Artist Analysis & Visualization

```
library(readr)

artists <- read_csv("/Users/Anton/Desktop/DSU Project/artist_compiled_complete.csv")
```

```
## Parsed with column specification:
## cols(
##   artist = col_character(),
##   DateTime = col_date(format = ""),
##   Total.Followers = col_double(),
##   followers_change = col_double(),
##   Monthly.Listeners = col_double(),
##   listeners_change = col_double(),
##   Popularity = col_double(),
##   pop_change = col_double(),
##   fol_pct_change = col_double(),
##   lis_pct_change = col_double(),
##   pop_pct_change = col_double(),
##   closest_fest_days = col_double(),
##   festival_within_4 = col_logical(),
##   festival_within_7 = col_logical(),
##   festival_within_14 = col_logical(),
##   closest_song_days = col_double(),
##   song_within_4 = col_logical(),
##   song_within_7 = col_logical(),
##   song_within_14 = col_logical()
## )
```

```
artists <- artists[artists$artist != "blastoyz" | artists$DateTime > as.Date("2018-07-19"),]
artists <- artists[artists$artist != "alessandra roncone",]

festivals <- read_csv("/Users/Anton/Desktop/DSU Project/Artists Compiled/songkick_data_initial_small_ar
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   Artist = col_character(),
##   Event = col_character(),
##   `Festival or Concert` = col_double(),
##   Date = col_character()
## )
```

```
library(stringr)
festivals$Date <- as.Date(str_remove(festivals$Date, "\\w*\\s"), format = "%d %B %Y")

festivals$Artist <- str_replace_all(tolower(festivals$Artist), "\\.", "")
```

```r
library(dplyr)
```

```
##
## 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
```

```r
festivals <- festivals %>% group_by(Artist, Event) %>% mutate(num_days = n())

festivals <- festivals[!duplicated(festivals[,c("Artist", "Event")]),]

songs <- read.csv("/Users/Anton/Desktop/DSU Project/Artists Compiled/spotify_song_data_initial_small_ar
songs <- songs[,c("artist_name", "name", "release_date", "total_tracks")]
songs$release_date <- as.Date(songs$release_date)
songs$artist_name <- str_replace_all(tolower(songs$artist_name), "\\.", "")
```

```r
str(artists)
```

```
## tibble [19,285 x 19] (S3: tbl_df/tbl/data.frame)
##  $ artist            : chr [1:19285] "alan fitzpatrick" "alan fitzpatrick" "alan fitzpatrick" "alan
##  $ DateTime          : Date[1:19285], format: "2018-04-05" "2018-04-06" ...
##  $ Total.Followers   : num [1:19285] 51168 51254 51298 51379 51459 ...
##  $ followers_change  : num [1:19285] 23 86 44 81 80 55 77 90 62 69 ...
##  $ Monthly.Listeners : num [1:19285] 254410 254830 257650 257650 258408 ...
##  $ listeners_change  : num [1:19285] 0 420 2820 0 758 0 0 0 0 -109 ...
##  $ Popularity        : num [1:19285] 50 50 50 50 50 50 50 50 49 49 ...
##  $ pop_change        : num [1:19285] 0 0 0 0 0 0 0 0 -1 0 ...
##  $ fol_pct_change    : num [1:19285] 0.045 0.1681 0.0858 0.1579 0.1557 ...
##  $ lis_pct_change    : num [1:19285] 0 0.165 1.107 0 0.294 ...
##  $ pop_pct_change    : num [1:19285] 0 0 0 0 0 0 0 0 -2 0 ...
##  $ closest_fest_days : num [1:19285] 27 28 29 30 0 1 2 3 4 5 ...
##  $ festival_within_4 : logi [1:19285] FALSE FALSE FALSE FALSE TRUE TRUE ...
##  $ festival_within_7 : logi [1:19285] FALSE FALSE FALSE FALSE TRUE TRUE ...
##  $ festival_within_14: logi [1:19285] FALSE FALSE FALSE FALSE TRUE TRUE ...
##  $ closest_song_days : num [1:19285] 104 0 1 2 3 4 5 6 7 8 ...
##  $ song_within_4     : logi [1:19285] FALSE TRUE TRUE TRUE TRUE TRUE ...
##  $ song_within_7     : logi [1:19285] FALSE TRUE TRUE TRUE TRUE TRUE ...
##  $ song_within_14    : logi [1:19285] FALSE TRUE TRUE TRUE TRUE TRUE ...
```

```r
str(as.data.frame(festivals))
```

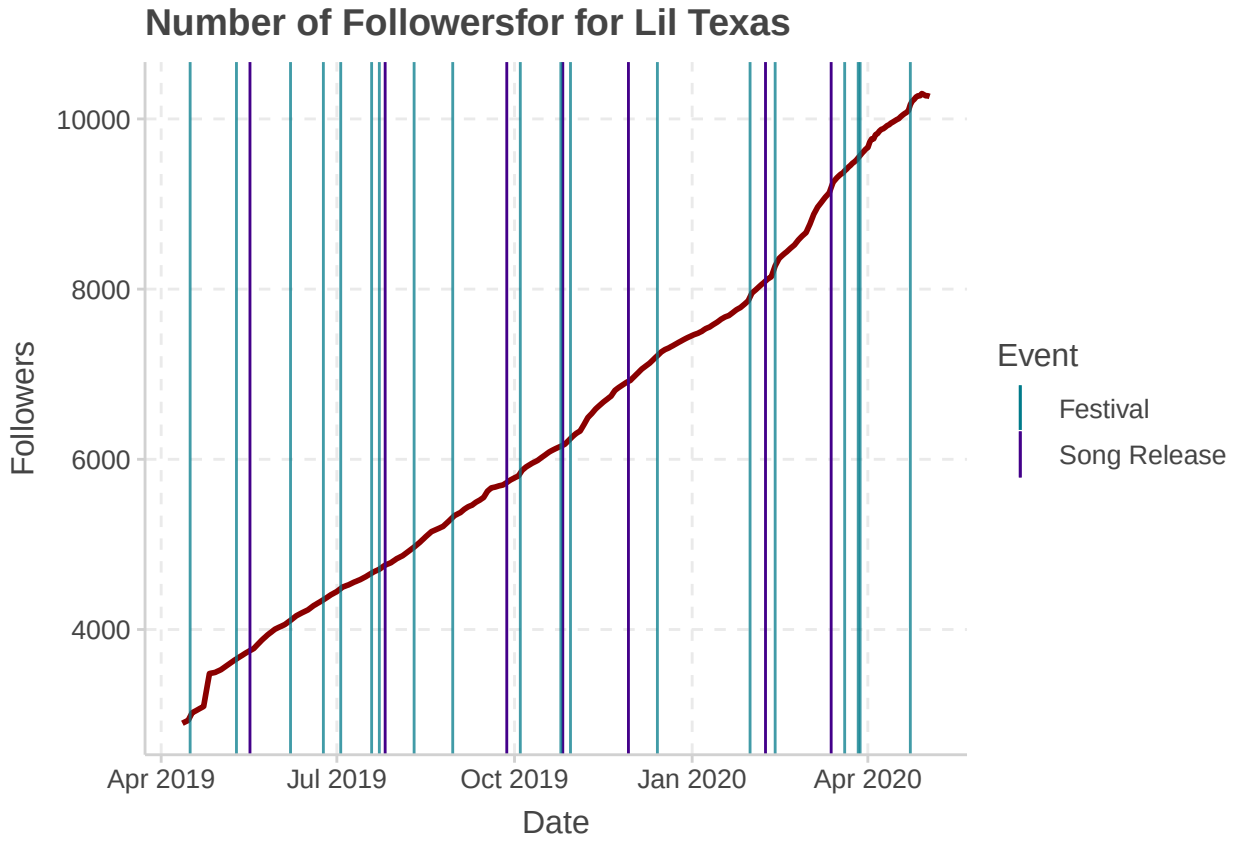
```
## 'data.frame':    1050 obs. of  5 variables:
##  $ Artist             : chr  "junkie kid" "junkie kid" "junkie kid" "junkie kid" ...
##  $ Event              : chr  "New Horizons 2019" "Parookaville Festival 2019" "Wasteland Mexico 2018
##  $ Festival or Concert: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Date               : Date, format: "2019-08-21" "2019-07-19" ...
##  $ num_days           : int  5 3 2 2 4 3 3 2 2 1 ...
```

```r
str(songs)
```

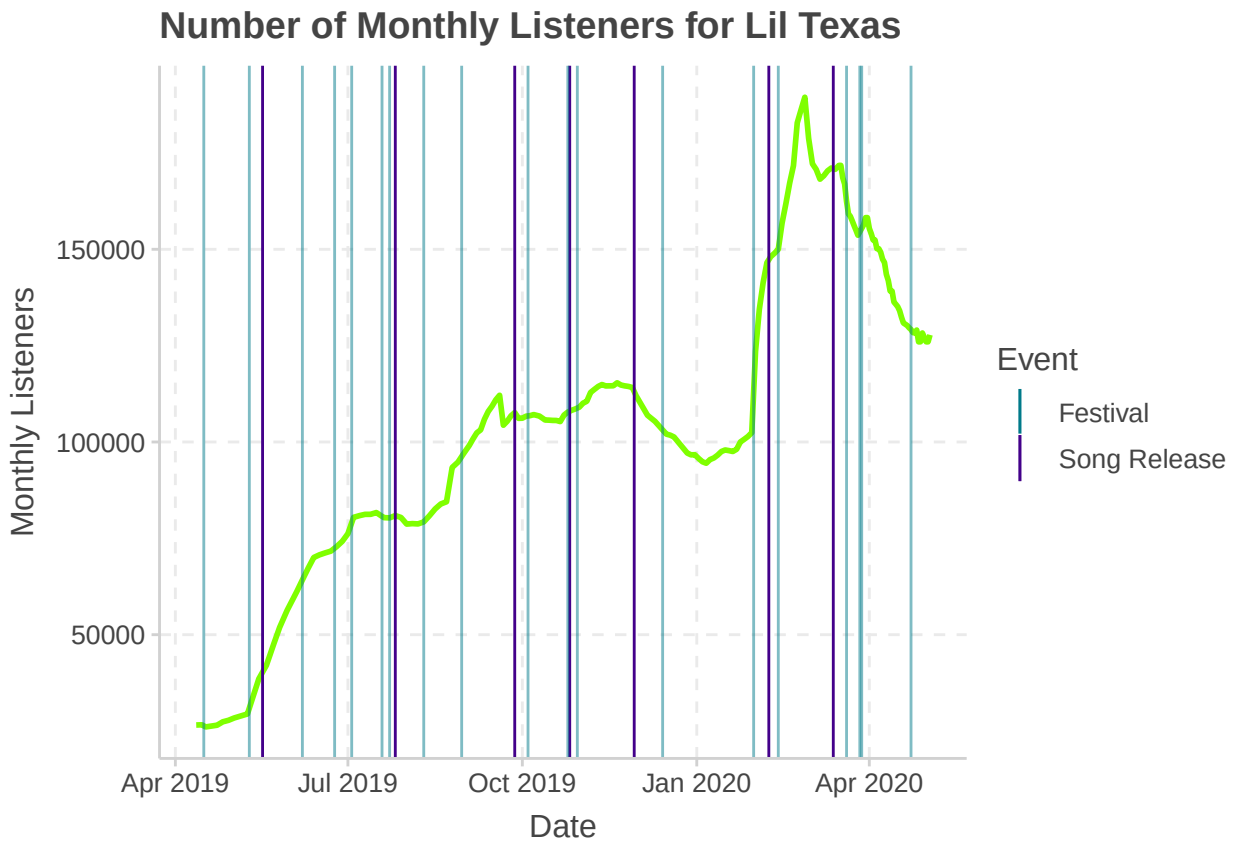
```
## 'data.frame':    1304 obs. of  4 variables:
```

```
##  $ artist_name : chr  "junkie kid" "junkie kid" "junkie kid" "junkie kid" ...
##  $ name        : chr  "Puro Pinche Hardstyle, Vol. 1" "Find Yourself" "Love Is Dead (LNY TNZ Remix)"
##  $ release_date: Date, format: "2019-10-18" "2018-05-04" ...
##  $ total_tracks: int  11 16 1 1 1 1 1 1 7 1 ...
```

```
library(ggplot2)
library(ggthemr)

ggthemr("pale")
#ggthemr_reset()

ggplot(artists[artists$artist == "lil texas",], aes(x = DateTime, y = Total.Followers)) +
  geom_line(size = 1, color = "red4") +
  geom_vline(data = festivals[festivals$Artist == "lil texas",], mapping = aes(xintercept = Date, color
  geom_vline(data = songs[songs$artist_name == "lil texas",], mapping = aes(xintercept = release_date,
  scale_color_manual(name = "Event", values = c("#43008a" = "#43008a", "#007a8a" = "#007a8a"), labels =
  xlab("Date") +
  ylab("Followers") +
  ggtitle("Number of Followersfor for Lil Texas")
```




```
ggplot(artists[artists$artist == "lil texas",], aes(x = DateTime, y = Monthly.Listeners)) +
  geom_line(size = 1, color = "chartreuse") +
  geom_vline(data = festivals[festivals$Artist == "lil texas",], mapping = aes(xintercept = Date, color
  geom_vline(data = songs[songs$artist_name == "lil texas",], mapping = aes(xintercept = release_date,
  scale_color_manual(name = "Event", values = c("#43008a" = "#43008a", "#007a8a" = "#007a8a"), labels =
  xlab("Date") +
  ylab("Monthly Listeners") +
  ggtitle("Number of Monthly Listeners for Lil Texas")
```



