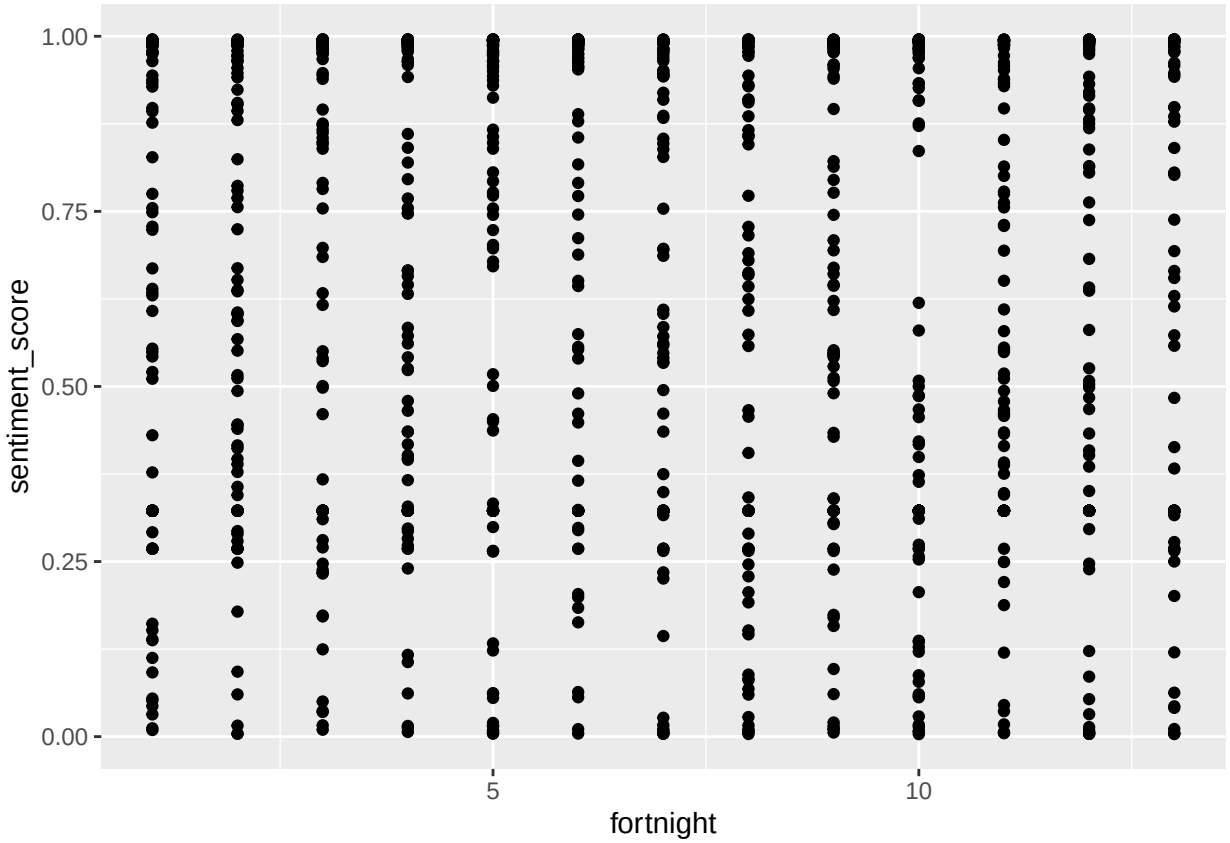
Datafest Data Analysis UK

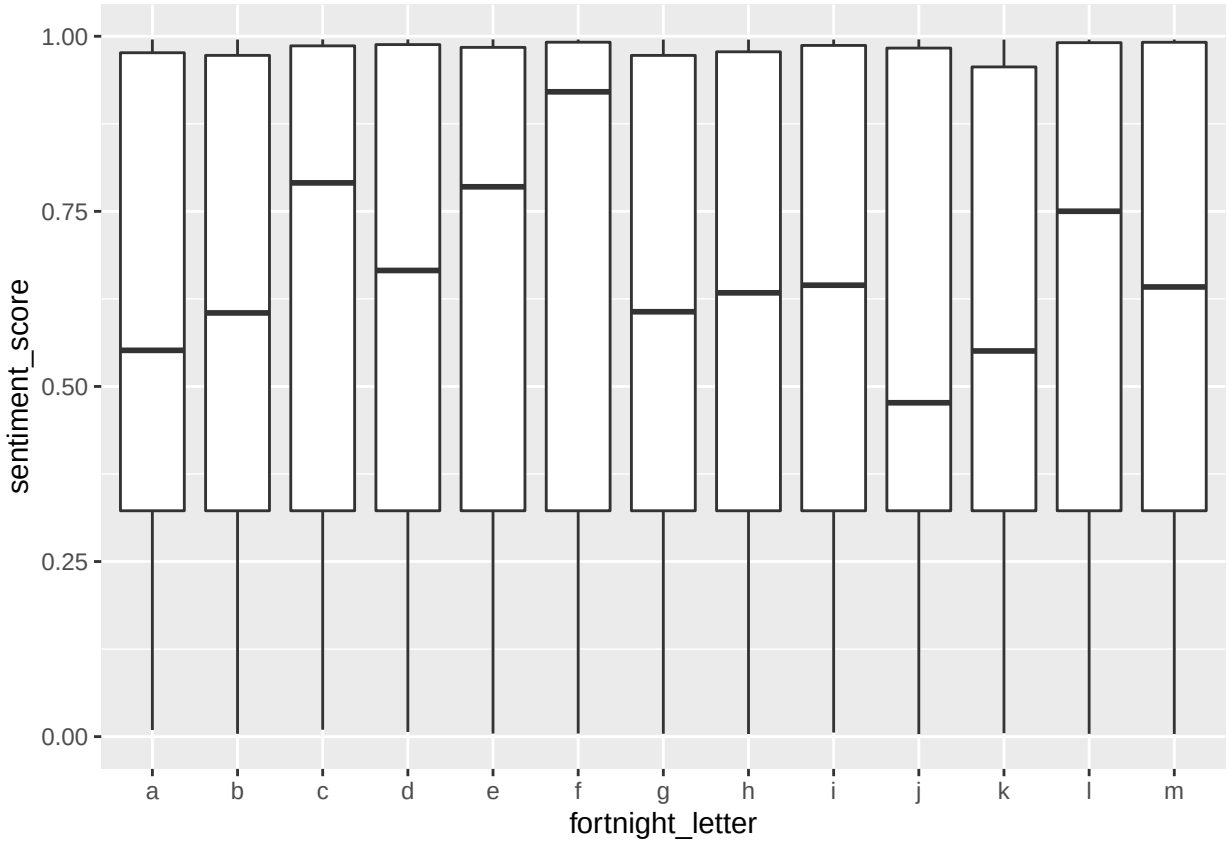
```
#UK analysis import
UK_analysis = read_excel("C:\\Users\\gtham\\OneDrive - Pomona College\\A - DATAFEST\\Analysis Datasets\\
```

```
UK_analysis_music <- UK_analysis %>%
  filter(video_category == "Music")

UK_analysis_travel <- UK_analysis %>%
  filter(video_category == "Travel and Events")

UK_analysis_people <- UK_analysis %>%
  filter(video_category == "People and Blogs")

UK_analysis_entertainment <- UK_analysis %>%
  filter(video_category == "Entertainment")

UK_analysis_news <- UK_analysis %>%
  filter(video_category == "News and Politics")

UK_analysis_how_to <- UK_analysis %>%
  filter(video_category == "How-to and Style")

UK_analysis_education <- UK_analysis %>%
  filter(video_category == "Education")

UK_analysis_science <- UK_analysis %>%
  filter(video_category == "Science and Technology")
```

```
#full UK data data summaries
ggplot(UK_analysis) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



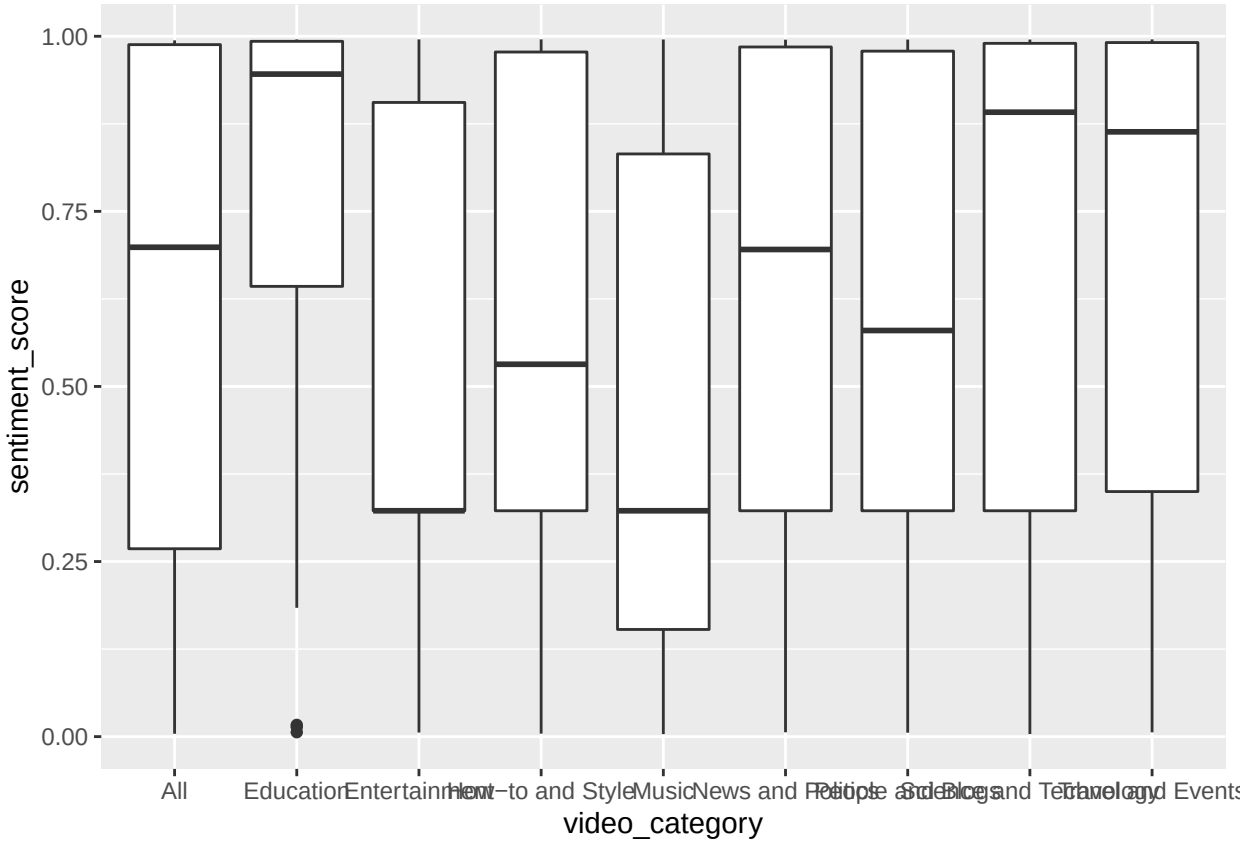

```
ggplot(UK_analysis) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.586
##  2         2                   0.626
##  3         3                   0.654
##  4         4                   0.655
##  5         5                   0.643
##  6         6                   0.704
##  7         7                   0.620
##  8         8                   0.588
##  9         9                   0.630
## 10        10                   0.576
## 11        11                   0.600
## 12        12                   0.635
## 13        13                   0.616
```

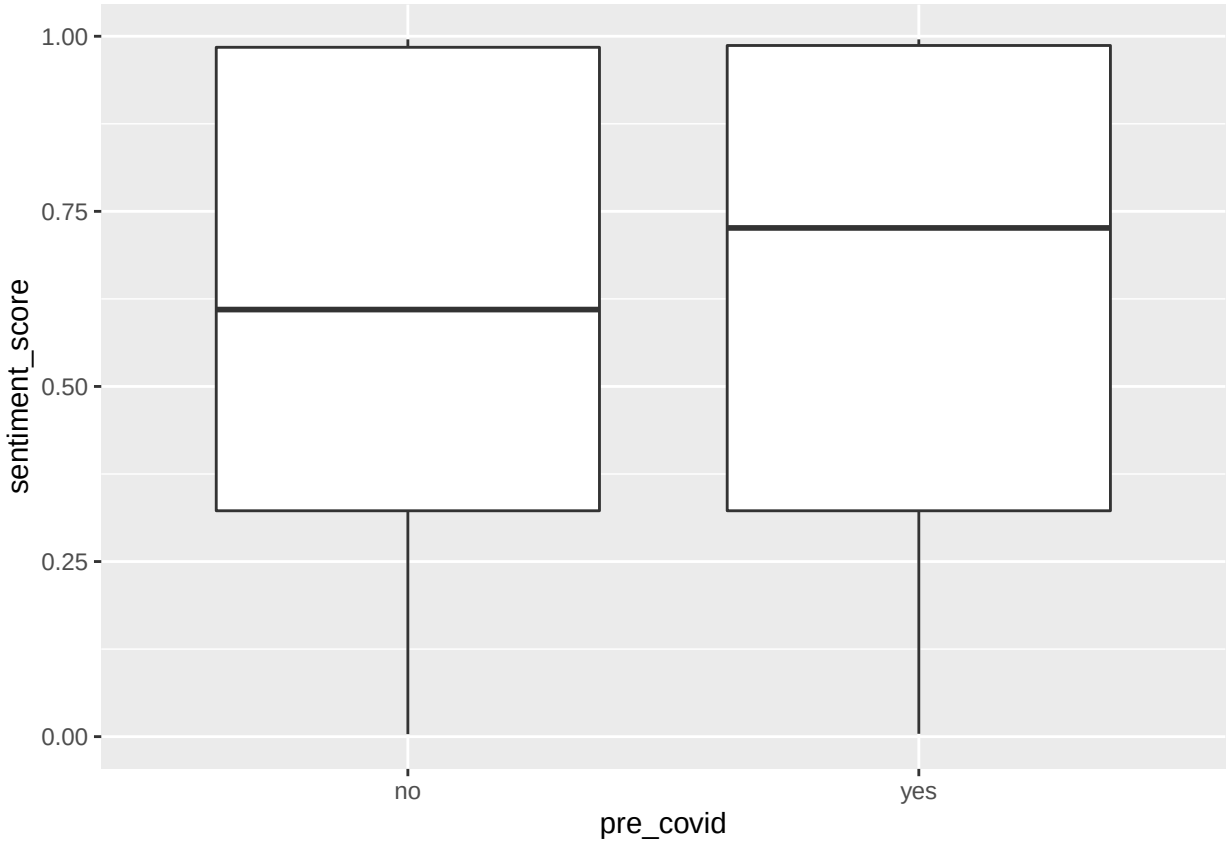
```
ggplot(UK_analysis) +
  geom_boxplot(aes(x = video_category, y = sentiment_score))
```




