# Using gplots

library(gplots) #library that contains functions for drawing plots like heatmap.2

#load and prepare data

msa1 <- read.csv("https://raw.githubusercontent.com/PineBiotech/omicslogic/master/SARS\_CoV\_2\_1.csv", header=TRUE)

#example 1: visualize 50 rows

msa1 <- msa1[,-1] #remove the "POS" column

msa1m <- data.matrix(msa1) #prepare a numeric matrix for msa1 data frame

#select only 50 rows

msa1sel <- msa1[21000:21100,] #select only 100 rows from the letter data frame

msa1msel <- data.matrix(msa1sel) #select only 100 rows from the numeric matrix

#transpose

msa1selT <- t(msa1sel)

msa1mselT <- t(msa1msel)

#set colors and margins for plot

TCGAcolors <- c("white", "lightgreen", "pink", "lightblue", "yellow")

names(TCGAcolors) = c(1,2,3,4,5) #labels = c("-","A","T","C","G")

par(cex.main=0.9, family="avenir") #set plot margins

#draw heatmap for first 50 positions of alignment

heatmap.2(msa1mselT, #data source matrix

#main settings

cexRow = 0.7, #row name font size

col = TCGAcolors, #set colors

dendrogram = "none", #remove dendrogram

Rowv = FALSE, #no reordering for rows

Colv = FALSE, #no reordering for columns

density.info="none", #remove density info

trace="none", #remove row and column lines

offsetRow=0.1, #change position of the row names

offsetCol=0.1, #change position of the column names

#add gray borders between cells

sepwidth=c(0.05,0.05), #sets separation width and height

sepcolor="gray", #color for border

colsep=1:ncol(msa1mselT), #add separation for number of columns in source data

rowsep=1:nrow(msa1mselT), #add separation for number of rows in source data

#plot title

main = "SARS\_CoV\_2\_Origin\_project\_FullTable\_1.csv", #heat map title

#plot margins

margins = c(5,10), #set margins

lwid=c(0.2,4),

lhei=c(0.9,3),

#adding letters inside the heatmap

notecex=1.0, #size of font inside each cell

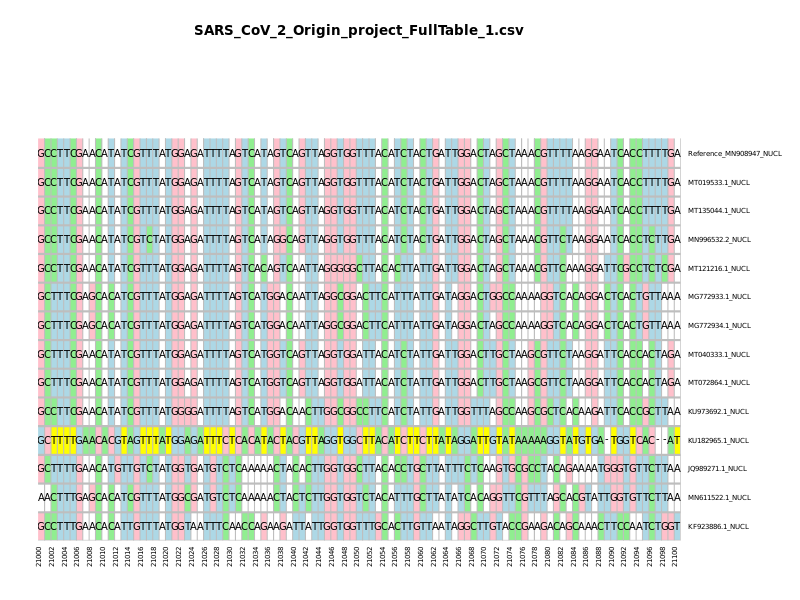
cellnote = msa1selT, #data to use in cells

notecol="black", #font color for cells

#legend

key = FALSE

)



