

DEG Gene Box Plots

Annika Jorgensen

2022-09-02

R Markdown

Title: "Box Plot Genes from Viral Etiology DEG Analysis"

Author: Annika Jorgensen

Date: 09/02/2022

Purpose: This document is for the author to create box plots for level of expression of genes taken from the Female HCV, Female HBV, Male HBV, and Female HBV.

Libraries

The first chunk of code is dedicated to installing the libraries. These libraries are to help execute the differential analysis and helps visualize the data. The code was not included for concision.

Environment parameters

This next section of code is dedicated to the environmental parameters. Environmental parameters are a series of variables and other code that will help make the rest of the script be easier to make and run later on.

Working Directory

A working directory is a code that iterates a file path on your computer th.t sets where the default location of any files that you read into R. Working directories work different in R files than R Markdowns. R Markdown files require directories to be defined at the end of each code chunk. Meaning from here on out you will see working directories being defined at the end of each code chunk.

```
setwd('~ / R')
```

Defining Colors

This chunk defines color palette variables that are going to be used in plots later on the script. These variables are defined by conversiting BrewerCode palettes into palettes that can be used in R.

```
viralPalette <- brewer.pal(8, "Set1")
hsvColor <- viralPalette[1]
hcvColor <- viralPalette[2]
bothColor <- viralPalette[3]
neitherColor <- viralPalette[4]

sexTissuePalette <- brewer.pal(12, "Paired")
maleTumorColor <- sexTissuePalette[4]
maleAdjacentColor <- sexTissuePalette[3]
femaleTumorColor <- sexTissuePalette[6]
femaleAdjacentColor <- sexTissuePalette[5]
setwd('~R')
```

Reading in Metadata

This chunk reads in the original salmon expression counts meta data. This data holds all of the expression counts from the tumor and tumor-adjacent samples. This data will be used to create box plots showing the expression of the gene in a variety of different samples.

```
metadata <- read.table("metadata_for_de.csv", row.names=1, header=TRUE, sep=",") #changing the name of the file
tumorAdjacentExp <- read.table("japan_all_samples_salmon_expression_counts.txt", row.names = 1, header=TRUE) #changing the name of the file
colnames(tumorAdjacentExp) <- gsub("\\.", "-", colnames(tumorAdjacentExp)) #changing the column names
```

The subsetting is making sure that the expression data is truly tied to the metadata by making expression counts to the metadata. Additionally, we are making sure the row names match the sample id, making sure the matrices of tumorAdjacentExpSubset and metadataSubset are aligned and we are adding an additional column called tumor where 0 means the sample is tumor adjacent and 1 means the sample is tumor

```
tumorAdjacentExpSubset <- tumorAdjacentExp[, colnames(tumorAdjacentExp) %in% metadata$sampleid]
metadataSubset <- metadata[metadata$sampleid %in% colnames(tumorAdjacentExpSubset),]
metadataSubset <- metadataSubset[match(colnames(tumorAdjacentExpSubset), metadataSubset$sampleid),]
identical(colnames(tumorAdjacentExpSubset), metadataSubset$sampleid)
```

```
## [1] TRUE
```

```
rownames(metadataSubset) <- metadataSubset$sampleid
metadataSubset$tumor <- as.numeric(grepl('tumor', metadataSubset$sampleid, ignore.case=T))
```

Box Plot Data Subsetting

This section is subsetting the datasets in order to show boxplots stratified by sex, etiology, and tumor tumor adjacent. This code chunk specifically subsetting by sex. The “which” command finds all samples that are marked “F” or “M” in the gender column of the metadataSubset file and then subsets those samples in the tumorAdjacentExpSubset file into the “female_expression” or “male_expression” variable.

```
female_dataset <- metadataSubset[ which((metadataSubset$Gender) == "F"), ]
male_dataset <- metadataSubset[ which((metadataSubset$Gender) == "M"), ]

female_expression <- tumorAdjacentExp[,colnames(tumorAdjacentExpSubset) %in% female_dataset$sampleid]
male_expression <- tumorAdjacentExp[,colnames(tumorAdjacentExpSubset) %in% male_dataset$sampleid]
```

This code chunk is specifically subsetting the data by etiology. The “which” command finds all samples that are marked “HBV” or “HCV” in the “Virus_Infection” column of the metadataSubset file and then subsets those samples in the tumorAdjacentExpSubset file into the “HBV_expression” and “HCV_expression” variable.

```
HBV_dataset <- metadataSubset[ which((metadataSubset$Virus_infection) == "HBV"), ]
HCV_dataset <- metadataSubset[ which((metadataSubset$Virus_infection) == "HCV"), ]

HBV_expression <- tumorAdjacentExp[,colnames(tumorAdjacentExpSubset) %in% HBV_dataset$sampleid]
HCV_expression <- tumorAdjacentExp[,colnames(tumorAdjacentExpSubset) %in% HCV_dataset$sampleid]
```

This code chunk is specifically subsetting by tumor and tumor-adjacent tissue type. The “which” command finds all samples that are marked by a zero or a one in the tumor column of the metadataSubset file. The tumor column has tumor-adjacent samples marked as zero and a tumor sample marked as one. These samples are then subsetting in the tumorAdjacentExpSubset file and are inputted into the “tumor_expression” and “adjacent_expression” variable.

```
tumor_dataset <- metadataSubset[ which((metadataSubset$tumor) == 1), ]
adjacent_dataset <- metadataSubset[ which((metadataSubset$tumor) == 0), ]

tumor_expression <- tumorAdjacentExp[,colnames(tumorAdjacentExpSubset) %in% tumor_dataset$sampleid]
adjacent_expression <- tumorAdjacentExp[,colnames(tumorAdjacentExpSubset) %in% adjacent_dataset$sampleid]
```

This code chunk is subsetting the sex-subsetted data by tumor. This is to create female tumor and male tumor datasets to create box plots with. The “which” command finds all samples that are marked by a zero or a one in the tumor column of the metadataSubset file. The tumor column has tumor-adjacent samples marked as zero and a tumor sample marked as one. These samples are then subsetting in the tumorAdjacentExpSubset file and are inputted into the “female_tumor_expression” and “male_tumor_expression” variable.

```
female_tumor_dataset <- female_dataset[ which((female_dataset$tumor) == 1), ]
male_tumor_dataset <- male_dataset[ which((male_dataset$tumor) == 1), ]

female_tumor_expression <- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% female_tumor_dataset$sampleid]
male_tumor_expression <- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% male_tumor_dataset$sampleid]
```

This code chunk is subsetting the sex-subsetted data by tumor-adjacent. This is to create female tumor-adjacent and male tumor-adjacent datasets to create box plots with. The “which” command finds all samples that are marked by a zero or a one in the tumor column of the metadataSubset file. The tumor column has tumor-adjacent

samples marked as zero and a tumor sample marked as one. These samples are then subsetted in the tumorAdjacentExpSubset file and are inputted into the “female_adjacent_expression” and “male_adjacent_expression” variable.

```
female_adjacent_dataset <- female_dataset[ which((female_dataset$tumor) == 0), ]
male_adjacent_dataset <- male_dataset[ which((male_dataset$tumor) == 0), ]

female_adjacent_expression <- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% female_adjacent_dataset$sampleid]
male_adjacent_expression <- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% male_adjacent_dataset$sampleid]
```

```
female_tumor_HBV_dataset<- female_tumor_dataset[ which((female_tumor_dataset$Virus_infection) == "HBV"), ]
male_tumor_HBV_dataset<- male_tumor_dataset[ which((male_tumor_dataset$Virus_infection) == "HBV"), ]

female_tumor_HBV_expression<- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% female_tumor_HBV_dataset$sampleid]
male_tumor_HBV_expression<- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% male_tumor_HBV_dataset$sampleid]
```

```
female_tumor_HCV_dataset<- female_tumor_dataset[ which((female_tumor_dataset$Virus_infection) == "HCV"), ]
male_tumor_HCV_dataset<- male_tumor_dataset[ which((male_tumor_dataset$Virus_infection) == "HCV"), ]

female_tumor_HCV_expression<- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% female_tumor_HCV_dataset$sampleid]
male_tumor_HCV_expression<- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% male_tumor_HCV_dataset$sampleid]
```

```
female_adjacent_HBV_dataset<- female_adjacent_dataset[ which((female_adjacent_dataset$Virus_infection) == "HBV"), ]
male_adjacent_HBV_dataset<- male_adjacent_dataset[ which((male_adjacent_dataset$Virus_infection) == "HBV"), ]

female_adjacent_HBV_expression<- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% female_adjacent_HBV_dataset$sampleid]
male_adjacent_HBV_expression<- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% male_adjacent_HBV_dataset$sampleid]
```

```
female_adjacent_HCV_dataset<- female_adjacent_dataset[ which((female_adjacent_dataset$Virus_infection) == "HCV"), ]
male_adjacent_HCV_dataset<- male_adjacent_dataset[ which((male_adjacent_dataset$Virus_infection) == "HCV"), ]

female_adjacent_HCV_expression<- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% female_adjacent_HCV_dataset$sampleid]
male_adjacent_HCV_expression<- tumorAdjacentExpSubset[,colnames(tumorAdjacentExpSubset) %in% male_adjacent_HCV_dataset$sampleid]
```

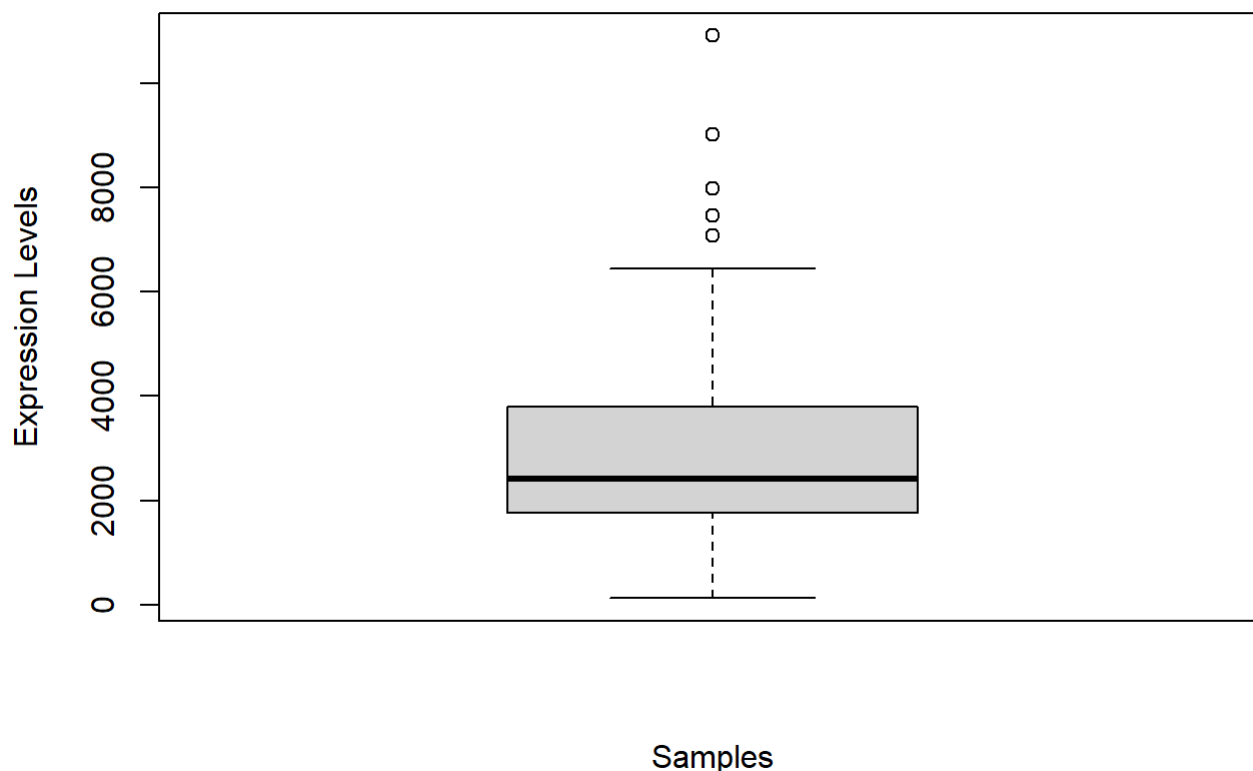
##Plots This section is creating the boxplots with the data read in and subsetting above.

Test Box Plot

The data variable takes the first row of the female_expression file and changes the file to all numerical values. The data variable is then plotted as a box plot with the gene (TSPAN 6 or ENSG00000000003.14) on the x axis and the expression levels on the y axis.

```
data <- as.numeric(female_expression[1, ])
boxplot(data, main= "Expression Levels of TSPAN6 in Female Samples", xlab= "Samples", ylab= "Expression Levels", horizontal= F)
```

Expression Levels of TSPAN6 in Female Samples



MUC13 Box Plots

This section is dedicated to creating and outputting a boxplot for the expression levels of the MUC13 gene.

This box plot has four categories: female tumor, female tumor-adjacent, male tumor, male tumor-adjacent on the same plot. The MUC13data_0 variable has all of samples that express the MUC13 gene in female tumor samples. The grep function was used to retrieve all of the rows in the female_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

MUC13data_1 is the same as the MUC13data_7 except that it is dealing with the female tumor-adjacent samples. The grep function was used to retrieve all of the rows in the female_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

MUC13data_2 is the same as the MUC13data_8 except that it is dealing with the male tumor samples. The grep function was used to retrieve all of the rows in the male_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

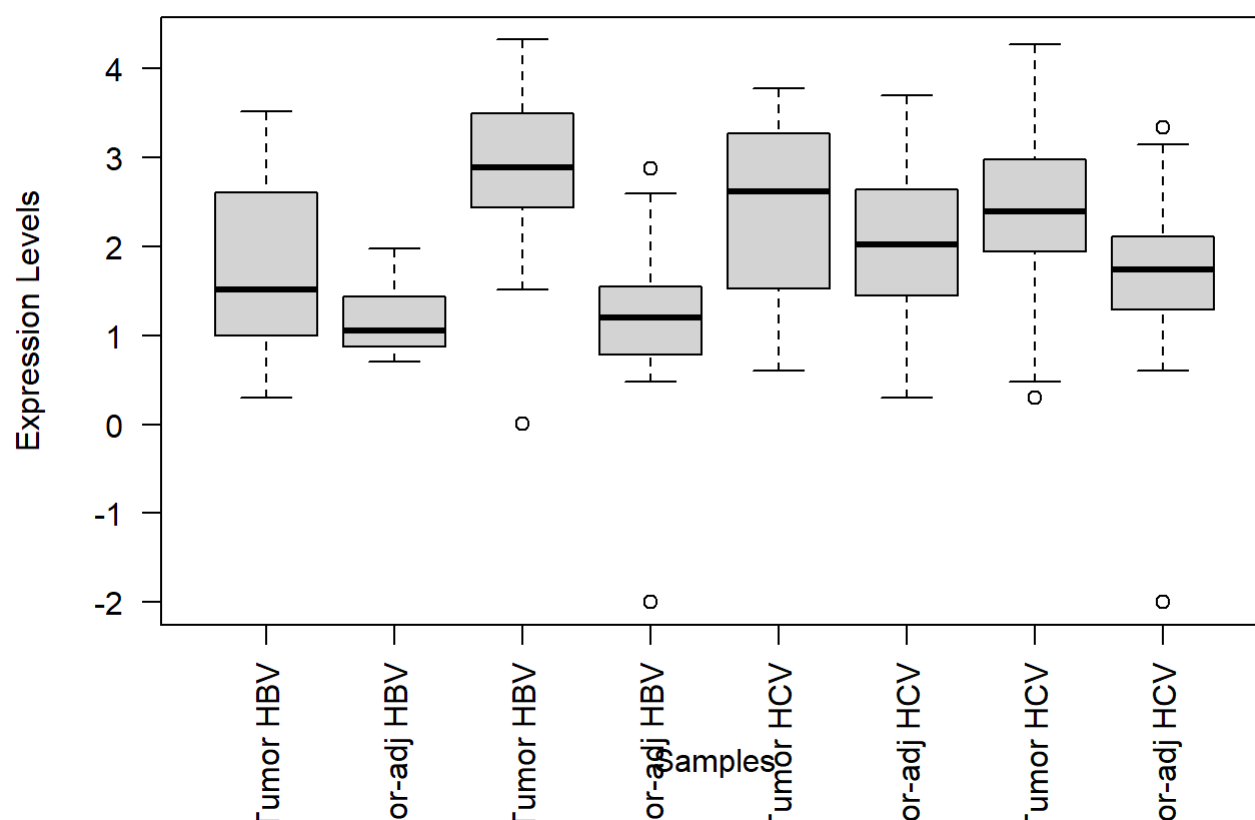
MUC13data_3 is the same as the MUC13data_9 except that it is dealing with the male tumor-adjacent samples. The grep function was used to retrieve all of the rows in the male_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

```
MUC13data_0 <- as.numeric(female_tumor_HBV_expression[ grep(pattern="^ENSG00000173702", rownames(
female_tumor_HBV_expression)), ])
MUC13data_1 <- as.numeric(female_adjacent_HBV_expression[ grep(pattern="^ENSG00000173702", rowna
mes(female_adjacent_HBV_expression)), ])
MUC13data_2 <- as.numeric(male_tumor_HBV_expression[ grep(pattern="^ENSG00000173702", rownames(m
ale_tumor_HBV_expression)), ])
MUC13data_3 <- as.numeric(male_adjacent_HBV_expression[ grep(pattern="^ENSG00000173702", rowname
s(male_adjacent_HBV_expression)), ])
MUC13data_4 <- as.numeric(female_tumor_HCV_expression[ grep(pattern="^ENSG00000173702", rownames
(female_tumor_HCV_expression)), ])
MUC13data_5 <- as.numeric(female_adjacent_HCV_expression[ grep(pattern="^ENSG00000173702", rowna
mes(male_adjacent_HCV_expression)), ])
MUC13data_6 <- as.numeric(male_tumor_HCV_expression[ grep(pattern="^ENSG00000173702", rownames(m
ale_tumor_HCV_expression)), ])
MUC13data_7 <- as.numeric(male_adjacent_HCV_expression[ grep(pattern="^ENSG00000173702", rowname
s(male_adjacent_HCV_expression)), ])
pdf("~/R/MUC13_expression_boxplot.pdf", width=12, height=12)
boxplot(log10(MUC13data_0+0.01), log10(MUC13data_1+0.01), log10(MUC13data_2+0.01), log10(MUC13da
ta_3+0.01), log10(MUC13data_4+0.01), log10(MUC13data_5+0.01),log10(MUC13data_6+0.01), log10(MUC1
3data_7+0.01), main= "Expression Levels of MUC13", xlab= "Samples", ylab= "Expression Levels", h
orizontal= F, names= c("F Tumor HBV", "Female Tumor-adj HBV", "M Tumor HBV", "Male Tumor-adj HB
V", "F Tumor HCV", "F Tumor-adj HCV", "M Tumor HCV", "Male Tumor-adj HCV"), las=2)
```