```
UK_analysis %>%
  group_by(video_category) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 9 x 2
##   video_category         `mean(sentiment_score)`
##   <chr>                                    <dbl>
## 1 All                                      0.629
## 2 Education                                0.793
## 3 Entertainment                            0.537
## 4 How-to and Style                         0.594
## 5 Music                                    0.456
## 6 News and Politics                        0.628
## 7 People and Blogs                         0.601
## 8 Science and Technology                   0.691
## 9 Travel and Events                        0.699
```

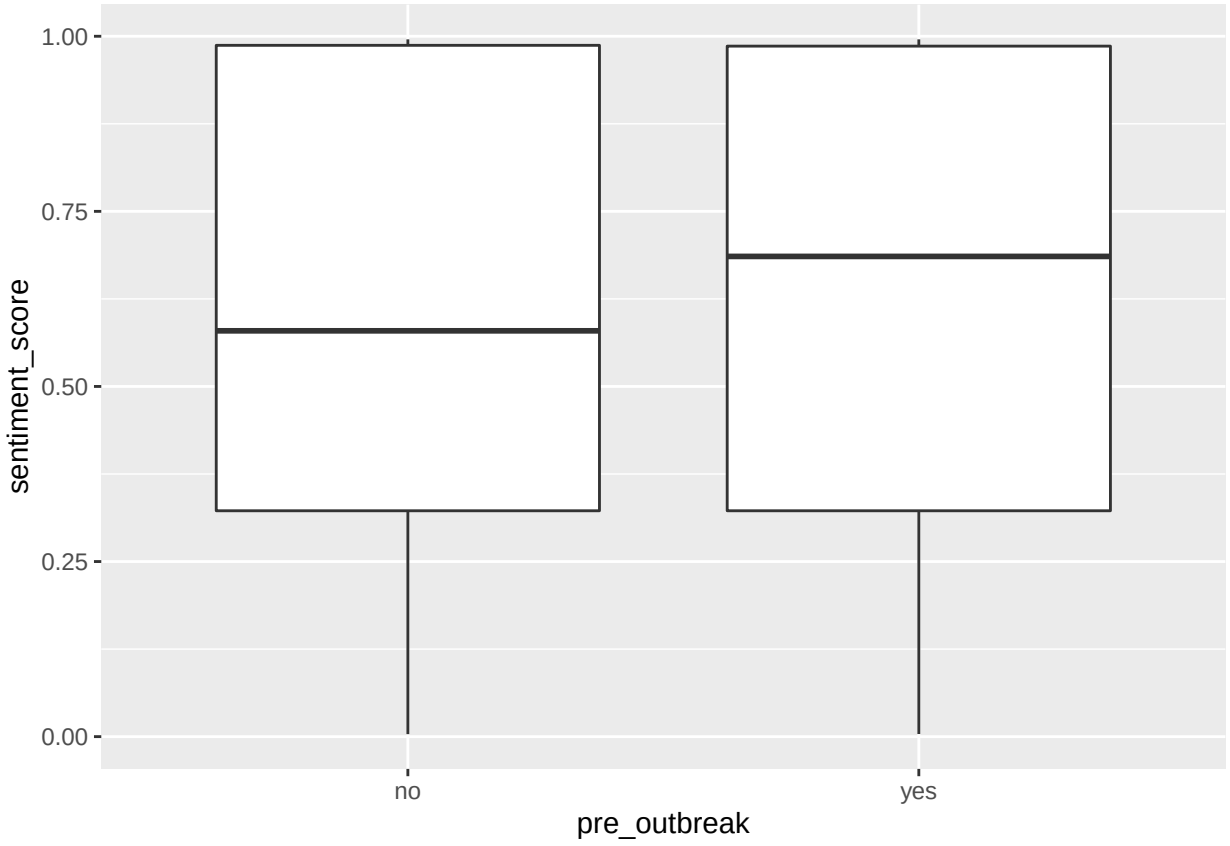
```
ggplot(UK_analysis) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.609
## 2 yes                         0.645
```

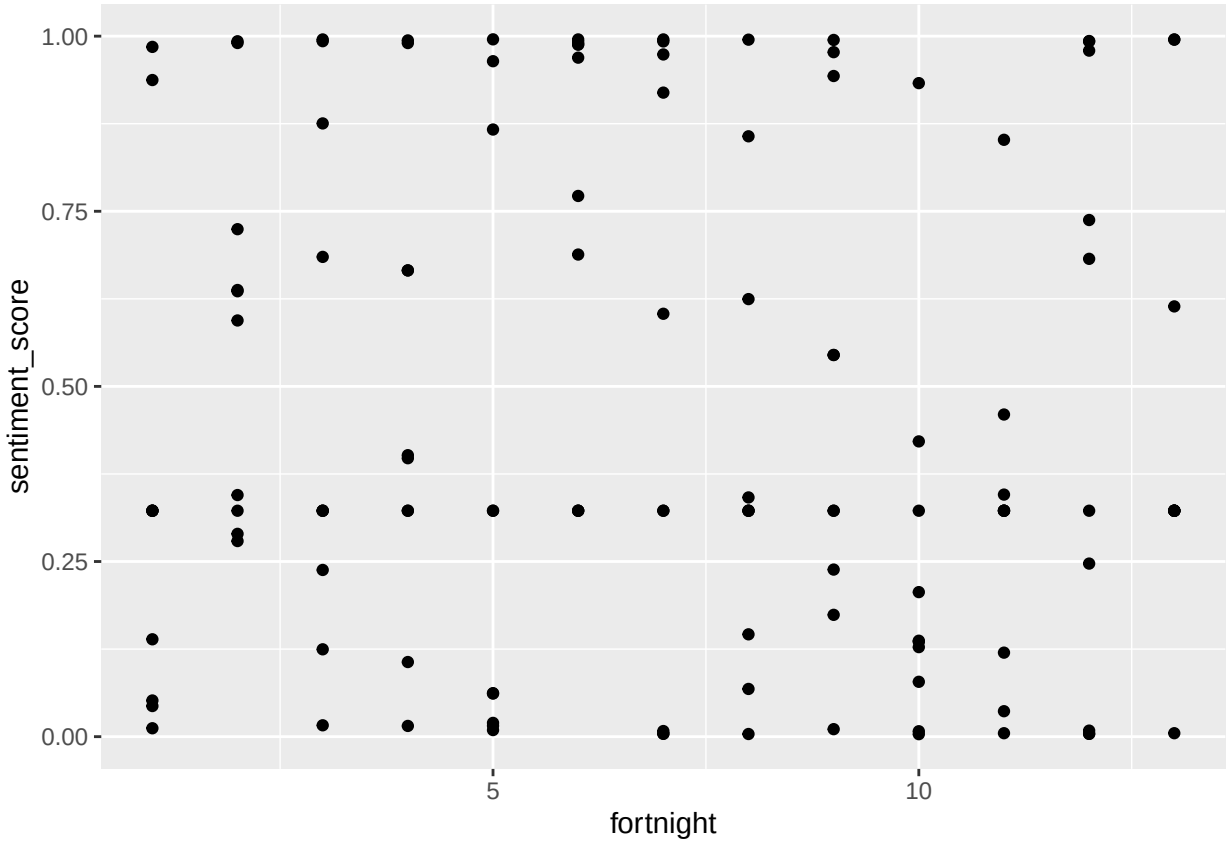
```
ggplot(UK_analysis) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```



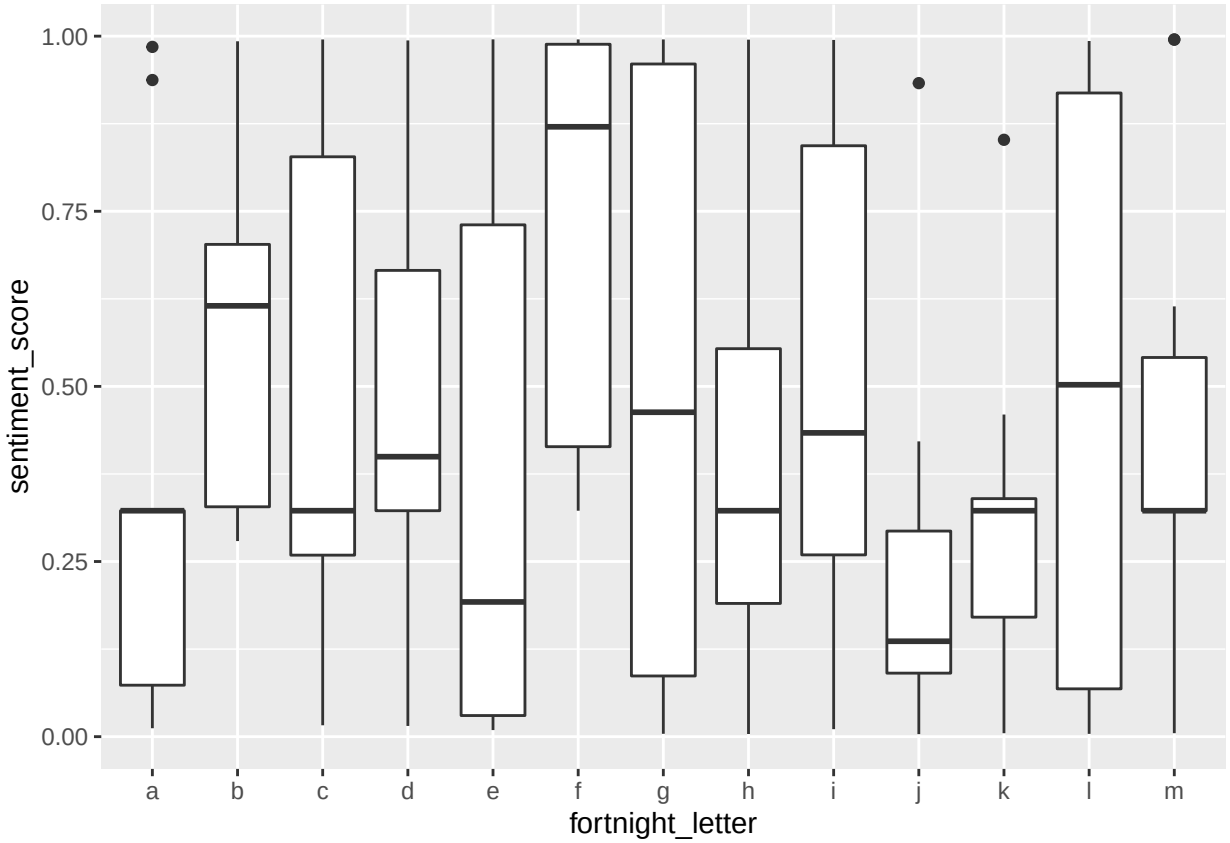

```
UK_analysis %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                           <dbl>
## 1 no                              0.607
## 2 yes                             0.634
```

```
#data summary and analysis for music dataset
ggplot(UK_analysis_music) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



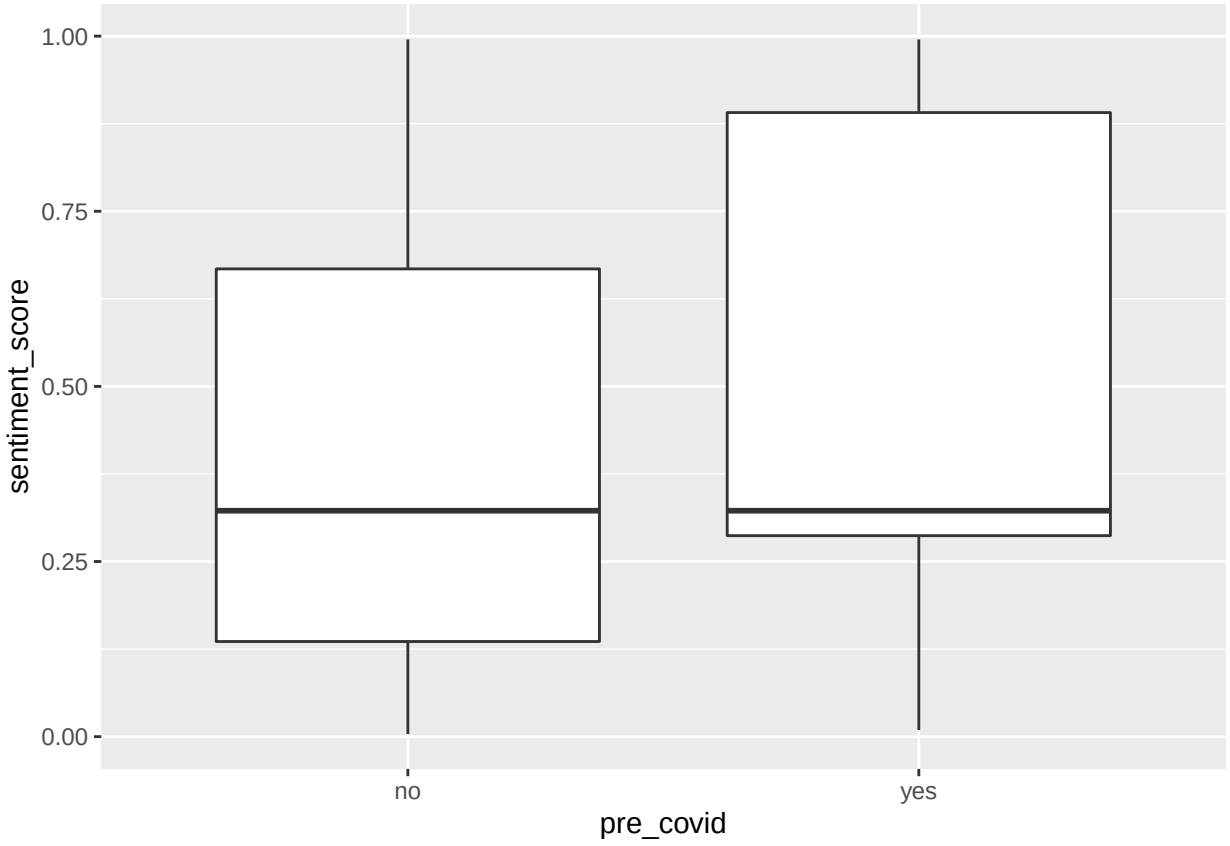

```
ggplot(UK_analysis_music) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_music %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.346
##  2         2                   0.581
##  3         3                   0.489
##  4         4                   0.488
##  5         5                   0.364
##  6         6                   0.736
##  7         7                   0.515
##  8         8                   0.400
##  9         9                   0.507
## 10        10                   0.237
## 11        11                   0.311
## 12        12                   0.497
## 13        13                   0.454
```

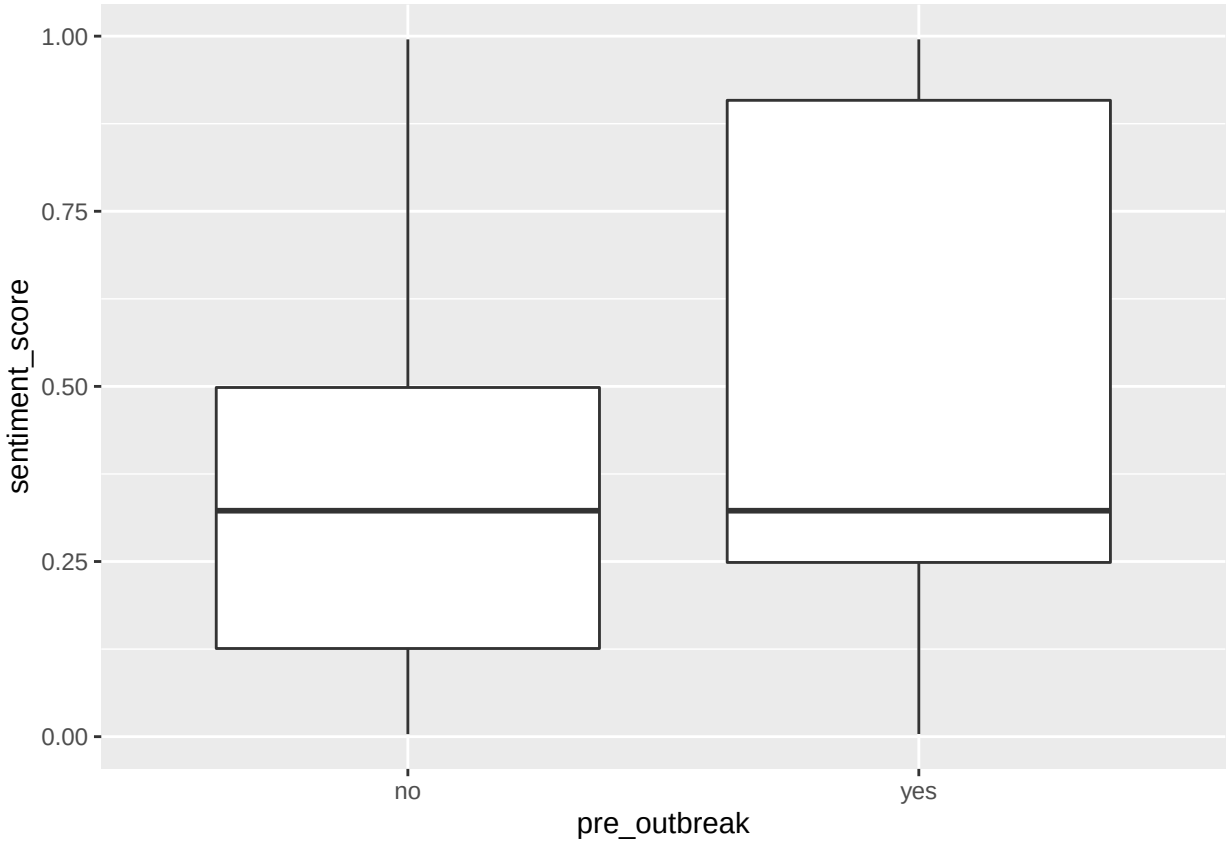
```
ggplot(UK_analysis_music) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_music %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.417
## 2 yes                         0.501
```

```
ggplot(UK_analysis_music) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_music %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.375
## 2 yes                            0.492
```

```
#two proportion z-test for music dataset
#null hypothesis: the true proportion of positive sentiment music videos published precovid and postcov

count(UK_analysis_music, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    60
```

