Emelie's Violins

Benjamin Moran

2018-01-15

Setup

Here you need to import the data for each network. Store the files separately in the data folder and copy the code below once for each file. (You could do this with a loop but this is simpler for now).

Discovery Time Data

```r
disc.dat1 <- read.delim("data/DiscoveryTime_Random.txt", # fileloc
                        header = FALSE, # no header
                        col.names = "Seconds") # label the column

disc.dat1$Network = factor("Random") # Add a column called 'Network' with the network name

disc.dat1$Seconds = disc.dat1$Seconds/1000 # convert to seconds

disc.dat2 <- read.delim("data/DiscoveryTime_B4.txt", # fileloc
                        header = FALSE, # no header
                        col.names = "Seconds") # label the column

disc.dat2$Network = factor("B4") # Add a column called 'Network' with the network name

disc.dat2$Seconds = disc.dat2$Seconds/1000 # convert to seconds
```

Next, combine all the data files together into a single data frame.

```r
discdata = rbind(disc.dat1, # Binds the data together by row (hence 'r'bind).
             disc.dat2) # add each dataset here.
```

Lastly, we need to calculate the median of each Network to plot later. Note: the `%>%` is a pipe (from the `magrittr` or `dplyr` packages in `R`). The `group_by` and `summarise` functions come from the `dplyr` package.

```r
discmed.data = discdata %>%            # Creates new object as a copy
  group_by(Network) %>%                # Groups by the network
  summarise(Median = median(Seconds)) # Returns the median value for each.
```

Messages Data

Here we will do the same for the messages data.

```r
mess.dat1 <- read.delim("data/MessagesSent_B4.txt", # fileloc
                        header = FALSE, # no header
                        col.names = "Messages") # label the column

mess.dat1$Network = factor("B4") # Add a column called 'Network' with the network name
mess.dat1$Messages = mess.dat1$Messages/12 # divides B4 message numbers by 12.
```

And we'll combine them in the same way.

```
# Comment out the below line and uncomment the last lines to add extra data).
messdata = mess.dat1

#messdata = rbind(mess.dat1, # Binds the data together by row (hence 'r'bind).
#                 mess.dat2) # add each dataset here.
```

And get the median.

```
messmed.data = messdata %>%            # Creates new object as a copy
  group_by(Network) %>%                # Groups by the network
  summarise(Median = median(Messages)) # Returns the median value for each.
```

Plotting

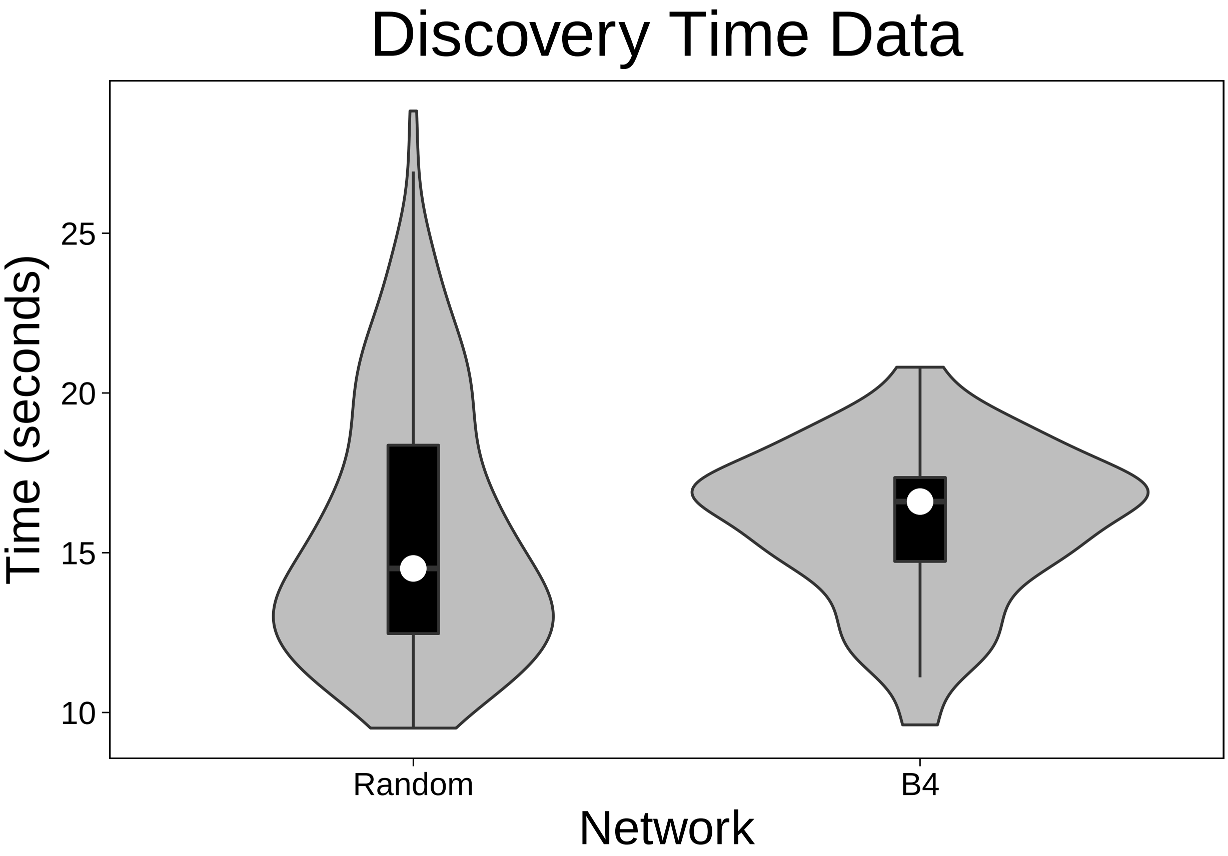
We'll use ggplo2 to plot the data becuase it has an easy-to-use geom_violin function for violin plots. Firtsly, you need to create a ggplot object with data. Then you call the various plotting functions. Remember to plot the median dots last, because you'll need to repoint data at the med.data dataframe we created earlier.