The box plot is the same as up above this chunk is just printing it out to the RMD file.

```
boxplot(log10(MUC13data_0+0.01), log10(MUC13data_1+0.01), log10(MUC13data_2+0.01), log10(MUC13da
ta_3+0.01), log10(MUC13data_4+0.01), log10(MUC13data_5+0.01),log10(MUC13data_6+0.01), log10(MUC1
3data_7+0.01), main= "Expression Levels of MUC13", xlab= "Samples", ylab= "Expression Levels", h
orizontal= F, names= c("F Tumor HBV", "Female Tumor-adj HBV", "M Tumor HBV", "Male Tumor-adj HB
V", "F Tumor HCV", "F Tumor-adj HCV", "M Tumor HCV", "Male Tumor-adj HCV"), las=2)
```

Expression Levels of MUC13



```
#boxplot(Log10(MUC13data_7), Log10(MUC13data_8), Log10(MUC13data_9), Log10(MUC13data_10), ylim=c(0,3000), main= "Expression Levels of MUC13", xlab= "Female Tumor, Female Tumor-adjacent, Male Tumor, Male Tumor-adjacent Samples", ylab= "Expression Levels", horizontal= F)
```

MAGEA6 Box Plot

This section is dedicated to creating and outputting a boxplot for the expression levels of the MAGEA6 gene.

This box plot has four categories: female tumor, female tumor-adjacent, male tumor, male tumor-adjacent on the same plot. The MAGEA6data_0 variable has all of samples that express the MAGEA6 gene in female tumor samples. The grep function was used to retrieve all of the rows in the female_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

MAGEA6data_1 is the same as the MAGEA6data_0 except that it is dealing with the female tumor-adjacent samples. The grep function was used to retrieve all of the rows in the female_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

MAGEA6data_2 is the same as the MAGEA6data_1 except that it is dealing with the male tumor samples. The grep function was used to retrieve all of the rows in the male_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

MAGEA6data_3 is the same as the MAGEA6data_2 except that it is dealing with the male tumor-adjacent samples. The grep function was used to retrieve all of the rows in the male_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

```

MAGEA6data_0 <- as.numeric(female_tumor_HBV_expression[ grep(pattern="^ENSG00000197172", rownames(female_tumor_HBV_expression)), ])
MAGEA6data_1 <- as.numeric(female_adjacent_HBV_expression[ grep(pattern="^ENSG00000197172", rownames(female_adjacent_HBV_expression)), ])
MAGEA6data_2 <- as.numeric(male_tumor_HBV_expression[ grep(pattern="^ENSG00000197172", rownames(male_tumor_HBV_expression)), ])
MAGEA6data_3 <- as.numeric(male_adjacent_HBV_expression[ grep(pattern="^ENSG00000197172", rownames(male_adjacent_HBV_expression)), ])
MAGEA6data_4 <- as.numeric(female_tumor_HCV_expression[ grep(pattern="^ENSG00000173702", rownames(female_tumor_HCV_expression)), ])
MAGEA6data_5 <- as.numeric(female_adjacent_HCV_expression[ grep(pattern="^ENSG00000173702", rownames(male_adjacent_HCV_expression)), ])
MAGEA6data_6 <- as.numeric(male_tumor_HCV_expression[ grep(pattern="^ENSG00000173702", rownames(male_tumor_HCV_expression)), ])
MAGEA6data_7 <- as.numeric(male_adjacent_HCV_expression[ grep(pattern="^ENSG00000173702", rownames(male_adjacent_HCV_expression)), ])

pdf("~/R/MAGEA6_expression_boxplot.pdf", width=12, height=12)
boxplot(log10(MAGEA6data_0+0.01), log10(MAGEA6data_1+0.01), log10(MAGEA6data_2+0.01), log10(MAGEA6data_3+0.01), log10(MAGEA6data_4+0.01), log10(MAGEA6data_5+0.01), log10(MAGEA6data_6+0.01), log10(MAGEA6data_7+0.01), main= "Expression Levels of MAGEA6", xlab= "Samples", ylab= "Expression Levels", horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Male Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male Tumor adjacent HCV"))

```

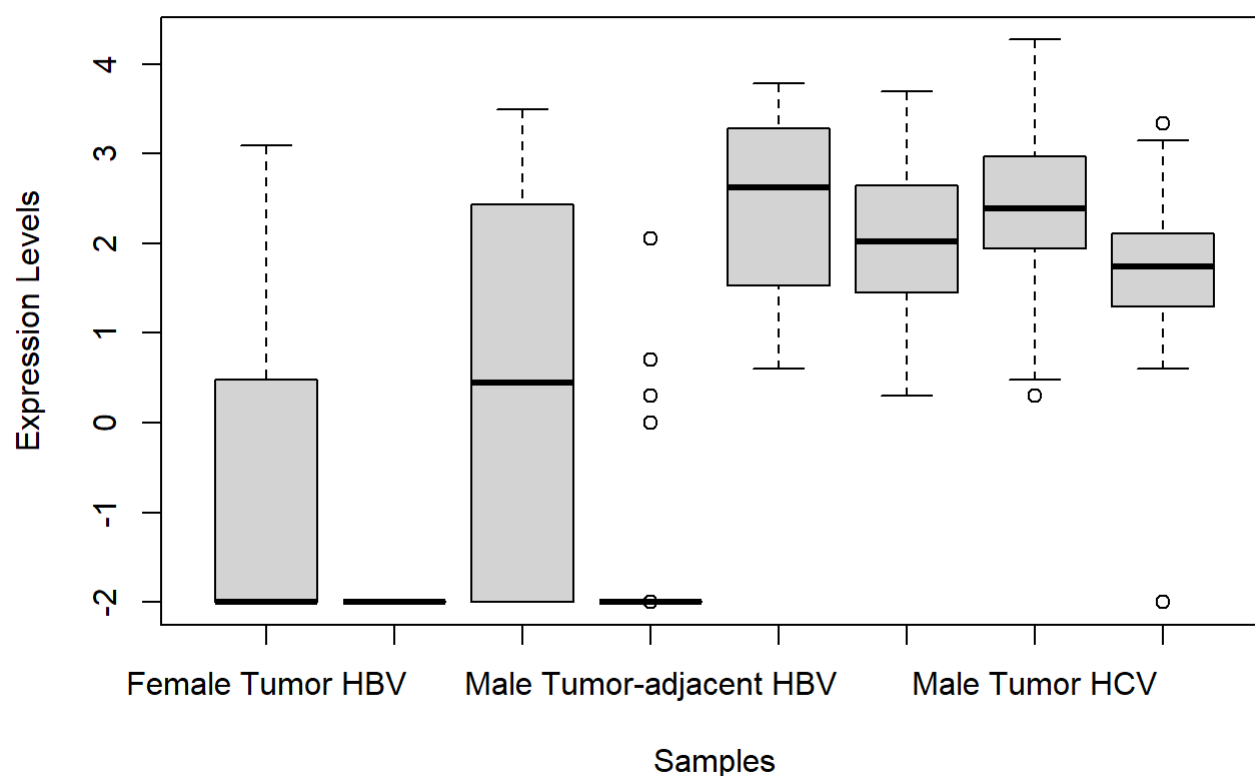
The box plot is the same as up above this chunk is just printing it out to the RMD file.

```

boxplot(log10(MAGEA6data_0+0.01), log10(MAGEA6data_1+0.01), log10(MAGEA6data_2+0.01), log10(MAGEA6data_3+0.01), log10(MAGEA6data_4+0.01), log10(MAGEA6data_5+0.01), log10(MAGEA6data_6+0.01), log10(MAGEA6data_7+0.01), main= "Expression Levels of MAGEA6", xlab= "Samples", ylab= "Expression Levels", horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Male Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male Tumor adjacent HCV"))

```


Expression Levels of MAGEA6



CTAG2 Box Plot

This section is dedicated to creating and outputting a boxplot for the expression levels of the CTAG2 gene.

This box plot has four categories: female tumor, female tumor-adjacent, male tumor, male tumor-adjacent on the same plot. The CTAG2data_0 variable has all of samples that express the CTAG2 gene in female tumor samples. The grep function was used to retrieve all of the rows in the female_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

CTAG2data_1 is the same as the CTAG2data_0 except that it is dealing with the female tumor-adjacent samples. The grep function was used to retrieve all of the rows in the female_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

CTAG2data_2 is the same as the CTAG2data_1 except that it is dealing with the male tumor samples. The grep function was used to retrieve all of the rows in the male_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

CTAG2data_3 is the same as the CTAG2data_2 except that it is dealing with the male tumor-adjacent samples. The grep function was used to retrieve all of the rows in the male_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

```

CTAG2data_0 <- as.numeric(female_tumor_HBV_expression[ grep(pattern="^ENSG00000126890", rownames
(female_tumor_HBV_expression)), ])
CTAG2data_1 <- as.numeric(female_adjacent_HBV_expression[ grep(pattern="^ENSG00000126890", rowna
mes(female_adjacent_HBV_expression)), ])
CTAG2data_2 <- as.numeric(male_tumor_HBV_expression[ grep(pattern="^ENSG00000126890", rownames(m
ale_tumor_HBV_expression)), ])
CTAG2data_3 <- as.numeric(male_adjacent_HBV_expression[ grep(pattern="^ENSG00000126890", rowname
s(male_adjacent_HBV_expression)), ])
CTAG2data_4 <- as.numeric(female_tumor_HCV_expression[ grep(pattern="^ENSG00000126890", rownames
(female_tumor_HCV_expression)), ])
CTAG2data_5 <- as.numeric(female_adjacent_HCV_expression[ grep(pattern="^ENSG00000126890", rowna
mes(female_adjacent_HCV_expression)), ])
CTAG2data_6 <- as.numeric(male_tumor_HCV_expression[ grep(pattern="^ENSG00000126890", rownames(m
ale_tumor_HCV_expression)), ])
CTAG2data_7 <- as.numeric(male_adjacent_HCV_expression[ grep(pattern="^ENSG00000126890", rowname
s(male_adjacent_HCV_expression)), ])
pdf("~/R/CTAG2_expression_boxplot.pdf", width=12, height=12)
boxplot(log10(CTAG2data_0+0.01), log10(CTAG2data_1+0.01), log10(CTAG2data_2+0.01), log10(CTAG2da
ta_3+0.01), log10(CTAG2data_4+0.01), log10(CTAG2data_5+0.01), log10(CTAG2data_6+0.01), log10(CTA
G2data_7+0.01), main= "Expression Levels of CTAG2", xlab= "Samples", ylab= "Expression Levels",
  horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Mal
e Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male
Tumor-adjacent HCV"))

```

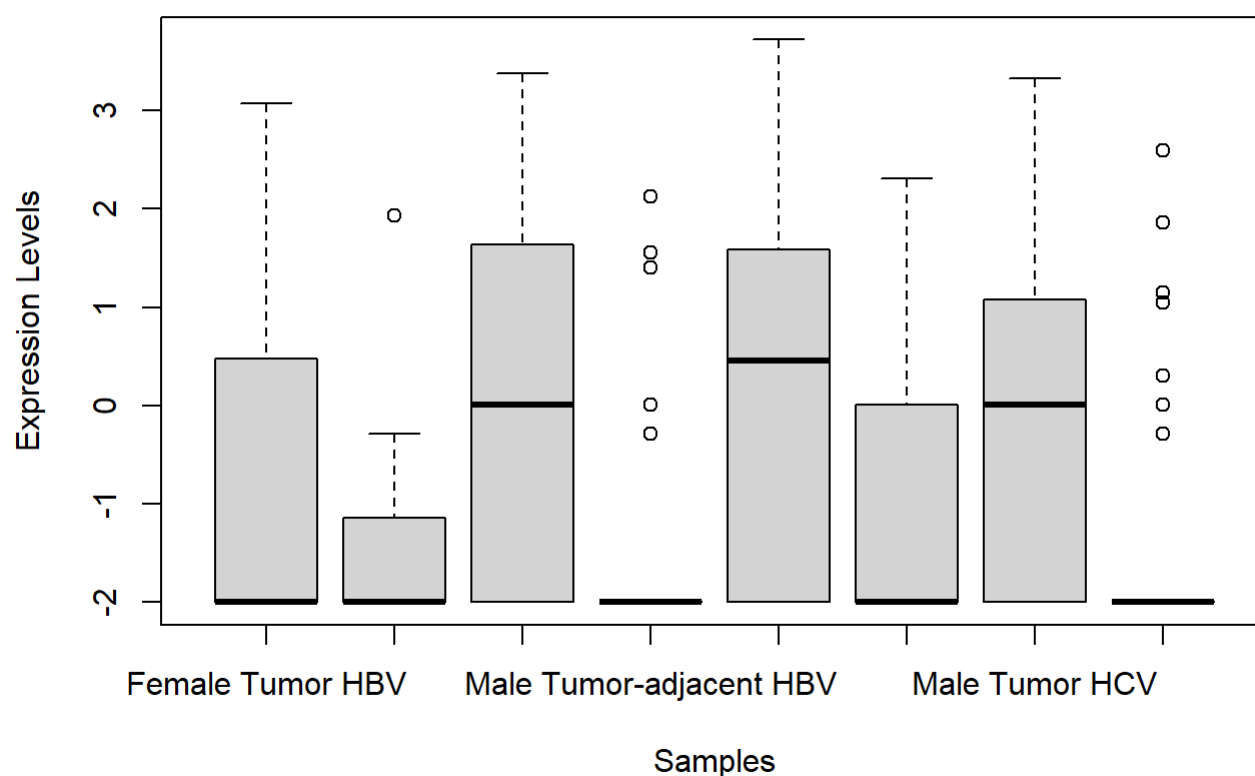
The box plot is the same as up above this chunk is just printing it out to the RMD file.

```

boxplot(log10(CTAG2data_0+0.01), log10(CTAG2data_1+0.01), log10(CTAG2data_2+0.01), log10(CTAG2da
ta_3+0.01), log10(CTAG2data_4+0.01), log10(CTAG2data_5+0.01), log10(CTAG2data_6+0.01), log10(CTA
G2data_7+0.01), main= "Expression Levels of CTAG2", xlab= "Samples", ylab= "Expression Levels",
  horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Mal
e Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male
Tumor-adjacent HCV"))

```

Expression Levels of CTAG2



EEF1A2 Box Plot

This section is dedicated to creating and outputting a boxplot for the expression levels of the EEF1A2 gene.