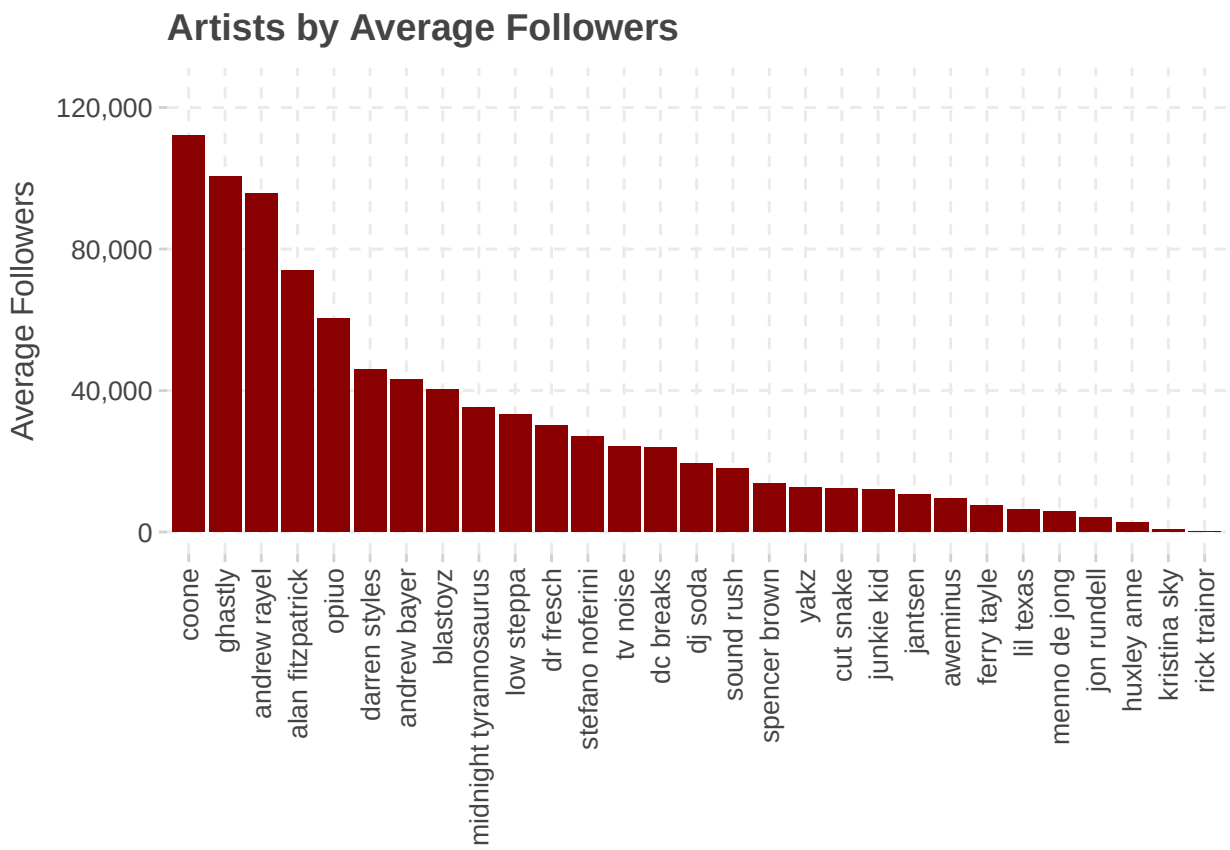

```
library(dplyr)

artists_summary <- artists %>% group_by(artist) %>% summarise(av_listeners = mean(Monthly.Listeners),
                                                              av_followers = mean(Total.Followers),
                                                              av_lis_change = mean(listeners_change),
                                                              av_fol_change = mean(followers_change),
                                                              av_pct_lis_change = mean(lis_pct_change),
                                                              av_pct_fol_change = mean(fol_pct_change),
                                                              num_fest = sum(festivals$Artist == artist),
                                                              num_fest_days = sum(festivals[festivals$Artist == a
                                                              av_fest_days = num_fest_days/num_fest)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

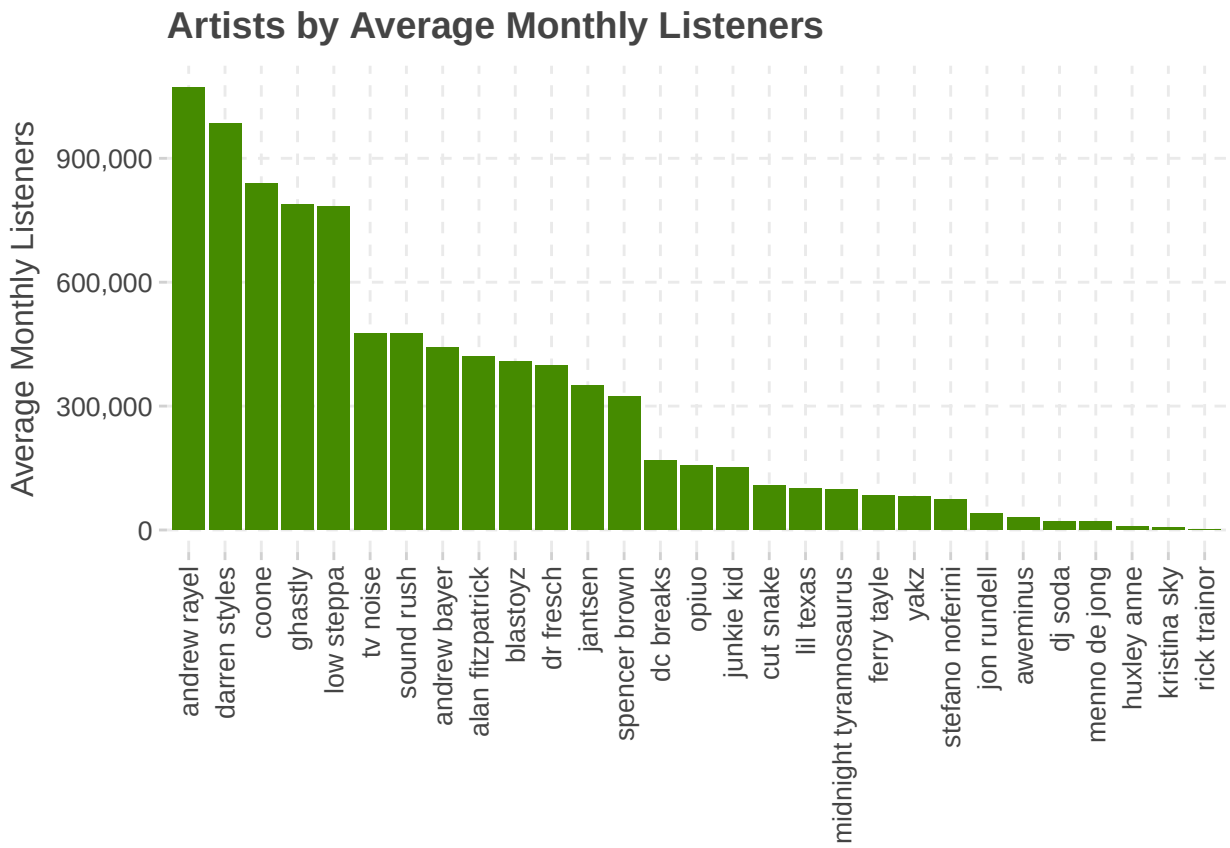
```
cat(unique(artists$artist), sep = ", ")
```