Lastly, we'll change some of the formatting. All of these options are available in the ggplot2 package.

```
# Create the ggplot object called 'p1'

p1 <-  ggplot(data = discdata,         # our combined data
             aes(x = Network,     # networks on the x axis
                 y = Seconds,     # Seconds on the y axis
                 group = Network) # plots each network individually
             ) +

      # Call each plot
      geom_violin(position = "dodge",   # Ignore
                  fill = "grey") +      # colour of violin
      geom_boxplot(width= 0.1,          # Width of the boxplot
                   outlier.colour = NA, # Outliers not plotted
                   position = "dodge",  # Ignore
                   fill = "black") +    # colour of box

      # plot the Median points
      geom_point(data = discmed.data,  # the median data
                 aes(x = Network,  # Networks on the x axis (same as above).
                     y = Median),  # the Median value on the y axis.
                 colour = "white", # colour of the dot
                 size = 4) +       # size of the dot

      # formatting options

      theme_linedraw() + # a basic ggplot theme

      labs(y = "Time (seconds)",              # y axis label
           x = "Network",                     # x axis label
           title = "Discovery Time Data") +  # Main title

      theme(axis.text = element_text(size=12),      # x / y axis label sizes
            axis.title = element_text(size = 18),   # x / y axis title sizes
            plot.title = element_text(size = 24,    # Main title size
                                       hjust = 0.5), # Centre main title
            panel.grid = element_blank())           # Removes all gridlines
```

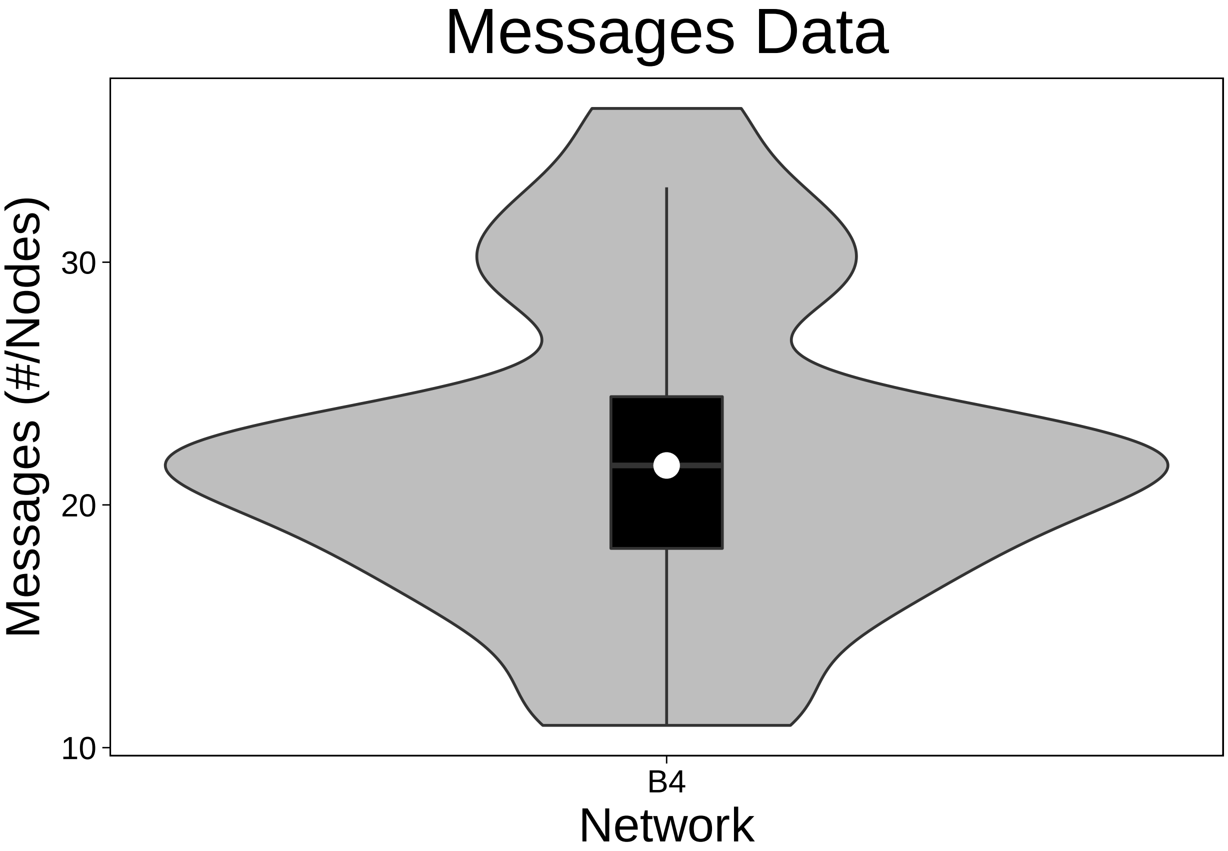
And here is the plot.

Discovery Plot



Messages Plots

Using the same code as the discovery time plots, we now plot the messages data.


Messages Data
Messages (#/Nodes)
B4
Network