This box plot has four categories: female tumor, female tumor-adjacent, male tumor, male tumor-adjacent on the same plot. The EEF1A2_data_0 variable has all of samples that express the EEF1A2 gene in female tumor samples. The grep function was used to retrieve all of the rows in the female_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

EEF1A2data_1 is the same as the EEF1A2data_0 except that it is dealing with the female tumor-adjacent samples. The grep function was used to retrieve all of the rows in the female_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

EEF1A2data_2 is the same as the EEF1A2data_1 except that it is dealing with the male tumor samples. The grep function was used to retrieve all of the rows in the male_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

EEF1A2data_3 is the same as the EEF1A2data_2 except that it is dealing with the male tumor-adjacent samples. The grep function was used to retrieve all of the rows in the male_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

```

EEF1A2data_0 <- as.numeric(female_tumor_HBV_expression[ grep(pattern="^ENSG00000101210", rownames(female_tumor_HBV_expression)), ])
EEF1A2data_1 <- as.numeric(female_adjacent_HBV_expression[ grep(pattern="^ENSG00000101210", rownames(female_adjacent_HBV_expression)), ])
EEF1A2data_2 <- as.numeric(male_tumor_HBV_expression[ grep(pattern="^ENSG00000101210", rownames(male_tumor_HBV_expression)), ])
EEF1A2data_3 <- as.numeric(male_adjacent_HBV_expression[ grep(pattern="^ENSG00000101210", rownames(male_adjacent_HBV_expression)), ])
EEF1A2data_4 <- as.numeric(female_tumor_HCV_expression[ grep(pattern="^ENSG00000101210", rownames(female_tumor_HCV_expression)), ])
EEF1A2data_5 <- as.numeric(female_adjacent_HCV_expression[ grep(pattern="^ENSG00000101210", rownames(female_adjacent_HCV_expression)), ])
EEF1A2data_6 <- as.numeric(male_tumor_HCV_expression[ grep(pattern="^ENSG00000101210", rownames(male_tumor_HCV_expression)), ])
EEF1A2data_7 <- as.numeric(male_adjacent_HCV_expression[ grep(pattern="^ENSG00000101210", rownames(male_adjacent_HCV_expression)), ])
pdf("~/R/EEF1A2_expression_boxplot.pdf", width=12, height=12)
boxplot(log10(EEF1A2data_0+0.01), log10(EEF1A2data_1+0.01), log10(EEF1A2data_2+0.01), log10(EEF1A2data_3+0.01), log10(EEF1A2data_4+0.01), log10(EEF1A2data_5+0.01), log10(EEF1A2data_6+0.01), log10(EEF1A2data_7+0.01), main= "Expression Levels of EEF1A2", xlab= "Samples", ylab= "Expression Levels", horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Male Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male Tumor-adjacent HCV"))

```

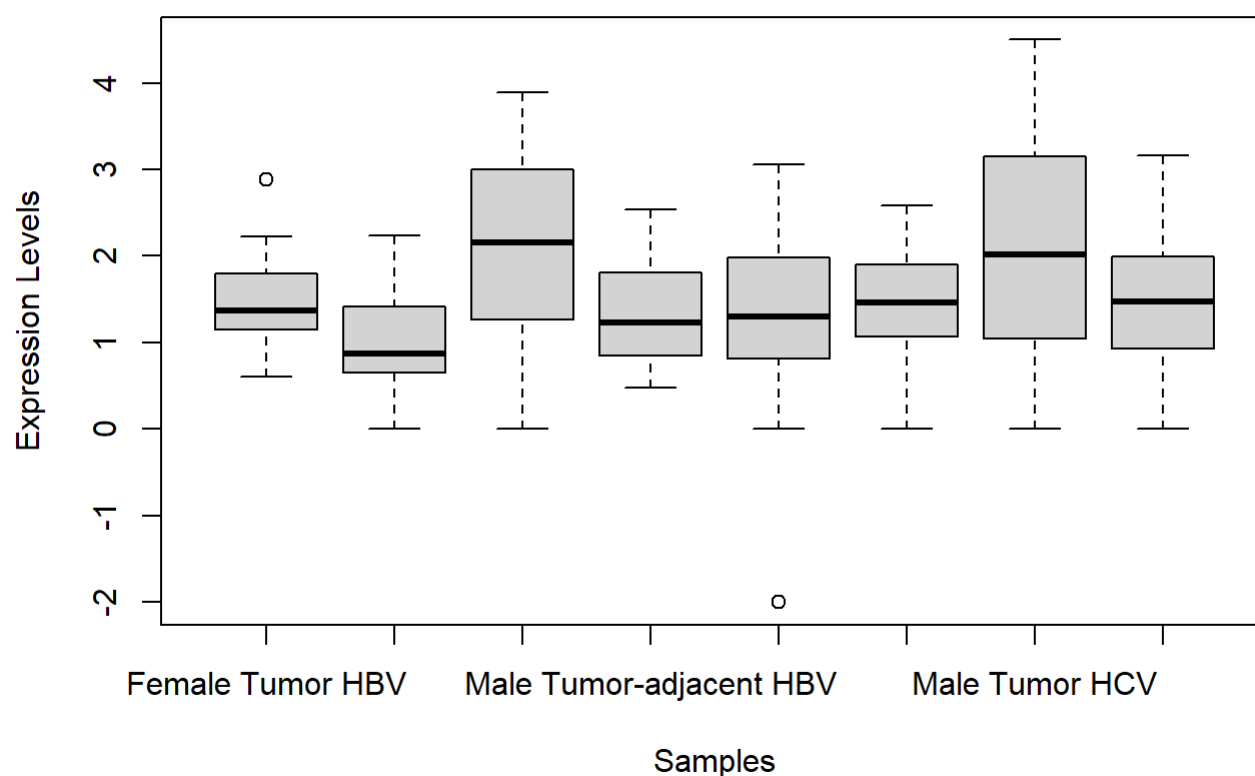
The box plot is the same as up above this chunk is just printing it out to the RMD file.

```

boxplot(log10(EEF1A2data_0+0.01), log10(EEF1A2data_1+0.01), log10(EEF1A2data_2+0.01), log10(EEF1A2data_3+0.01), log10(EEF1A2data_4+0.01), log10(EEF1A2data_5+0.01), log10(EEF1A2data_6+0.01), log10(EEF1A2data_7+0.01), main= "Expression Levels of EEF1A2", xlab= "Samples", ylab= "Expression Levels", horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Male Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male Tumor-adjacent HCV"))

```

Expression Levels of EEF1A2



###CDHR2 Box Plot

This section is dedicated to creating and outputting a boxplot for the expression levels of the CDHR2 gene.

This box plot has four categories: female tumor, female tumor-adjacent, male tumor, male tumor-adjacent on the same plot. The CDHR2_data_0 variable has all of samples that express the CDHR2 gene in female tumor samples. The grep function was used to retrieve all of the rows in the female_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

CDHR2data_1 is the same as the CDHR2data_0 except that it is dealing with the female tumor-adjacent samples. The grep function was used to retrieve all of the rows in the female_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

CDHR2data_2 is the same as the CDHR2data_1 except that it is dealing with the male tumor samples. The grep function was used to retrieve all of the rows in the male_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

CDHR2data_3 is the same as the CDHR2data_2 except that it is dealing with the male tumor-adjacent samples. The grep function was used to retrieve all of the rows in the male_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

```

CDHR2data_0 <- as.numeric(female_tumor_HBV_expression[ grep(pattern="^ENSG00000074276", rownames
(female_tumor_HBV_expression)), ])
CDHR2data_1 <- as.numeric(female_adjacent_HBV_expression[ grep(pattern="^ENSG00000074276", rowna
mes(female_adjacent_HBV_expression)), ])
CDHR2data_2 <- as.numeric(male_tumor_HBV_expression[ grep(pattern="^ENSG00000074276", rownames(m
ale_tumor_HBV_expression)), ])
CDHR2data_3 <- as.numeric(male_adjacent_HBV_expression[ grep(pattern="^ENSG00000074276", rowname
s(male_adjacent_HBV_expression)), ])
CDHR2data_4 <- as.numeric(female_tumor_HCV_expression[ grep(pattern="^ENSG00000074276", rownames
(female_tumor_HCV_expression)), ])
CDHR2data_5 <- as.numeric(female_adjacent_HCV_expression[ grep(pattern="^ENSG00000074276", rowna
mes(female_adjacent_HCV_expression)), ])
CDHR2data_6 <- as.numeric(male_tumor_HCV_expression[ grep(pattern="^ENSG00000074276", rownames(m
ale_tumor_HCV_expression)), ])
CDHR2data_7 <- as.numeric(male_adjacent_HCV_expression[ grep(pattern="^ENSG00000074276", rowname
s(male_adjacent_HCV_expression)), ])

pdf("~/R/CDHR2_expression_boxplot.pdf", width=12, height=12)
boxplot(log10(CDHR2data_0+0.01), log10(CDHR2data_1+0.01), log10(CDHR2data_2+0.01), log10(CDHR2da
ta_3+0.01), log10(CDHR2data_4+0.01), log10(CDHR2data_5+0.01), log10(CDHR2data_6+0.01), log10(CDH
R2data_7+0.01), main= "Expression Levels of CDHR2", xlab= "Samples", ylab= "Expression Levels",
  horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Mal
e Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male
Tumor-adjacent HCV"))

```

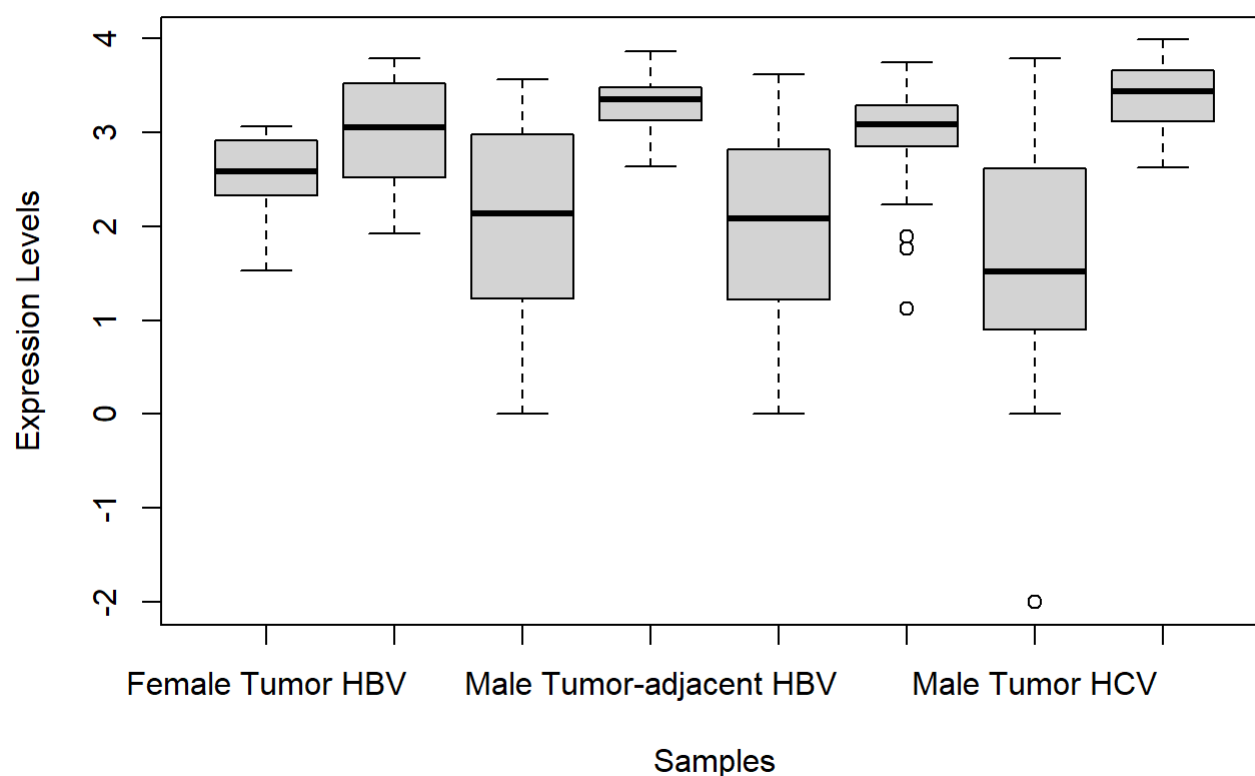
The box plot is the same as up above this chunk is just printing it out to the RMD file.

```

boxplot(log10(CDHR2data_0+0.01), log10(CDHR2data_1+0.01), log10(CDHR2data_2+0.01), log10(CDHR2da
ta_3+0.01), log10(CDHR2data_4+0.01), log10(CDHR2data_5+0.01), log10(CDHR2data_6+0.01), log10(CDH
R2data_7+0.01), main= "Expression Levels of CDHR2", xlab= "Samples", ylab= "Expression Levels",
  horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Mal
e Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male
Tumor-adjacent HCV"))

```

Expression Levels of CDHR2



###IGF2 Box Plot

This section is dedicated to creating and outputting a boxplot for expression levels of the IGF2 gene.

This box plot has four categories: female tumor, female tumor-adjacent, male tumor, male tumor-adjacent on the same plot. The IGF2_data_0 variable has all of samples that express the IGF2 gene in female tumor samples. The grep function was used to retrieve all of the rows in the female_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

IGF2data_1 is the same as the IGF2data_0 except that it is dealing with the female tumor-adjacent samples. The grep function was used to retrieve all of the rows in the female_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

IGF2data_2 is the same as the IGF2data_1 except that it is dealing with the male tumor samples. The grep function was used to retrieve all of the rows in the male_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

IGF2data_3 is the same as the IGF2data_2 except that it is dealing with the male tumor-adjacent samples. The grep function was used to retrieve all of the rows in the male_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

```

IGF2data_0 <- as.numeric(female_tumor_HBV_expression[ grep(pattern="^ENSG00000167244", rownames
(female_tumor_HBV_expression)), ])
IGF2data_1 <- as.numeric(female_adjacent_HBV_expression[ grep(pattern="^ENSG00000167244", rownam
es(female_adjacent_HBV_expression)), ])
IGF2data_2 <- as.numeric(male_tumor_HBV_expression[ grep(pattern="^ENSG00000167244", rownames(ma
le_tumor_HBV_expression)), ])
IGF2data_3 <- as.numeric(male_adjacent_HBV_expression[ grep(pattern="^ENSG00000167244", rownames
(male_adjacent_HBV_expression)), ])
IGF2data_4 <- as.numeric(female_tumor_HCV_expression[ grep(pattern="^ENSG00000167244", rownames
(female_tumor_HCV_expression)), ])
IGF2data_5 <- as.numeric(female_adjacent_HCV_expression[ grep(pattern="^ENSG00000167244", rownam
es(female_adjacent_HCV_expression)), ])
IGF2data_6 <- as.numeric(male_tumor_HCV_expression[ grep(pattern="^ENSG00000167244", rownames(ma
le_tumor_HCV_expression)), ])
IGF2data_7 <- as.numeric(male_adjacent_HCV_expression[ grep(pattern="^ENSG00000167244", rownames
(male_adjacent_HCV_expression)), ])
pdf("~/R/IGF2_expression_boxplot.pdf", width=12, height=12)
boxplot(log10(IGF2data_0+0.01), log10(IGF2data_1+0.01), log10(IGF2data_2+0.01), log10(IGF2data_3
+0.01), log10(IGF2data_4+0.01), log10(IGF2data_5+0.01), log10(IGF2data_6+0.01), log10(IGF2data_7
+0.01), main= "Expression Levels of IGF2", xlab= "Samples", ylab= "Expression Levels", horizonta
l= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Male Tumor-ad
jacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male Tumor-adja
cent HCV"))

