      ca2    ca1_v             1998.44
## 20 ENSMUSG00000049775 Tmsb4x      ca2    ca1_v             1715.59
## 21 ENSMUSG00000090223   Pcp4      ca2    ca1_d             1687.60
## 22 ENSMUSG00000090223   Pcp4      ca2    ca1_v             1687.60
##    CA1_gene_expression log2.fold_change. p_value
## 3           1828.60000         -0.767051   5e-05
## 4            538.22200         -2.531520   5e-05
## 5           1791.61000         -0.738467   5e-05
## 11          1409.98000         -0.649887   5e-05
## 12          1462.34000         -0.597289   5e-05
## 15          1264.44000         -0.702349   5e-05
## 18          1155.62000         -0.790206   5e-05
## 20           966.86600         -0.827314   5e-05
## 21             1.43521        -10.199500   5e-05
## 22            96.77030         -4.124270   5e-05
```

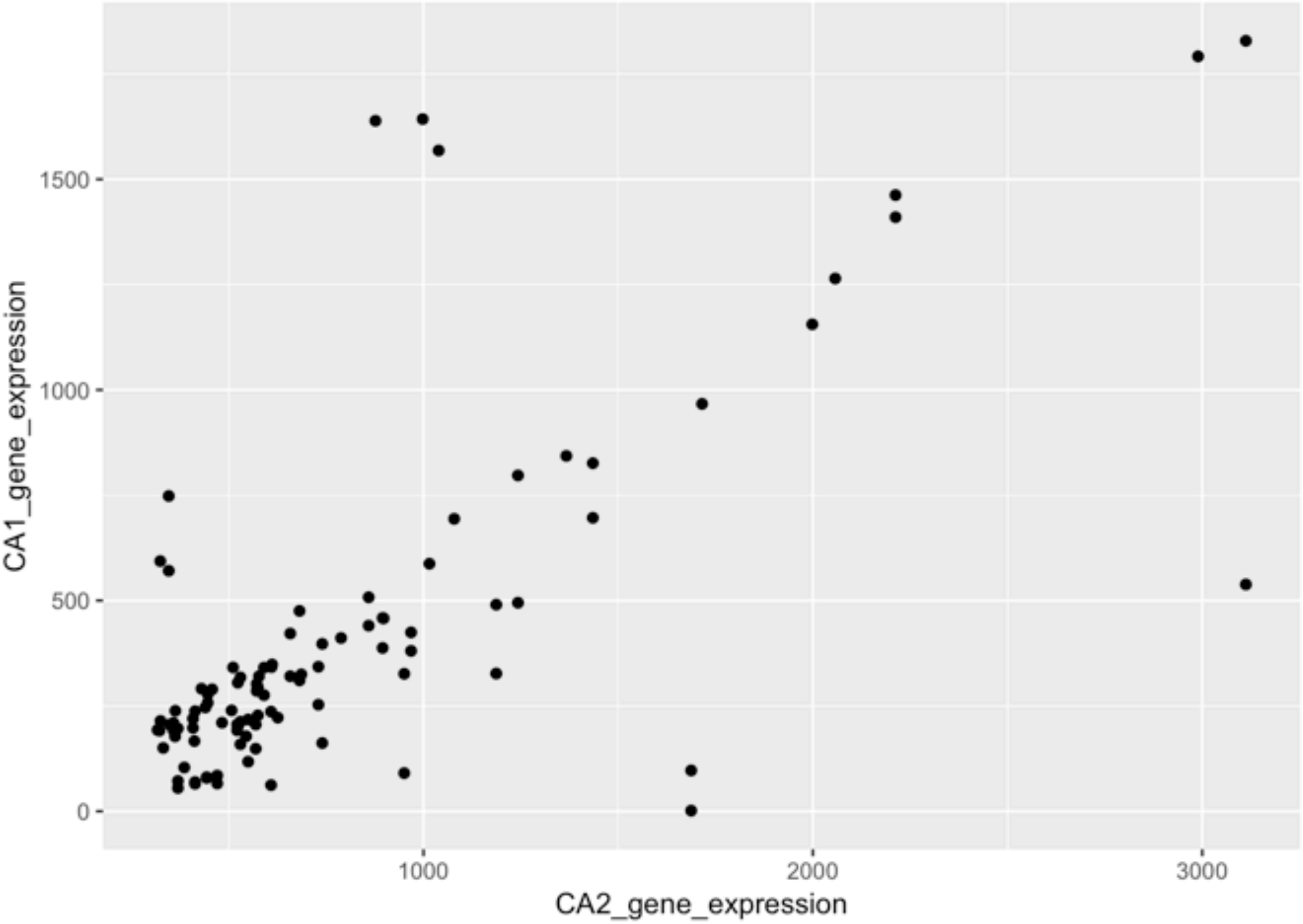
Data summary of CA2_gene_expression vs CA1_gene_expression for 100 datasets that are ordered by P-value.