```
m_num_precovid = 60
m_num_postcovid = 70
m_num = 130

UK_analysis_music %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

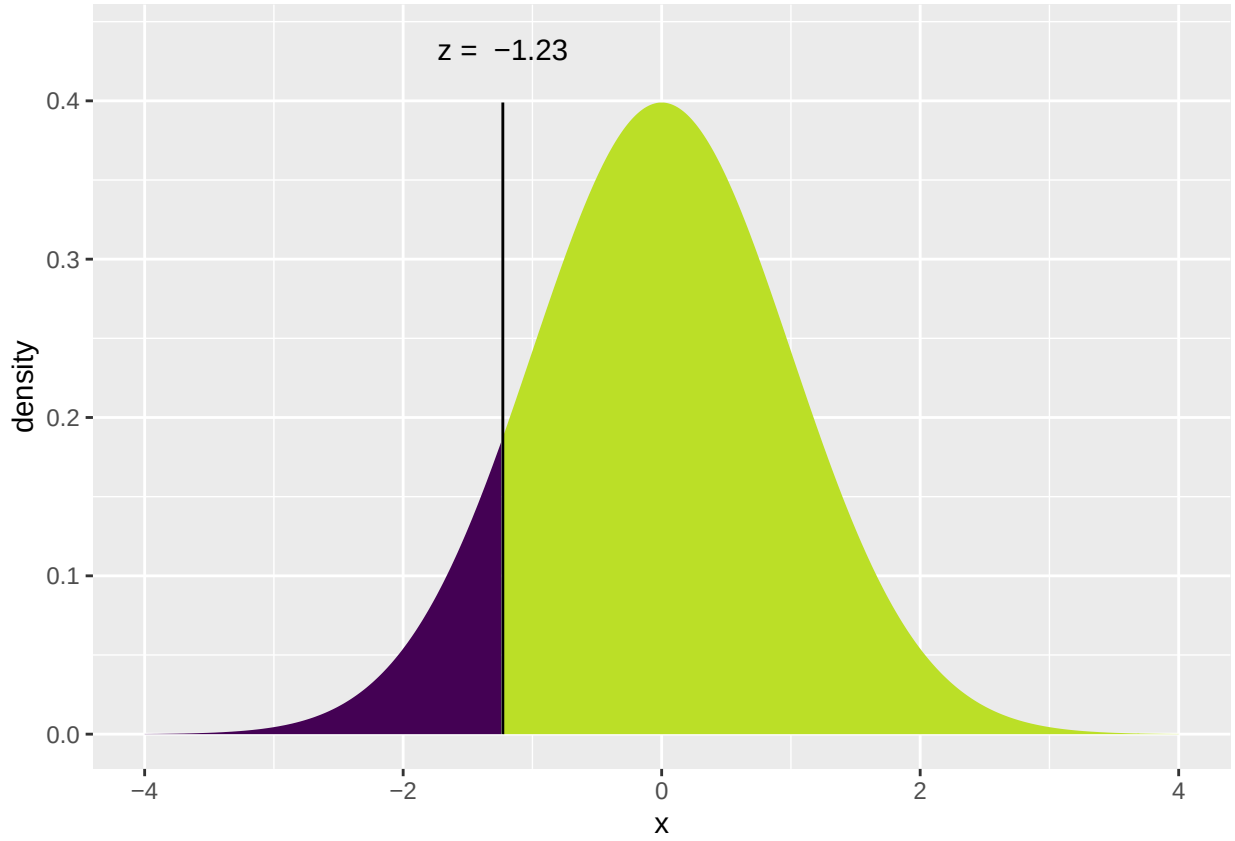
```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      34
## 2 TRUE                       26
```

```
p_hat_1_m_pos = 26/60

UK_analysis_music %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      47
## 2 TRUE                       23
```

```
p_hat_2_m_pos = 23/70

p_hat_m_pos = (26+23)/(60+70)

sd <- sqrt((((p_hat_m_pos)*(1-p_hat_m_pos))/60)+(((p_hat_m_pos)*(1-p_hat_m_pos))/70))
z_score <- ((p_hat_2_m_pos-p_hat_1_m_pos)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -1.229) = P(Z <= -1.229) = 0.1096
##  P(X >  -1.229) = P(Z >  -1.229) = 0.8904
##
```




```
## [1] 0.2191702
```

```
#outbreak music
count(UK_analysis_music, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       90
```

```
m_num_preoutbreak = 90
m_num_postoutbreak = 40
m_num = 130

UK_analysis_music %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      51
## 2 TRUE                       39
```

```
p_hat_1_m_pos = 39/90

UK_analysis_music %>%
```

```
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

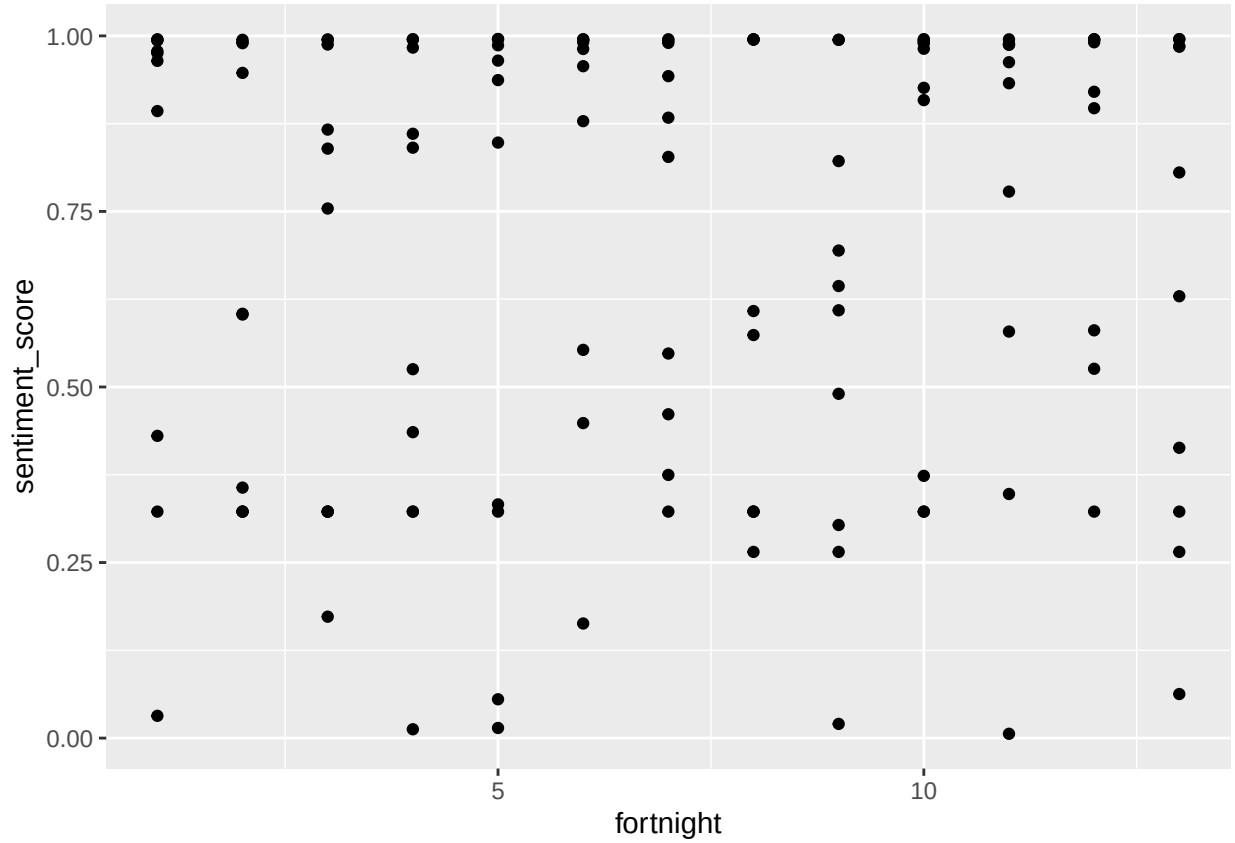
```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      30
## 2 TRUE                       10
```

```
p_hat_2_m_pos = 10/40

p_hat_m_pos = (39+10)/(90+40)

sd <- sqrt((((p_hat_m_pos)*(1-p_hat_m_pos))/90)+(((p_hat_m_pos)*(1-p_hat_m_pos))/40))
z_score <- ((p_hat_2_m_pos-p_hat_1_m_pos)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

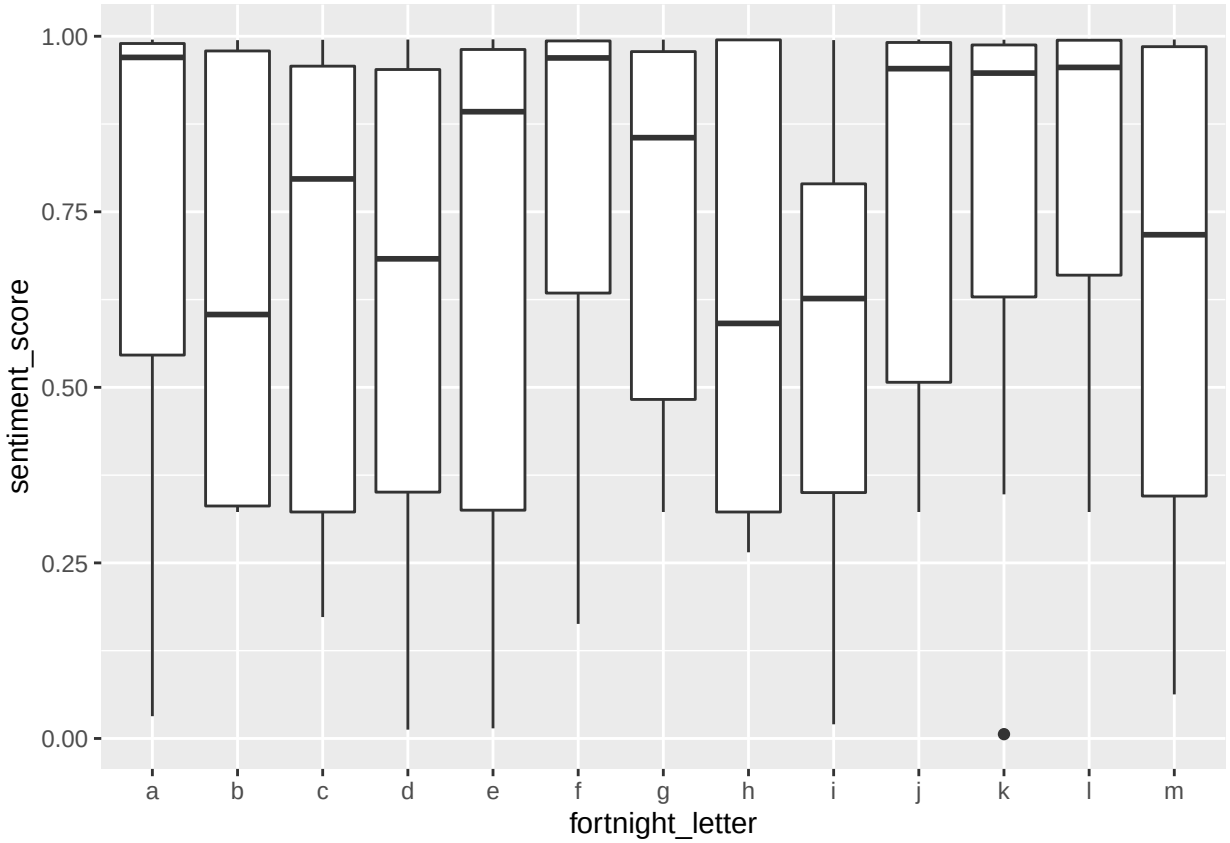
```
##
## If X ~ N(0, 1), then
##  P(X <= -1.991) = P(Z <= -1.991) = 0.02325
##  P(X >  -1.991) = P(Z >  -1.991) = 0.9767
##
```

```
## [1] 0.04650481
```

```
#data summary travel
ggplot(UK_analysis_travel) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



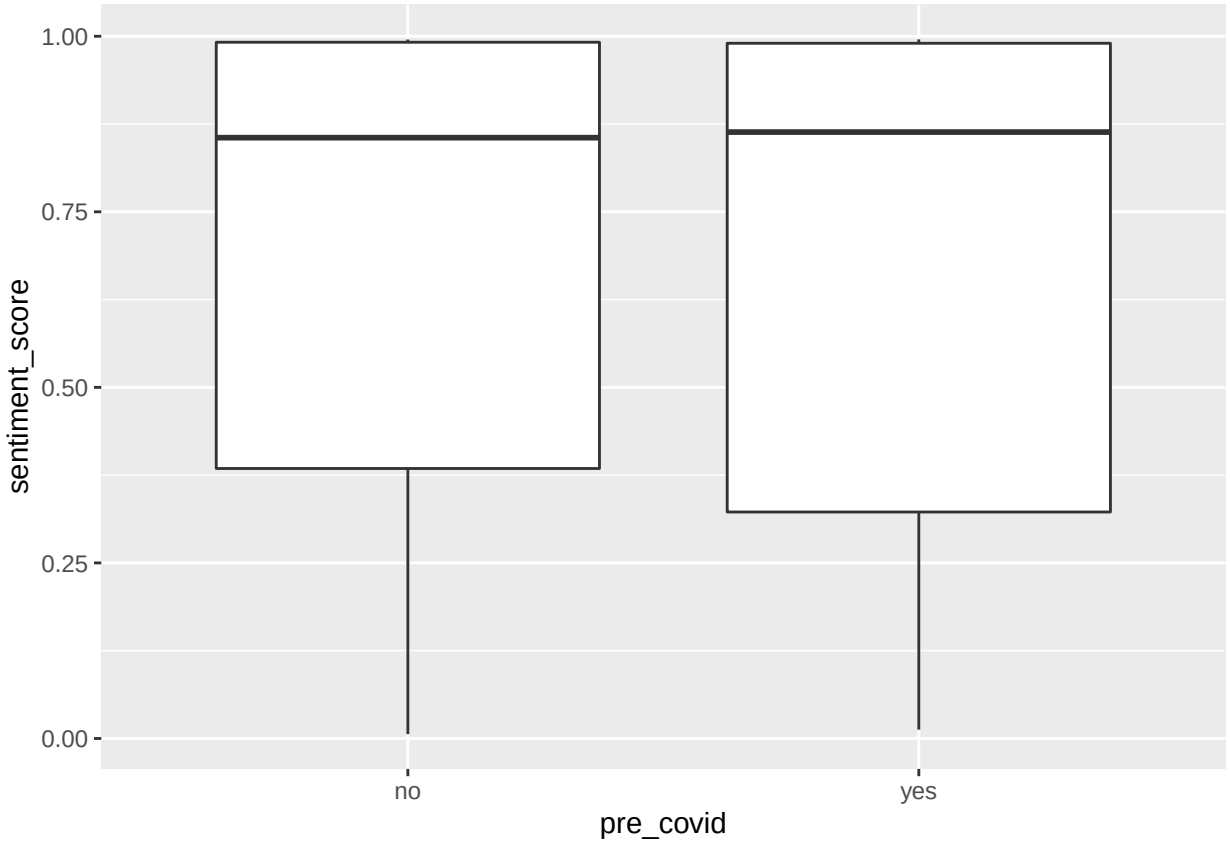

```
ggplot(UK_analysis_travel) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_travel %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.758
##  2         2                   0.645
##  3         3                   0.658
##  4         4                   0.629
##  5         5                   0.645
##  6         6                   0.796
##  7         7                   0.734
##  8         8                   0.639
##  9         9                   0.584
## 10        10                   0.780
## 11        11                   0.756
## 12        12                   0.822
## 13        13                   0.646
```

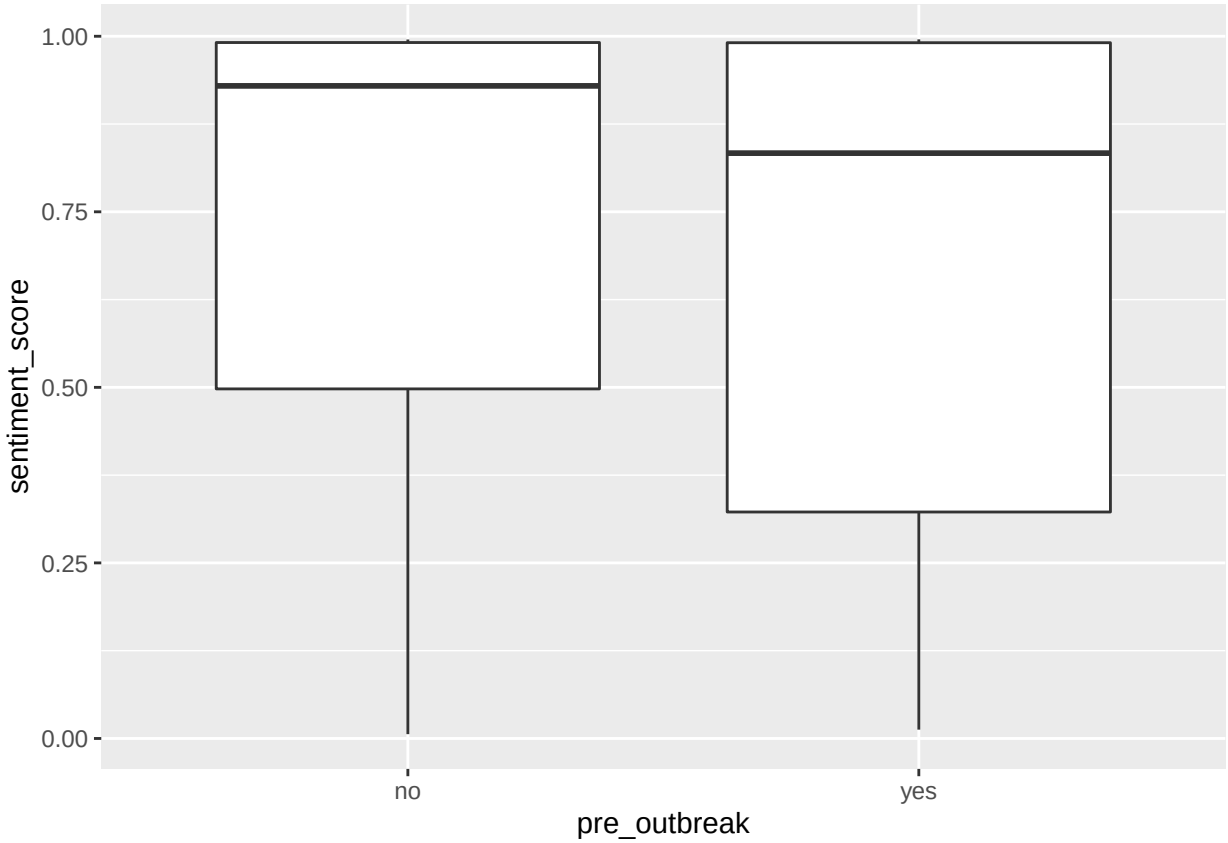
```
ggplot(UK_analysis_travel) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_travel %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.709
## 2 yes                         0.689
```

```
ggplot(UK_analysis_travel) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_travel %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.751
## 2 yes                            0.676
```

```
#precovid travel
#null hypothesis: the true proportion of positive sentiment travel videos published precovid and postco

count(UK_analysis_travel, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    60
```

```
t_num_precovid = 60
t_num_postcovid = 70
t_num = 130

UK_analysis_travel %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

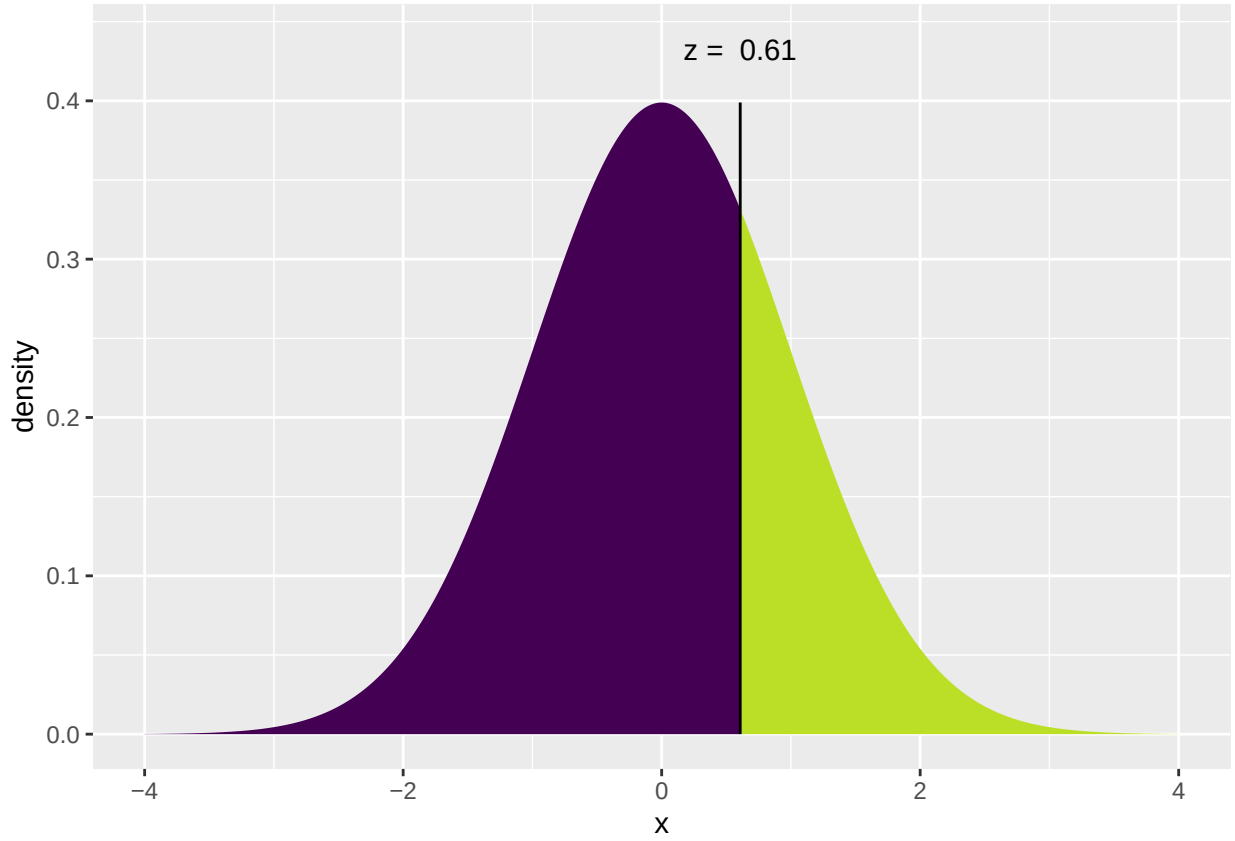
```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      21
## 2 TRUE                       39
```

```
p_hat_1_t_pos = 39/60

UK_analysis_travel %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      21
## 2 TRUE                       49
```

```
p_hat_2_t_pos = 49/70

p_hat_t_pos = (39+49)/(60+70)

sd <- sqrt((((p_hat_t_pos)*(1-p_hat_t_pos))/60)+(((p_hat_t_pos)*(1-p_hat_t_pos))/70))
z_score <- ((p_hat_2_t_pos-p_hat_1_t_pos)-0)/sd

#p-value
2* (1-xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= 0.6077) = P(Z <= 0.6077) = 0.7283
##  P(X >  0.6077) = P(Z >  0.6077) = 0.2717
##
```




```
## [1] 0.5433762
```

```
#outbreak travel
count(UK_analysis_travel, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       90
```

```
t_num_preoutbreak = 90
t_num_postoutbreak = 40
t_num = 130

UK_analysis_travel %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      32
## 2 TRUE                       58
```