```

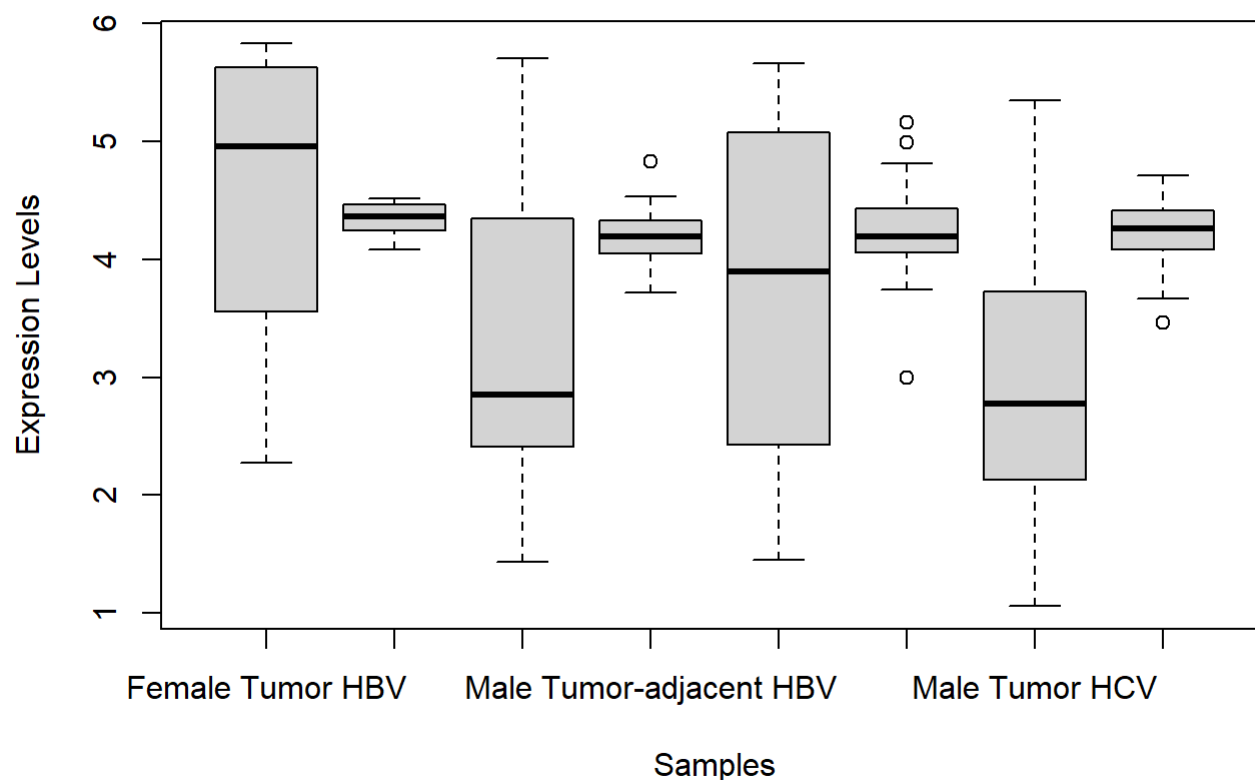
The box plot is the same as up above this chunk is just printing it out to the RMD file.

```

boxplot(log10(IGF2data_0+0.01), log10(IGF2data_1+0.01), log10(IGF2data_2+0.01), log10(IGF2data_3
+0.01), log10(IGF2data_4+0.01), log10(IGF2data_5+0.01), log10(IGF2data_6+0.01), log10(IGF2data_7
+0.01), main= "Expression Levels of IGF2", xlab= "Samples", ylab= "Expression Levels", horizonta
l= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Male Tumor-ad
jacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male Tumor-adja
cent HCV"))

```


Expression Levels of IGF2



###H19 Box Plot

This section is dedicated to creating and outputting a boxplot for the expression levels of the H19 gene.

This box plot has four categories: female tumor, female tumor-adjacent, male tumor, male tumor-adjacent on the same plot. The H19_data_0 variable has all of samples that express the H19 gene in female tumor samples. The grep function was used to retrieve all of the rows in the female_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

H19data_1 is the same as the H19data_0 except that it is dealing with the female tumor-adjacent samples. The grep function was used to retrieve all of the rows in the female_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

H19data_2 is the same as the H19data_1 except that it is dealing with the male tumor samples. The grep function was used to retrieve all of the rows in the male_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

H19data_3 is the same as the H19data_2 except that it is dealing with the male tumor-adjacent samples. The grep function was used to retrieve all of the rows in the male_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

```

H19data_0 <- as.numeric(female_tumor_HBV_expression[ grep(pattern="^ENSG00000130600", rownames(female_tumor_HBV_expression)), ])
H19data_1 <- as.numeric(female_adjacent_HBV_expression[ grep(pattern="^ENSG00000130600", rownames(female_adjacent_HBV_expression)), ])
H19data_2 <- as.numeric(male_tumor_HBV_expression[ grep(pattern="^ENSG00000130600", rownames(male_tumor_HBV_expression)), ])
H19data_3 <- as.numeric(male_adjacent_HBV_expression[ grep(pattern="^ENSG00000130600", rownames(male_adjacent_HBV_expression)), ])
H19data_4 <- as.numeric(female_tumor_HCV_expression[ grep(pattern="^ENSG00000130600", rownames(female_tumor_HCV_expression)), ])
H19data_5 <- as.numeric(female_adjacent_HCV_expression[ grep(pattern="^ENSG00000130600", rownames(female_adjacent_HCV_expression)), ])
H19data_6 <- as.numeric(male_tumor_HCV_expression[ grep(pattern="^ENSG00000130600", rownames(male_tumor_HCV_expression)), ])
H19data_7 <- as.numeric(male_adjacent_HCV_expression[ grep(pattern="^ENSG00000130600", rownames(male_adjacent_HCV_expression)), ])

pdf("~/R/H19_expression_boxplot.pdf", width=12, height=12)
boxplot(log10(H19data_0+0.01), log10(H19data_1+0.01), log10(H19data_2+0.01), log10(H19data_3+0.01), log10(H19data_4+0.01), log10(H19data_5+0.01), log10(H19data_6+0.01), log10(H19data_7+0.01),
  main= "Expression Levels of H19", xlab= "Samples", ylab= "Expression Levels", horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Male Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male Tumor-adjacent HCV"))

```

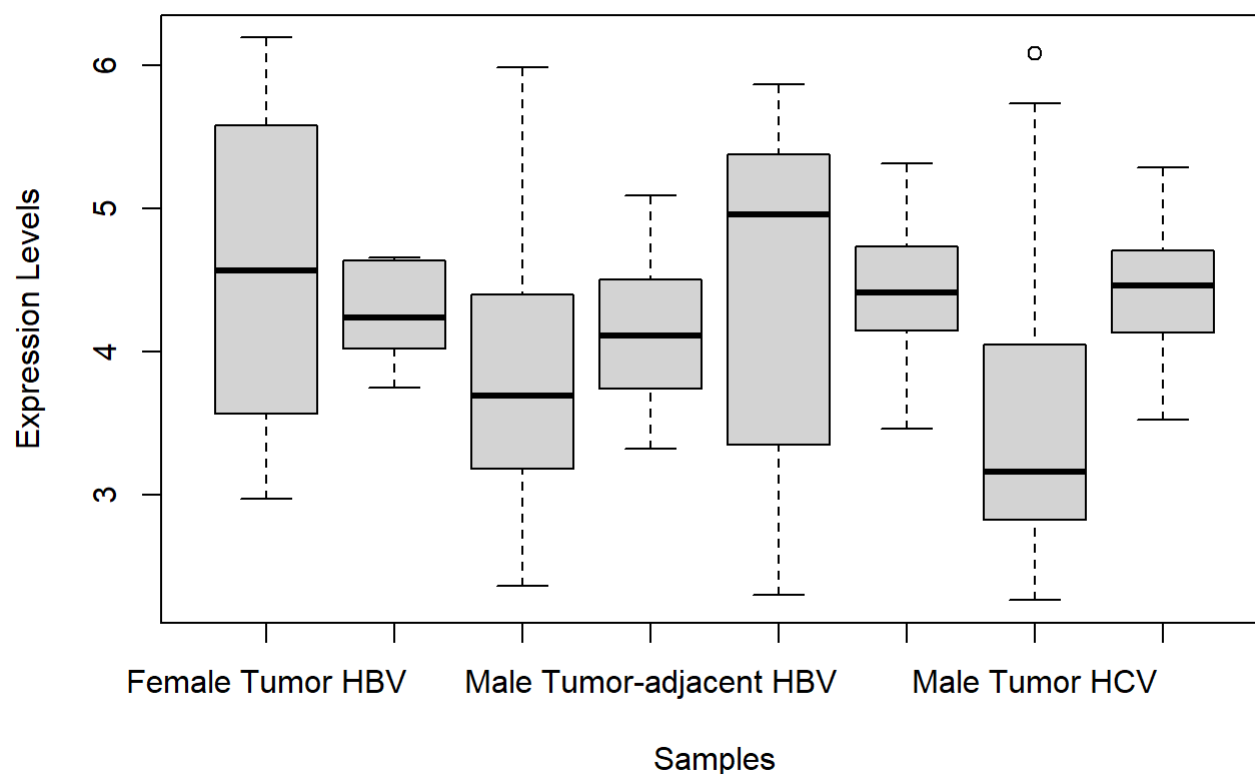
The box plot is the same as up above this chunk is just printing it out to the RMD file.

```

boxplot(log10(H19data_0+0.01), log10(H19data_1+0.01), log10(H19data_2+0.01), log10(H19data_3+0.01), log10(H19data_4+0.01), log10(H19data_5+0.01), log10(H19data_6+0.01), log10(H19data_7+0.01),
  main= "Expression Levels of H19", xlab= "Samples", ylab= "Expression Levels", horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Male Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male Tumor-adjacent HCV"))

```

Expression Levels of H19



###MEP1A Box Plot

This section is dedicated to creating and outputting a boxplot for the expression levels of the MEP1A gene.

This box plot has four categories: female tumor, female tumor-adjacent, male tumor, male tumor-adjacent on the same plot. The MEP1A_data_0 variable has all of samples that express the MEP1A gene in female tumor samples. The grep function was used to retrieve all of the rows in the female_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

MEP1A_1 is the same as the MEP1Adata_0 except that it is dealing with the female tumor-adjacent samples. The grep function was used to retrieve all of the rows in the female_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

MEP1Adata_2 is the same as the MEP1Adata_1 except that it is dealing with the male tumor samples. The grep function was used to retrieve all of the rows in the male_tumor_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

MEP1Adata_3 is the same as the MEP1Adata_2 except that it is dealing with the male tumor-adjacent samples. The grep function was used to retrieve all of the rows in the male_adjacent_expression file that start with the ensembl gene version. The variable is then changed to report all numeric values.

```

MEP1Adata_0 <- as.numeric(female_tumor_HBV_expression[ grep(pattern="^ENSG00000112818", rownames
(female_tumor_HBV_expression)), ])
MEP1Adata_1 <- as.numeric(female_adjacent_HBV_expression[ grep(pattern="^ENSG00000112818", rowna
mes(female_adjacent_HBV_expression)), ])
MEP1Adata_2 <- as.numeric(male_tumor_HBV_expression[ grep(pattern="^ENSG00000112818", rownames(m
ale_tumor_HBV_expression)), ])
MEP1Adata_3 <- as.numeric(male_adjacent_HBV_expression[ grep(pattern="^ENSG00000112818", rowname
s(male_adjacent_HBV_expression)), ])
MEP1Adata_4 <- as.numeric(female_tumor_HCV_expression[ grep(pattern="^ENSG00000112818", rownames
(female_tumor_HCV_expression)), ])
MEP1Adata_5 <- as.numeric(female_adjacent_HCV_expression[ grep(pattern="^ENSG00000112818", rowna
mes(female_adjacent_HCV_expression)), ])
MEP1Adata_6 <- as.numeric(male_tumor_HCV_expression[ grep(pattern="^ENSG00000112818", rownames(m
ale_tumor_HCV_expression)), ])
MEP1Adata_7 <- as.numeric(male_adjacent_HCV_expression[ grep(pattern="^ENSG00000112818", rowname
s(male_adjacent_HCV_expression)), ])

pdf("~/R/MEP1A_expression_boxplot.pdf", width=12, height=12)
boxplot(log10(MEP1Adata_0+0.01), log10(MEP1Adata_1+0.01), log10(MEP1Adata_2+0.01), log10(MEP1Ada
ta_3+0.01), log10(MEP1Adata_4+0.01), log10(MEP1Adata_5+0.01), log10(MEP1Adata_6+0.01), log10(MEP
1Adata_7+0.01), main= "Expression Levels of MEP1A", xlab= "Samples", ylab= "Expression Levels",
  horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Mal
e Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male
Tumor-adjacent HCV"))

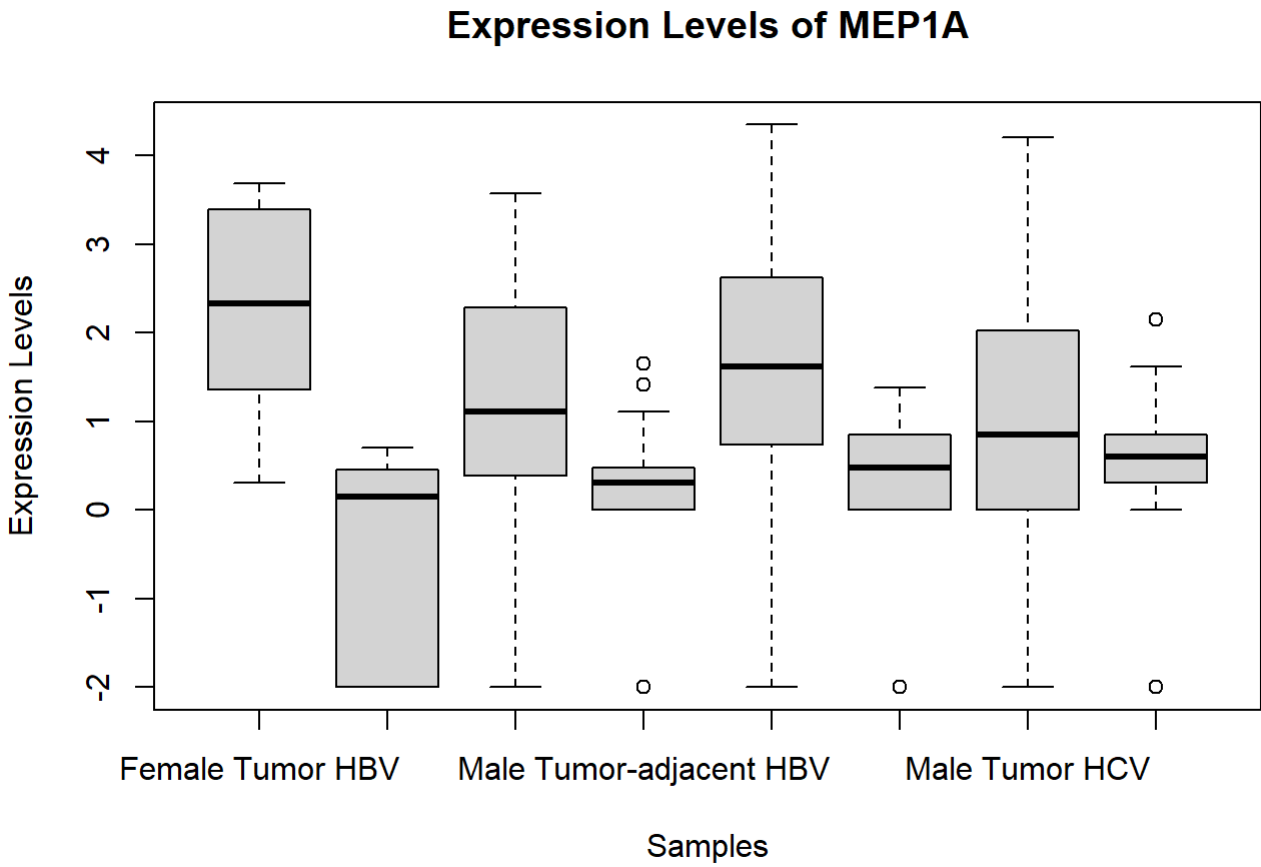
```

The box plot is the same as up above this chunk is just printing it out to the RMD file.

```

boxplot(log10(MEP1Adata_0+0.01), log10(MEP1Adata_1+0.01), log10(MEP1Adata_2+0.01), log10(MEP1Ada
ta_3+0.01), log10(MEP1Adata_4+0.01), log10(MEP1Adata_5+0.01), log10(MEP1Adata_6+0.01), log10(MEP
1Adata_7+0.01), main= "Expression Levels of MEP1A", xlab= "Samples", ylab= "Expression Levels",
  horizontal= F, names= c("Female Tumor HBV", "Female Tumor-adjacent HBV", "Male Tumor HBV", "Mal
e Tumor-adjacent HBV", "Female Tumor HCV", "Female Tumor-adjacent HCV", "Male Tumor HCV", "Male
Tumor-adjacent HCV"))