```
## alan fitzpatrick, andrew bayer, andrew rayel, aweminus, blastoyz, coone, cut snake, darren styles, d
```

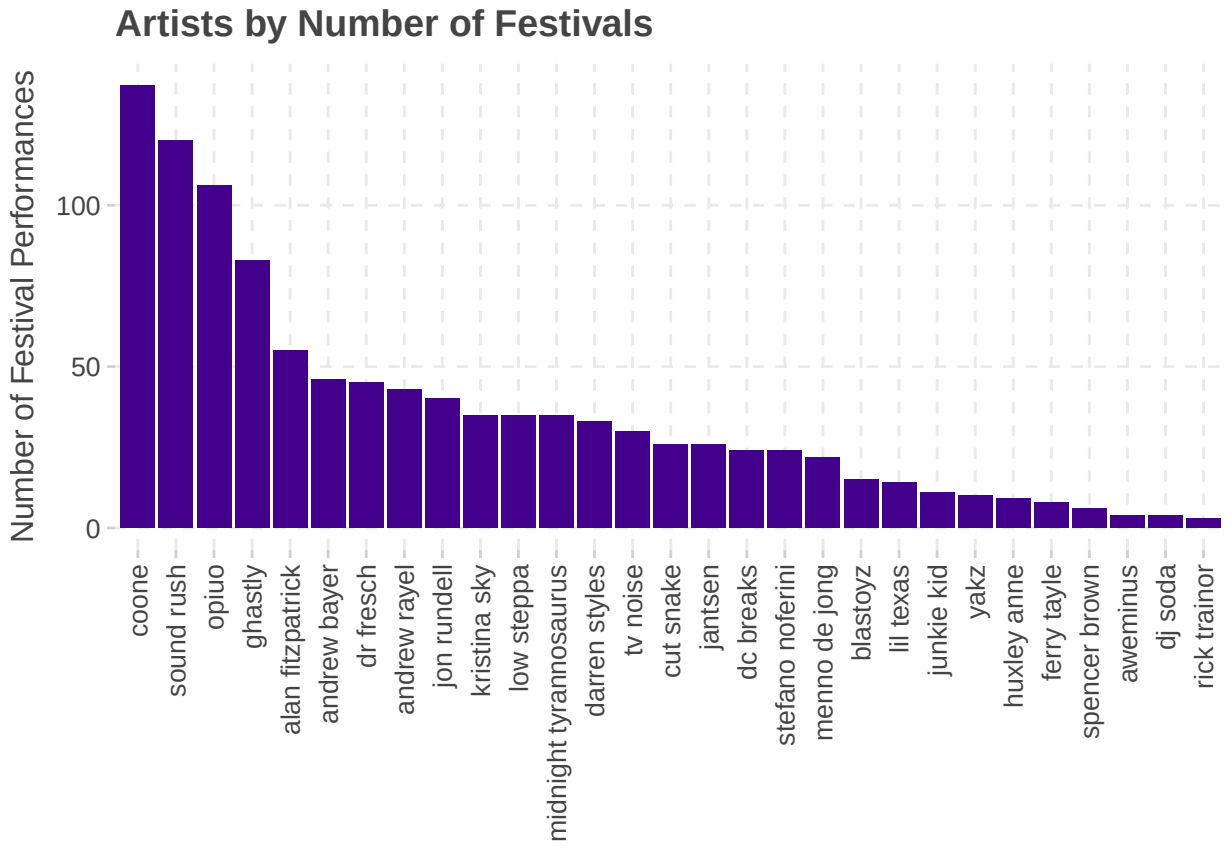
```
ggplot(artists_summary, aes(y = av_followers, x = reorder(artist, -av_followers))) +
  geom_bar(stat = "identity", fill = "red4") +
  scale_y_continuous(limits = c(0, 125000), labels = scales::comma) +
  ylab("Average Followers") +
  ggtitle("Artists by Average Followers") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1),
        axis.line = element_blank(),
        axis.title.x = element_blank())
```




```
ggplot(artists_summary, aes(y = av_listeners, x = reorder(artist, -av_listeners))) +
  geom_bar(stat = "identity", fill = "chartreuse4") +
  scale_y_continuous(labels = scales::comma) +
  ylab("Average Monthly Listeners") +
  ggtitle("Artists by Average Monthly Listeners") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1),
        axis.line = element_blank(),
        axis.title.x = element_blank())
```

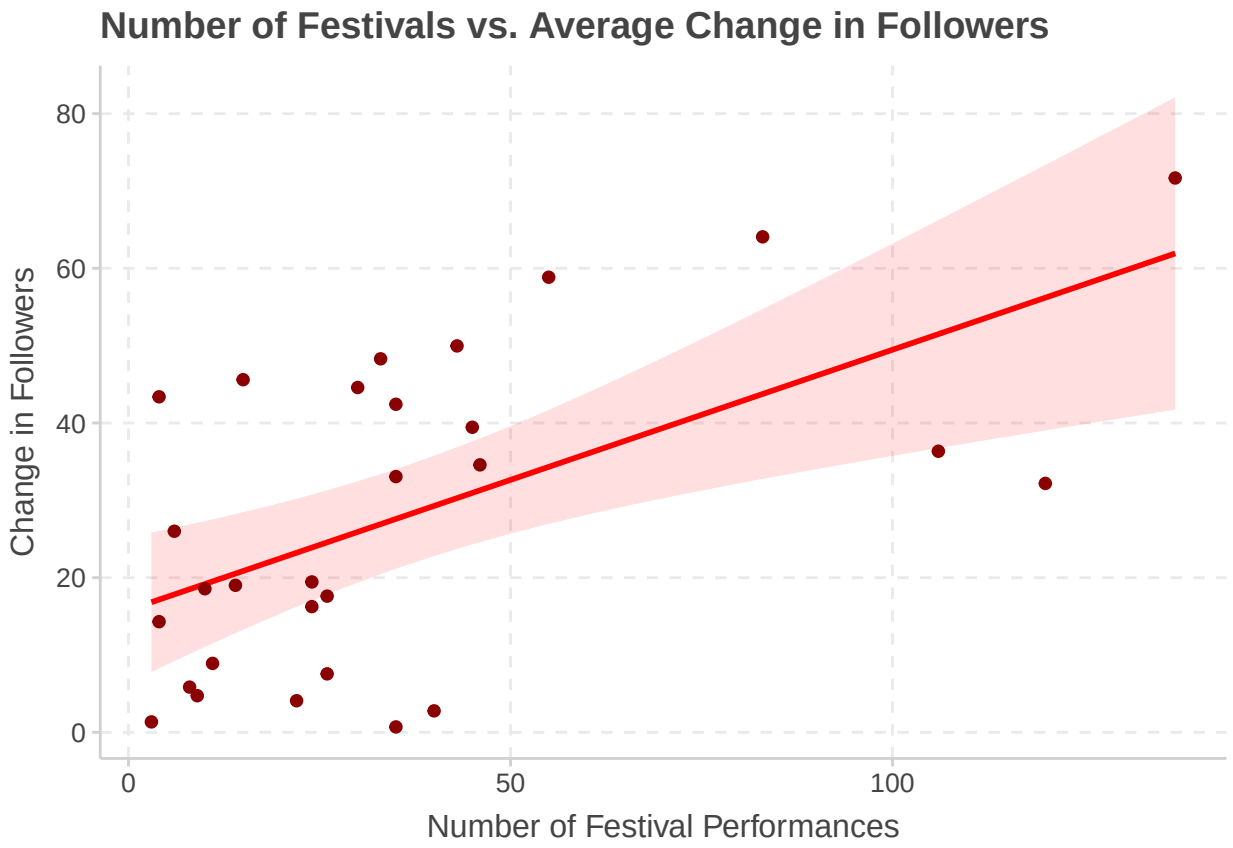



```
ggplot(artists_summary, aes(y = num_fest, x = reorder(artist, -num_fest))) +
  geom_bar(stat = "identity", fill = "#43008a") +
  scale_y_continuous(labels = scales::comma) +
  ylab("Number of Festival Performances") +
  ggtitle("Artists by Number of Festivals") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1),
        axis.line = element_blank(),
        axis.title.x = element_blank())
```



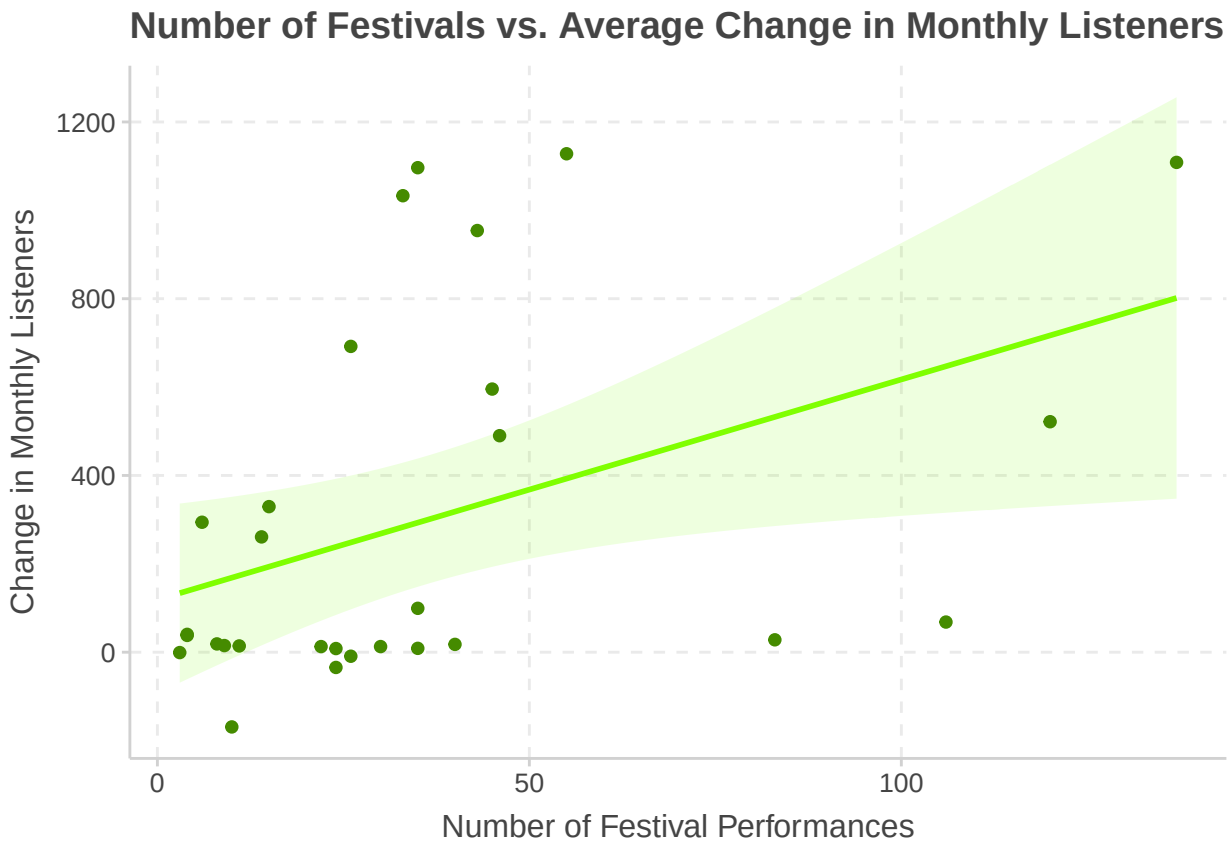

```
ggplot(artists_summary, aes(x = num_fest, y = av_fol_change)) +
  geom_smooth(method = "lm", fill = "red", color = "red", alpha = 0.13) +
  geom_point(color = "red4", size = 1.7) +
  xlab("Number of Festival Performances") +
  ylab("Change in Followers") +
  ggtitle("Number of Festivals vs. Average Change in Followers")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
ggplot(artists_summary, aes(x = num_fest, y = av_lis_change)) +
  geom_smooth(method = "lm", fill = "chartreuse", color = "chartreuse", alpha = 0.13) +
  geom_point(color = "chartreuse4", size = 1.7) +
  xlab("Number of Festival Performances") +
  ylab("Change in Monthly Listeners") +
  ggtitle("Number of Festivals vs. Average Change in Monthly Listeners")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
lm(av_fol_change ~ num_fest, data = artists_summary) %>% summary()
```