## select only data where any sample has a variant

#select only data where any sample has a variant

new\_df <- msa1sel[!(msa1sel[2] == msa1sel[3] & msa1sel[2] == msa1sel[4] & msa1sel[2] == msa1sel[5] & msa1sel[2] == msa1sel[6] & msa1sel[2] == msa1sel[7] & msa1sel[2] == msa1sel[8] & msa1sel[2] == msa1sel[9] & msa1sel[2] == msa1sel[10] & msa1sel[2] == msa1sel[11] & msa1sel[2] == msa1sel[12] & msa1sel[2] == msa1sel[13] & msa1sel[2] == msa1sel[14]),]

new\_dfT <- t(new\_df)

new\_matrix <- msa1msel[!(msa1msel[,2] == msa1msel[,3] & msa1msel[,2] == msa1msel[,4] & msa1msel[,2] == msa1msel[,5] & msa1msel[,2] == msa1msel[,6] & msa1msel[,2] == msa1msel[,7] & msa1msel[,2] == msa1msel[,8] & msa1msel[,2] == msa1msel[,9] & msa1msel[,2] == msa1msel[,10] & msa1msel[,2] == msa1msel[,11] & msa1msel[,2] == msa1msel[,12] & msa1msel[,2] == msa1msel[,13] & msa1msel[,2] == msa1msel[,14]),]

new\_matrixT <- t(new\_matrix)

#draw heatmap for only variants

heatmap.2(new\_matrixT, #data source

#main settings

cexRow = 0.7, #row name font size

col = TCGAcolors, #set colors

dendrogram = "none", #remove dendrogram

Rowv = FALSE, #no reordering for rows

Colv = FALSE, #no reordering for columns

density.info="none", #remove density info

trace="none", #remove row and column lines

offsetRow=0.1, #change position of the row names

offsetCol=0.1, #change position of the column names

#add gray borders between cells

sepwidth=c(0.05,0.05), #sets separation width and height

sepcolor="gray", #color for border

colsep=1:ncol(new\_matrixT), #add separation for number of columns in source data

rowsep=1:nrow(new\_matrixT), #add separation for number of rows in source data

#plot title

main = cbind("Number of Variants Found: ",ncol(new\_matrixT)), #heat map title

#plot margins

margins = c(5,10), #set margins

lwid=c(0.2,4),

lhei=c(0.9,3),

#adding letters inside the heatmap

notecex=1.0, #size of font inside each cell

cellnote = new\_dfT, #data to use in cells

notecol="black", #font color for cells

#legend

key = FALSE

)

## 

## 

# 

# using ggplot

library(ggplot2)

data <- read.csv("https://raw.githubusercontent.com/PineBiotech/omicslogic/master/SARS\_CoV\_2\_1.csv", header=TRUE)

#example 1: visualize 50 rows

msa <- data[,-1] #remove the "POS" column

#select only 50 rows

msa\_sel <- msa[21000:21100,] #select only 100 rows from the letter data frame

msa\_selT <- t(msa\_sel)

#data1f <- t(data1[1:50, 2:15])

#data2 <- data.matrix(data1[10:50,2:15])

#data2t <- t(data2)

library(reshape2)

melted\_mat <- melt(msa\_selT)

#head(melted\_mat)

plot1 <- ggplot(data = melted\_mat, aes(x=Var2, y=Var1, fill=value)) + geom\_tile() + geom\_text(aes(label = value)) + xlab("Nucleotide Position") + ylab("Samples" ) + labs(fill = "Nucleotide") + labs(title="Complete MSA plot for the selected sequence region") + scale\_fill\_manual(values=c("white", "lemonchiffon", "lightgreen", "lightblue", "pink")) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))

plot(plot1)