```



Violin Plot

```

#Female HBV tumor adjacent (data_1) tumor(data_0)
MUC13data_1 <- data.frame(MUC13data_1,1:8)
MUC13data_0 <-data.frame(MUC13data_0,1:9)

#Male HBV tumor adjacent(data_3) tumor(data_2)
MUC13data_3<-data.frame(MUC13data_3,1:33)
MUC13data_2<-data.frame(MUC13data_2,1:40)

#Female HCV tumor adjacent (data_5) tumor (data_4)
MUC13data_5<-data.frame(MUC13data_5,1:34)
MUC13data_4<-data.frame(MUC13data_4,1:36)

#Male HCV tumor adjacent (data_7) tumor(data_6)
MUC13data_7<-data.frame(MUC13data_7,1:59)
MUC13data_6<- data.frame(MUC13data_6,1:71)

#Remaning columns

#Female HBV
colnames(MUC13data_1)<- c("Expression","ind")
colnames(MUC13data_0)<-c("Expression","ind")

#Male HBV
colnames(MUC13data_3)<- c("Expression","ind")
colnames(MUC13data_2)<-c("Expression","ind")

#Female HCV
colnames(MUC13data_5)<-c("Expression","ind")
colnames(MUC13data_4)<-c("Expression","ind")

#Male HCV
colnames(MUC13data_7)<- c("Expression","ind")
colnames(MUC13data_6)<-c("Expression","ind")

#Plotting

#Female HBV
p <- ggplot(MUC13data_1, aes(x=ind, y=Expression)) + geom_violin()
p1<- ggplot(MUC13data_0,aes(x= ind, y=Expression)) + geom_violin()

#Male HBV
p2<- ggplot(MUC13data_3,aes(x=ind, y=Expression)) +geom_violin()
p20<-ggplot(MUC13data_2,aes(x=ind, y=Expression))+geom_violin()

#Female HCV
p3<- ggplot(MUC13data_5,aes(x=ind, y=Expression)) + geom_violin()
p30<-ggplot(MUC13data_4,aes(x=ind, y=Expression)) + geom_violin()

#Male HCV
p4<-ggplot(MUC13data_7, aes(x=ind, y=Expression)) + geom_violin()
p40<-ggplot(MUC13data_6, aes(x=ind, y=Expression)) + geom_violin()

```

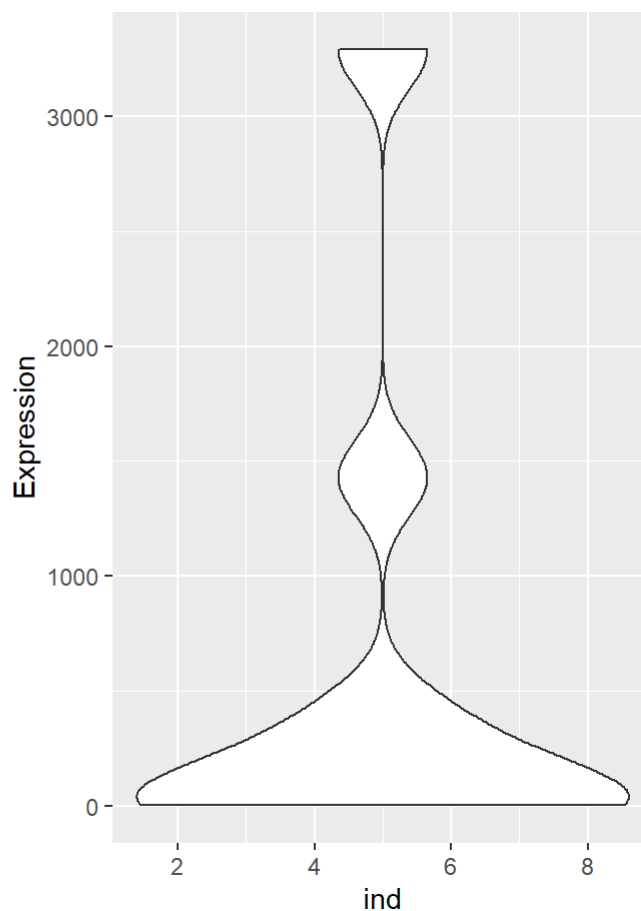
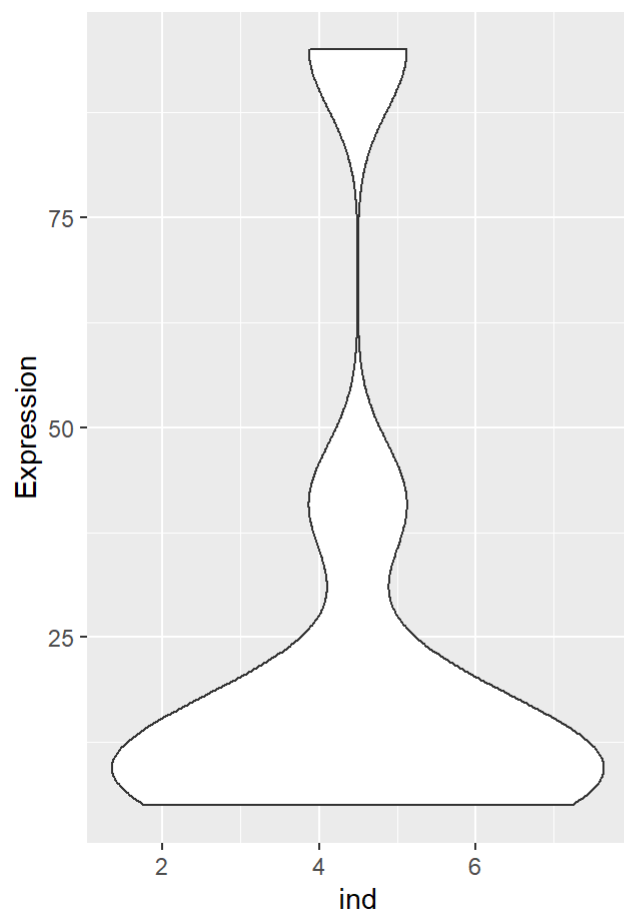
```
p0 <- p+p1
pdf("~/R/MUC13_expression_violin_F_HBV.pdf", width=12, height=12)
p0
```

```
p21<- p20+p2
pdf("~/R/MUC13_expression_violin_M_HBV.pdf", width=12, height=12)
p21
```

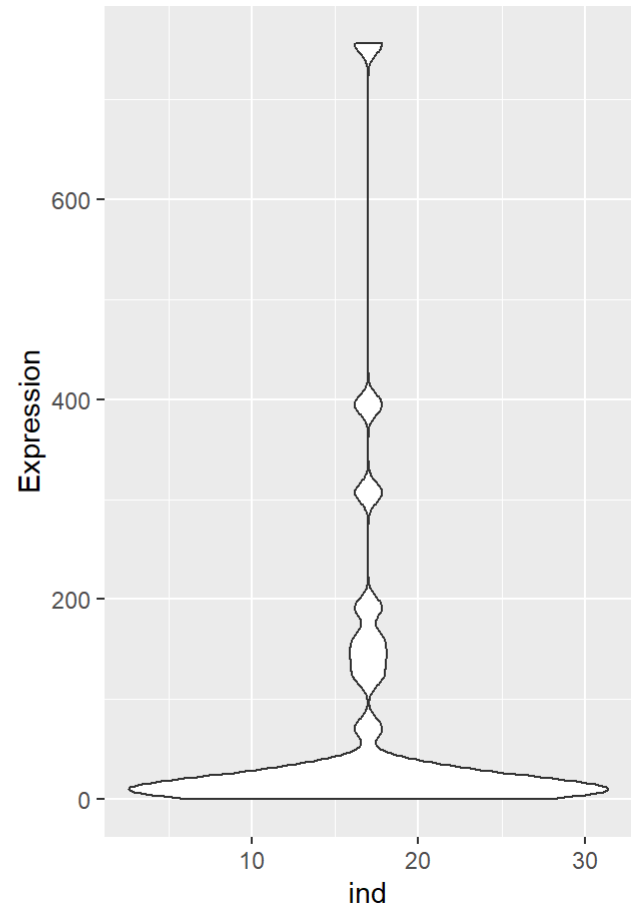
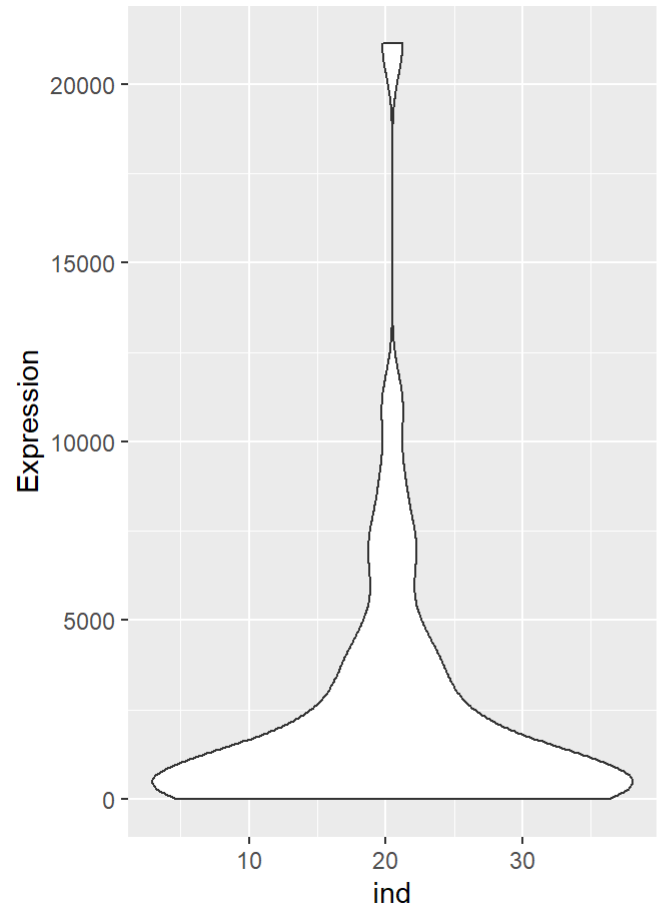
```
p31<-p30+p3
pdf("~/R/MUC13_expression_violin_F_HCV.pdf", width=12, height=12)
p31
```

```
p41<-p40+p4
pdf("~/R/MUC13_expression_violin_M_HCV.pdf", width=12, height=12)
p41
```

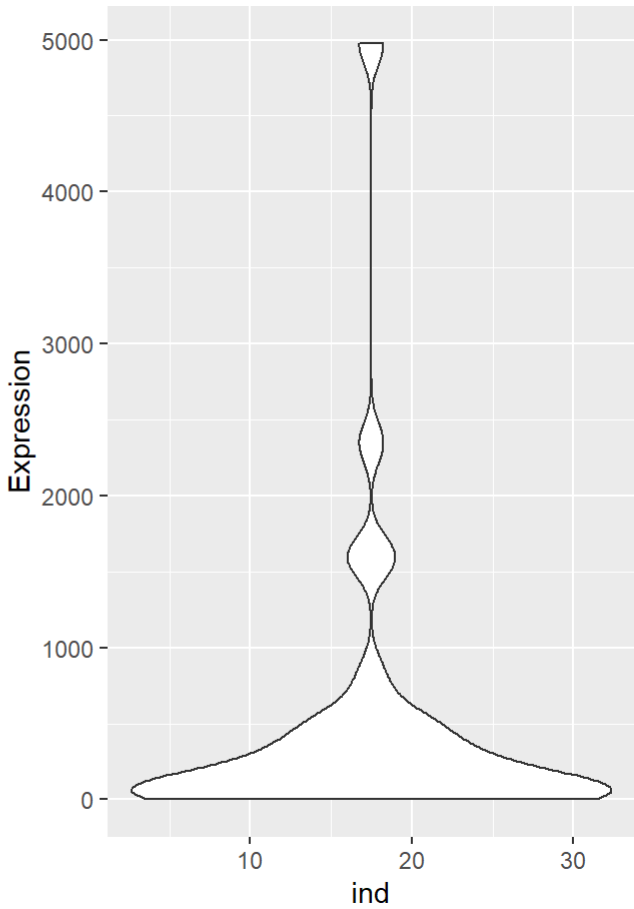
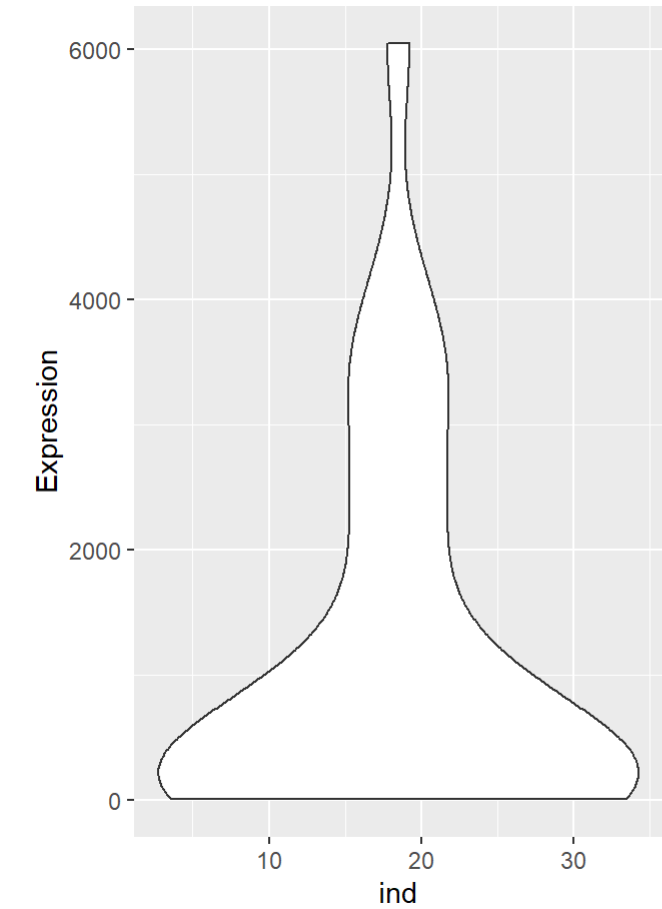
p0