```
##
## Call:
## lm(formula = av_fol_change ~ num_fest, data = artists_summary)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -26.896 -14.109  -1.511  14.827  26.227
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 15.81436    4.59142   3.444  0.00188 **
## num_fest     0.33658    0.09252   3.638  0.00114 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.93 on 27 degrees of freedom
## Multiple R-squared:  0.329,  Adjusted R-squared:  0.3041
## F-statistic: 13.24 on 1 and 27 DF,  p-value: 0.001143
```

```
lm(av_fol_change ~ num_fest + av_followers, data = artists_summary) %>% summary()
```

```
##
## Call:
## lm(formula = av_fol_change ~ num_fest + av_followers, data = artists_summary)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -15.650  -9.145  -1.795   7.363  21.529
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.071e+01  2.939e+00   3.645  0.00117 **
## num_fest     -5.715e-03  7.679e-02  -0.074  0.94124
## av_followers  5.757e-04  8.617e-05   6.681 4.35e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.47 on 26 degrees of freedom
## Multiple R-squared:  0.753,  Adjusted R-squared:  0.734
## F-statistic: 39.64 on 2 and 26 DF,  p-value: 1.272e-08
```

```
lm(av_lis_change ~ num_fest, data = artists_summary) %>% summary()
```

```
##
## Call:
## lm(formula = av_lis_change ~ num_fest, data = artists_summary)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -578.9 -255.6 -139.9  145.5  803.3
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  118.741    103.272   1.150   0.2603
## num_fest       4.985      2.081   2.395   0.0238 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 380.8 on 27 degrees of freedom
## Multiple R-squared:  0.1753, Adjusted R-squared:  0.1447
## F-statistic: 5.738 on 1 and 27 DF,  p-value: 0.0238
```

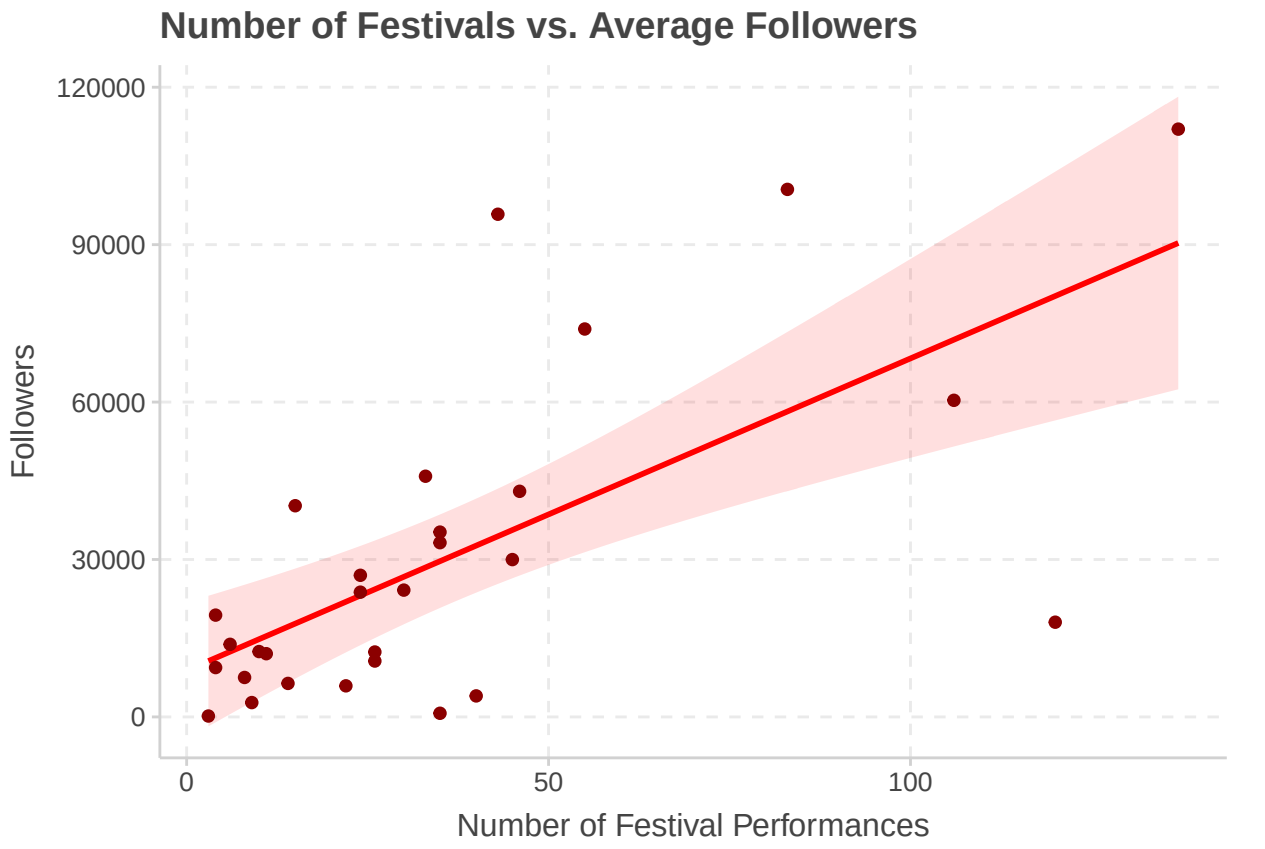
```
lm(av_lis_change ~ num_fest + av_listeners, data = artists_summary) %>% summary()
```

```
##
## Call:
## lm(formula = av_lis_change ~ num_fest + av_listeners, data = artists_summary)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -772.90 -103.50    6.61   52.57  706.20
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.498e+01  7.613e+01  -0.328    0.745
## num_fest      5.667e-01  1.649e+00   0.344    0.734
## av_listeners  9.885e-04  1.796e-04   5.505 8.91e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 263.7 on 26 degrees of freedom
```

```
## Multiple R-squared:  0.6192, Adjusted R-squared:  0.5899
## F-statistic: 21.13 on 2 and 26 DF,  p-value: 3.545e-06
```

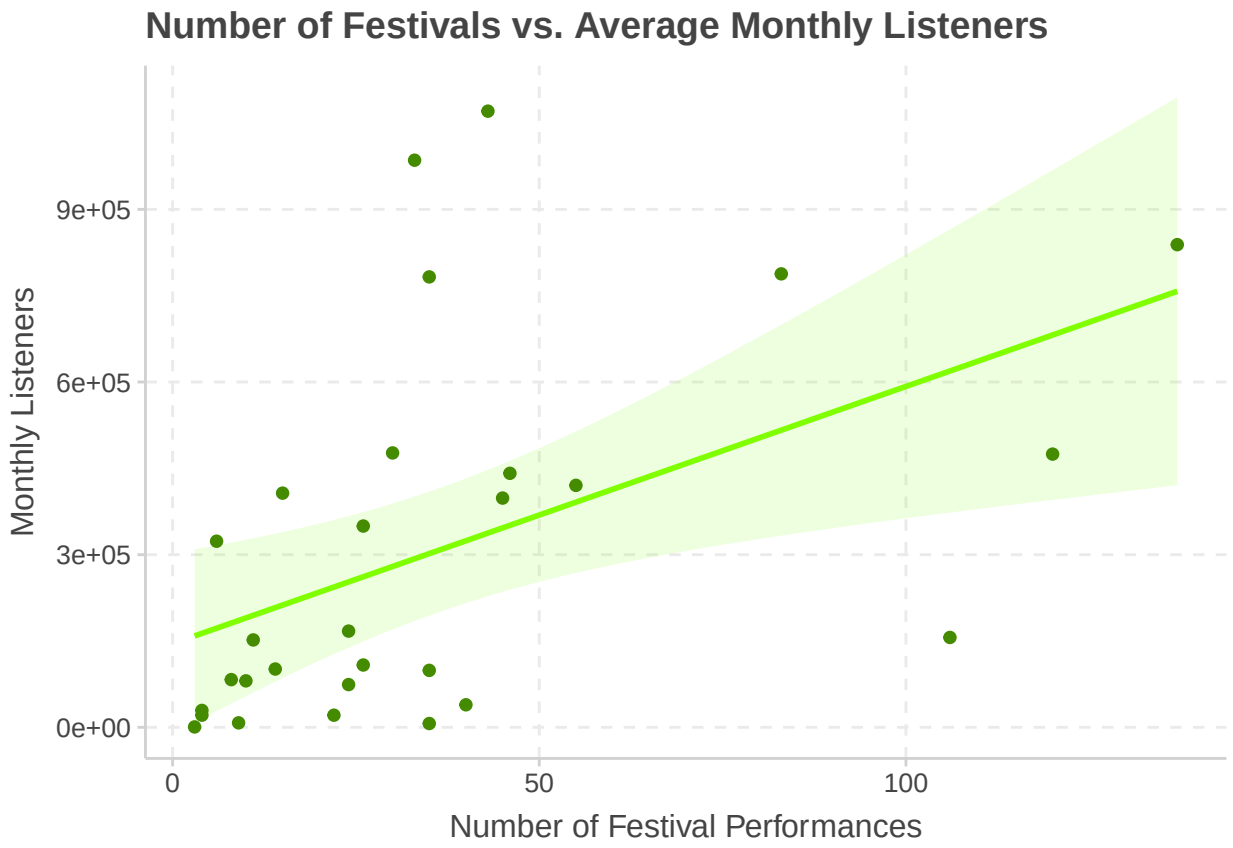
```
ggplot(artists_summary, aes(x = num_fest, y = av_followers)) +
  geom_smooth(method = "lm", fill = "red", color = "red", alpha = 0.13) +
  geom_point(color = "red4", size = 1.7) +
  xlab("Number of Festival Performances") +
  ylab("Followers") +
  ggtitle("Number of Festivals vs. Average Followers")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