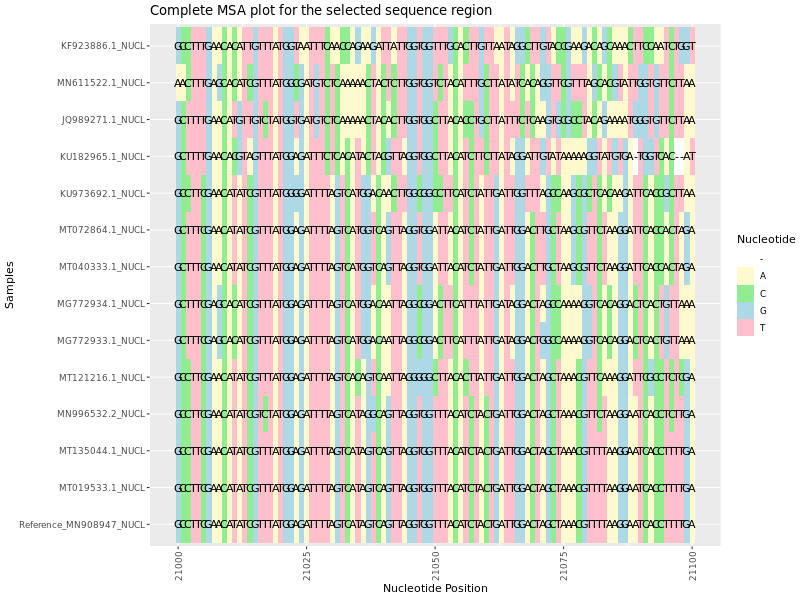


Figure: MSA plot for the selected region of sequence.

## select only data where any sample has a variant

**######MSA plot for only mutated region ########**

#select only data where any sample has a variant,

new\_df <- msa\_sel[!(msa\_sel[2] == msa\_sel[3] & msa\_sel[2] == msa\_sel[4] & msa\_sel[2] == msa\_sel[5] & msa\_sel[2] == msa\_sel[6] & msa\_sel[2] == msa\_sel[7] & msa\_sel[2] == msa\_sel[8] & msa\_sel[2] == msa\_sel[9] & msa\_sel[2] == msa\_sel[10] & msa\_sel[2] == msa\_sel[11] & msa\_sel[2] == msa\_sel[12] & msa\_sel[2] == msa\_sel[13] & msa\_sel[2] == msa\_sel[14]),]

#Transpose

new\_dfT <- t(new\_df)

#Melt data (Transform into new object)

melted\_mat2 <- melt(new\_dfT)

#head(melted\_mat)

title1 <- paste("Number of Variants Found ", ncol(new\_dfT))

plot2 <- ggplot(data = melted\_mat2, aes(x=Var2, y=Var1, fill=value)) + geom\_tile() + geom\_text(aes(label = value)) + xlab("Nucleotide Position") + ylab("Samples" ) + labs(fill = "Nucleotide") + labs(title= title1) + scale\_fill\_manual(values=c("white", "lemonchiffon", "lightgreen", "lightblue", "pink")) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5)) + ggtitle(as.character(title1))

plot(plot2)