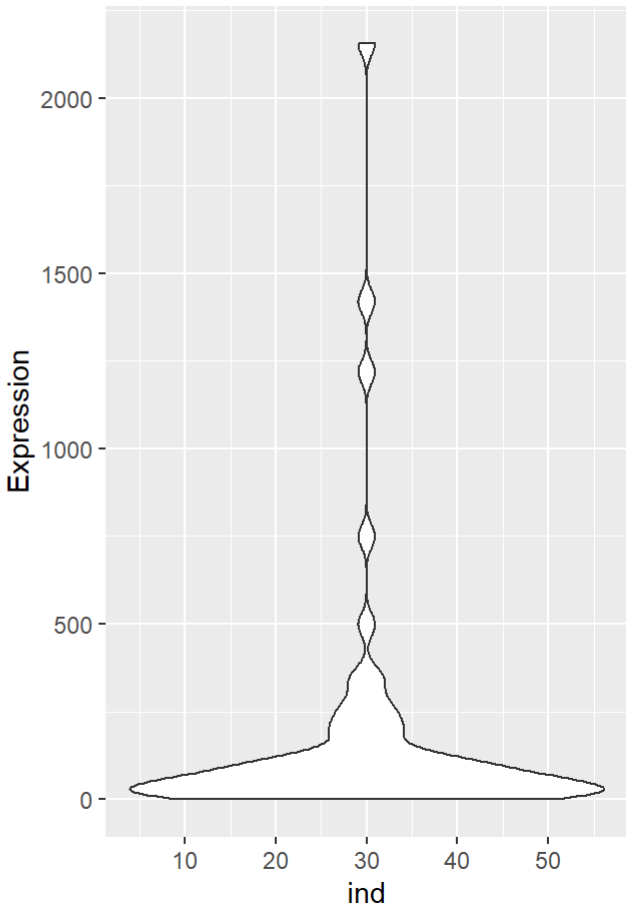
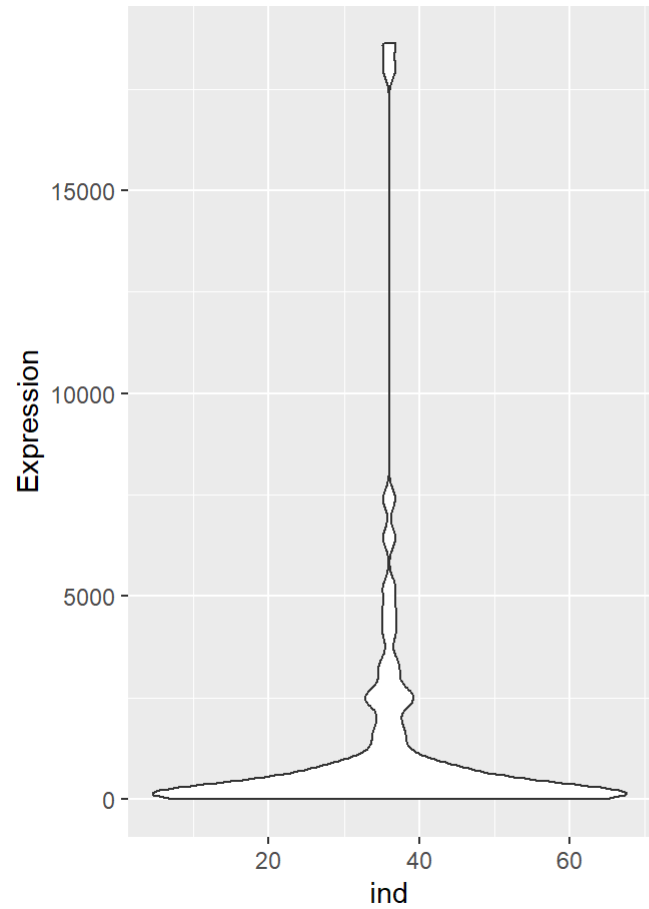
p21



p31



p41



```

#Female HBV Tumor-adjacent (data_1) tumor (data_0)
MAGEA6data_1<- data.frame(MAGEA6data_1, 1:8)
MAGEA6data_0<-data.frame(MAGEA6data_0,1:9)

#Male HBV Tumor-adjacent(data_3) tumor (data_2)
MAGEA6data_3<-data.frame(MAGEA6data_3,1:33)
MAGEA6data_2<-data.frame(MAGEA6data_2,1:40)

#Female HCV Tumor-adjacent (data_5) tumor(data_4)
MAGEA6data_5<-data.frame(MAGEA6data_5,1:34)
MAGEA6data_4<-data.frame(MAGEA6data_4,1:36)

#Male HCV Tumor-adjacent (data_7) tumor(data_6)
MAGEA6data_7<-data.frame(MAGEA6data_7,1:59)
MAGEA6data_6<-data.frame(MAGEA6data_6,1:71)
#Female HBV
colnames(MAGEA6data_1)<- c("Expression","ind")
colnames(MAGEA6data_0)<- c("Expression","ind")

#Male HBV
colnames(MAGEA6data_3)<- c("Expression","ind")
colnames(MAGEA6data_2)<- c("Expression","ind")

#Female HCV
colnames(MAGEA6data_5)<- c("Expression","ind")
colnames(MAGEA6data_4)<- c("Expression","ind")

#Male HCV
colnames(MAGEA6data_7)<- c("Expression","ind")
colnames(MAGEA6data_6)<- c("Expression","ind")

#Female HBV
p<- ggplot(MAGEA6data_1, aes(x=ind, y=Expression)) + geom_violin()
p1<- ggplot(MAGEA6data_0, aes(x=ind, y=Expression)) + geom_violin()

#Male HBV
p2<- ggplot(MAGEA6data_3, aes(x=ind, y=Expression)) + geom_violin()
p20<- ggplot(MAGEA6data_2, aes(x=ind, y=Expression)) + geom_violin()

#Female HCV
p3<- ggplot(MAGEA6data_5, aes(x=ind, y=Expression)) + geom_violin()
p30<- ggplot(MAGEA6data_4, aes(x=ind, y=Expression)) + geom_violin()

#Male HCV
p4<-ggplot(MAGEA6data_7, aes(x=ind, y=Expression)) + geom_violin()
p40<- ggplot(MAGEA6data_6, aes(x=ind, y=Expression)) + geom_violin()

```

```

p0<-p+p1
pdf("~/R/MAGEA6_expression_violin_F_HBV.pdf", width=12, height=12)
p0

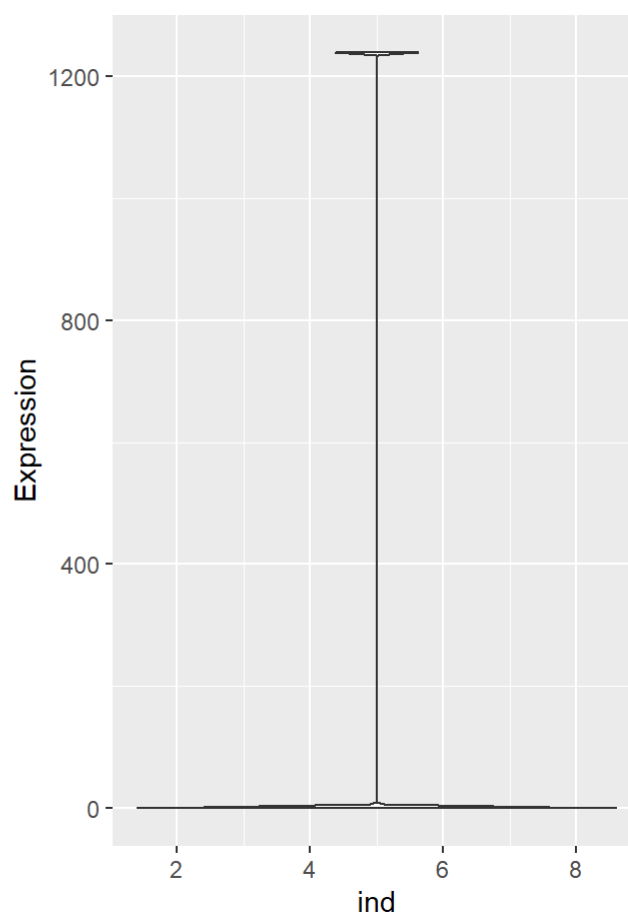
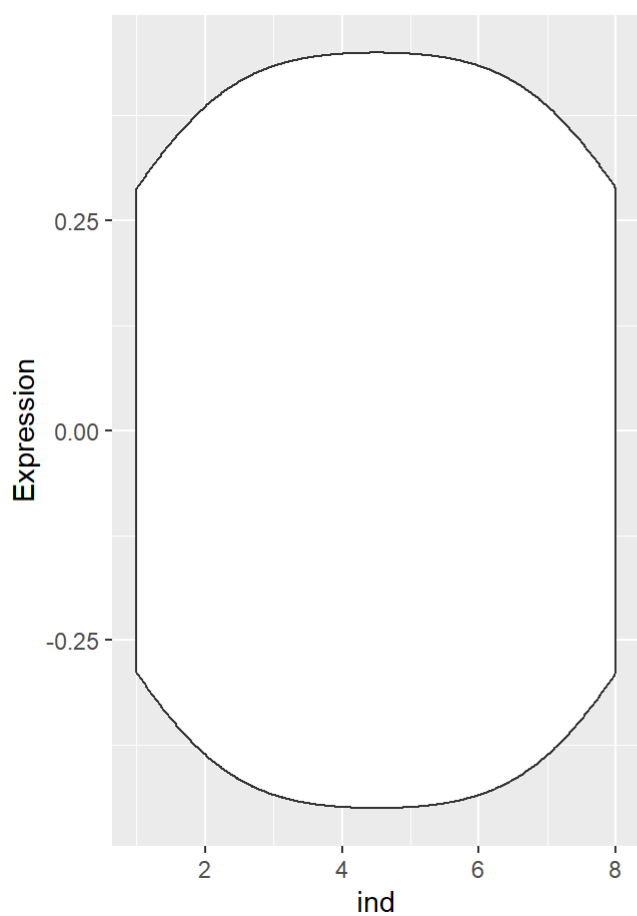
```

```
p21<-p2+p20
pdf("~/R/MAGEA6_expression_violin_M_HBV.pdf", width=12, height=12)
p21
```

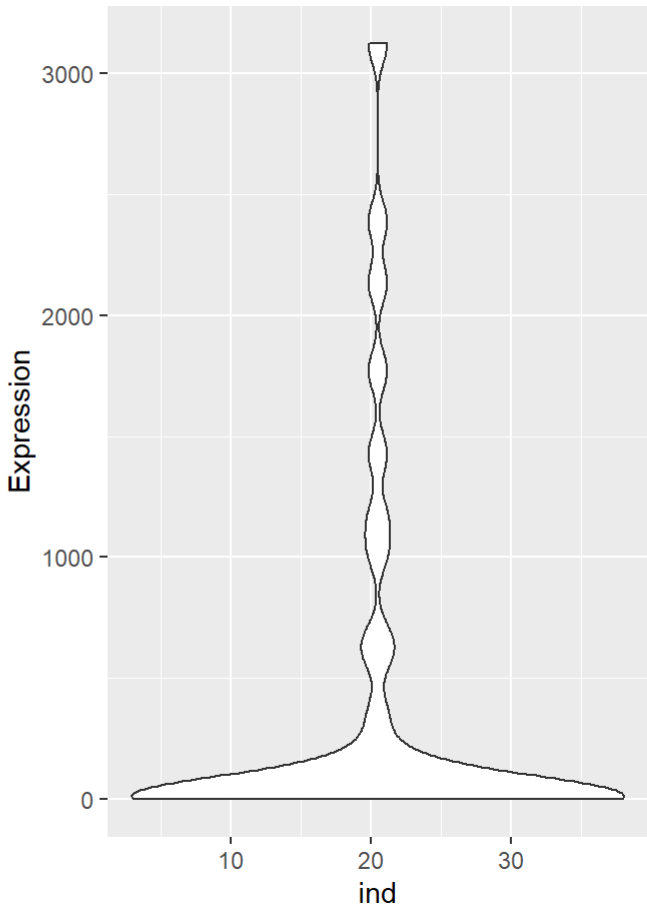
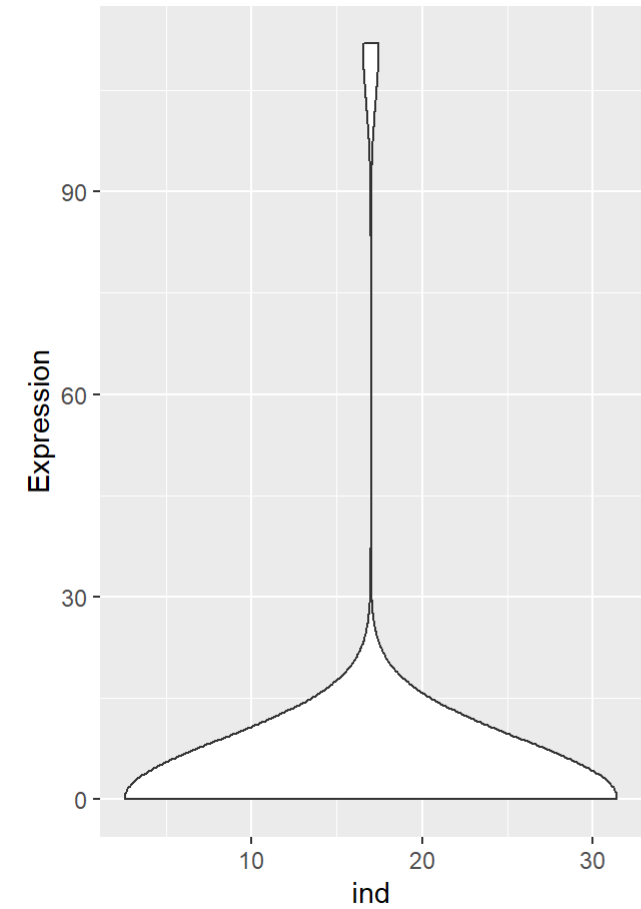
```
p31<-p3+p30
pdf("~/R/MAGEA6_expression_violin_F_HCV.pdf", width=12, height=12)
p31
```

```
p41<-p4+p40
pdf("~/R/MAGEA6_expression_violin_M_HCV.pdf", width=12, height=12)
p41
```

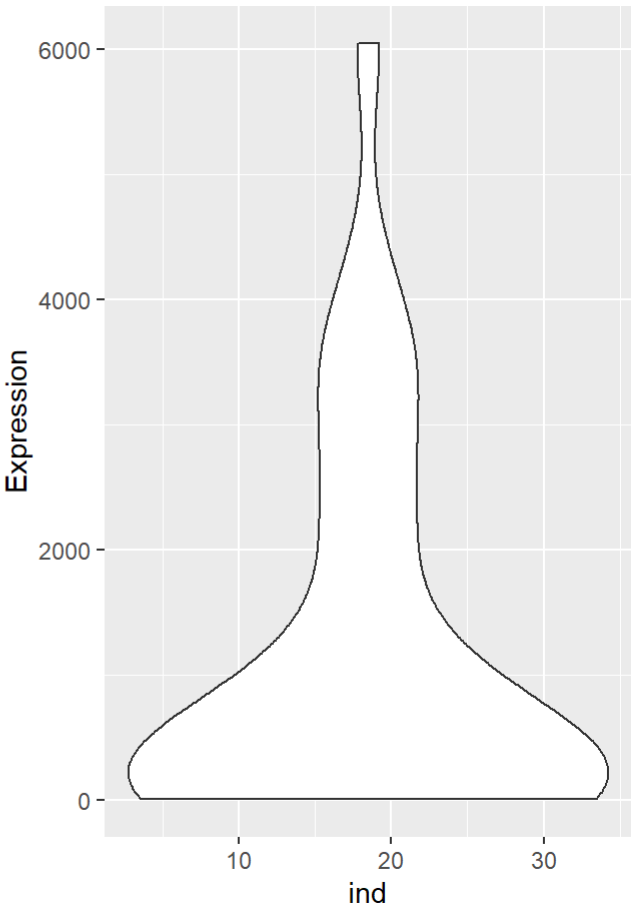
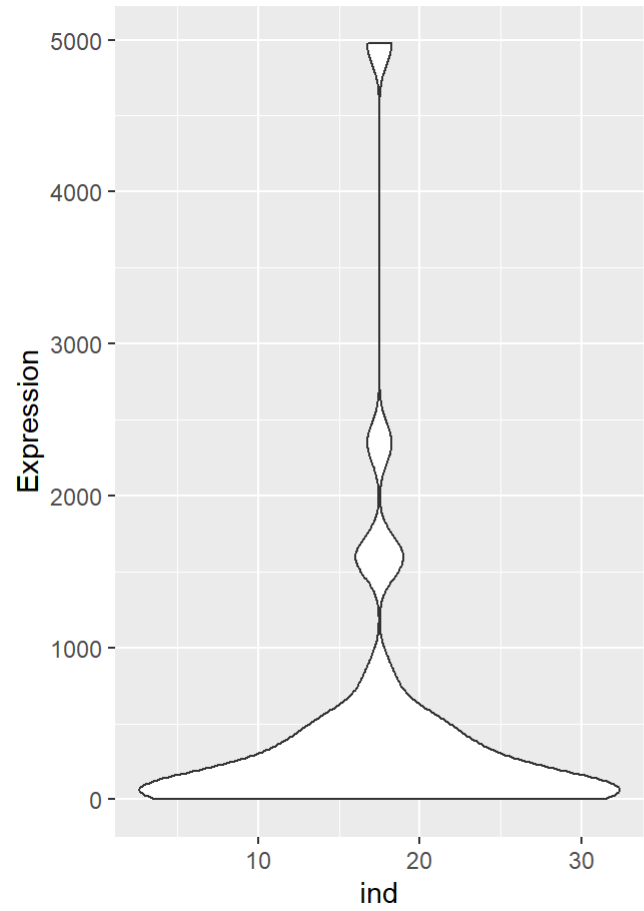
```
p0
```