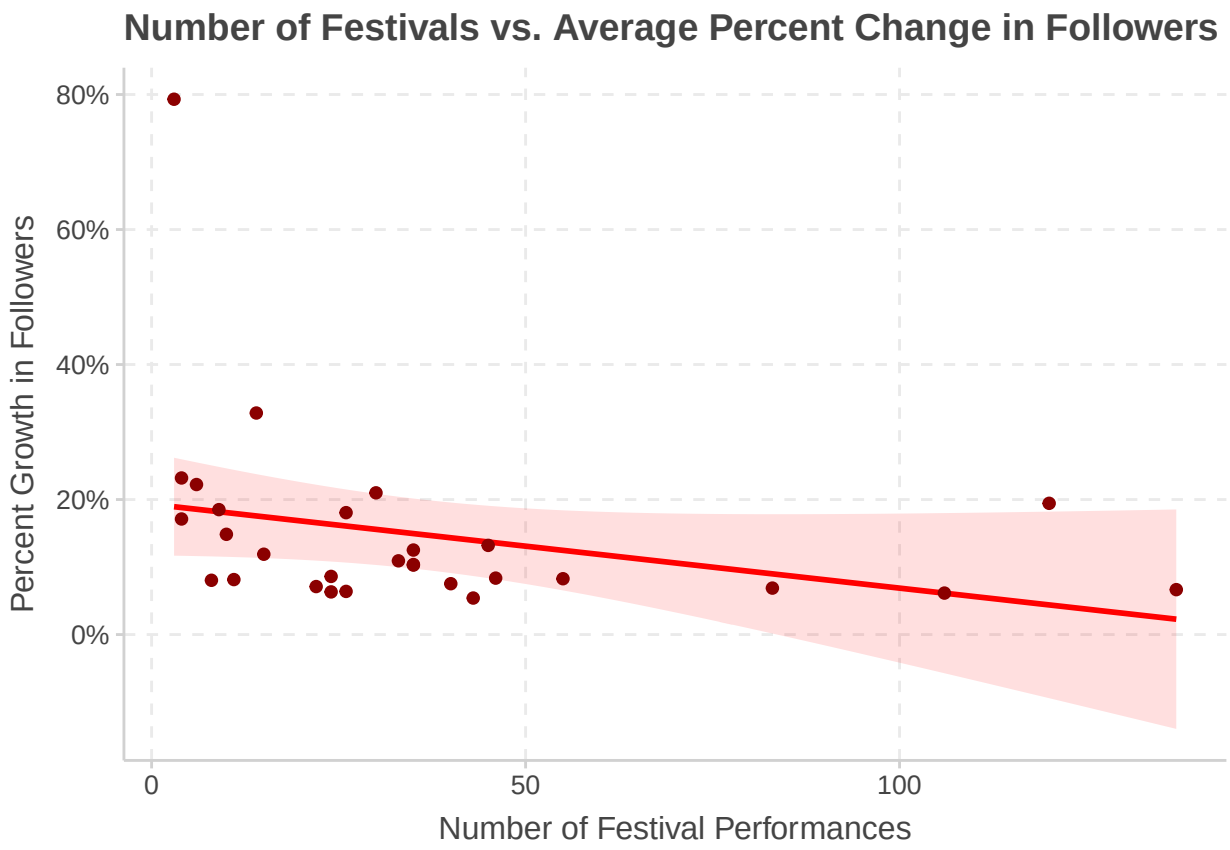

```
ggplot(artists_summary, aes(x = num_fest, y = av_listeners)) +
  geom_smooth(method = "lm", fill = "chartreuse", color = "chartreuse", alpha = 0.13) +
  geom_point(color = "chartreuse4", size = 1.7) +
  xlab("Number of Festival Performances") +
  ylab("Monthly Listeners") +
  ggtitle("Number of Festivals vs. Average Monthly Listeners")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



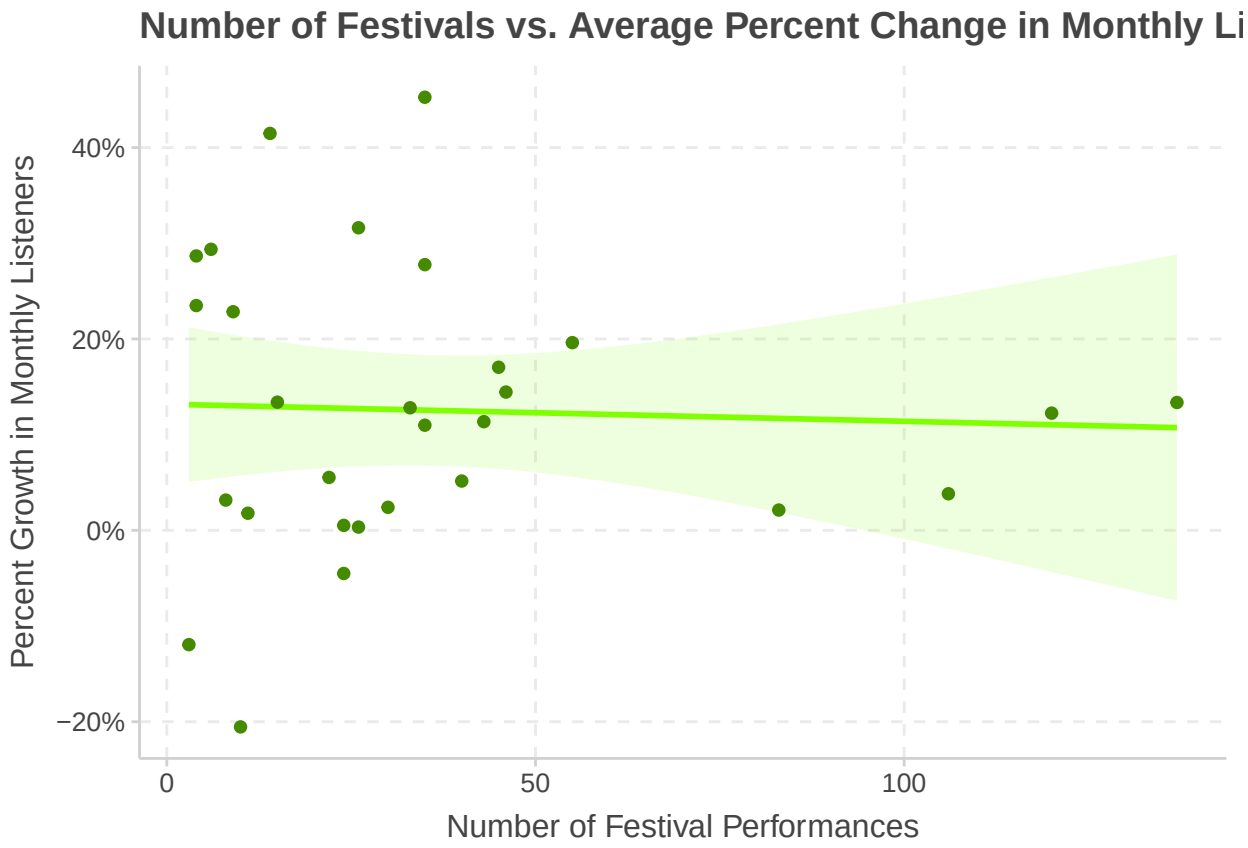

```
ggplot(artists_summary, aes(x = num_fest, y = av_pct_fol_change)) +
  geom_smooth(method = "lm", fill = "red", color = "red", alpha = 0.13) +
  geom_point(color = "red4", size = 1.7) +
  scale_y_continuous(labels = scales::percent) +
  xlab("Number of Festival Performances") +
  ylab("Percent Growth in Followers") +
  ggtitle("Number of Festivals vs. Average Percent Change in Followers")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
ggplot(artists_summary, aes(x = num_fest, y = av_pct_lis_change)) +
  geom_smooth(method = "lm", fill = "chartreuse", color = "chartreuse", alpha = 0.13) +
  geom_point(color = "chartreuse4", size = 1.7) +
  scale_y_continuous(labels = scales::percent) +
  xlab("Number of Festival Performances") +
  ylab("Percent Growth in Monthly Listeners") +
  ggtitle("Number of Festivals vs. Average Percent Change in Monthly Listeners")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
lm(av_pct_fol_change ~ num_fest, data = artists_summary) %>% summary()
```

```
##
## Call:
## lm(formula = av_pct_fol_change ~ num_fest, data = artists_summary)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.10277 -0.06806 -0.03217  0.01980  0.60362
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.1931114  0.0369565   5.225 1.66e-05 ***
## num_fest    -0.0012434  0.0007447  -1.670    0.107
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1363 on 27 degrees of freedom
## Multiple R-squared:  0.09359,    Adjusted R-squared:  0.06002
## F-statistic: 2.788 on 1 and 27 DF,  p-value: 0.1065
```

```
lm(av_pct_fol_change ~ num_fest + av_followers, data = artists_summary) %>% summary()
```

```
##
## Call:
## lm(formula = av_pct_fol_change ~ num_fest + av_followers, data = artists_summary)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -0.11631 -0.07629 -0.01692  0.03608  0.58947
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.051e-01  3.794e-02   5.406 1.16e-05 ***
## num_fest     -4.402e-04  9.910e-04  -0.444    0.661
## av_followers -1.351e-06  1.112e-06  -1.215    0.235
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1351 on 26 degrees of freedom
## Multiple R-squared:  0.1423, Adjusted R-squared:  0.07628
## F-statistic: 2.156 on 2 and 26 DF,  p-value: 0.136
```

```
lm(av_pct_lis_change ~ num_fest, data = artists_summary) %>% summary()
```

```
##
## Call:
## lm(formula = av_pct_lis_change ~ num_fest, data = artists_summary)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33549 -0.09877  0.00212  0.09821  0.32702
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.1318492  0.0411488   3.204  0.00346 **
## num_fest    -0.0001789  0.0008291  -0.216  0.83081
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1517 on 27 degrees of freedom
## Multiple R-squared:  0.001721,   Adjusted R-squared:  -0.03525
## F-statistic: 0.04655 on 1 and 27 DF,  p-value: 0.8308
```

```
lm(av_pct_lis_change ~ num_fest + av_followers, data = artists_summary) %>% summary()
```

```
##
## Call:
## lm(formula = av_pct_lis_change ~ num_fest + av_followers, data = artists_summary)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33575 -0.09944  0.00402  0.09696  0.32762
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.328e-01  4.342e-02   3.059   0.0051 **
## num_fest     -1.142e-04  1.134e-03  -0.101   0.9206
## av_followers -1.088e-07  1.273e-06  -0.086   0.9325
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1546 on 26 degrees of freedom
```

```
## Multiple R-squared:  0.002002,   Adjusted R-squared:  -0.07477
## F-statistic: 0.02607 on 2 and 26 DF,  p-value: 0.9743
```

```
artists <-  artists %>% group_by(artist, DateTime) %>%
  mutate(days_before_next_fest = max(as.numeric(DateTime - unlist(festivals[festivals$Artist == artist &
                                                                              festivals$Date > DateTim
         days_before_next_song = max(as.numeric(DateTime - unlist(songs[songs$artist_name == artist &
                                                                        songs$release_date > Dat

artists <- artists %>% mutate(festival_before_4 = days_before_next_fest >= -4,
                              festival_before_7 = days_before_next_fest >= -7,
                              festival_before_14 = days_before_next_fest >= -14,
                              song_within_31 = closest_song_days <= 31,
                              song_within_45 = closest_song_days <= 45,
                              song_before_4 = days_before_next_song >= -4,
                              song_before_7 = days_before_next_song >= -7,
                              song_before_14 = days_before_next_song >= -14)
```