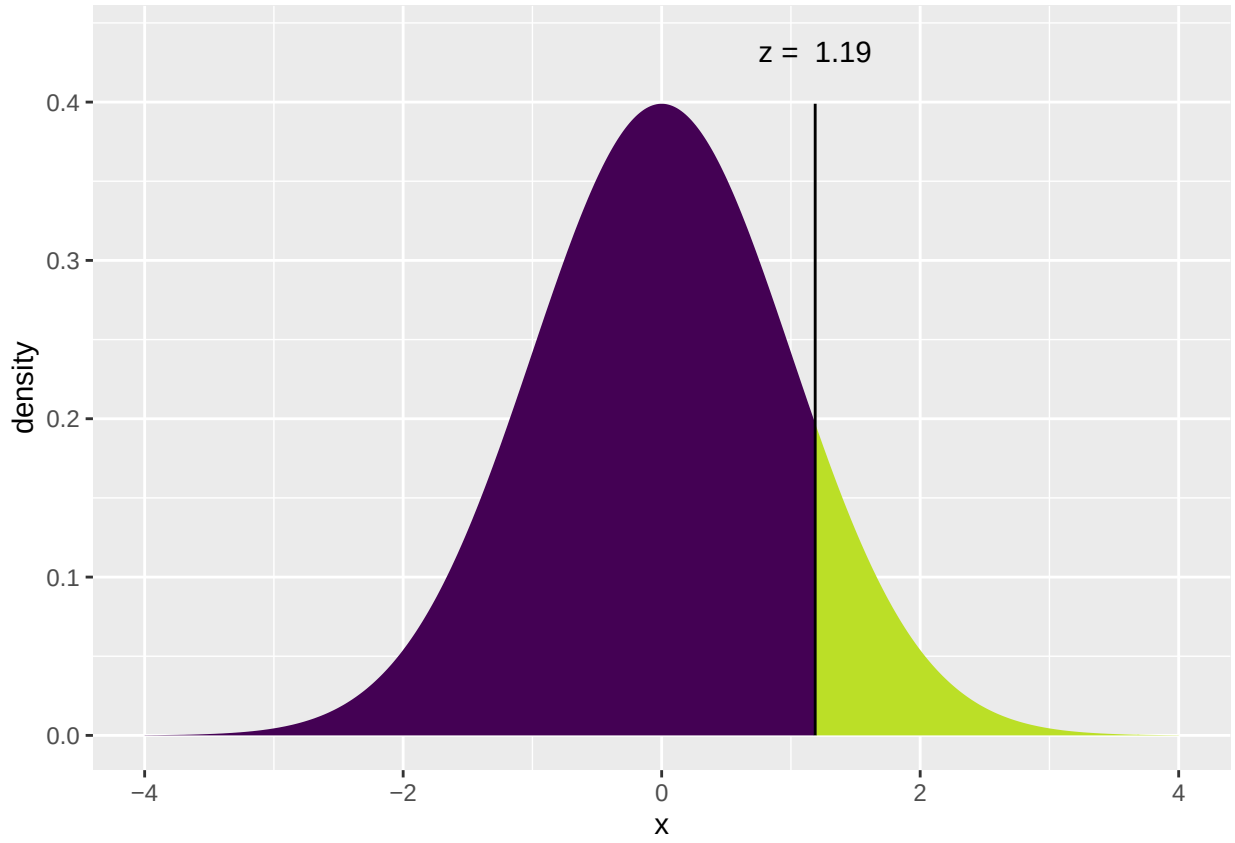
```
p_hat_1_t_pos = 58/90

UK_analysis_travel %>%
```

```
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      10
## 2 TRUE                       30
```

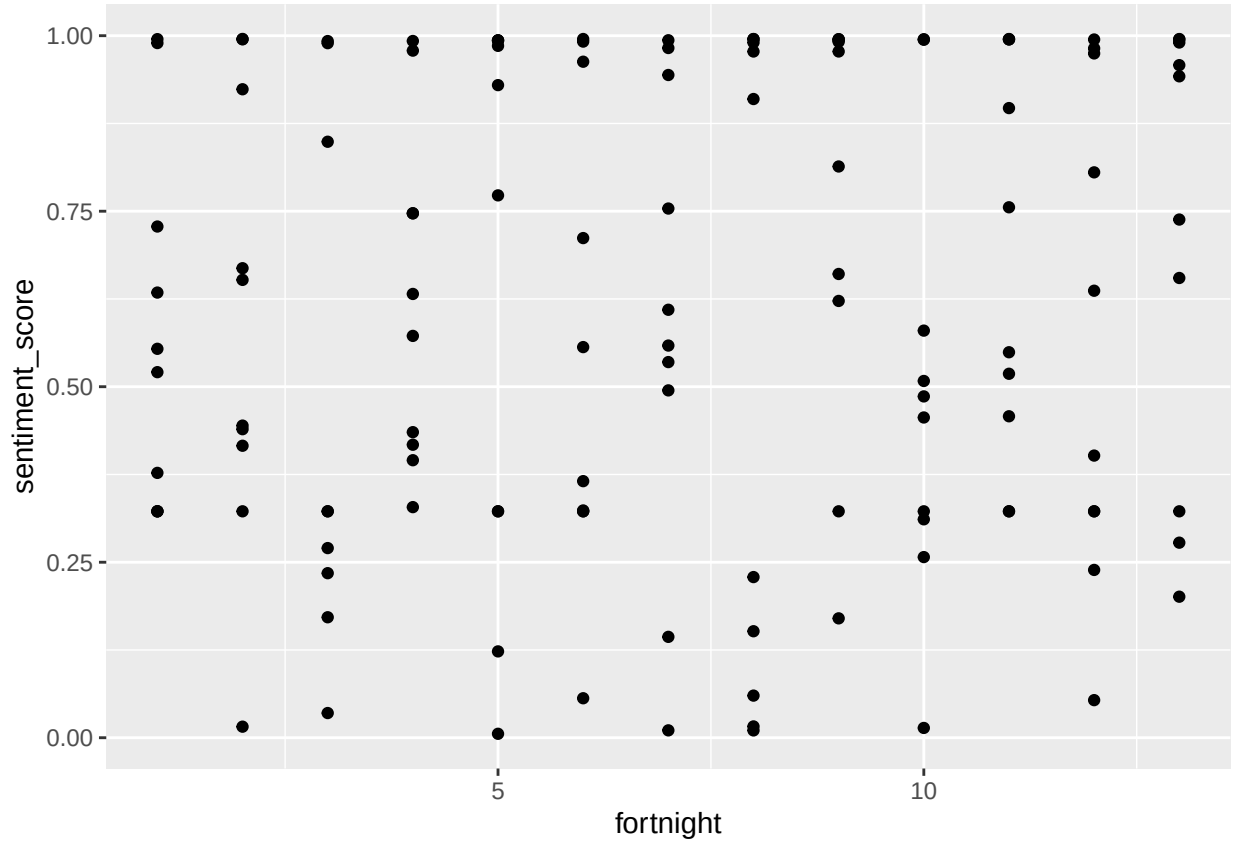
```
p_hat_2_t_pos = 30/40

p_hat_t_pos = (58+30)/(90+40)

sd <- sqrt((((p_hat_t_pos)*(1-p_hat_t_pos))/90)+(((p_hat_t_pos)*(1-p_hat_t_pos))/40))
z_score <- ((p_hat_2_t_pos-p_hat_1_t_pos)-0)/sd

#p-value
2* (1-xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= 1.188) = P(Z <= 1.188) = 0.8825
##  P(X >  1.188) = P(Z >  1.188) = 0.1175
##
```

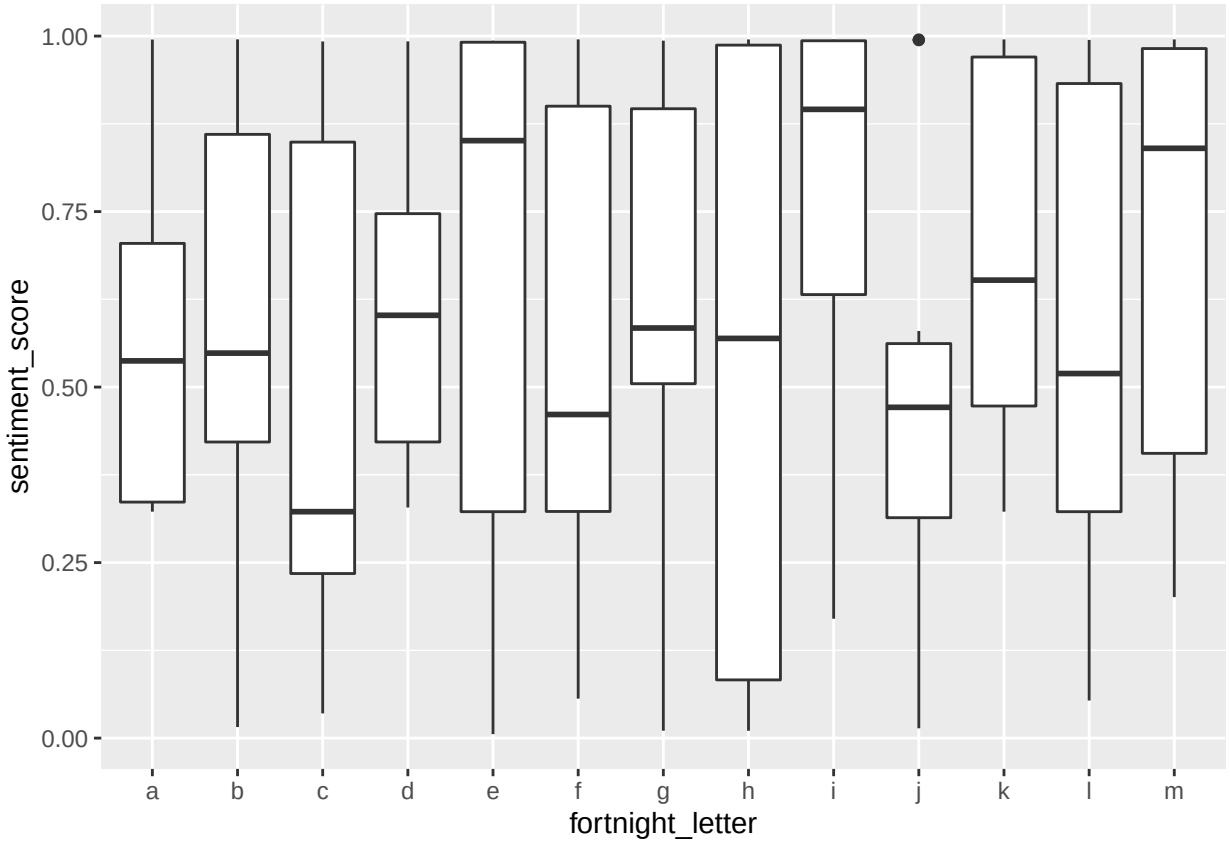


```
## [1] 0.2349179
```

```
#data summary people and blogs
ggplot(UK_analysis_people) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



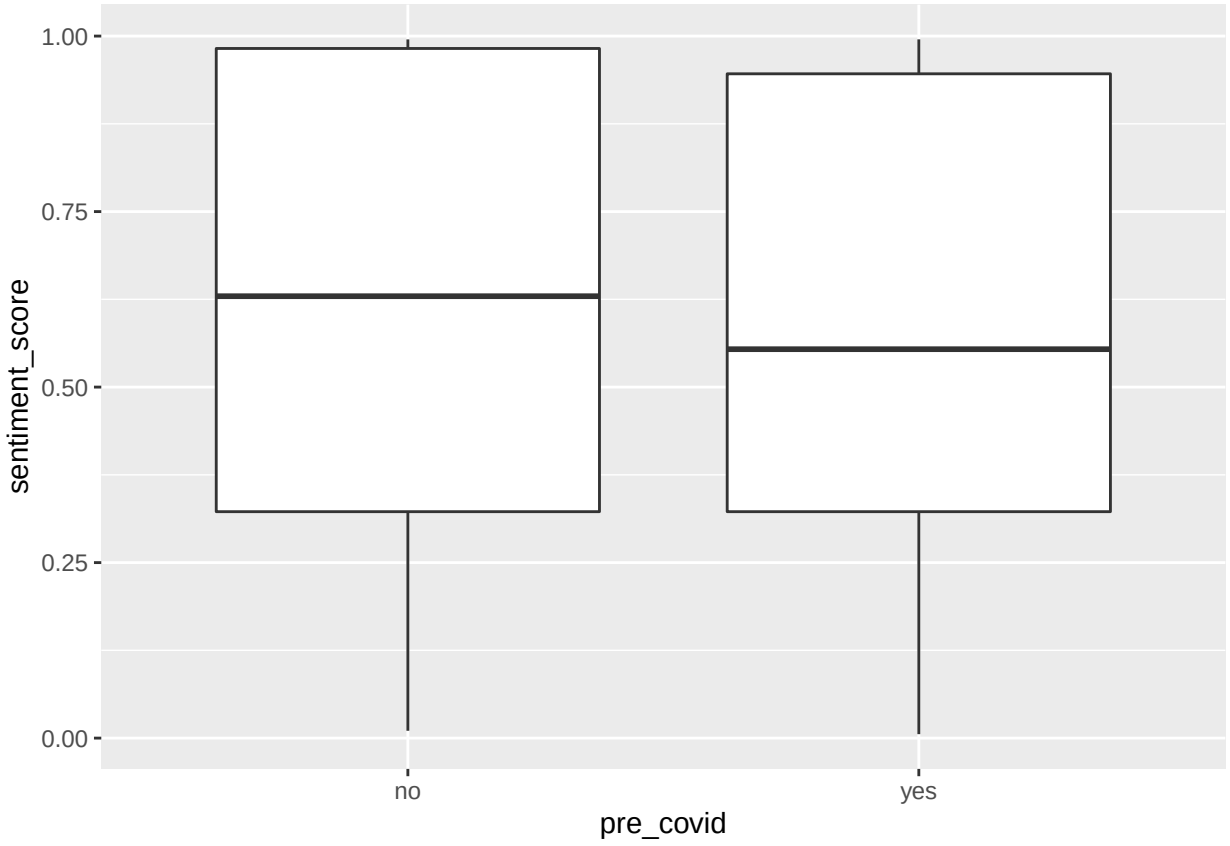

```
ggplot(UK_analysis_people) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_people %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.577
##  2         2                   0.587
##  3         3                   0.465
##  4         4                   0.625
##  5         5                   0.644
##  6         6                   0.561
##  7         7                   0.603
##  8         8                   0.534
##  9         9                   0.754
## 10        10                   0.492
## 11        11                   0.681
## 12        12                   0.573
## 13        13                   0.707
```

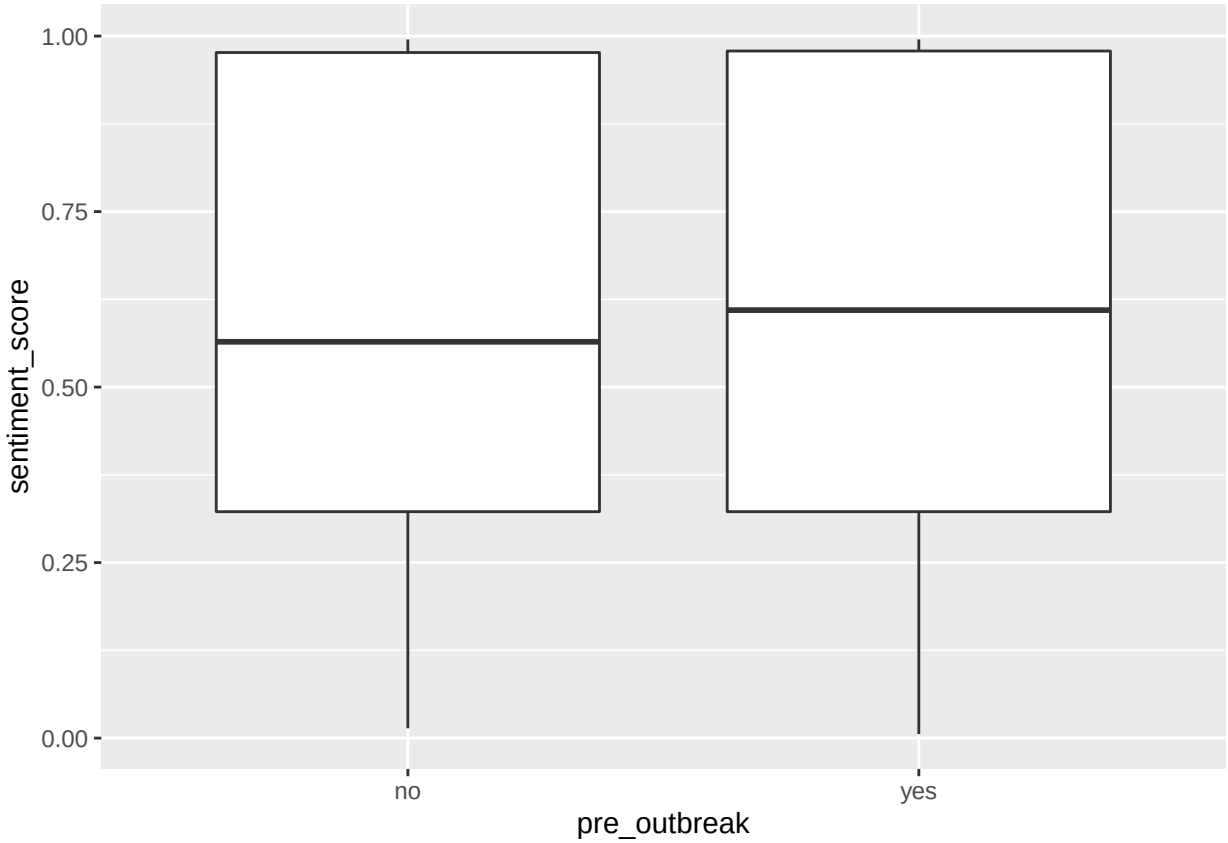
```
ggplot(UK_analysis_people) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_people %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.621
## 2 yes                         0.578
```