```
summary(newdata7[,c(5,6)])
```

```
##  CA2_gene_expression CA1_gene_expression
##  Min.   : 317.2      Min.   :   1.435   
##  1st Qu.: 442.5      1st Qu.: 195.619   
##  Median : 574.8      Median : 287.445   
##  Mean   : 797.8      Mean   : 414.633   
##  3rd Qu.: 910.9      3rd Qu.: 458.030   
##  Max.   :3111.9      Max.   :1828.600   
```

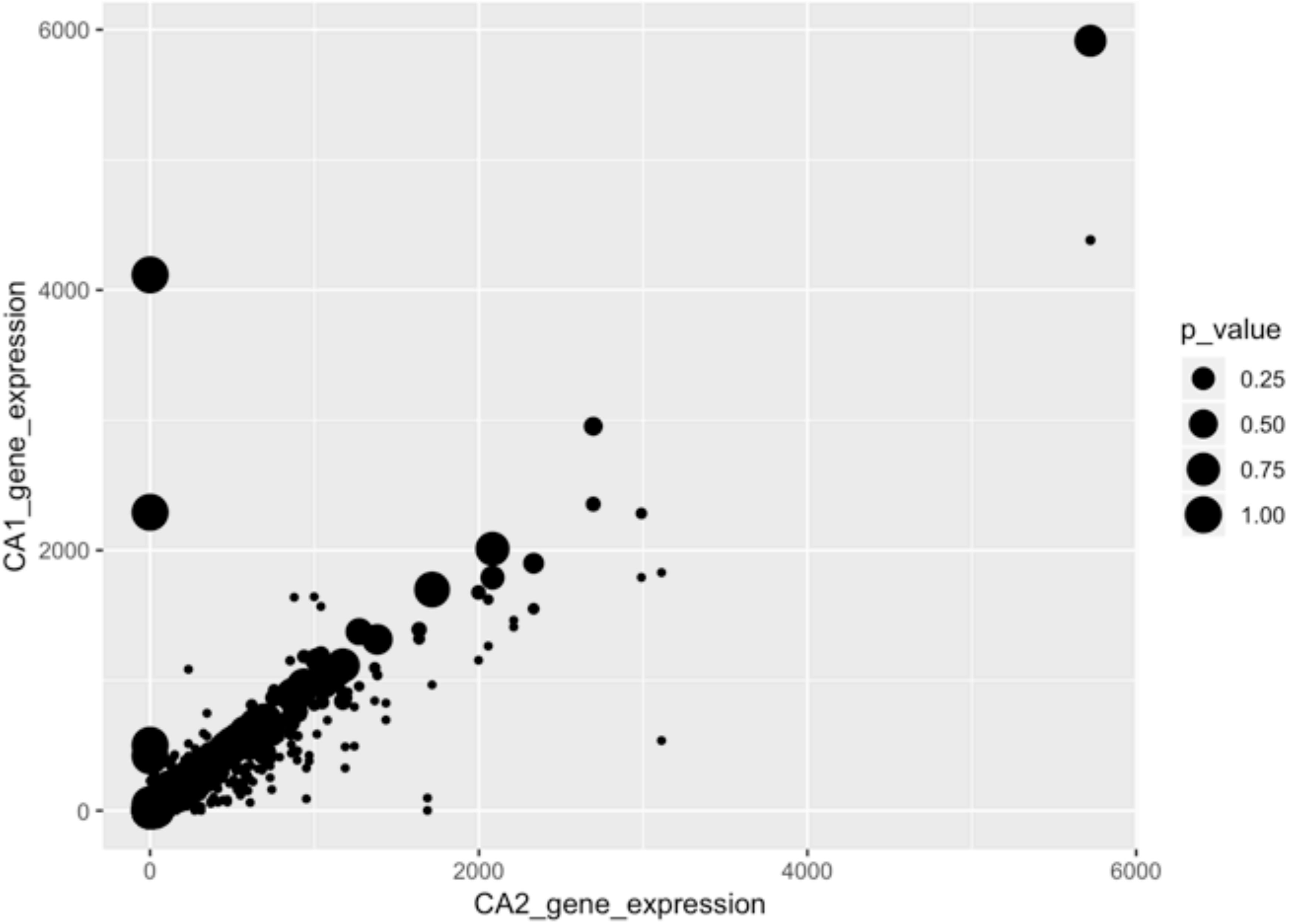
Scatter plot to see CA2 gene expression vs CA1 gene expression