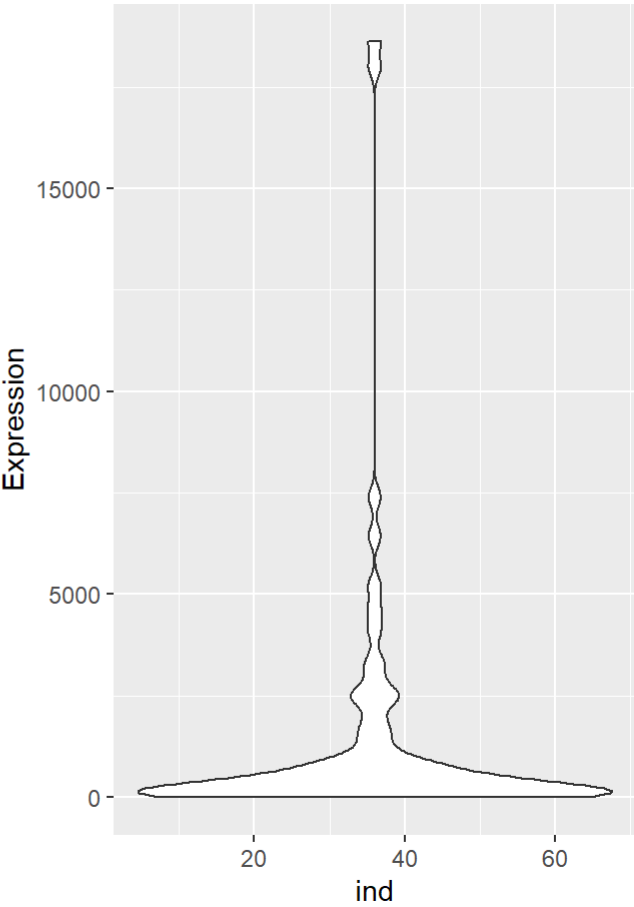
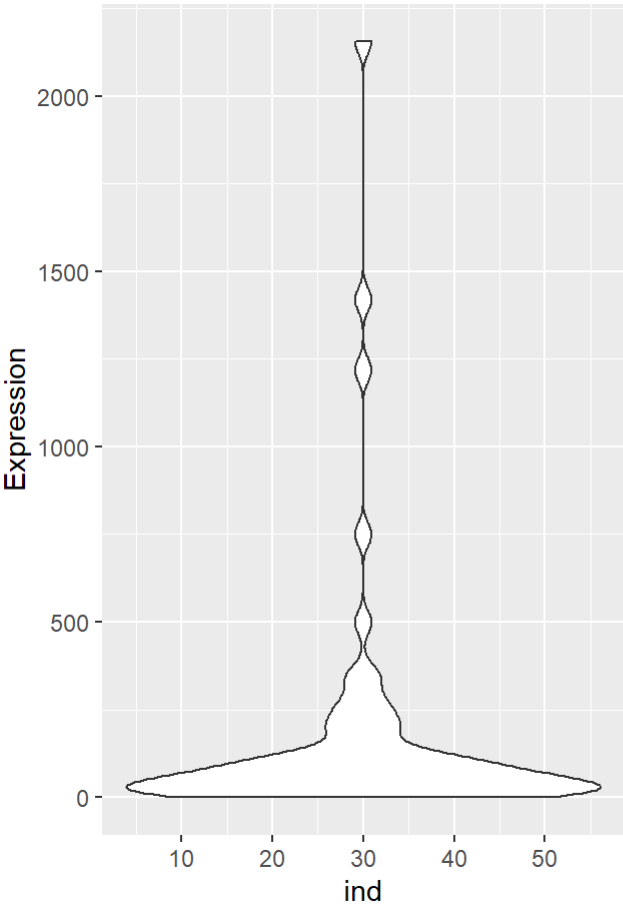
```
p21
```



p31



p41



```

#Female HBV Tumor-adjacent (data_1) tumor (data_0)
H19data_1<- data.frame(H19data_1, 1:8)
H19data_0<-data.frame(H19data_0,1:9)

#Male HBV Tumor-adjacent(data_3) tumor (data_2)
H19data_3<-data.frame(H19data_3,1:33)
H19data_2<-data.frame(H19data_2,1:40)

#Female HCV Tumor-adjacent (data_5) tumor(data_4)
H19data_5<-data.frame(H19data_5,1:34)
H19data_4<-data.frame(H19data_4,1:36)

#Male HCV Tumor-adjacent (data_7) tumor(data_6)
H19data_7<-data.frame(H19data_7,1:59)
H19data_6<-data.frame(H19data_6,1:71)
#Female HBV
colnames(H19data_1)<- c("Expression","ind")
colnames(H19data_0)<- c("Expression","ind")

#Male HBV
colnames(H19data_3)<- c("Expression","ind")
colnames(H19data_2)<- c("Expression","ind")

#Female HCV
colnames(H19data_5)<- c("Expression","ind")
colnames(H19data_4)<- c("Expression","ind")

#Male HCV
colnames(H19data_7)<- c("Expression","ind")
colnames(H19data_6)<- c("Expression","ind")

#Female HBV
p<- ggplot(H19data_1, aes(x=ind, y=Expression)) + geom_violin()
p1<- ggplot(H19data_0, aes(x=ind, y=Expression)) + geom_violin()

#Male HBV
p2<- ggplot(H19data_3, aes(x=ind, y=Expression)) + geom_violin()
p20<- ggplot(H19data_2, aes(x=ind, y=Expression)) + geom_violin()

#Female HCV
p3<- ggplot(H19data_5, aes(x=ind, y=Expression)) + geom_violin()
p30<- ggplot(H19data_4, aes(x=ind, y=Expression)) + geom_violin()

#Male HCV
p4<- ggplot(H19data_7, aes(x=ind, y=Expression)) + geom_violin()
p40<- ggplot(H19data_6, aes(x=ind, y=Expression)) + geom_violin()

```

```

p0<-p+p1
pdf("~/R/H19_expression_violin_F_HBV.pdf", width=12, height=12)
p0

```

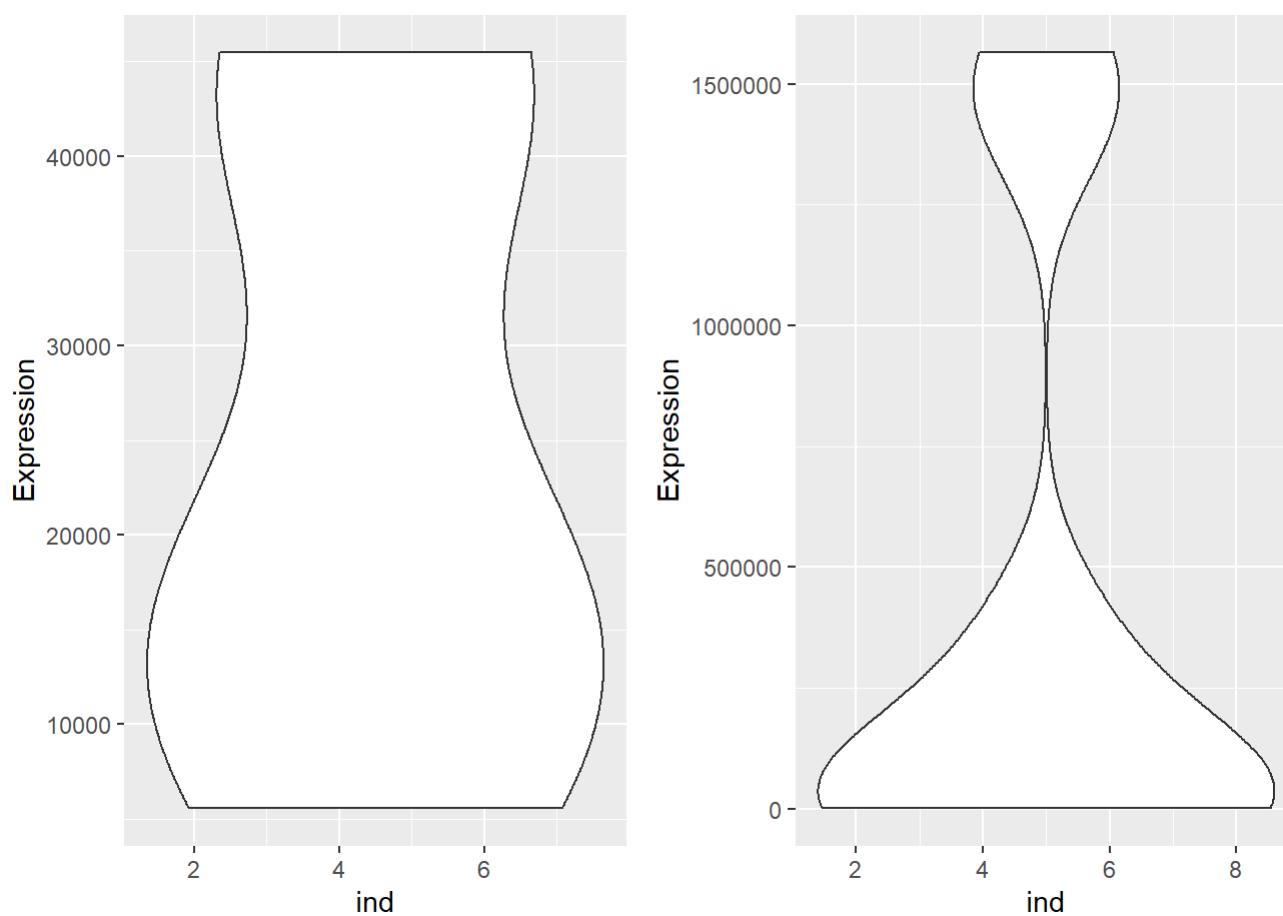


```
p21<-p2+p20  
pdf("~/R/H19_expression_violin_M_HBV.pdf", width=12, height=12)  
p21
```

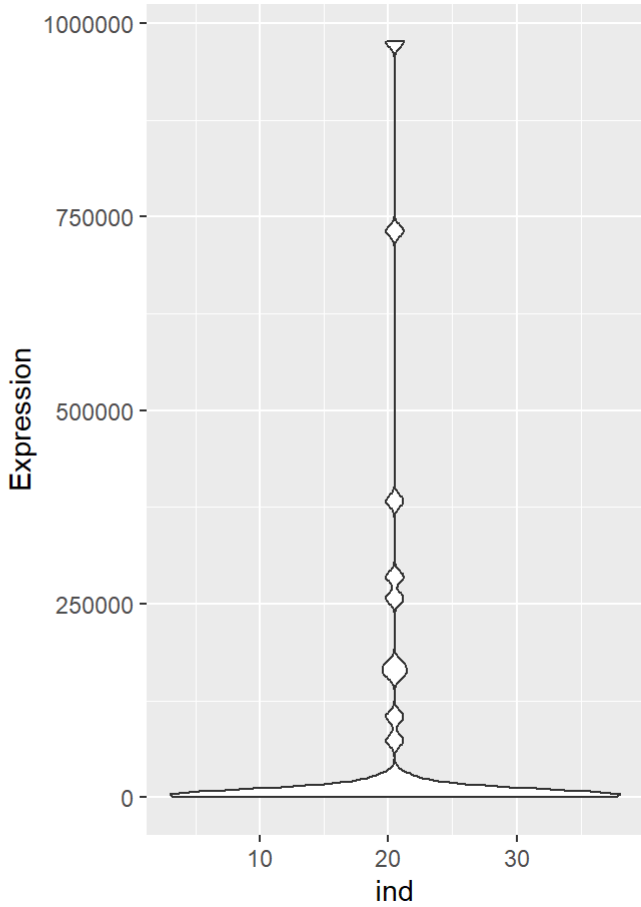
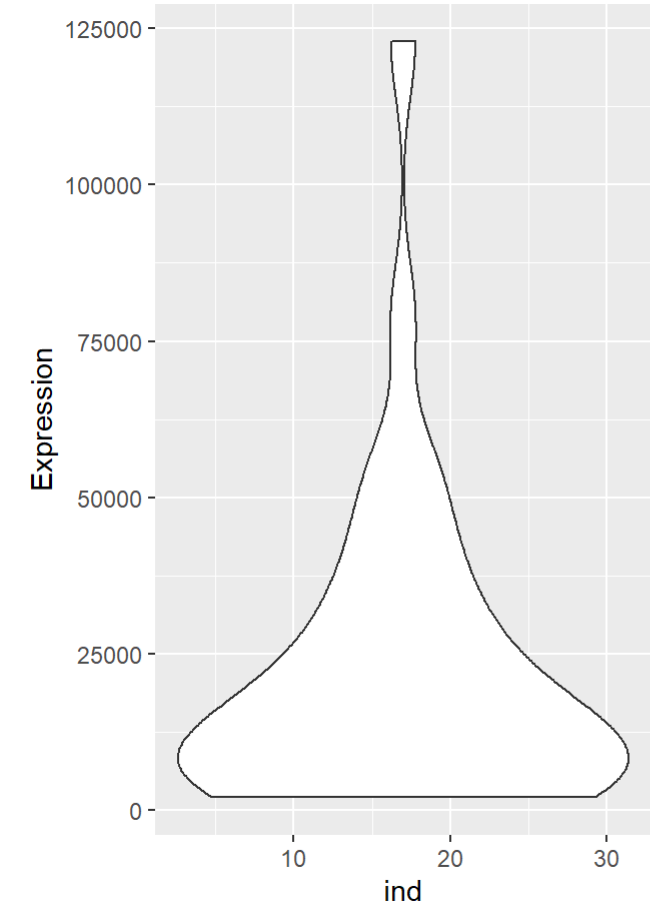
```
p31<-p3+p30  
pdf("~/R/H19_expression_violin_F_HCV.pdf", width=12, height=12)  
p31
```

```
p41<-p4+p40  
pdf("~/R/H19_expression_violin_M_HCV.pdf", width=12, height=12)  
p41
```