```
ggplot(UK_analysis_people) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_people %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.613
## 2 yes                            0.596
```

```
#precovid people
count(UK_analysis_people, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    59
```

```
p_num_precovid = 59
p_num_postcovid = 70
p_num = 129

UK_analysis_people %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
```

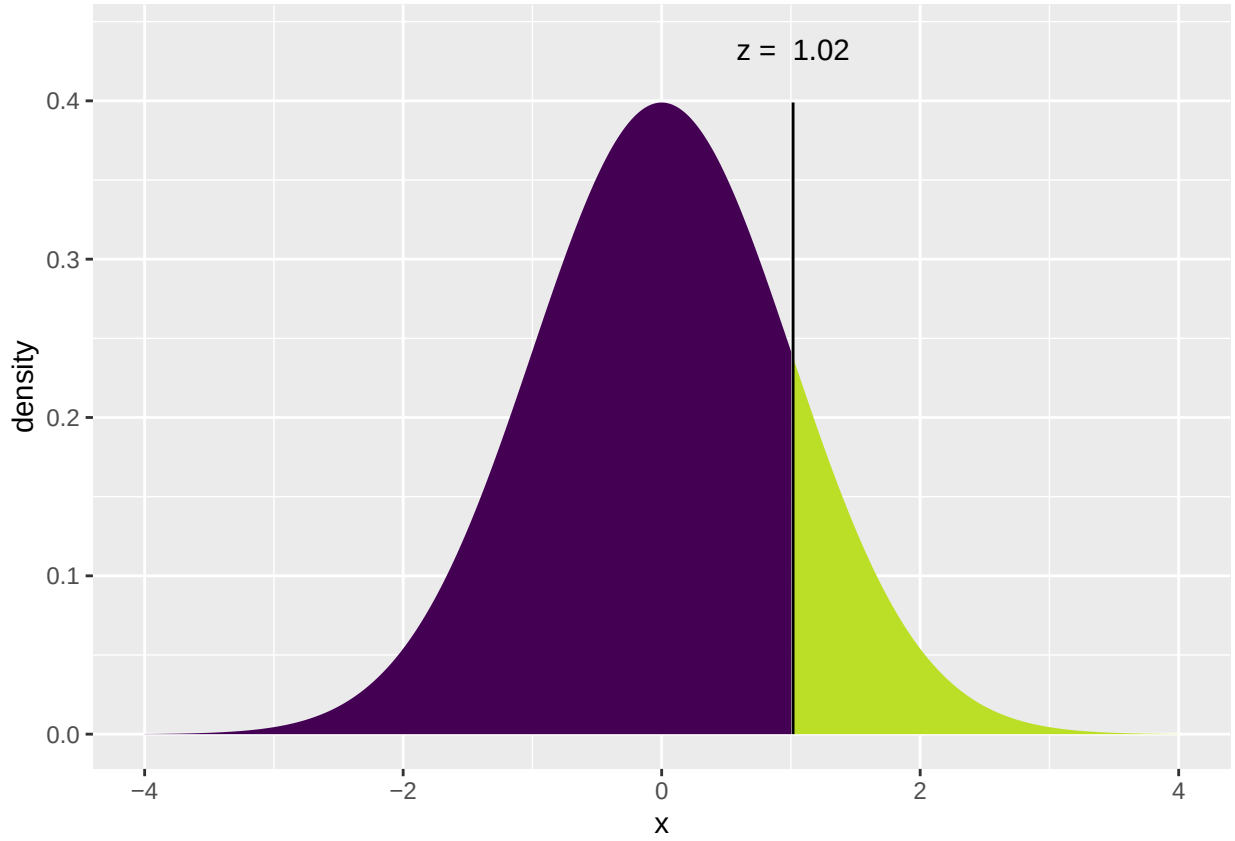
```
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      28
## 2 TRUE                       31
```

```
p_hat_1_p_pos = 31/59

UK_analysis_people %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      27
## 2 TRUE                       43
```

```
p_hat_2_p_pos = 43/70

p_hat_p_pos = (31+43)/(59+70)

sd <- sqrt((((p_hat_p_pos)*(1-p_hat_p_pos))/59)+(((p_hat_p_pos)*(1-p_hat_p_pos))/70))
z_score <- ((p_hat_2_p_pos-p_hat_1_p_pos)-0)/sd

#p-value
2* (1-xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= 1.017) = P(Z <= 1.017) = 0.8453
##  P(X >  1.017) = P(Z >  1.017) = 0.1547
##
```




```
## [1] 0.3093008
```

```
#outbreak people
count(UK_analysis_people, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       89
```

```
p_num_preoutbreak = 89
p_num_postoutbreak = 40
p_num = 129

UK_analysis_people %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      38
## 2 TRUE                       51
```

```
p_hat_1_p_pos = 51/79

UK_analysis_people %>%
```

```
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

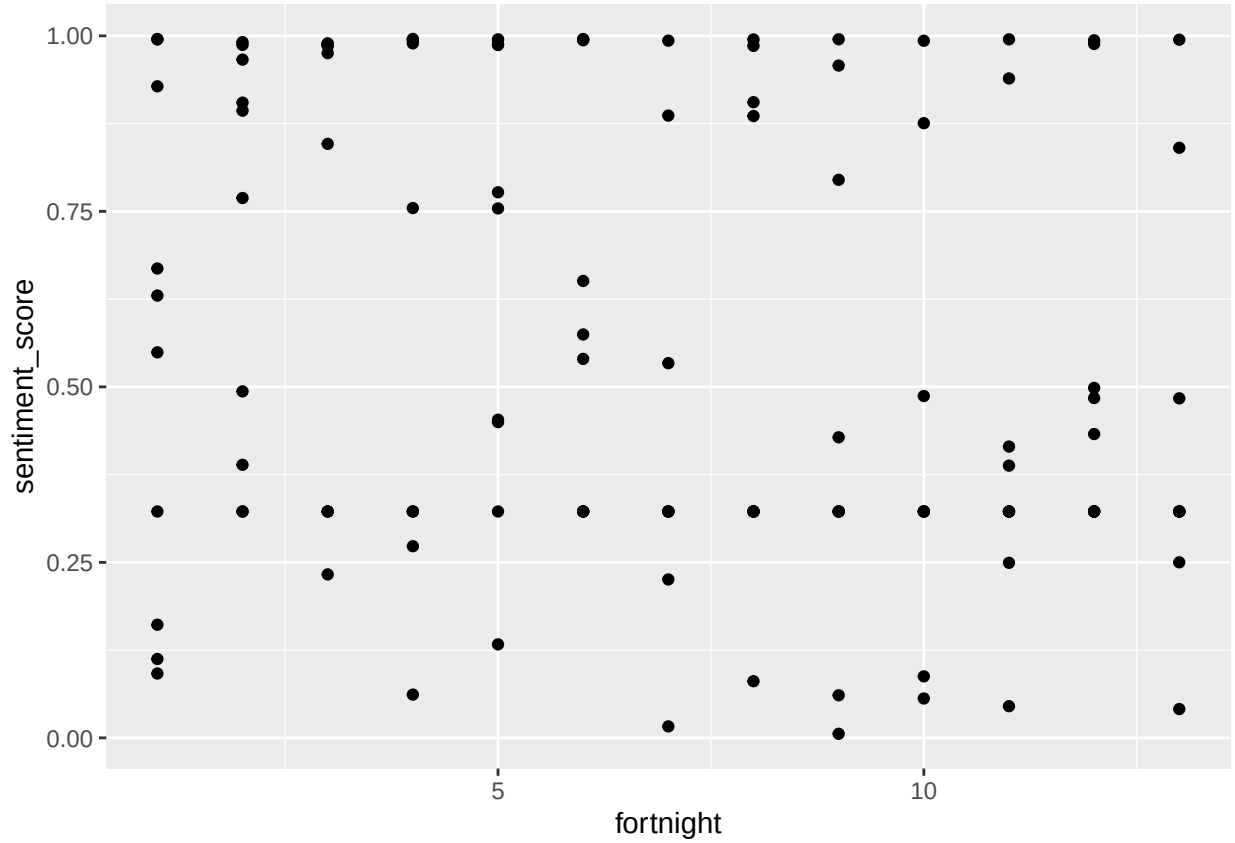
```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      17
## 2 TRUE                       23
```

```
p_hat_2_p_pos = 23/50

p_hat_p_pos = (51+23)/(89+40)

sd <- sqrt((((p_hat_p_pos)*(1-p_hat_p_pos))/89)+(((p_hat_p_pos)*(1-p_hat_p_pos))/40))
z_score <- ((p_hat_2_p_pos-p_hat_1_p_pos)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

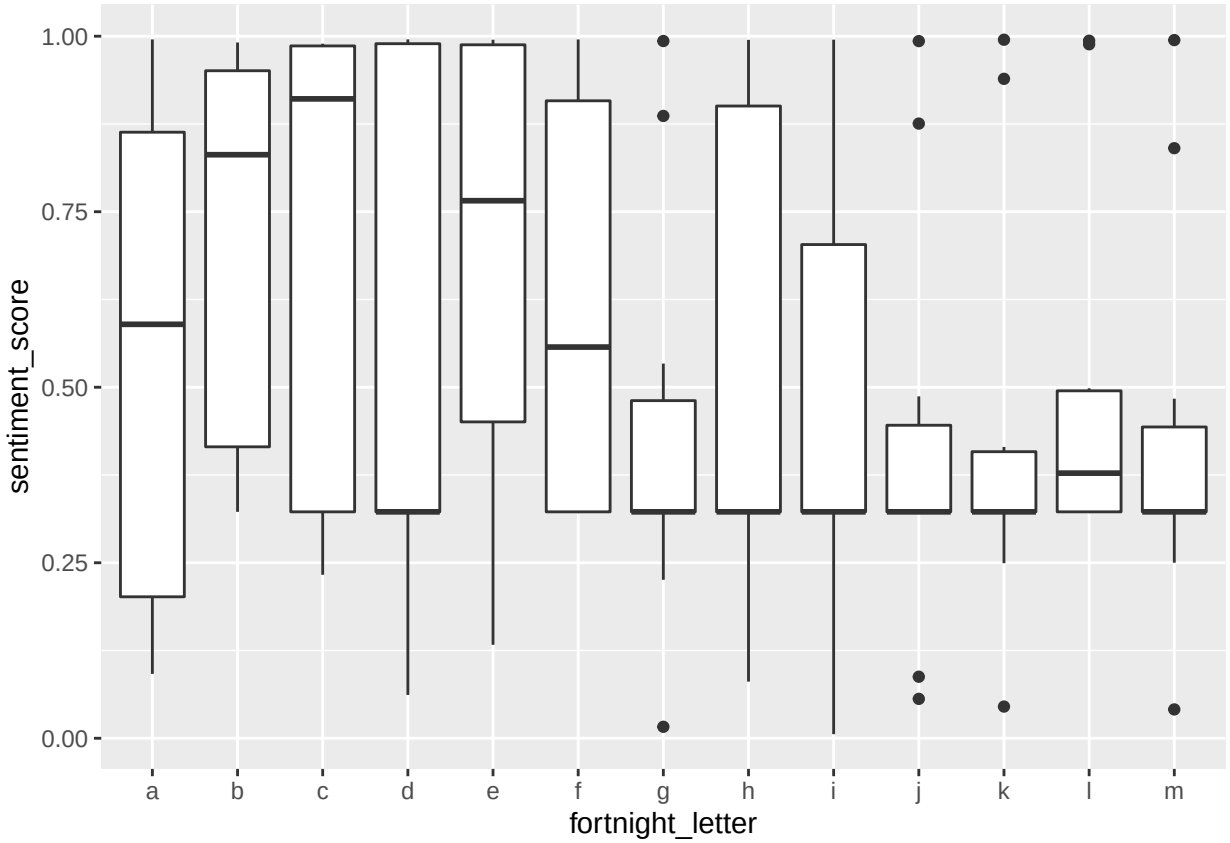
```
##
## If X ~ N(0, 1), then
##  P(X <= -1.971) = P(Z <= -1.971) = 0.02435
##  P(X >  -1.971) = P(Z >  -1.971) = 0.9756
##
```

```
## [1] 0.04870161
```

```
#data summary entertainment
ggplot(UK_analysis_entertainment) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



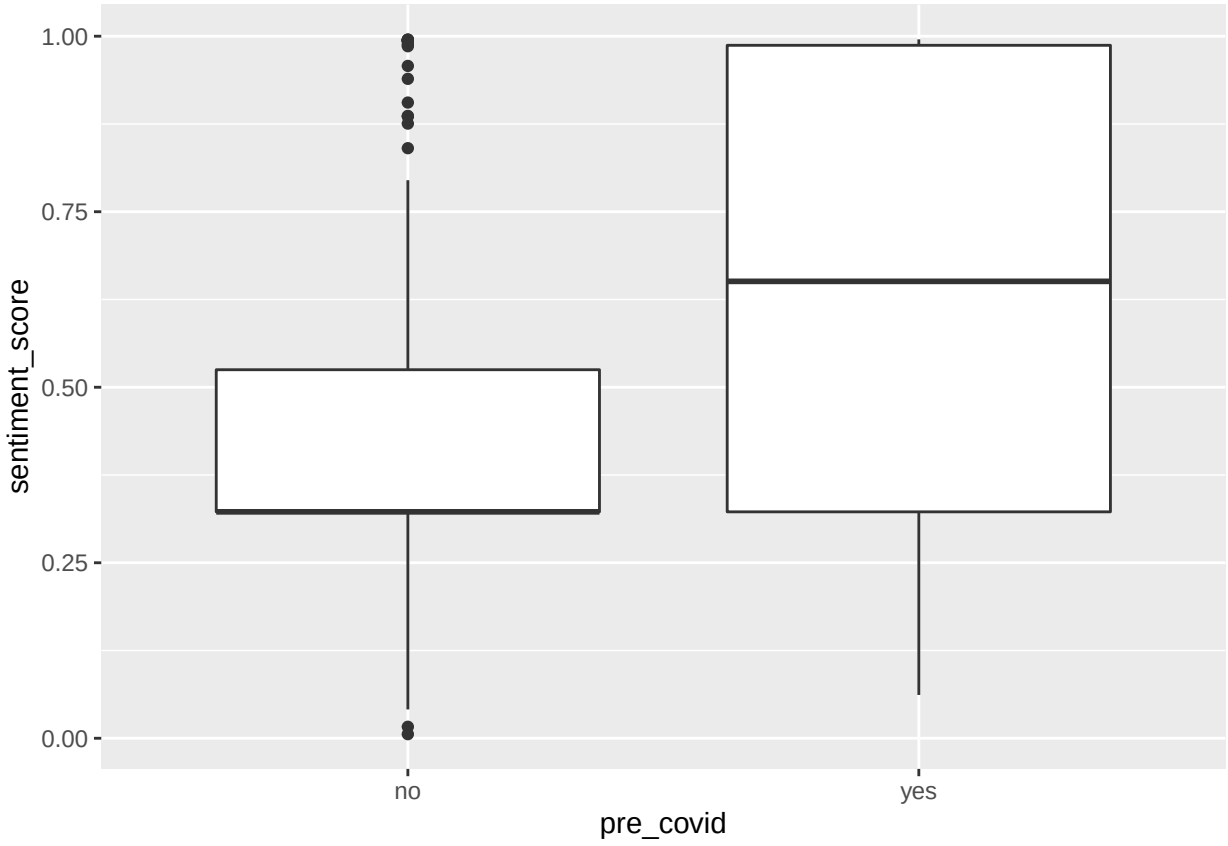

```
ggplot(UK_analysis_entertainment) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_entertainment %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.545
##  2         2                   0.704
##  3         3                   0.697
##  4         4                   0.560
##  5         5                   0.685
##  6         6                   0.604
##  7         7                   0.427
##  8         8                   0.547
##  9         9                   0.453
## 10        10                   0.411
## 11        11                   0.432
## 12        12                   0.501
## 13        13                   0.422
```