```
change_fol_fest <- artists %>% group_by(artist) %>%
  summarise(before_fest_pct_change =
              mean(unlist(artists[artists$artist == artist & artists$festival_before_7 & !artists$song_
            after_fest_pct_change =
              mean(unlist(artists[artists$artist == artist & artists$festival_within_7 & !artists$song_
            fest_change_in_pct_change =
              after_fest_pct_change - before_fest_pct_change) %>%
  arrange(-fest_change_in_pct_change)
```

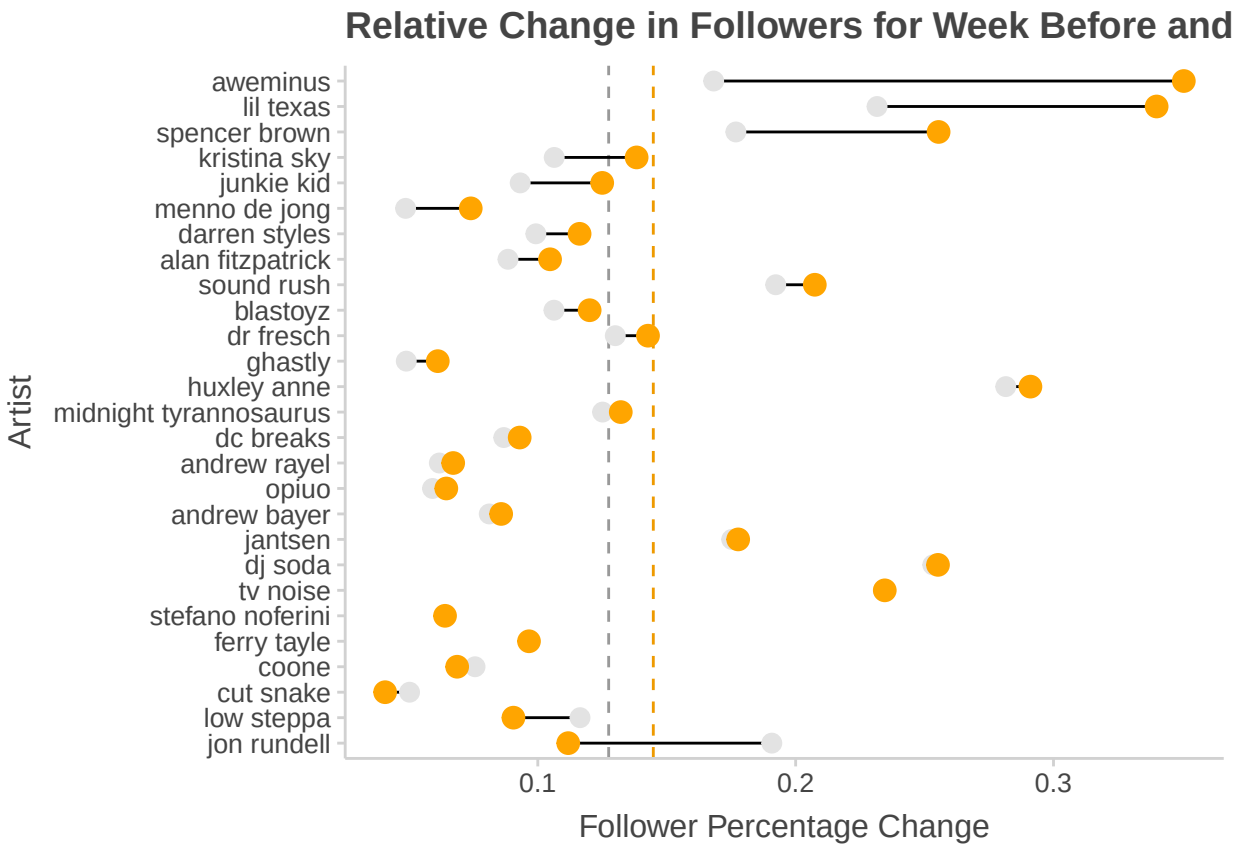
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
change_fol_fest <- na.omit(change_fol_fest)
```

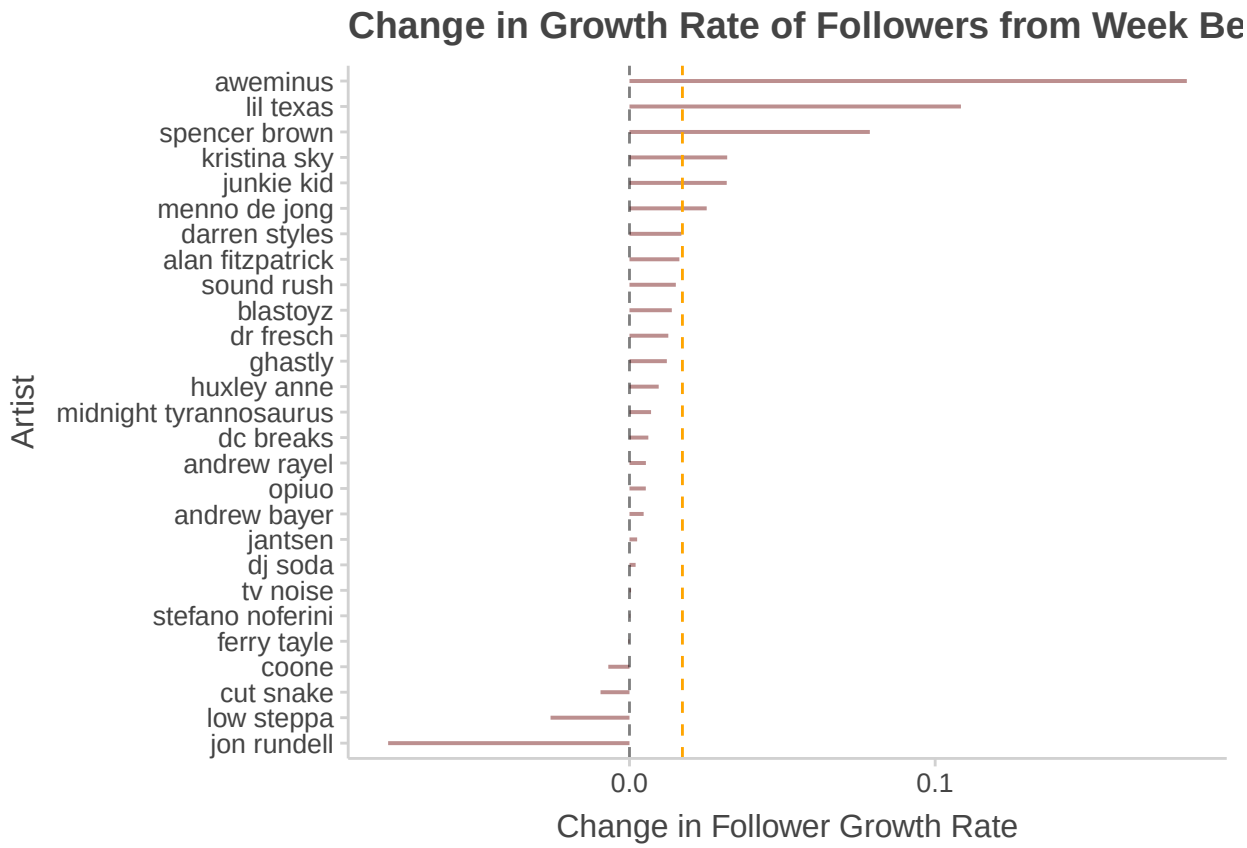
```
library(ggalt)
```

```
## Registered S3 methods overwritten by 'ggalt':
##   method                  from
##   grid.draw.absoluteGrob  ggplot2
##   grobHeight.absoluteGrob ggplot2
##   grobWidth.absoluteGrob  ggplot2
##   grobX.absoluteGrob      ggplot2
##   grobY.absoluteGrob      ggplot2
```

```
#815 530
ggplot(change_fol_fest, aes(x=before_fest_pct_change, xend=after_fest_pct_change, y=reorder(artist, fes
  geom_vline(xintercept = mean(change_fol_fest$before_fest_pct_change), color = "grey60", linetype = 2)
  geom_vline(xintercept = mean(change_fol_fest$after_fest_pct_change), color = "orange2", linetype = 2)
  geom_dumbbell(colour_x="grey89",
                colour_xend = "orange1",
                size_x = 3,
                size_xend = 3.5) +
  xlab("Follower Percentage Change") +
  ylab("Artist") +
  ggtitle("Relative Change in Followers for Week Before and Week After Festival") +
  theme(panel.grid = element_blank())
```




```
#875 530
ggplot(change_fol_fest, aes(x = fest_change_in_pct_change,
                            y = reorder(artist, fest_change_in_pct_change))) +
  geom_segment(data = change_fol_fest,
               aes(x=0,
                   xend=fest_change_in_pct_change,
                   y=reorder(artist, fest_change_in_pct_change),
                   yend=reorder(artist, fest_change_in_pct_change)),
               size = 0.7,
               color = "rosybrown") +
  geom_vline(xintercept = 0, color = "black", linetype = 2, alpha = 0.5) +
  geom_vline(xintercept = mean(change_fol_fest$fest_change_in_pct_change), linetype = 2, color = "orang
  xlab("Change in Follower Growth Rate") +
  ylab("Artist") +
  ggtitle("Change in Growth Rate of Followers from Week Before to Week After Festival") +
  theme(panel.grid = element_blank())
```




```
mean(change_fol_fest$before_fest_pct_change)
```

```
## [1] 0.1274761
```

```
mean(change_fol_fest$after_fest_pct_change)
```

```
## [1] 0.1447912
```

```
mean(change_fol_fest$fest_change_in_pct_change)
```

```
## [1] 0.01731518
```

```
sd(change_fol_fest$fest_change_in_pct_change)
```

```
## [1] 0.04572213
```

```
mean(artists$fol_pct_change)
```