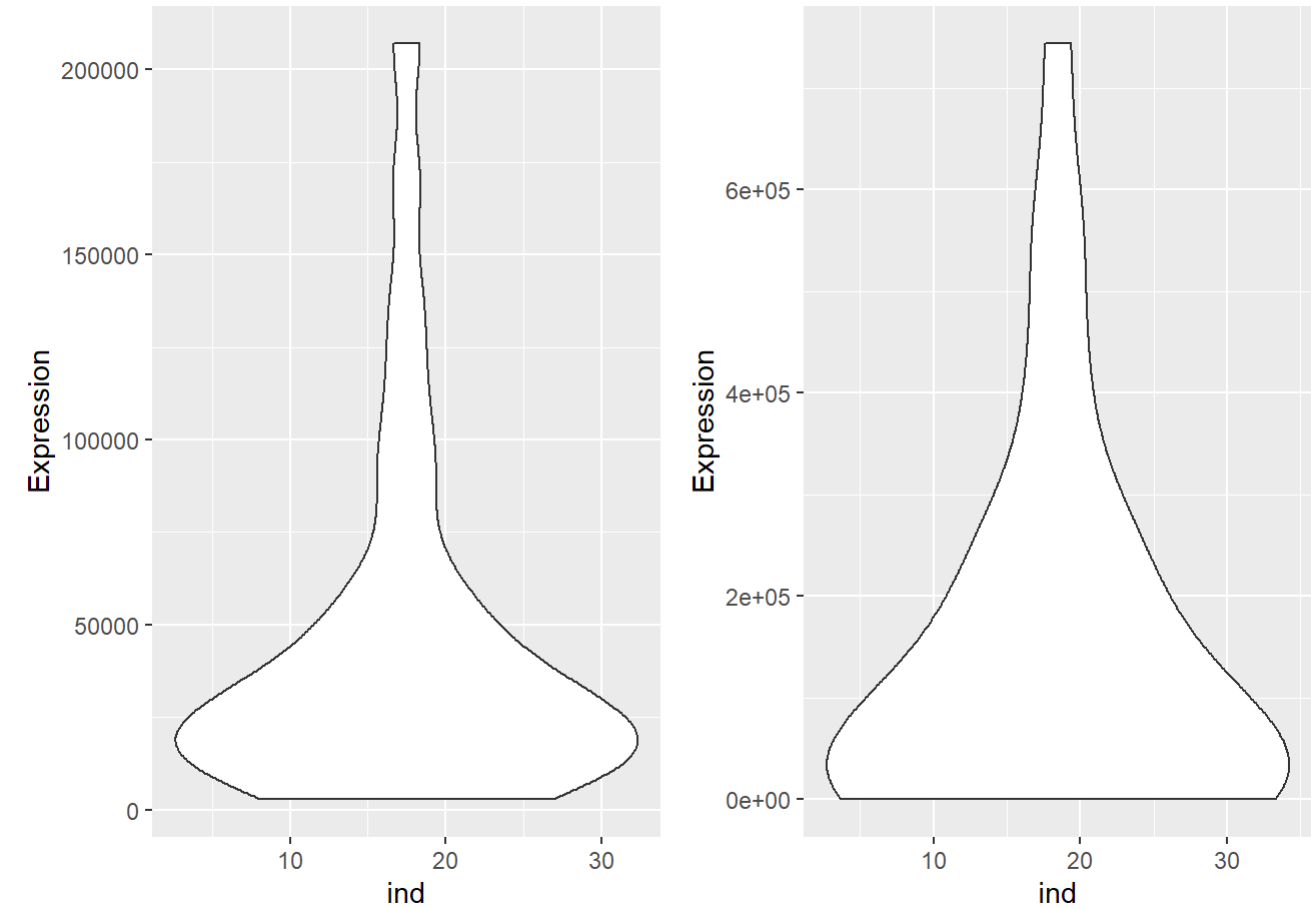
p0



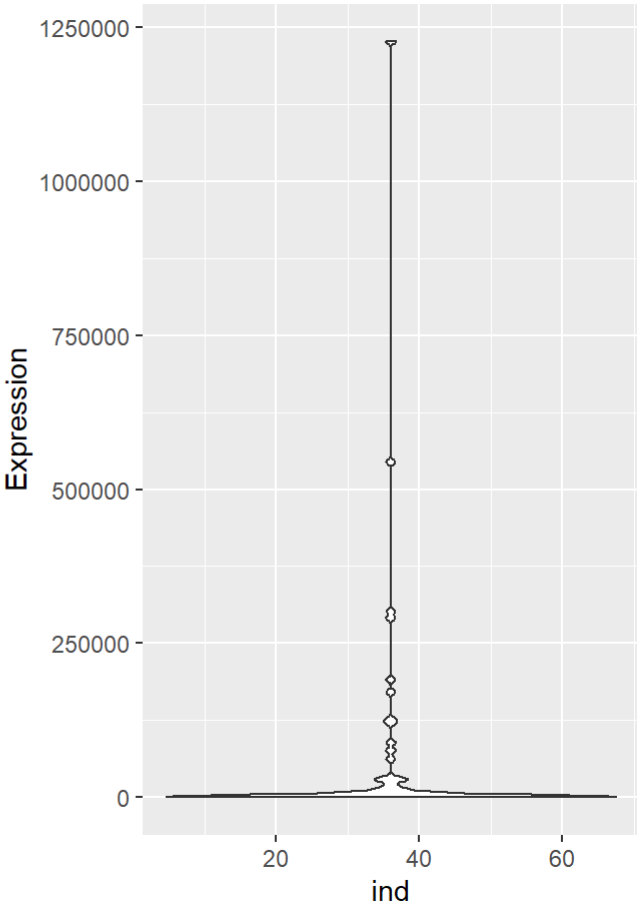
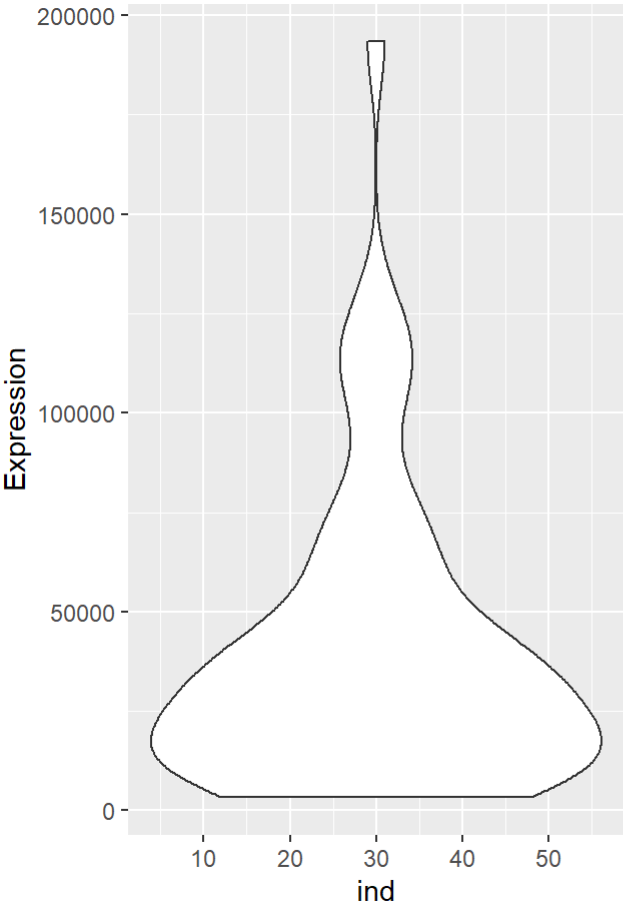
p21



p31



p41



```
#Female HBV Tumor-adjacent (data_1) tumor (data_0)
CDHR2data_1<- data.frame(CDHR2data_1, 1:8)
CDHR2data_0<-data.frame(CDHR2data_0,1:9)

#Male HBV Tumor-adjacent(data_3) tumor (data_2)
CDHR2data_3<-data.frame(CDHR2data_3,1:33)
CDHR2data_2<-data.frame(CDHR2data_2,1:40)

#Female HCV Tumor-adjacent (data_5) tumor(data_4)
CDHR2data_5<-data.frame(CDHR2data_5,1:34)
CDHR2data_4<-data.frame(CDHR2data_4,1:36)

#Male HCV Tumor-adjacent (data_7) tumor(data_6)
CDHR2data_7<-data.frame(CDHR2data_7,1:59)
CDHR2data_6<-data.frame(CDHR2data_6,1:71)

#Female HBV
colnames(CDHR2data_1)<- c("Expression","ind")
colnames(CDHR2data_0)<- c("Expression","ind")

#Male HBV
colnames(CDHR2data_3)<- c("Expression","ind")
colnames(CDHR2data_2)<- c("Expression","ind")

#Female HCV
colnames(CDHR2data_5)<- c("Expression","ind")
colnames(CDHR2data_4)<- c("Expression","ind")

#Male HCV
colnames(CDHR2data_7)<- c("Expression","ind")
colnames(CDHR2data_6)<- c("Expression","ind")

#Female HBV
p<- ggplot(CDHR2data_1, aes(x=ind, y=Expression)) + geom_violin()
p1<- ggplot(CDHR2data_0, aes(x=ind, y=Expression)) + geom_violin()

#Male HBV
p2<-ggplot(CDHR2data_3, aes(x=ind, y=Expression)) + geom_violin()
p20<- ggplot(CDHR2data_2, aes(x=ind, y=Expression)) + geom_violin()

#Female HCV
p3<-ggplot(CDHR2data_5, aes(x=ind, y=Expression)) + geom_violin()
p30<- ggplot(CDHR2data_4, aes(x=ind, y=Expression)) + geom_violin()

#Male HCV
p4<- ggplot(CDHR2data_7, aes(x=ind, y=Expression)) + geom_violin()
p40<- ggplot(CDHR2data_6, aes(x=ind, y=Expression)) + geom_violin()
```

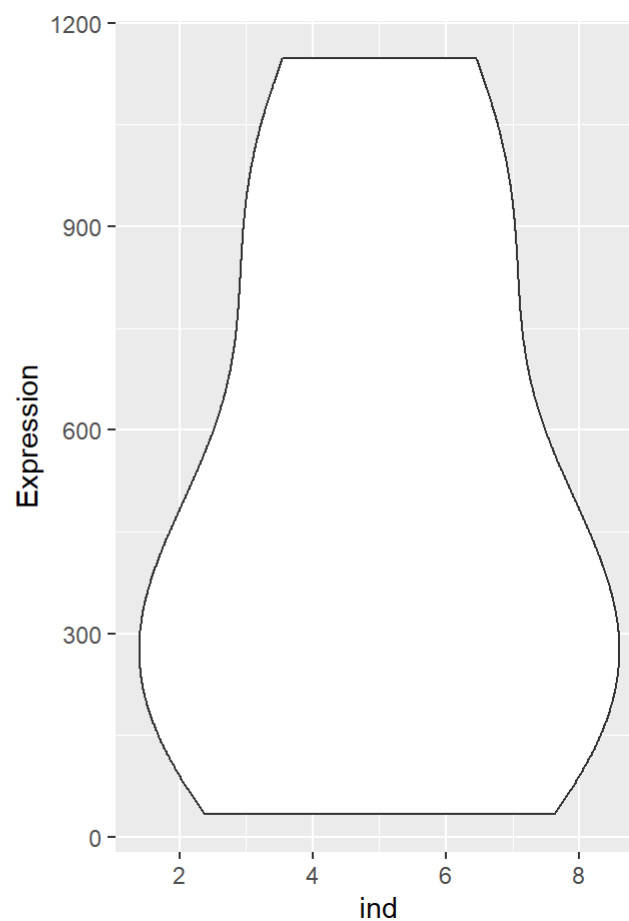
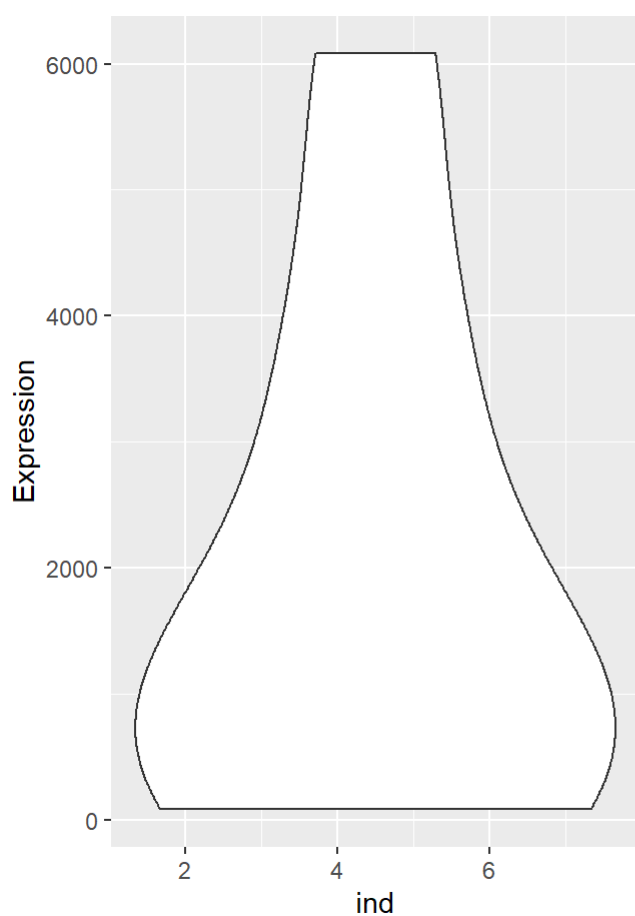
```
p0<-p+p1
pdf("~/R/CDHR2_expression_violin_F_HBV.pdf", width=12, height=12)
p0
```

```
p21<-p2+p20
pdf("~/R/CDHR2_expression_violin_M_HBV.pdf", width=12, height=12)
p21
```

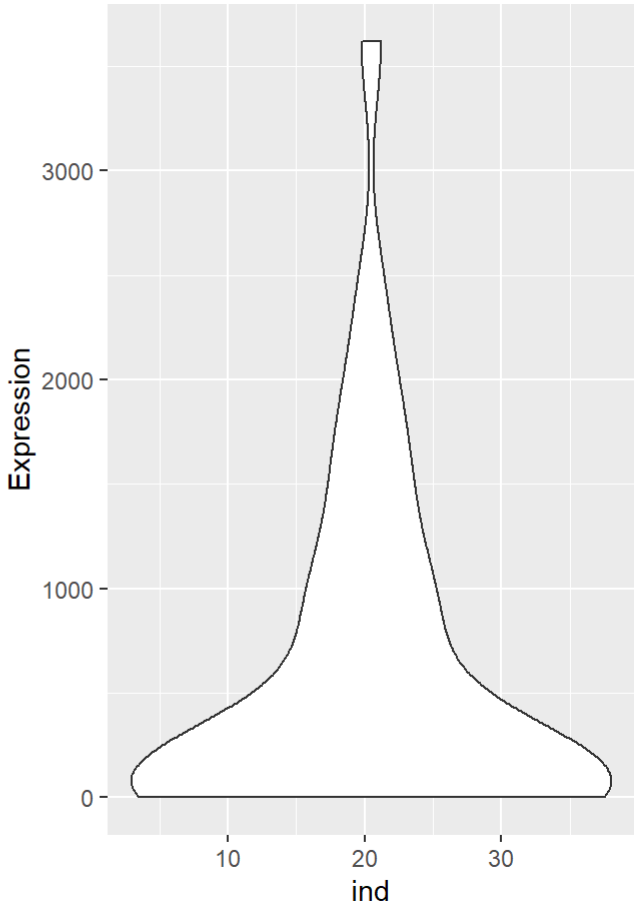
```
p31<-p3+p30
pdf("~/R/CDHR2_expression_violin_F_HCV.pdf", width=12, height=12)
p31
```

```
p41<-p4+p40
pdf("~/R/CDHR2_expression_violin_M_HCV.pdf", width=12, height=12)
p41
```

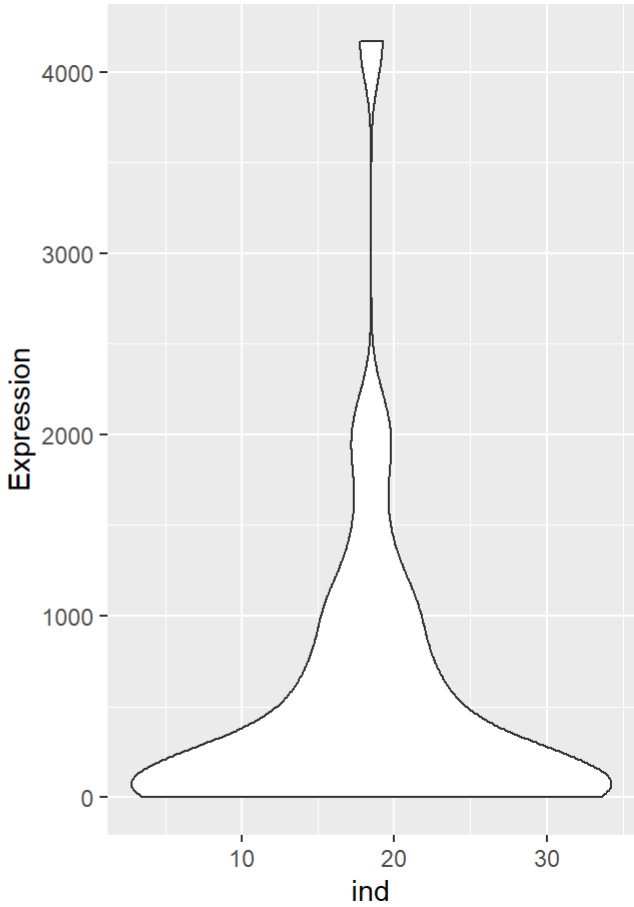
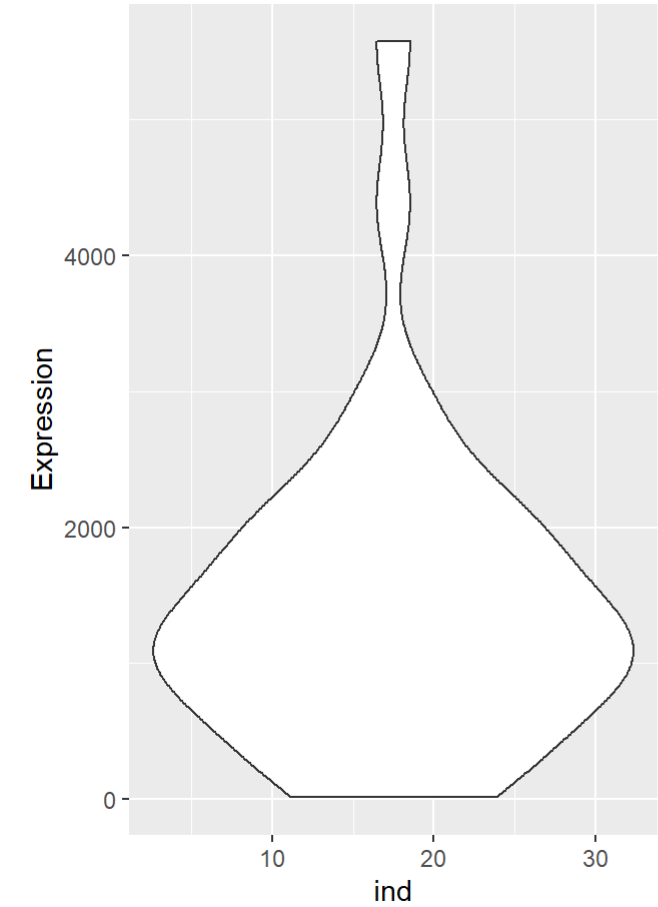
p0



p21



p31



p41