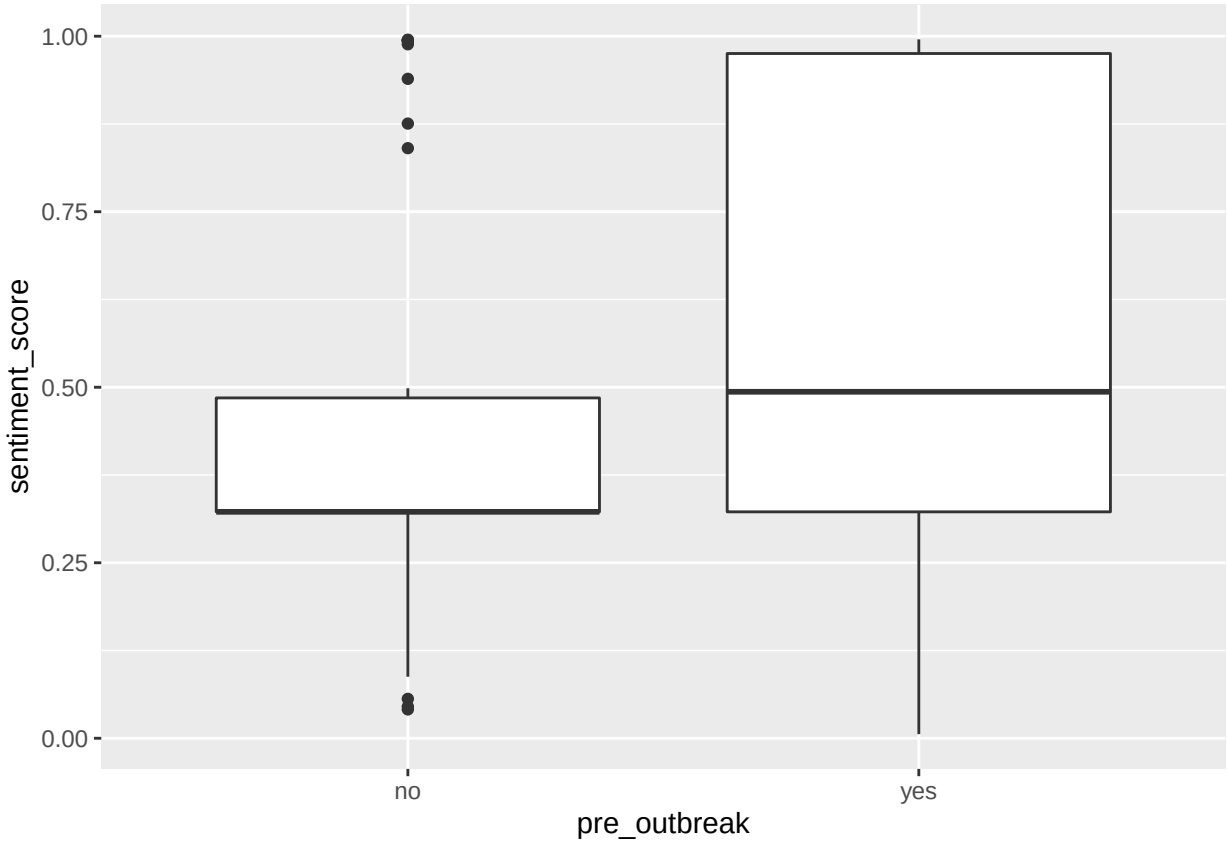
```
ggplot(UK_analysis_entertainment) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_entertainment %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.456
## 2 yes                         0.634
```

```
ggplot(UK_analysis_entertainment) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_entertainment %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.442
## 2 yes                            0.580
```

```
#precovid entertainment
count(UK_analysis_entertainment, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    59
```

```
num_precovid = 59
num_postcovid = 70
num = 129

UK_analysis_entertainment %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
```

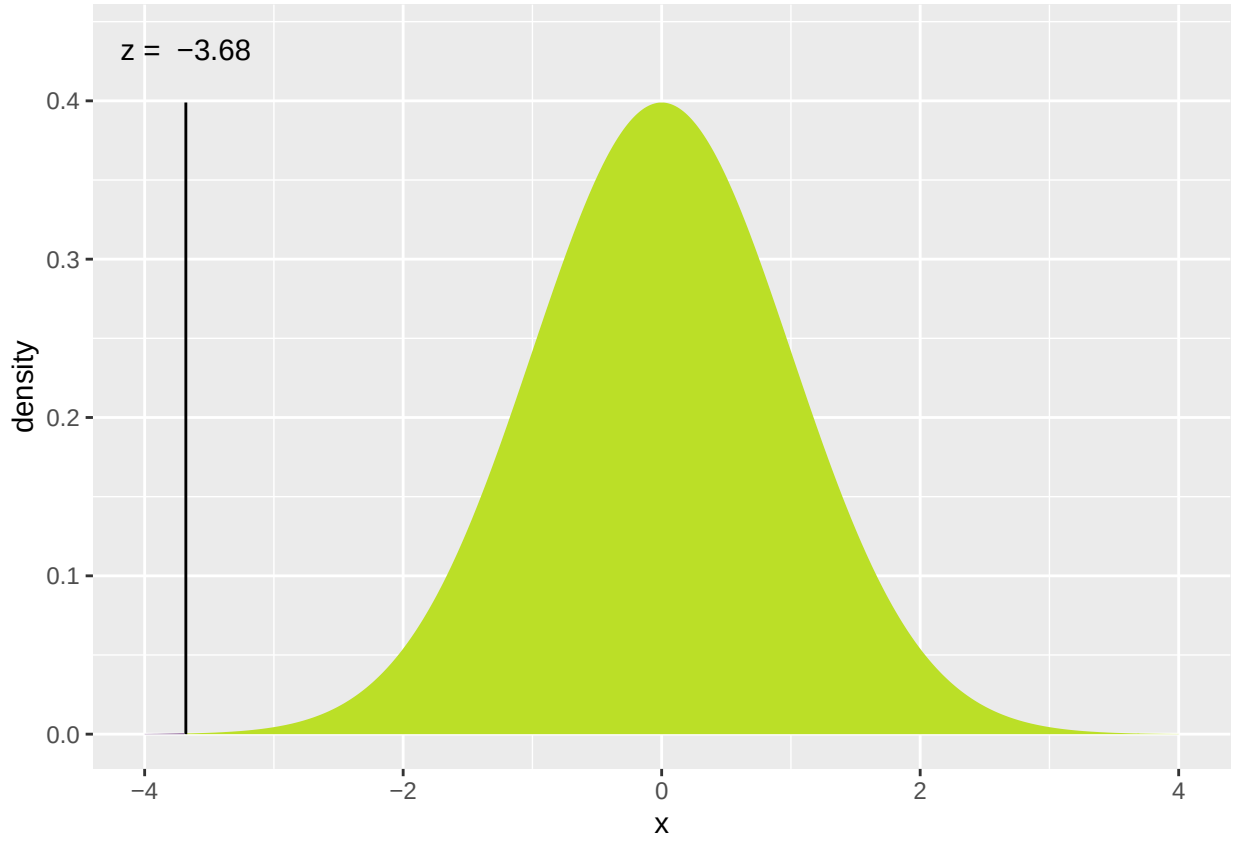
```
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      25
## 2 TRUE                       34
```

```
#proportion of positive sentiment videos precovid from sample
p_hat1 = 34/59

UK_analysis_entertainment %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      52
## 2 TRUE                       18
```

```
#proportion of positive sentiment videos postcovid from sample
p_hat2 = 18/70

p_hat = (34+18)/(59+70)

sd <- sqrt((((p_hat)*(1-p_hat))/59)+(((p_hat)*(1-p_hat))/70))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -3.681) = P(Z <= -3.681) = 0.0001161
##  P(X >  -3.681) = P(Z >  -3.681) = 0.9999
##
```




```
## [1] 0.0002321481
```

```
#outbreak entertainment
count(UK_analysis_entertainment, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       89
```

```
num_preoutbreak = 89
num_postoutbreak = 40
num = 129

UK_analysis_entertainment %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      45
## 2 TRUE                       44
```

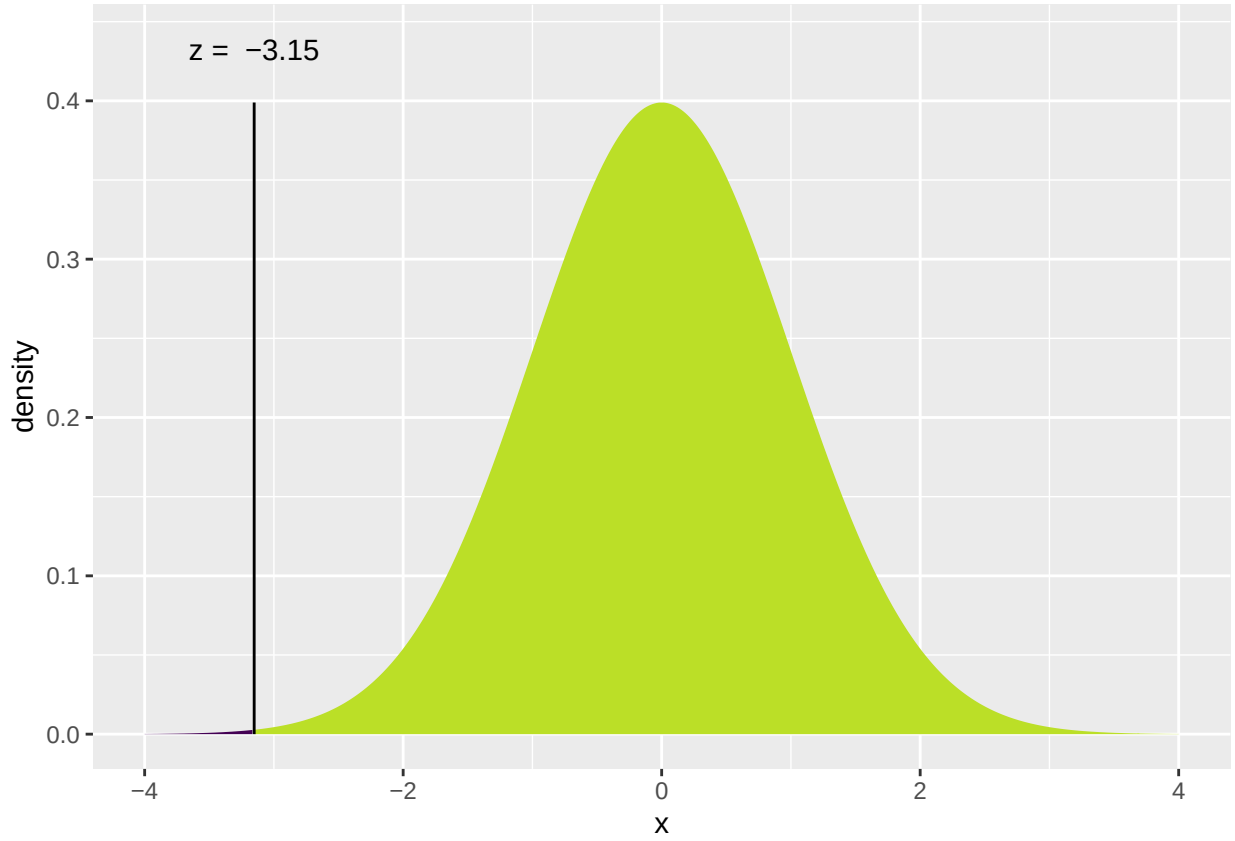
```
#proportion of positive sentiment videos preoutbreak from sample
p_hat1 = 44/89
```

```r
UK_analysis_entertainment %>%
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      32
## 2 TRUE                        8
```

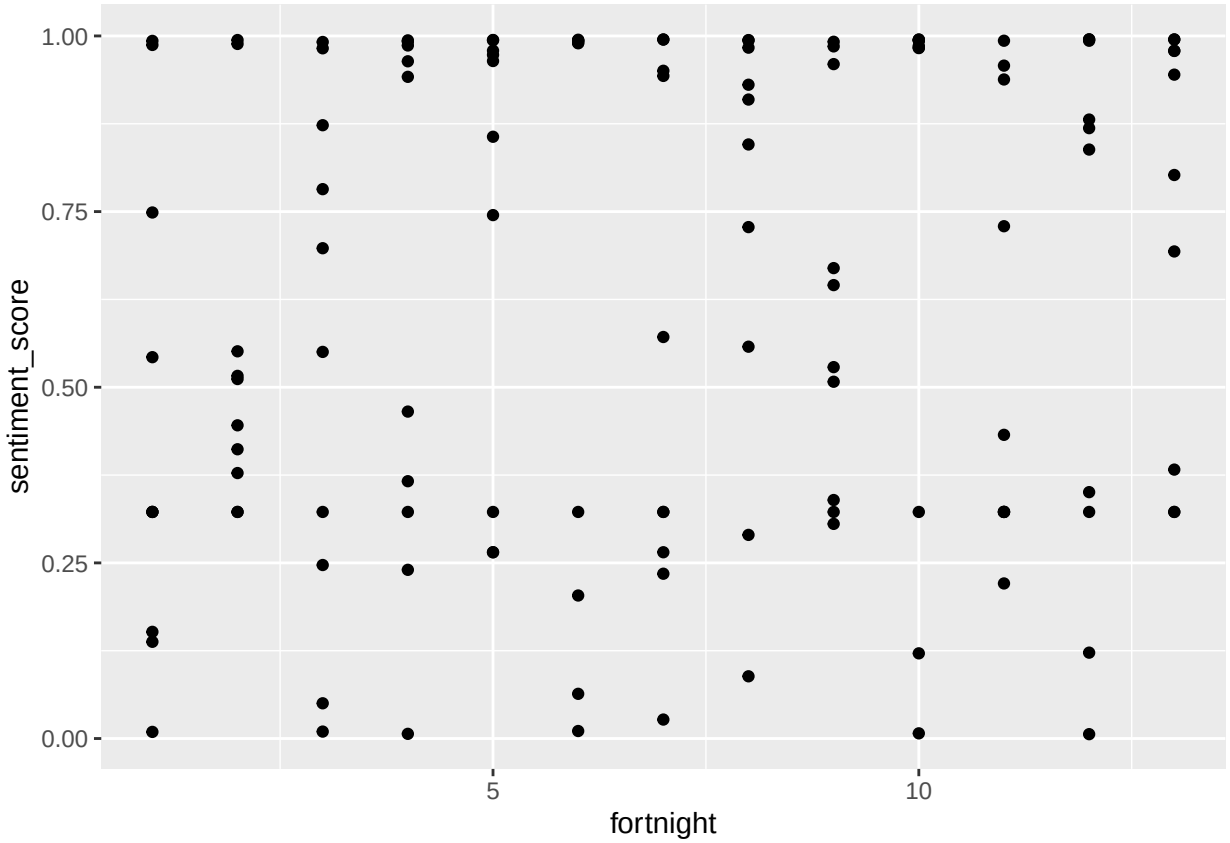
```r
#proportion of positive sentiment videos postoutbreak from sample
p_hat2 = 8/40

p_hat = (44+8)/(89+40)

sd <- sqrt((((p_hat)*(1-p_hat))/89)+(((p_hat)*(1-p_hat))/40))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -3.153) = P(Z <= -3.153) = 0.0008088
##  P(X >  -3.153) = P(Z >  -3.153) = 0.9992
##
```

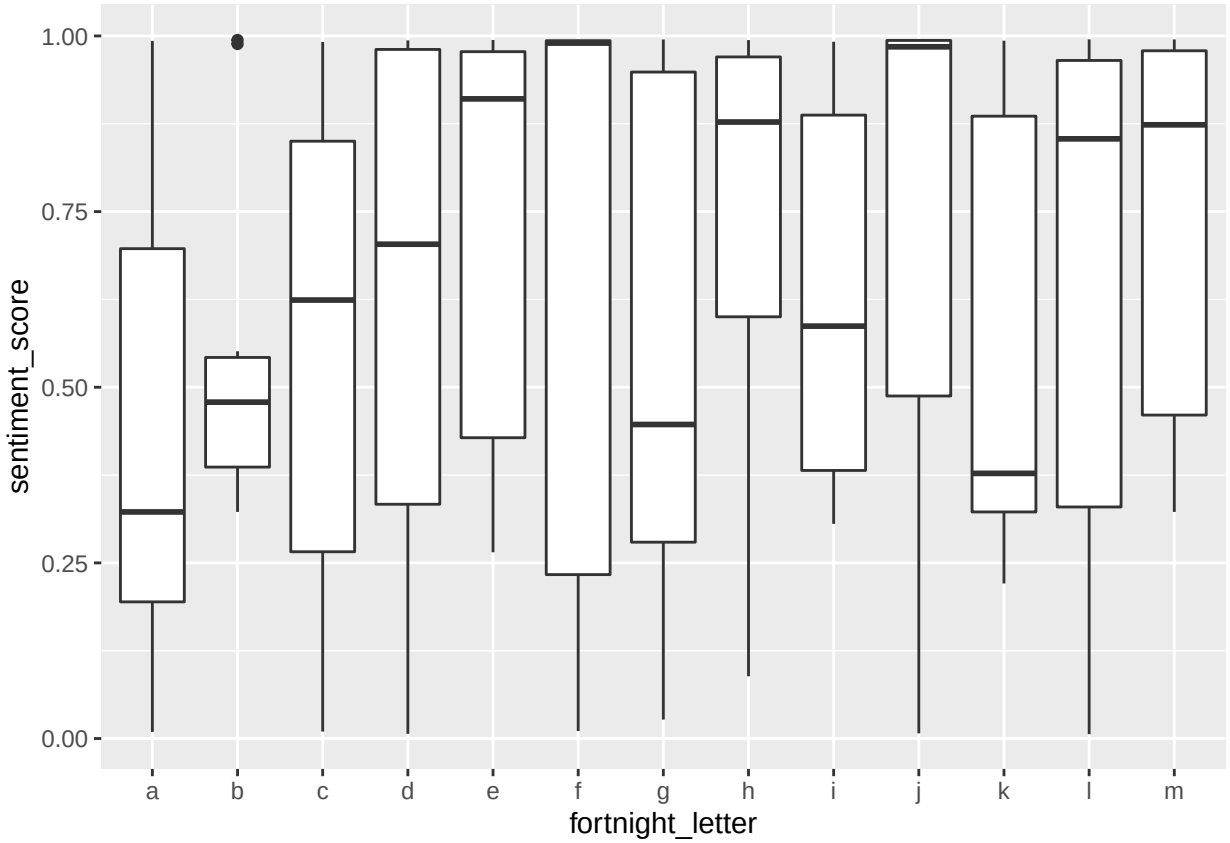



```
## [1] 0.001617621
```

```
#data summary news and politics
ggplot(UK_analysis_news) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



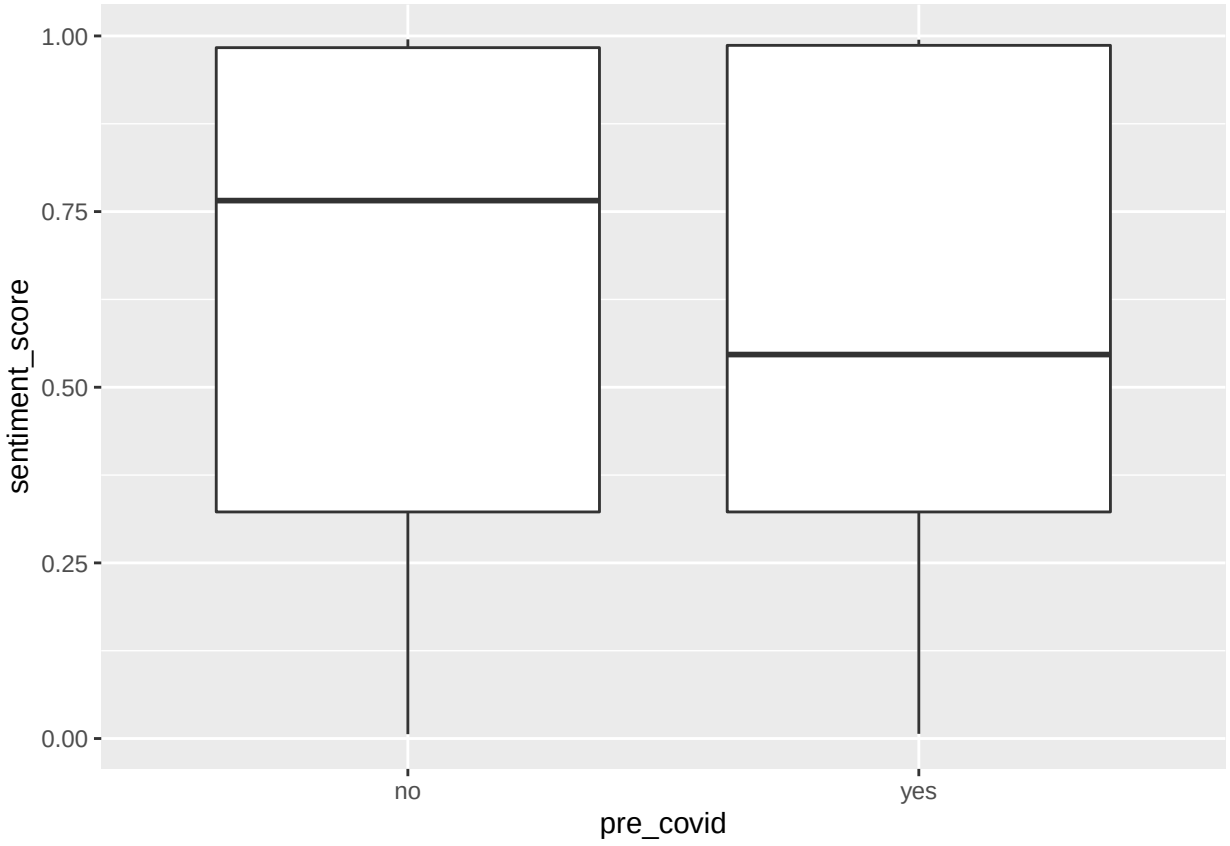