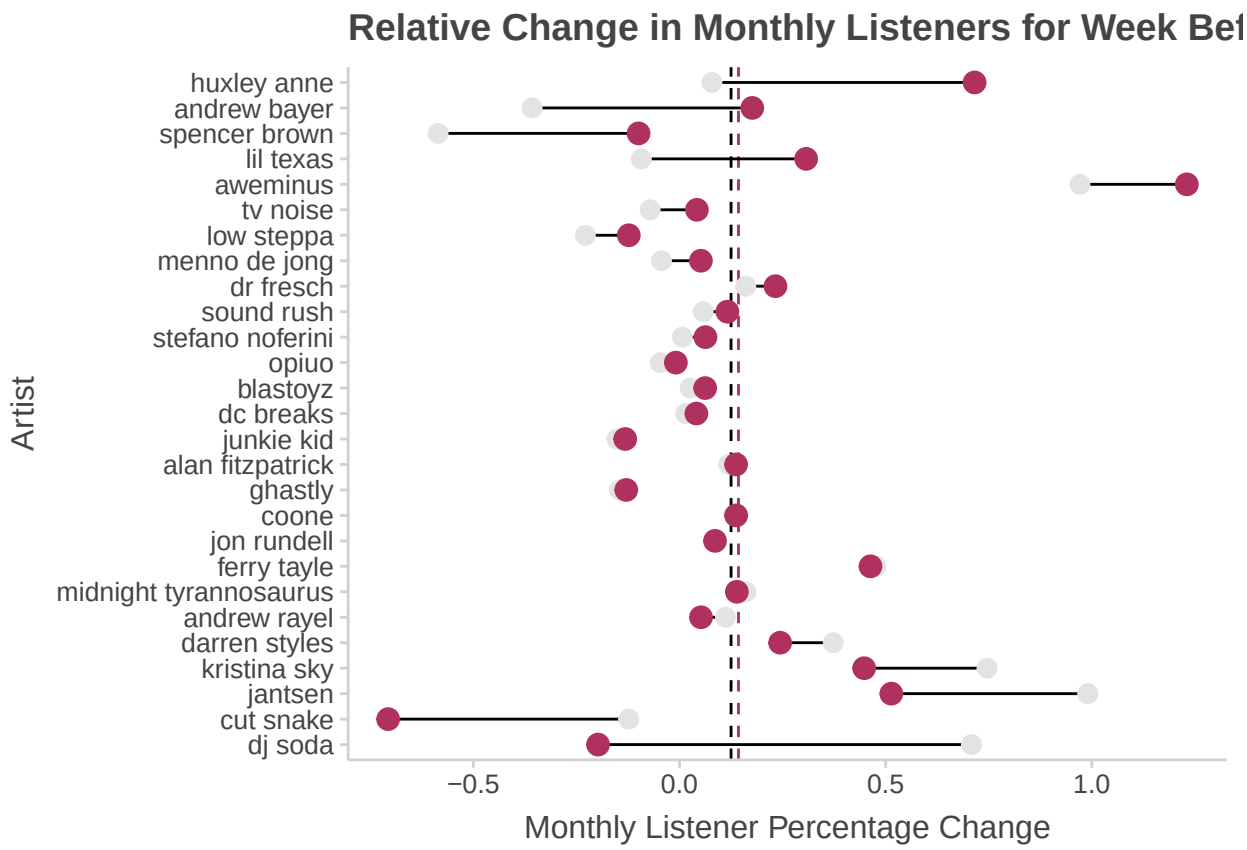
```
## [1] 0.1175139
```

```
change_lis_fest <- artists %>% group_by(artist) %>%
  summarise(before_fest_pct_change =
              mean(unlist(artists[artists$artist == artist &
                                  artists$festival_before_7 &
                                  !artists$song_within_7, "lis_pct_change"])),
            after_fest_pct_change =
              mean(unlist(artists[artists$artist == artist & artists$festival_within_7 & !artists$song_
            fest_change_in_pct_change =
              after_fest_pct_change - before_fest_pct_change) %>%
  arrange(-fest_change_in_pct_change)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

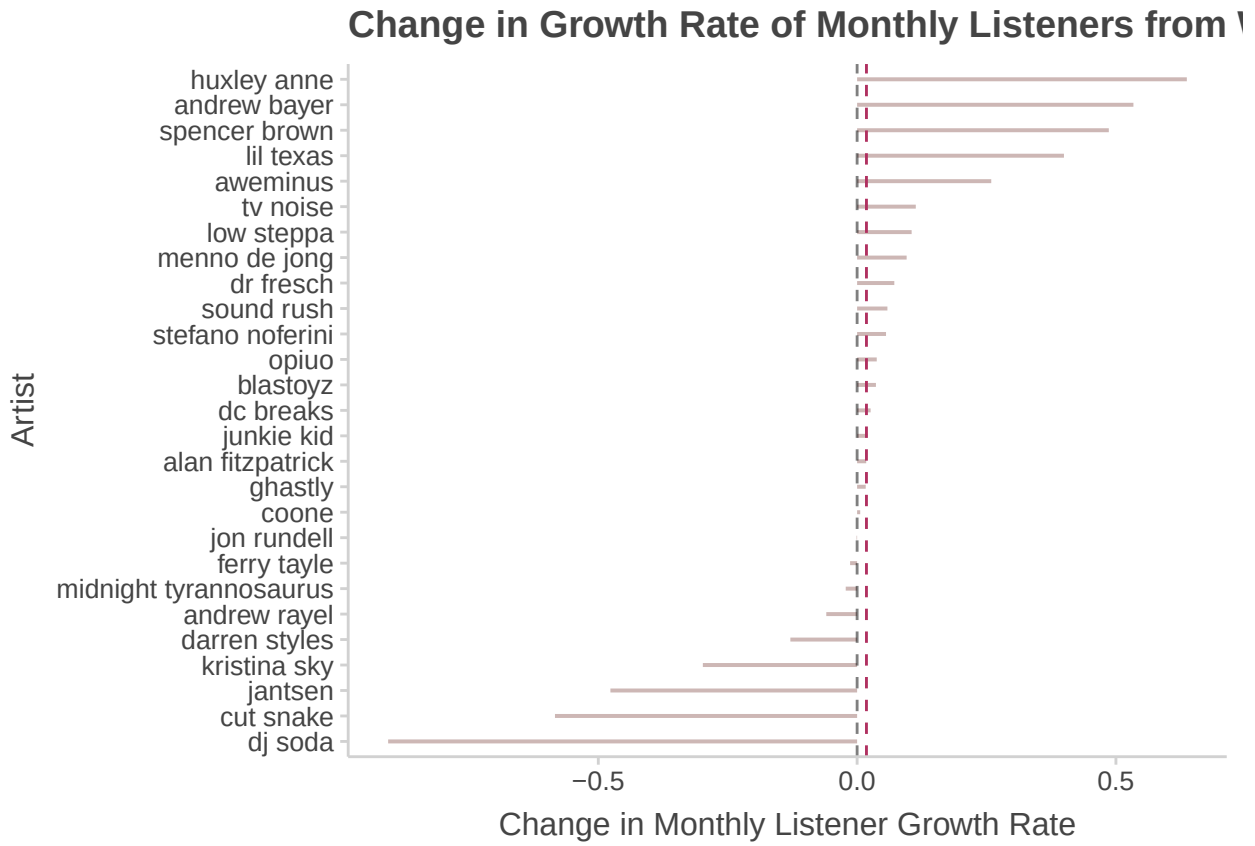
```
change_lis_fest <- na.omit(change_lis_fest)
```

```
ggplot(change_lis_fest, aes(x=before_fest_pct_change, xend=after_fest_pct_change, y=reorder(artist, fes
  geom_vline(xintercept = mean(change_lis_fest$before_fest_pct_change), color = "black", linetype = 2) +
  geom_vline(xintercept = mean(change_lis_fest$after_fest_pct_change), color = "maroon", linetype = 2) +
  geom_dumbbell(colour_x="grey89",
                colour_xend = "maroon",
                size_x = 3,
                size_xend = 3.5) +
  xlab("Monthly Listener Percentage Change") +
  ylab("Artist") +
  ggtitle("Relative Change in Monthly Listeners for Week Before and Week After Festival") +
  theme(panel.grid = element_blank())
```




```
ggplot(change_lis_fest, aes(x = fest_change_in_pct_change,
                            y = reorder(artist, fest_change_in_pct_change))) +
  geom_segment(data = change_lis_fest,
               aes(x=0,
                   xend=fest_change_in_pct_change,
                   y=reorder(artist, fest_change_in_pct_change),
                   yend=reorder(artist, fest_change_in_pct_change)),
               size = 0.7,
               color = "mistyrose3") +
  geom_vline(xintercept = 0, color = "black", linetype = 2, alpha = 0.5) +
  geom_vline(xintercept = mean(change_lis_fest$fest_change_in_pct_change), linetype = 2, color = "maroo
  xlab("Change in Monthly Listener Growth Rate") +
```

```
  ylab("Artist") +
  ggtitle("Change in Growth Rate of Monthly Listeners from Week Before to Week After Festival") +
  theme(panel.grid = element_blank())
```




```
mean(change_lis_fest$before_fest_pct_change)
```

```
## [1] 0.1250052
```

```
mean(change_lis_fest$after_fest_pct_change)
```

```
## [1] 0.1430691
```

```
mean(change_lis_fest$fest_change_in_pct_change)
```

```
## [1] 0.01806389
```

```
sd(change_lis_fest$fest_change_in_pct_change)
```

```
## [1] 0.3218304
```

```
mean(artists$lis_pct_change)
```

```
## [1] 0.1369414
```

```
lm(fol_pct_change ~ festival_within_7 + song_within_7 + festival_before_7, data = artists) %>% summary()
```

```
##
## Call:
## lm(formula = fol_pct_change ~ festival_within_7 + song_within_7 +
##     festival_before_7, data = artists)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5291 -0.0570 -0.0237  0.0288  6.1426
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.109515   0.001366  80.161  < 2e-16 ***
## festival_within_7TRUE 0.020076   0.003420   5.870 4.44e-09 ***
## song_within_7TRUE     0.024150   0.002956   8.169 3.30e-16 ***
## festival_before_7TRUE 0.009724   0.003608   2.695  0.00704 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1572 on 19281 degrees of freedom
## Multiple R-squared:  0.005861,   Adjusted R-squared:  0.005706
## F-statistic: 37.89 on 3 and 19281 DF,  p-value: < 2.2e-16
```

```
lm(fol_pct_change ~ festival_within_14 + song_within_14 + festival_before_14, data = artists) %>% summa
```

```
##
## Call:
## lm(formula = fol_pct_change ~ festival_within_14 + song_within_14 +
##     festival_before_14, data = artists)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5250 -0.0563 -0.0240  0.0292  6.1485
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.105461   0.001535  68.708  < 2e-16 ***
## festival_within_14TRUE 0.016940   0.002815   6.018  1.8e-09 ***
## song_within_14TRUE     0.020769   0.002467   8.418  < 2e-16 ***
## festival_before_14TRUE 0.011001   0.002889   3.808 0.000141 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1571 on 19281 degrees of freedom
## Multiple R-squared:  0.007028,   Adjusted R-squared:  0.006873
## F-statistic: 45.49 on 3 and 19281 DF,  p-value: < 2.2e-16
```

```
artists$weeks_af_fest <- ifelse(artists$closest_fest_days == 0, 0, ifelse(artists$closest_fest_days %% 
artists$weeks_af_song <- ifelse(artists$closest_song_days == 0, 0, ifelse(artists$closest_song_days %% 
```

```
artists$week_af_fest_1 <- artists$weeks_af_fest == 0
artists$week_af_fest_2 <- artists$weeks_af_fest == 1
artists$week_af_fest_3 <- artists$weeks_af_fest == 2
artists$week_af_song_1 <- artists$weeks_af_song == 0
artists$week_af_song_2 <- artists$weeks_af_song == 1
artists$week_af_song_3 <- artists$weeks_af_song == 2
artists$week_af_song_4 <- artists$weeks_af_song == 3
artists$week_af_song_5 <- artists$weeks_af_song == 4
artists$week_af_song_6 <- artists$weeks_af_song == 5
artists$week_af_song_7 <- artists$weeks_af_song == 6
```

```r
artists$week_af_song_8 <- artists$weeks_af_song == 7
artists$week_af_song_9 <- artists$weeks_af_song == 8
artists$week_af_song_10 <- artists$weeks_af_song == 9

lm(fol_pct_change ~ week_af_fest_1 + week_af_fest_2 + week_af_fest_3 +
     week_af_song_1 + week_af_song_2 + week_af_song_3 + week_af_song_4 +
     festival_before_7, data = artists) %>% summary()
```

