> table(tweets.emojis$hashtag == "husky")

FALSE

226

> tweets.emojis$hashtag == "Goldenretriever"

[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[19] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[55] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[73] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[91] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[109] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[127] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[145] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[163] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[199] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[217] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

> which(tweets.emojis$hashtag == "Goldenretriever")

integer(0)

> tail(tweets.emojis)

tweetid

221 221

222 222

223 223

224 224

225 225

226 226

text

221 Semi Flush Mount Crystal Pendant Chandeliers #pitbull #textureandstylehomemarket #bulldog #decorative #pillowcases<e2><80><a6> https://t.co/9rY0hrwkHf

222 Pink Diamond Canvas Wall Art #bulldog #goldedoodle #goldenretriever #decorative #pillowcasecover #pug #pillowcases<e2><80><a6> https://t.co/QW06l8YCJ6

223 If you're not following the @TheGoldenRatio4 on the Snaps, you're missing out on adorable pictures like this.<e2><80><a6> https://t.co/JD1pQ1Cxwv

224 <ed><a0><bd><ed><b8><8d><ed><a0><bd><ed><b8><ac> #ilovepuppies and am especially a sucker for #goldenretriever #puppies like<e2><80><a6> https://t.co/YLFFa5qjZZ

225 Couldn't resist! \n#husky #huskylife #itsahuskything #sofunny https://t.co/yyHMb1RLyQ

226 <e2><80><9c>What are you lookin at?<e2><80><9d> @beaglefacts #beagles https://t.co/tQz1g8atD7

created url latitude longitude retweets

221 2018-01-30 18:23:01 https://twitter.com/TSHomeMarket/status/958405205388820492 NA NA 0

222 2018-01-30 18:26:46 https://twitter.com/TSHomeMarket/status/958406148260016129 NA NA 0

223 2018-01-30 15:40:23 https://twitter.com/wx\_becks/status/958364273826246658 NA NA 0

224 2018-01-30 22:01:54 https://twitter.com/YPDogTrainer/status/958460288856154114 NA NA 0

225 2018-01-30 21:16:53 https://twitter.com/yvetteposkitt/status/958448958308864002 NA NA 0

226 2018-01-30 08:43:39 https://twitter.com/zxfxr/status/958259403009900547 NA NA 0

hashtag username z num.emojis emoji.names

221 #Goldenretriever TSHomeMarket 1 0

222 #Goldenretriever TSHomeMarket 1 0

223 #Goldenretriever wx\_becks 1 0

224 #Goldenretriever YPDogTrainer 1 2 smiling face with heart-shaped eyes, grimacing face

225 #husky yvetteposkitt 1 0

226 #beagles zxfxr 1 0

> df.2 <- subset(tweets.emojis, grepl(paste(c('#beagles'), collapse = '|'), tolower(tweets.emojis$text)));

>

> df.a <- subset(subset(df.2, emoji.names != ''), select = c(tweetid, emoji.names)); df.a$emoji.names <- as.character(df.a$emoji.names);

> df.b <- data.frame(table(unlist(strsplit(df.a$emoji.names, ',')))); names(df.b) <- c('var', 'freq'); df.b$var <- trimws(df.b$var, 'both'); df.b <- subset(df.b, var != '');

> df.c <- aggregate(freq ~ var, data = df.b, function(x) sum(x)); df.c <- df.c[with(df.c, order(-freq)), ]; row.names(df.c) <- NULL;

>

> nrow(df.c)

[1] 5

>

> df.d <- subset(df.c, freq > 1); df.d$dens <- round(100 \* (df.d$freq / nrow(df)), 1); df.d$dens.sm <- (df.d$freq + 1) / (nrow(df) + 1); df.d$rank <- as.numeric(row.names(df.d)); df.d <- rename(df.d, c(var = 'name'));

>

> nrow(df.d)

[1] 0

> ncol(df.d)

[1] 5

> df.d <- subset(df.2, freq > 1); df.d$dens <- round(100 \* (df.d$freq / nrow(df)), 1); df.d$dens.sm <- (df.d$freq + 1) / (nrow(df) + 1); df.d$rank <- as.numeric(row.names(df.d)); df.d <- rename(df.d, c(var = 'name'));

Error in eval(expr, envir, enclos) : object 'freq' not found

>

> df.d <- subset(df.2); df.d$dens <- round(100 \* (df.d$freq / nrow(df)), 1); df.d$dens.sm <- (df.d$freq + 1) / (nrow(df) + 1); df.d$rank <- as.numeric(row.names(df.d)); df.d <- rename(df.d, c(var = 'name'));

Error in `$<-.data.frame`(`\*tmp\*`, "dens", value = numeric(0)) :

replacement has 0 rows, data has 46

>

> nrow(df.b)

[1] 5

> df.2 <- subset(tweets.emojis, grepl(paste(c('#goldenretriever'), collapse = '|'), tolower(tweets.emojis$text)));

> >

Error: unexpected '>' in ">"