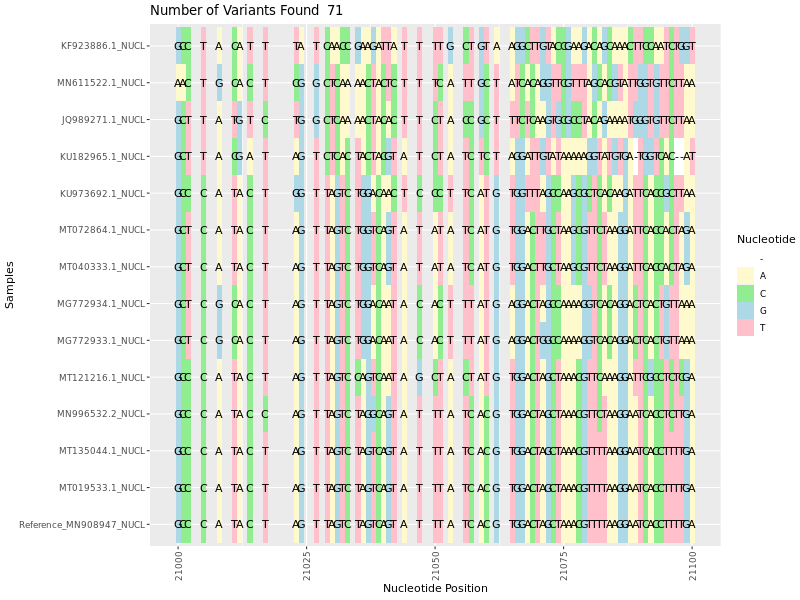
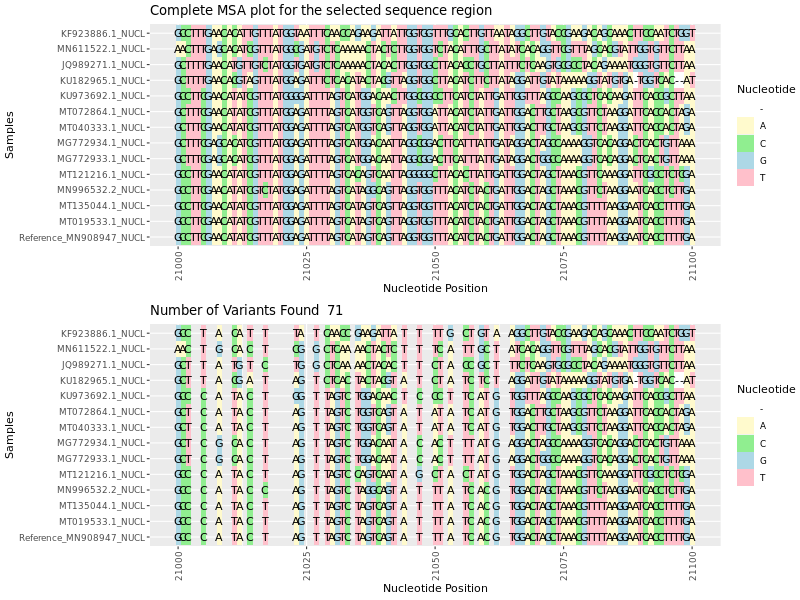


Figure: MSA plot variants.

#Combined plots together

library("gridExtra")

grid.arrange(plot1, plot2 , ncol=1, nrow = 2)



#Print title

title1

## Calculate no of mutations in each samples ##

# Write 1 if there is mutation else 0 for each sample

col1 <- ifelse(msa\_sel[2]== msa\_sel[1],0,1)

col2 <- ifelse(msa\_sel[3]== msa\_sel[1],0,1)

col3 <- ifelse(msa\_sel[4]== msa\_sel[1],0,1)

col4 <- ifelse(msa\_sel[5]== msa\_sel[1],0,1)

col5 <- ifelse(msa\_sel[6]== msa\_sel[1],0,1)

col6 <- ifelse(msa\_sel[7]== msa\_sel[1],0,1)

col7 <- ifelse(msa\_sel[8]== msa\_sel[1],0,1)

col8 <- ifelse(msa\_sel[9]== msa\_sel[1],0,1)

col9 <- ifelse(msa\_sel[10]== msa\_sel[1],0,1)

col10 <- ifelse(msa\_sel[11]== msa\_sel[1],0,1)

col11 <- ifelse(msa\_sel[12]== msa\_sel[1],0,1)

col12 <- ifelse(msa\_sel[13]== msa\_sel[1],0,1)

col13 <- ifelse(msa\_sel[14]== msa\_sel[1],0,1)

#Combind all the columns together to get single dataframe

dataframe <- cbind(col1, col2, col3, col4,col5, col6, col7, col8, col9, col10, col11, col12,col13)

#head(dataframe)

# Compute the some of mutations in each sample

colSums(dataframe)

#Write into a variable

sample\_mutations\_res <- as.data.frame(colSums(dataframe))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gene symbol | Start | Stop | Length |  |
| ORF1ab | 266 | 21555 | 21289 |  |
| S | 21563 | 25384 | 3821 |  |
| ORF3a | 25393 | 26220 | 827 |  |
| E | 26245 | 26472 | 227 |  |
| M | 26523 | 27191 | 668 |  |
| ORF6 | 27202 | 27387 | 185 |  |
| ORF7a | 27394 | 27759 | 365 |  |
| ORF7b | 27756 | 27887 | 131 |  |
| ORF8 | 27894 | 28259 | 365 |  |
| N | 28274 | 29533 | 1259 |  |
| ORF10 | 29558 | 29674 | 116 |  |