```
##
## Call:
## lm(formula = fol_pct_change ~ week_af_fest_1 + week_af_fest_2 +
##     week_af_fest_3 + week_af_song_1 + week_af_song_2 + week_af_song_3 +
##     week_af_song_4 + festival_before_7, data = artists)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5230 -0.0567 -0.0239  0.0288  6.1376
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.103411   0.001717  60.236  < 2e-16 ***
## week_af_fest_1TRUE    0.021237   0.003475   6.111 1.01e-09 ***
## week_af_fest_2TRUE    0.013294   0.004132   3.217  0.00130 **
## week_af_fest_3TRUE    0.004913   0.004510   1.089  0.27600
## week_af_song_1TRUE    0.028654   0.003102   9.236  < 2e-16 ***
## week_af_song_2TRUE    0.016105   0.003597   4.477 7.63e-06 ***
## week_af_song_3TRUE    0.011658   0.003940   2.959  0.00309 **
## week_af_song_4TRUE    0.009835   0.004345   2.264  0.02361 *
## festival_before_7TRUE 0.007974   0.003626   2.199  0.02787 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1571 on 19276 degrees of freedom
## Multiple R-squared:  0.00783,    Adjusted R-squared:  0.007418
## F-statistic: 19.01 on 8 and 19276 DF,  p-value: < 2.2e-16
```

```r
lm(lis_pct_change ~ week_af_fest_1 + week_af_fest_2 + week_af_fest_3 +
     week_af_song_1 + week_af_song_2 + week_af_song_3 + week_af_song_4 +
     festival_before_7, data = artists) %>% summary()
```

```
##
## Call:
## lm(formula = lis_pct_change ~ week_af_fest_1 + week_af_fest_2 +
##     week_af_fest_3 + week_af_song_1 + week_af_song_2 + week_af_song_3 +
##     week_af_song_4 + festival_before_7, data = artists)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -44.68  -0.38  -0.09   0.17 392.33
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -0.06137    0.04165  -1.473 0.140635
## week_af_fest_1TRUE     0.06538    0.08431   0.775 0.438077
```

```
## week_af_fest_2TRUE     0.31837    0.10024   3.176 0.001496 **
## week_af_fest_3TRUE     0.06363    0.10942   0.582 0.560903
## week_af_song_1TRUE     0.40201    0.07526   5.341 9.33e-08 ***
## week_af_song_2TRUE     0.28755    0.08727   3.295 0.000986 ***
## week_af_song_3TRUE     0.31347    0.09559   3.279 0.001042 **
## week_af_song_4TRUE     0.21492    0.10541   2.039 0.041466 *
## festival_before_7TRUE  0.03523    0.08796   0.401 0.688740
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.81 on 19276 degrees of freedom
## Multiple R-squared:  0.002597,   Adjusted R-squared:  0.002183
## F-statistic: 6.273 on 8 and 19276 DF,  p-value: 3.859e-08
```

```
lm(fol_pct_change ~ festival_within_7 + song_within_7, data = artists) %>% summary()
```

```
##
## Call:
## lm(formula = fol_pct_change ~ festival_within_7 + song_within_7,
##     data = artists)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5300 -0.0569 -0.0235  0.0290  6.1500
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.110409   0.001325  83.301  < 2e-16 ***
## festival_within_7TRUE 0.021445   0.003383   6.339 2.36e-10 ***
## song_within_7TRUE     0.024339   0.002956   8.234  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1572 on 19282 degrees of freedom
## Multiple R-squared:  0.005486,   Adjusted R-squared:  0.005383
## F-statistic: 53.19 on 2 and 19282 DF,  p-value: < 2.2e-16
```

```
library(plm)
```

```
##
## Attaching package: 'plm'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, lag, lead
```

```
fixed_model_fol <- plm(fol_pct_change ~ week_af_fest_1 + week_af_fest_2 + week_af_fest_3 +
     week_af_song_1 + week_af_song_2 + week_af_song_3 + week_af_song_4 +
     festival_before_7,
     data = artists,
     index = ("artist"),
     model = "within",
     effect = "individual")

summary(fixed_model_fol)
```

```
## Warning in Ops.pseries(y, bX): indexes of pseries have same length but not same
```

```
## content: result was assigned first operand's index

## Warning in Ops.pseries(y, bX): indexes of pseries have same length but not same
## content: result was assigned first operand's index
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = fol_pct_change ~ week_af_fest_1 + week_af_fest_2 +
##     week_af_fest_3 + week_af_song_1 + week_af_song_2 + week_af_song_3 +
##     week_af_song_4 + festival_before_7, data = artists, effect = "individual",
##     model = "within", index = ("artist"))
##
## Unbalanced Panel: n = 29, T = 20-761, N = 19285
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -2.6186546 -0.0319250 -0.0063595  0.0217194  6.0769152
##
## Coefficients:
##                         Estimate Std. Error t-value  Pr(>|t|)
## week_af_fest_1TRUE    0.0202670  0.0033546  6.0415 1.555e-09 ***
## week_af_fest_2TRUE    0.0134913  0.0038722  3.4841 0.0004948 ***
## week_af_fest_3TRUE    0.0042030  0.0041897  1.0032 0.3157848
## week_af_song_1TRUE    0.0241546  0.0029704  8.1319 4.482e-16 ***
## week_af_song_2TRUE    0.0120240  0.0033965  3.5402 0.0004008 ***
## week_af_song_3TRUE    0.0091545  0.0036954  2.4773 0.0132472 *
## week_af_song_4TRUE    0.0086591  0.0040503  2.1379 0.0325394 *
## festival_before_7TRUE 0.0067482  0.0034233  1.9712 0.0487109 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    400.11
## Residual Sum of Squares: 397.71
## R-Squared:      0.005992
## Adj. R-Squared: 0.0041328
## F-statistic: 14.5036 on 8 and 19248 DF, p-value: < 2.22e-16
```

```
fixef(fixed_model_fol)
```

```
##       alan fitzpatrick           andrew bayer           andrew rayel
##               0.070128               0.067289               0.041323
##               aweminus               blastoyz                  coone
##               0.159439               0.100904               0.049304
##              cut snake          darren styles              dc breaks
##               0.056146               0.097215               0.080081
##                dj soda              dr fresch            ferry tayle
##               0.229277               0.116993               0.071507
##                ghastly            huxley anne                jantsen
##               0.052563               0.177839               0.171487
##            jon rundell             junkie kid           kristina sky
##               0.065567               0.075741               0.093512
##              lil texas             low steppa          menno de jong
##               0.308294               0.089770               0.067332
## midnight tyrannosaurus                  opiuo           rick trainor
```

```
##               0.107624              0.045416              0.792998
##             sound rush         spencer brown      stefano noferini
##               0.168808              0.211965              0.054520
##               tv noise                  yakz
##               0.199078              0.129177
```

```
artists %>% filter(artist == "andrew bayer") %>% group_by(artist) %>% summarise(mean(fol_pct_change), me
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 1 x 4
##   artist      `mean(fol_pct_change~ `mean(Total.Follower~ `mean(followers_chang~
##   <chr>                       <dbl>                 <dbl>                   <dbl>
## 1 andrew bay~                0.0836                42992.                    34.6
```

```
artists %>% filter(artist == "andrew bayer", week_af_fest_1) %>% group_by(artist) %>% summarise(mean(fo
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 1 x 4
##   artist      `mean(fol_pct_change~ `mean(Total.Follower~ `mean(followers_chang~
##   <chr>                       <dbl>                 <dbl>                   <dbl>
## 1 andrew bay~                 0.107                38699.                      40
```

```
artists %>% filter(artist == "lil texas") %>% group_by(artist) %>% summarise(mean(fol_pct_change), mean
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 1 x 4
##   artist    `mean(fol_pct_change~ `mean(Total.Followers~ `mean(followers_change~
##   <chr>                     <dbl>                  <dbl>                   <dbl>
## 1 lil texas                 0.328                  6385.                    19.0
```

```
artists %>% filter(artist == "lil texas", week_af_fest_1) %>% group_by(artist) %>% summarise(mean(fol_p
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 1 x 4
##   artist    `mean(fol_pct_change~ `mean(Total.Followers~ `mean(followers_change~
##   <chr>                     <dbl>                  <dbl>                   <dbl>
## 1 lil texas                 0.346                  6446.                    22.1
```

```
fixed_model_lis <- plm(lis_pct_change ~ week_af_fest_1 + week_af_fest_2 + week_af_fest_3 +
     week_af_song_1 + week_af_song_2 + week_af_song_3 + week_af_song_4 +
     festival_before_7,
     data = artists,
     index = ("artist"),
     model = "within",
     effect = "individual")

summary(fixed_model_lis)
```

```
## Warning in Ops.pseries(y, bX): indexes of pseries have same length but not same
## content: result was assigned first operand's index

## Warning in Ops.pseries(y, bX): indexes of pseries have same length but not same
## content: result was assigned first operand's index
```

```
## Oneway (individual) effect Within Model
##
```

```
## Call:
## plm(formula = lis_pct_change ~ week_af_fest_1 + week_af_fest_2 +
##     week_af_fest_3 + week_af_song_1 + week_af_song_2 + week_af_song_3 +
##     week_af_song_4 + festival_before_7, data = artists, effect = "individual",
##     model = "within", index = ("artist"))
##
## Unbalanced Panel: n = 29, T = 20-761, N = 19285
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -44.931101  -0.405384  -0.082021   0.184328 392.085105
##
## Coefficients:
##                       Estimate Std. Error t-value  Pr(>|t|)
## week_af_fest_1TRUE    0.087185   0.088929  0.9804 0.3269048
## week_af_fest_2TRUE    0.322781   0.102650  3.1445 0.0016662 **
## week_af_fest_3TRUE    0.061788   0.111066  0.5563 0.5780009
## week_af_song_1TRUE    0.439221   0.078743  5.5779 2.467e-08 ***
## week_af_song_2TRUE    0.323358   0.090039  3.5913 0.0003298 ***
## week_af_song_3TRUE    0.350542   0.097963  3.5783 0.0003467 ***
## week_af_song_4TRUE    0.252424   0.107372  2.3509 0.0187365 *
## festival_before_7TRUE 0.054811   0.090750  0.6040 0.5458657
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    280240
## Residual Sum of Squares: 279490
## R-Squared:      0.0026532
## Adj. R-Squared: 0.00078785
## F-statistic: 6.4006 on 8 and 19248 DF, p-value: 2.4608e-08
```

```
fixef(fixed_model_lis)
```

```
##       alan fitzpatrick           andrew bayer           andrew rayel
##              -0.022484              -0.172068              -0.135111
##               aweminus               blastoyz                  coone
##               0.026312              -0.223102              -0.151128
##              cut snake          darren styles              dc breaks
##              -0.162497              -0.094070              -0.118863
##                dj soda              dr fresch            ferry tayle
##               0.204629              -0.074371              -0.163825
##                ghastly            huxley anne                jantsen
##              -0.241337               0.122577               0.165972
##            jon rundell             junkie kid           kristina sky
##              -0.147313              -0.087886               0.109762
##              lil texas             low steppa          menno de jong
##               0.028316              -0.158315              -0.011914
## midnight tyrannosaurus                  opiuo           rick trainor
##               0.144643              -0.199713              -0.119436
##             sound rush          spencer brown       stefano noferini
##              -0.247186               0.089896              -0.226230
##               tv noise                   yakz
##              -0.197692              -0.603019
```