```
ggplot(UK_analysis_news) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_news %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.454
##  2         2                   0.544
##  3         3                   0.551
##  4         4                   0.628
##  5         5                   0.736
##  6         6                   0.655
##  7         7                   0.563
##  8         8                   0.732
##  9         9                   0.626
## 10        10                   0.738
## 11        11                   0.556
## 12        12                   0.637
## 13        13                   0.742
```

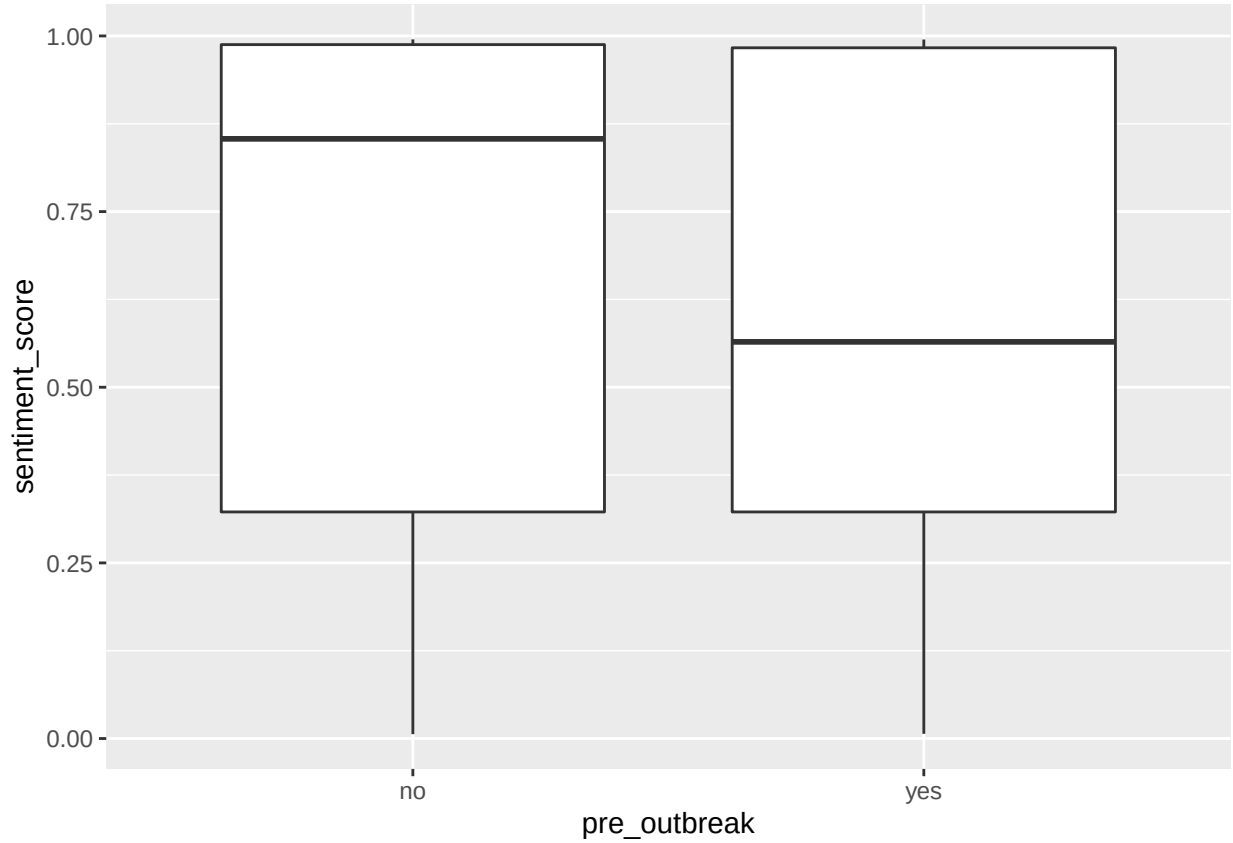
```
ggplot(UK_analysis_news) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_news %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.656
## 2 yes                         0.595
```

```
ggplot(UK_analysis_news) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_news %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.668
## 2 yes                            0.610
```

```
#precovid news
count(UK_analysis_news, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    60
```

```
num_precovid = 60
num_postcovid = 70
num = 129

UK_analysis_news %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
```

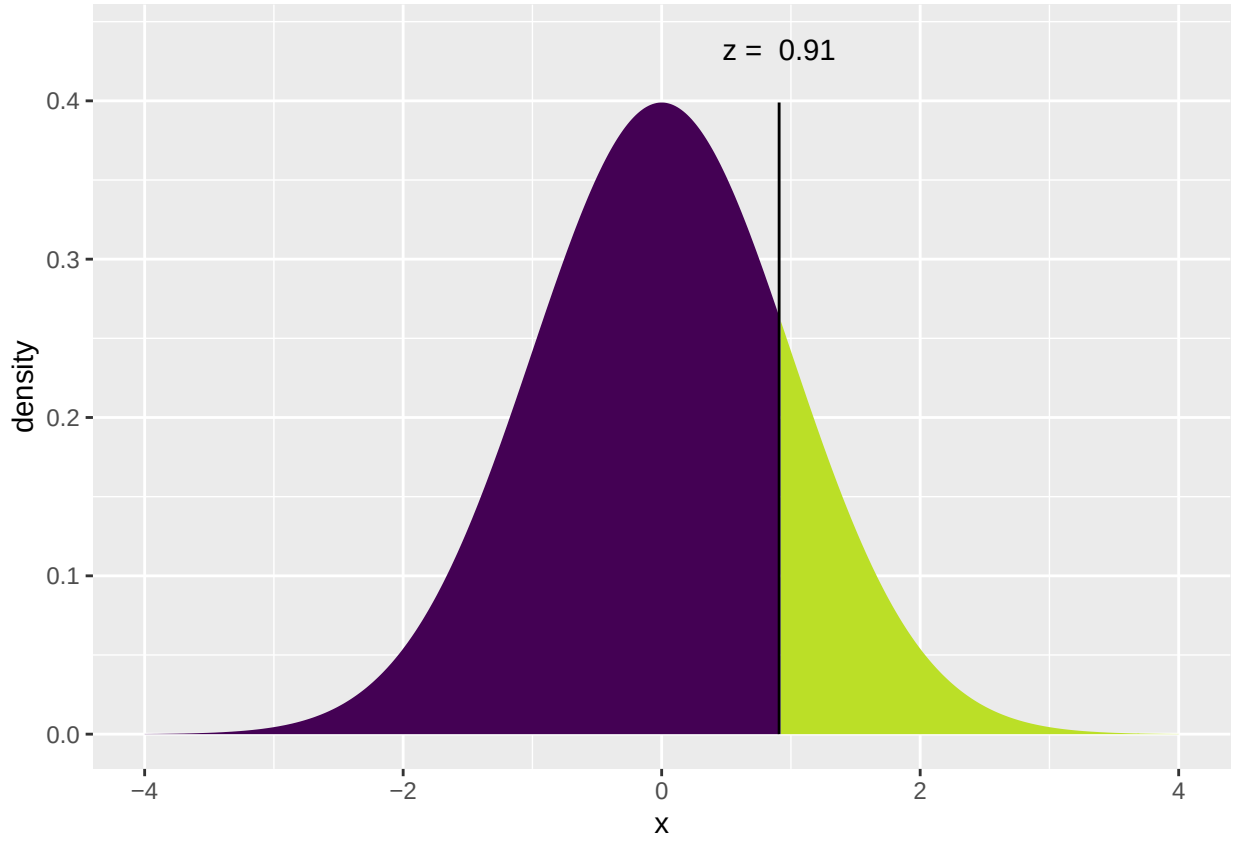
```
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      27
## 2 TRUE                       33
```

```
#proportion of positive sentiment videos precovid from sample
p_hat1 = 33/60

UK_analysis_news %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      26
## 2 TRUE                       44
```

```
#proportion of positive sentiment videos postcovid from sample
p_hat2 = 44/70

p_hat = (33+44)/(60+70)

sd <- sqrt((((p_hat)*(1-p_hat))/60)+(((p_hat)*(1-p_hat))/70))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (1-xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= 0.9088) = P(Z <= 0.9088) = 0.8183
##  P(X >  0.9088) = P(Z >  0.9088) = 0.1817
##
```




```
## [1] 0.3634454
```

```
#outbreak news
count(UK_analysis_news, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       90
```

```
num_preoutbreak = 90
num_postoutbreak = 40
num = 130

UK_analysis_news %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      37
## 2 TRUE                       53
```

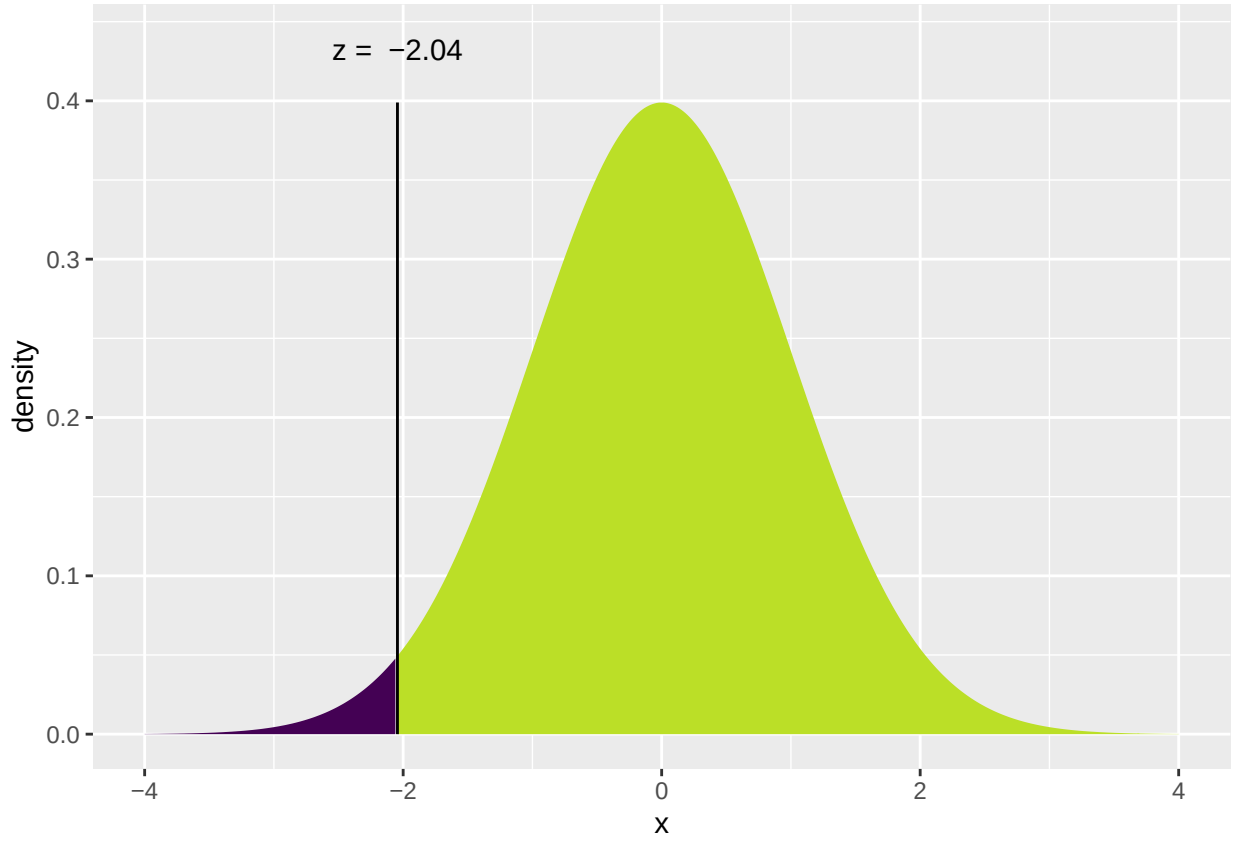
```
#proportion of positive sentiment videos preoutbreak from sample
p_hat1 = 53/79
```

```r
UK_analysis_news %>%
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      16
## 2 TRUE                       24
```

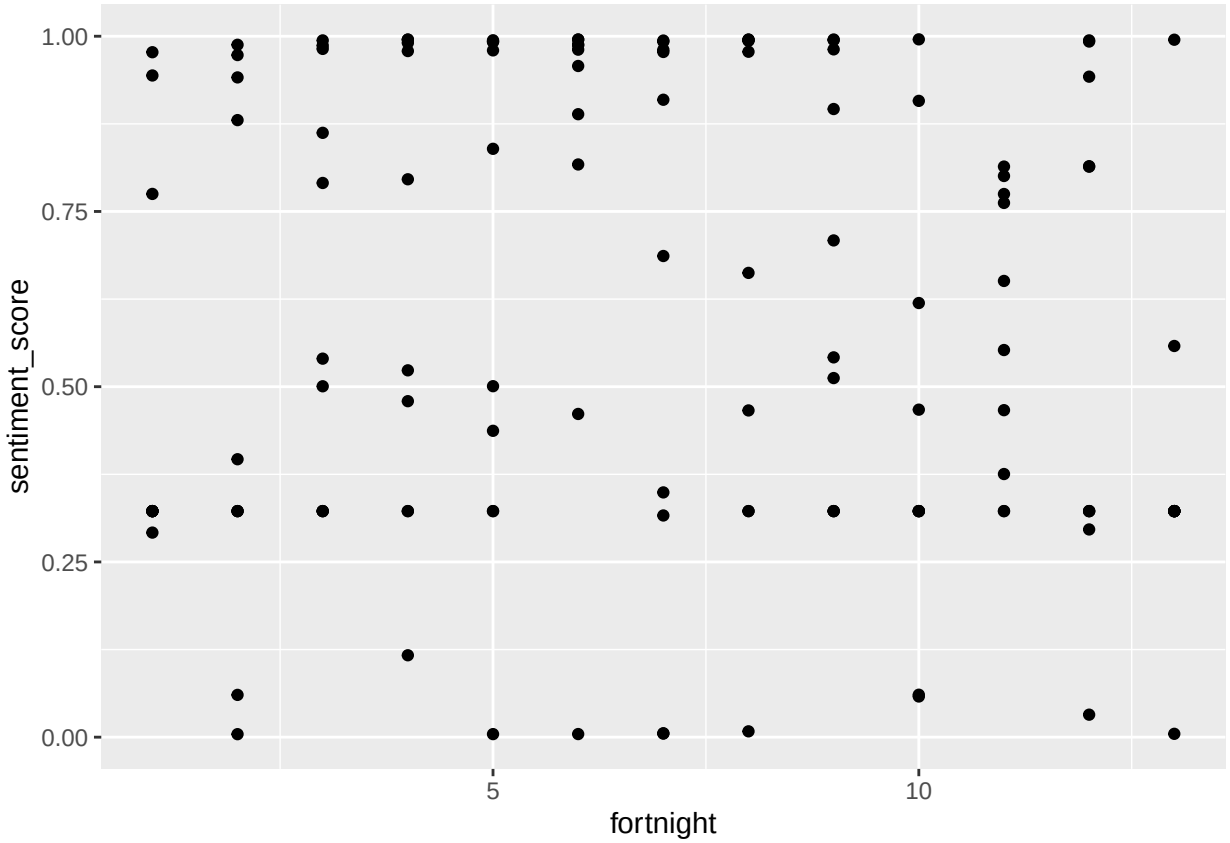
```r
#proportion of positive sentiment videos postoutbreak from sample
p_hat2 = 24/50

p_hat = (53+24)/(90+40)