> df.2 <- subset(tweets.emojis, grepl(paste(c('#goldenretriever'), collapse = '|'), tolower(tweets.emojis$text)));

>

> nrow(df.2)

[1] 58

> df.2 <- subset(tweets.emojis, grepl(paste(c('#oldenretriever'), collapse = '|'), tolower(tweets.emojis$text)));

>

> nrow(df.2)

[1] 0

> df.2 <- subset(tweets.emojis, grepl(paste(c('#goldenretriever'), collapse = '|'), tolower(tweets.emojis$text)));

>

> df.a <- subset(subset(df.2, emoji.names != ''), select = c(tweetid, emoji.names)); df.a$emoji.names <- as.character(df.a$emoji.names);

> df.b <- data.frame(table(unlist(strsplit(df.a$emoji.names, ',')))); names(df.b) <- c('var', 'freq'); df.b$var <- trimws(df.b$var, 'both'); df.b <- subset(df.b, var != '');

>

> nrow(df.b)

[1] 20

> df.c <- aggregate(freq ~ var, data = df.b, function(x) sum(x)); df.c <- df.c[with(df.c, order(-freq)), ]; row.names(df.c) <- NULL;

>

> nrow(df.c)

[1] 19

> df.d <- subset(df.c, freq > 1); df.d$dens <- round(1000 \* (df.d$freq / nrow(df)), 1); df.d$dens.sm <- (df.d$freq + 1) / (nrow(df) + 1); df.d$rank <- as.numeric(row.names(df.d)); df.d <- rename(df.d, c(var = 'name'));

> d

Error: object 'd' not found

> df.d <- subset(df.c, freq > 1); df.d$dens <- round(1000 \* (df.d$freq / nrow(df)), 1); df.d$dens.sm <- (df.d$freq + 1) / (nrow(df) + 1); df.d$rank <- as.numeric(row.names(df.d)); df.d <- rename(df.d, c(var = 'name'));

>

> nrow(df.d)

[1] 7

> df.e <- subset(df.d, select = c(name, dens, dens.sm, freq, rank))

> df.e$ht <- as.character(arrange(data.frame(table(tolower(unlist(str\_extract\_all(df.2$text, '#\\w+'))))), -Freq)$Var1[1]);

> df.e[1:10, ]; emojis.count.2 <- df.e

name dens dens.sm freq rank ht

1 heavy black heart 26.5 0.03083700 6 1 #goldenretriever

2 dog face 13.3 0.01762115 3 2 #goldenretriever

3 paw prints 13.3 0.01762115 3 3 #goldenretriever

4 face with tears of joy 8.8 0.01321586 2 4 #goldenretriever

5 purple heart 8.8 0.01321586 2 5 #goldenretriever

6 smiling face with heart-shaped eyes 8.8 0.01321586 2 6 #goldenretriever

7 smiling face with smiling eyes 8.8 0.01321586 2 7 #goldenretriever

NA <NA> NA NA NA NA <NA>

NA.1 <NA> NA NA NA NA <NA>

NA.2 <NA> NA NA NA NA <NA>

> names(emojis.count.1)[-1] <- paste0(names(emojis.count.1)[-1], '.1'); names(emojis.count.2)[-1] <- paste0(names(emojis.count.2)[-1], '.2');

> df.a <- merge(emojis.count.1, emojis.count.2, by = 'name', all.x = TRUE, all.y = TRUE)

> df.a[, c(2:4, 6:8)][is.na(df.a[, c(2:4, 6:8)])] <- 0; df.a <- df.a[with (df.a, order(-dens.1)), ]

> df.a$index <- ifelse(df.a$dens.1 > 0 & df.a$dens.2 > 0 & (df.a$dens.1 > df.a$dens.2), round(100 \* ((df.a$dens.1 / df.a$dens.2) - 1), 0),

+ ifelse(df.a$dens.1 > 0 & df.a$dens.2 > 0 & (df.a$dens.2 > df.a$dens.1), -1 \* round(100 \* ((df.a$dens.2 / df.a$dens.1) - 1), 0), NA))

>

> df.a$logor <- log(df.a$dens.sm.1 / df.a$dens.sm.2)

> df.a$dens.mean <- 0.5 \* (df.a$dens.1 + df.a$dens.2)

> k <- 50; df.b <- subset(df.a, (rank.1 <= k | rank.2 <= k) &

+ (freq.1 >= 5 | freq.2 >= 5) &

+ (freq.1 > 0 & freq.2 > 0) & dens.mean > 0); nrow(df.b)

[1] 3

> df.c <- subset(df.b, select = c(name, dens.1, dens.2, freq.1, freq.2, dens.mean, round(logor, 2)))

> df.c <- df.c[with(df.c, order(-logor)), ]; row.names(df.c) <- NULL; nrow(df.c); df.c;