```
#ggplot
temporary <- newdata7
rownames(temporary) <- make.names(temporary$gene, TRUE)
ggplot(newdata7, aes(CA2_gene_expression, CA1_gene_expression)) + geom_point()
```



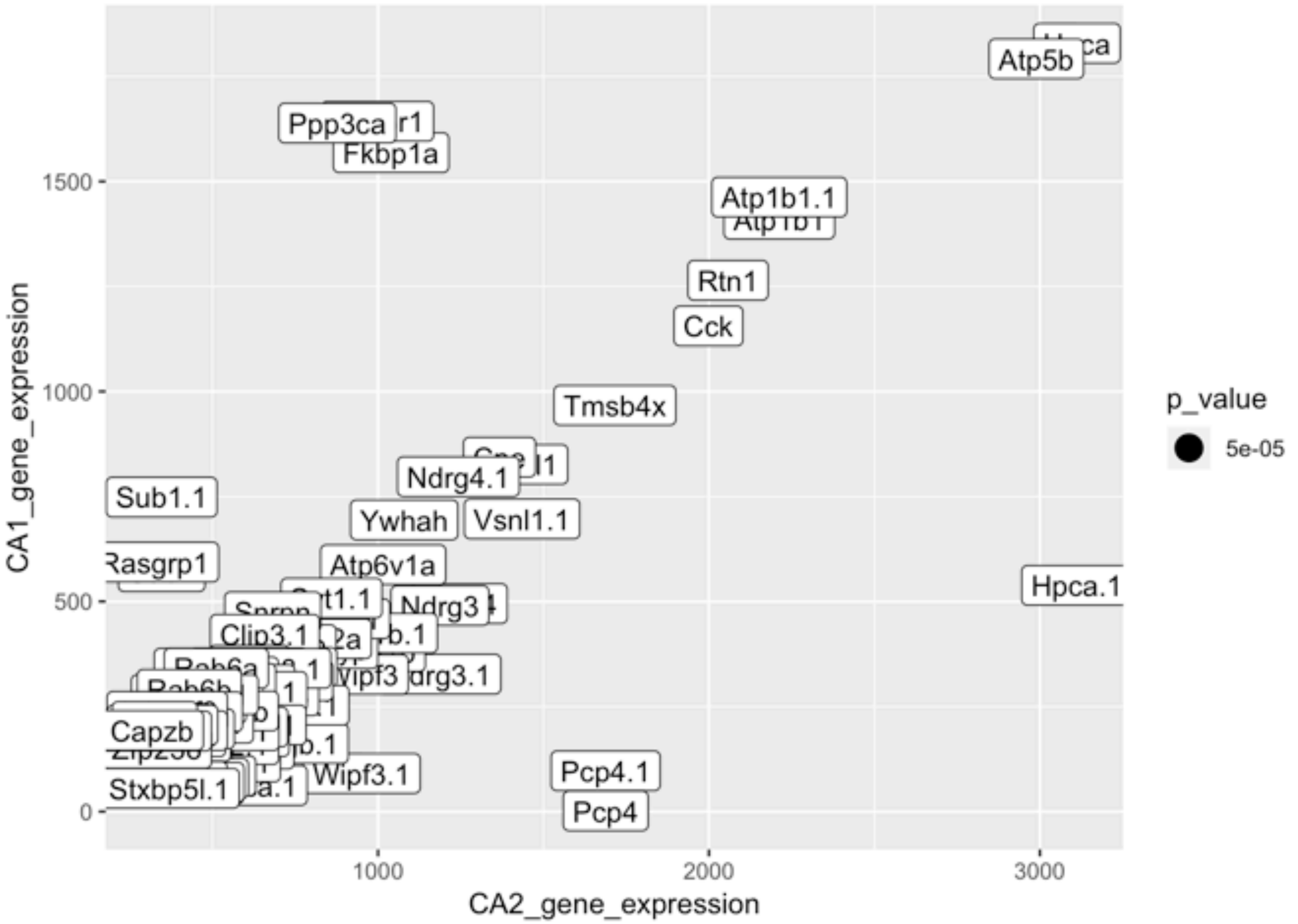
Scatterplot of gene expression between CA2 and CA1 with display of p-value

```
ggplot(newdata6, aes(x = CA2_gene_expression, CA1_gene_expression)) + geom_point(aes(
size = p_value))
```



Scattr plot displaying gene expression of CA2 vs CA1 with the gene name label displayed

```
ggplot(temporary, aes(x = CA2_gene_expression, y = CA1_gene_expression)) + geom_point
(aes(size = p_value)) + geom_label(label=rownames(temporary), nudge_x = 0.25, nudge_y
= 0.2)
```



Correlation between CA2 and CA1 gene expression

```
cor(newdata7$CA2_gene_expression, newdata7$CA1_gene_expression, method="pearson")
```

```
## [1] 0.7043021
```

Conclusion: Cembrowski et al. have analysed data by identifying three-fold gene expression difference pair-wise comparison using FDR values. Here, we order genes based on the gene expression differences and p-values. Based on p-values, genes such as hpca and pcp4 has highest gene expression in CA2 region and significantly different to dorsal and ventral CA1 regions. Based n pearson correlation, CA2 gene expression is highly correlatd with CA1 gene expression. Further CA2 markers should be identified by gene expression level between CA2 and other hippocampal regions.

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.