[1] 3

name dens.1 dens.2 freq.1 freq.2 dens.mean logor

1 dog face 31.0 13.3 7 3 22.15 0.6931472

2 smiling face with heart-shaped eyes 22.1 8.8 5 2 15.45 0.6931472

3 heavy black heart 17.7 26.5 4 6 22.10 -0.3364722

> emojis.comp.p <- df.c

> bind(head(emojis.comp.p), tail(emojis.comp.p))

Error: could not find function "bind"

> rbind(head(emojis.comp.p), tail(emojis.comp.p))

name dens.1 dens.2 freq.1 freq.2 dens.mean logor

1 dog face 31.0 13.3 7 3 22.15 0.6931472

2 smiling face with heart-shaped eyes 22.1 8.8 5 2 15.45 0.6931472

3 heavy black heart 17.7 26.5 4 6 22.10 -0.3364722

4 dog face 31.0 13.3 7 3 22.15 0.6931472

5 smiling face with heart-shaped eyes 22.1 8.8 5 2 15.45 0.6931472

6 heavy black heart 17.7 26.5 4 6 22.10 -0.3364722

>

>

>

>

> setwd('/Users/Asad/ios\_9\_3\_emoji\_files')

> df.t <- arrange(emojis.comp.p, name)

> imgs <- lapply(paste0(df.t$name, '.png'), png::readPNG)

> g <- lapply(imgs, grid::rasterGrob)

> df.t <- arrange(emojis.comp.p, logor)

> xlab <- paste0('Emoji Valence: Log Odds Ratio (', paste0(unique(emojis.count.2$ht), ' <--> ', unique(emojis.count.1$ht), ')'));

> y

[1] 2 3 5 7 11 13

> ylab <- 'Overall Frequency (Per 100 Tweets)'

> k <- 8

> xsize <- (k/100) \* (max(df.t$logor) - min(df.t$logor)); ysize <- (k/100) \* (max(df.t$dens.mean) - min(df.t$dens.mean));

>

> df.t$xsize <- xsize; df.t$ysize <- ysize

> df.t$dens.m <- ifelse(df.t$dens.mean > median(df.t$dens.mean), round(sqrt((df.t$dens.mean / min(df.t$dens.mean))), 2), 1)

> df.t$xsize <- df.t$dens.m \* df.t$xsize; df.t$ysize <- df.t$dens.m \* df.t$ysize

> df.t <- arrange(df.t, name)

> g1 <- ggplot(df.t, aes(jitter(logor), dens.mean)) +

+ xlab(xlab) + ylab(ylab) +

+ mapply(function(x, y, i) {

+ annotation\_custom(g[[i]], xmin = x-0.5\*df.t$xsize[i], xmax = x+0.5\*df.t$xsize[i],

+ ymin = y-0.5\*df.t$ysize[i], ymax = y+0.5\*df.t$ysize[i])},

+ jitter(df.t$logor), df.t$dens.mean, seq\_len(nrow(df.t))) +

+ scale\_x\_continuous(limits = c(1.15 \* min(df.t$logor), 1.15 \* max(df.t$logor))) +

+ scale\_y\_continuous(limits = c(0, 1.20 \* max(df.t$dens.mean))) +

+ geom\_vline(xintercept = 0) + geom\_hline(yintercept = 0) +

+ theme\_bw() +

+ theme(axis.title.x = element\_text(size = 10), axis.title.y = element\_text(size = 10),

+ axis.text.x = element\_text(size = 8, colour = 'black'), axis.text.y = element\_text(size = 8, colour = 'black'))

> setwd('/Users/Asad/Emoji\_tutorial')

> png(paste0('emojis.comp.p\_', Sys.Date(), '\_', format(Sys.time(), '%H-%M-%S'), '.png'),

+ width = 6600, height = 4000, units = 'px', res = 1000)

> g1

> g1;

> g1

> dev.off()

quartz\_off\_screen

4

> g1

> dev.off()

quartz\_off\_screen

5

>

>

>

> g1 <- ggplot(df.t, aes(jitter(logor), dens.mean)) +

+ xlab(xlab) + ylab(ylab) +

+ mapply(function(x, y, i) {

+ annotation\_custom(g[[i]], xmin = x-0.5\*df.t$xsize[i], xmax = x+0.5\*df.t$xsize[i],

+ ymin = y-0.5\*df.t$ysize[i], ymax = y+0.5\*df.t$ysize[i])},

+ jitter(df.t$logor), df.t$dens.mean, seq\_len(nrow(df.t))) +

+ scale\_x\_continuous(limits = c(1.15 \* min(df.t$logor), 1.15 \* max(df.t$logor))) +

+ scale\_y\_continuous(limits = c(0, 1.20 \* max(df.t$dens.mean))) +

+ geom\_vline(xintercept = 0) + geom\_hline(yintercept = 0) +

+ theme\_bw() +

+ theme(axis.title.x = element\_text(size = 10), axis.title.y = element\_text(size = 14),

+ axis.text.x = element\_text(size = 8, colour = 'black'), axis.text.y = element\_text(size = 8, colour = 'black'))

> g1

> dev.off()