sd <- sqrt((((p_hat)*(1-p_hat))/90)+(((p_hat)*(1-p_hat))/40))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -2.044) = P(Z <= -2.044) = 0.02047
##  P(X >  -2.044) = P(Z >  -2.044) = 0.9795
##
```

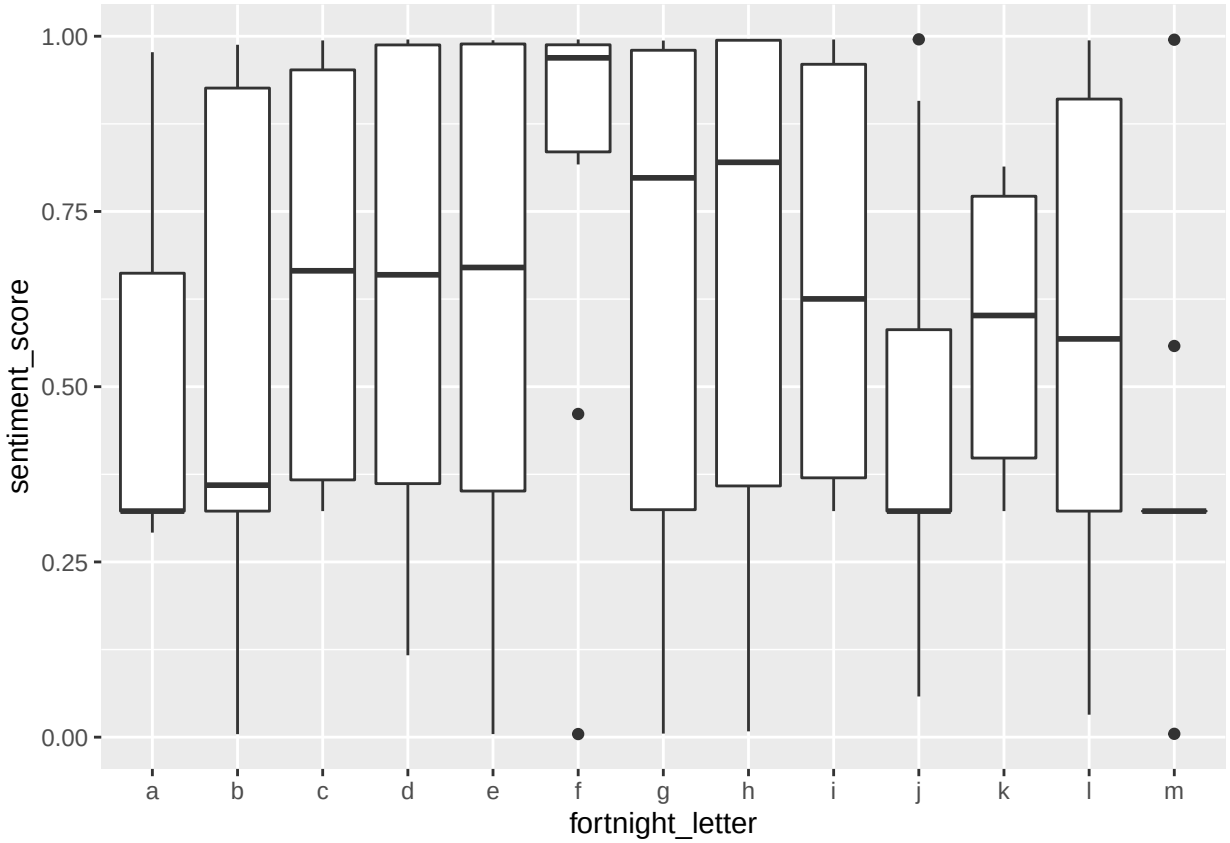



```
## [1] 0.04093821
```

```
#data summary how-to and style
ggplot(UK_analysis_how_to) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



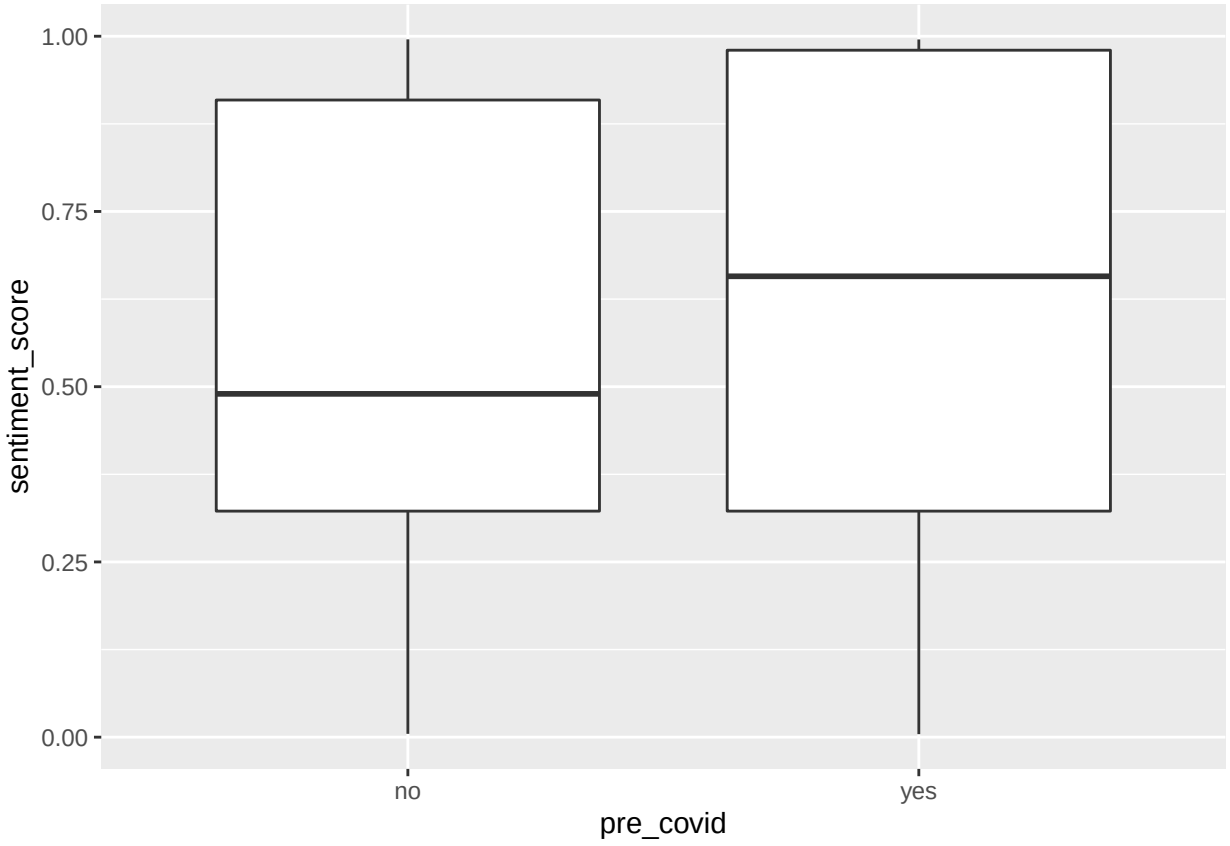

```
ggplot(UK_analysis_how_to) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_how_to %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.492
##  2         2                   0.521
##  3         3                   0.662
##  4         4                   0.652
##  5         5                   0.639
##  6         6                   0.808
##  7         7                   0.622
##  8         8                   0.674
##  9         9                   0.660
## 10        10                   0.440
## 11        11                   0.584
## 12        12                   0.585
## 13        13                   0.382
```

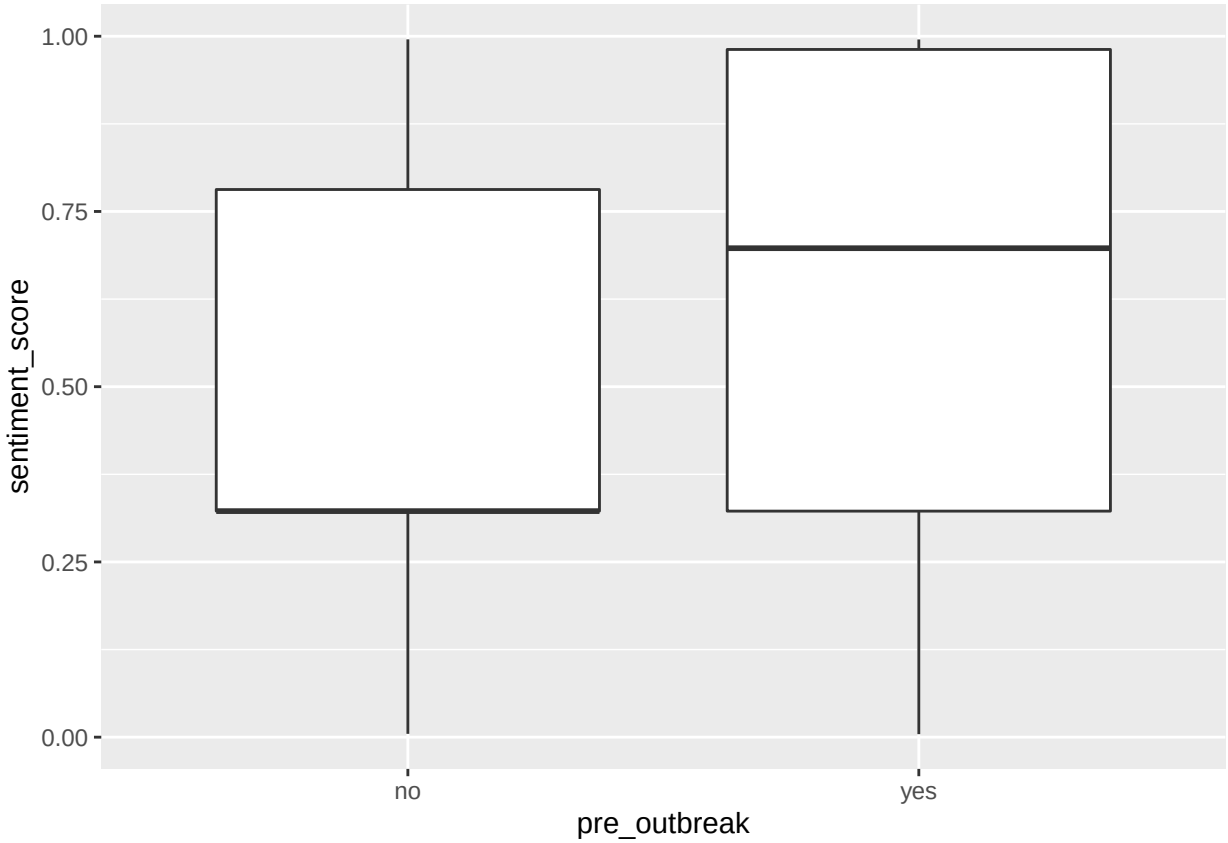
```
ggplot(UK_analysis_how_to) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_how_to %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.564
## 2 yes                         0.629
```

```
ggplot(UK_analysis_how_to) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_how_to %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.498
## 2 yes                            0.637
```

```
#precovid how-to
count(UK_analysis_how_to, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    60
```

```
num_precovid = 60
num_postcovid = 70
num = 130

UK_analysis_how_to %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
```

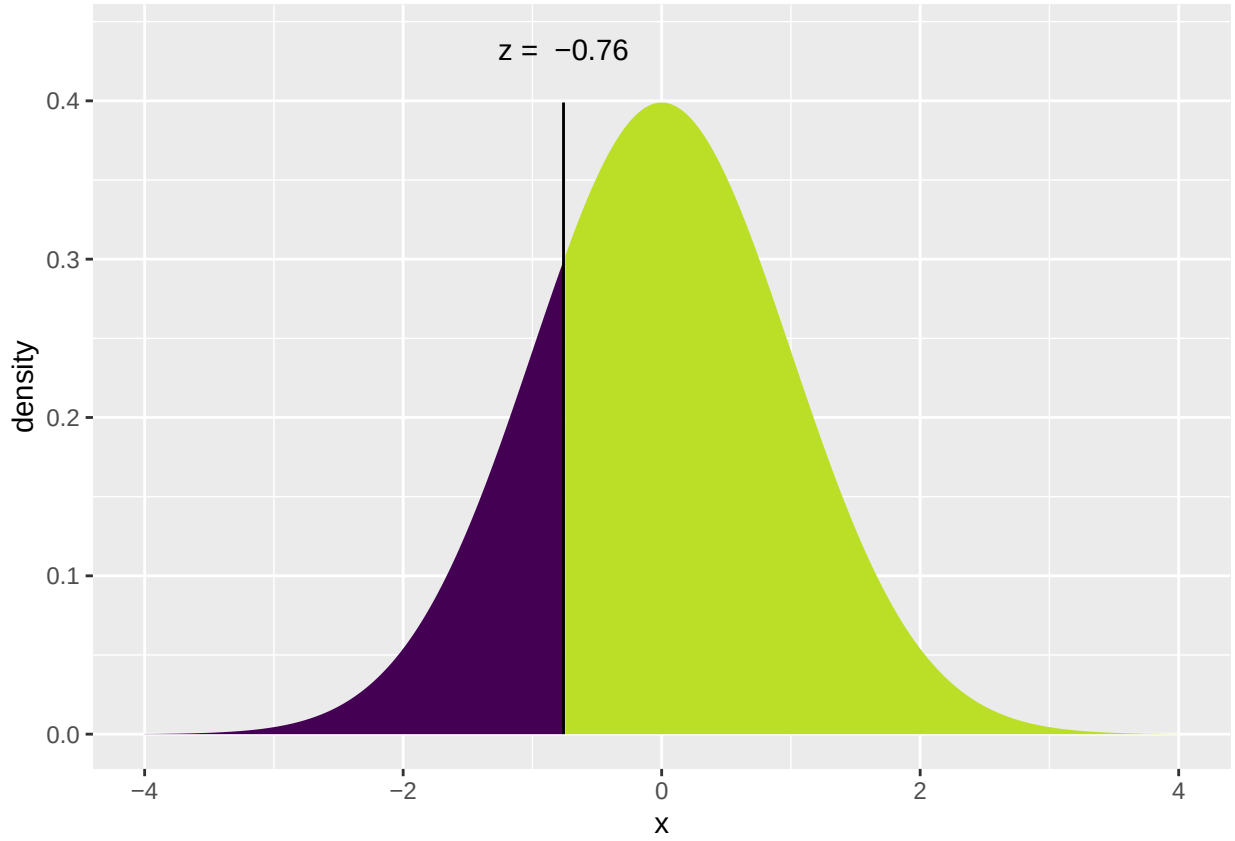
```
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      26
## 2 TRUE                       34
```

```
#proportion of positive sentiment videos precovid from sample
p_hat1 = 34/60

UK_analysis_how_to %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      35
## 2 TRUE                       35
```

```
#proportion of positive sentiment videos postcovid from sample
p_hat2 = 35/70

p_hat = (34+35)/(60+70)

sd <- sqrt((((p_hat)*(1-p_hat))/60)+(((p_hat)*(1-p_hat))/70))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -0.7593) = P(Z <= -0.7593) = 0.2238
##  P(X >  -0.7593) = P(Z >  -0.7593) = 0.7762
##
```




```
## [1] 0.4476708
```

```
#outbreak how-to
count(UK_analysis_how_to, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       90
```

```
num_preoutbreak = 90
num_postoutbreak = 40
num = 130

UK_analysis_how_to %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      37
## 2 TRUE                       53
```

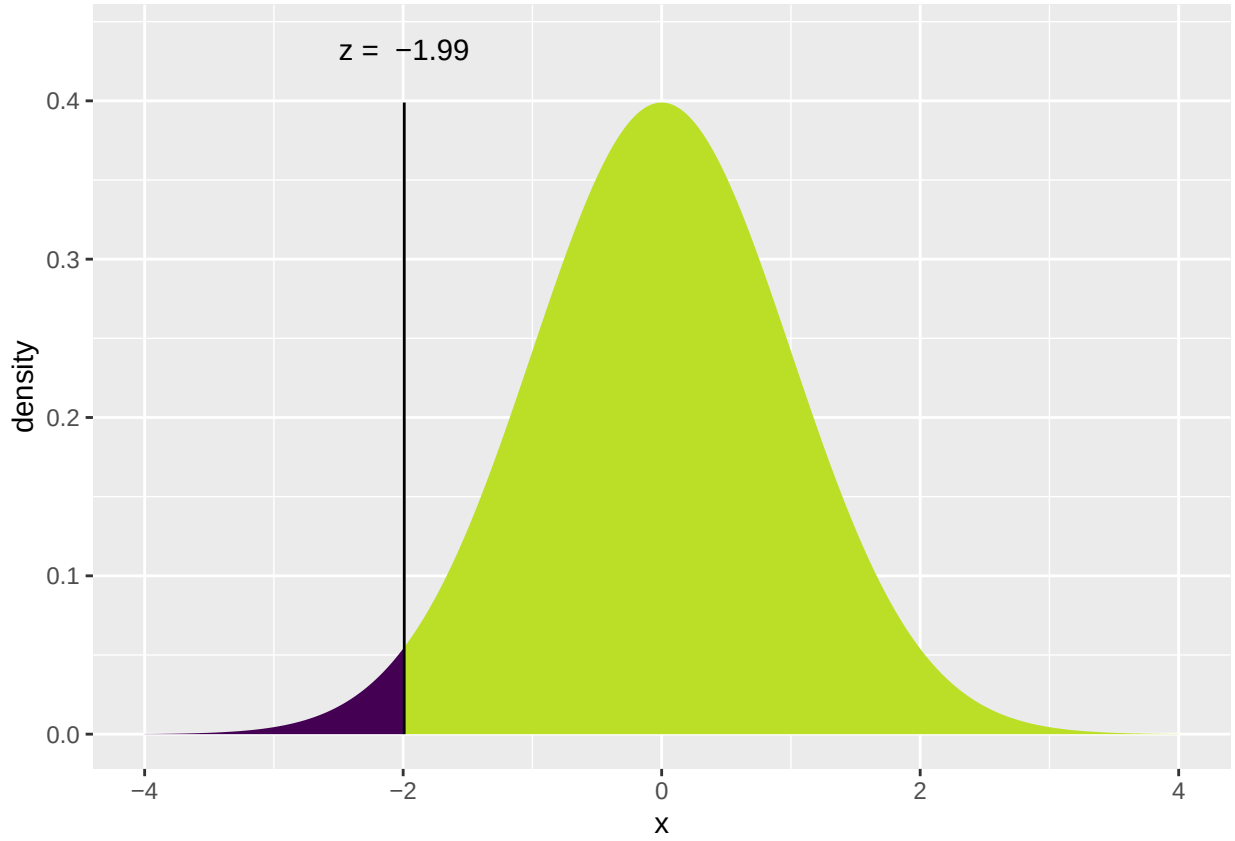
```
#proportion of positive sentiment videos preoutbreak from sample
p_hat1 = 53/90
```

```r
UK_analysis_how_to %>%
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      24
## 2 TRUE                       16
```

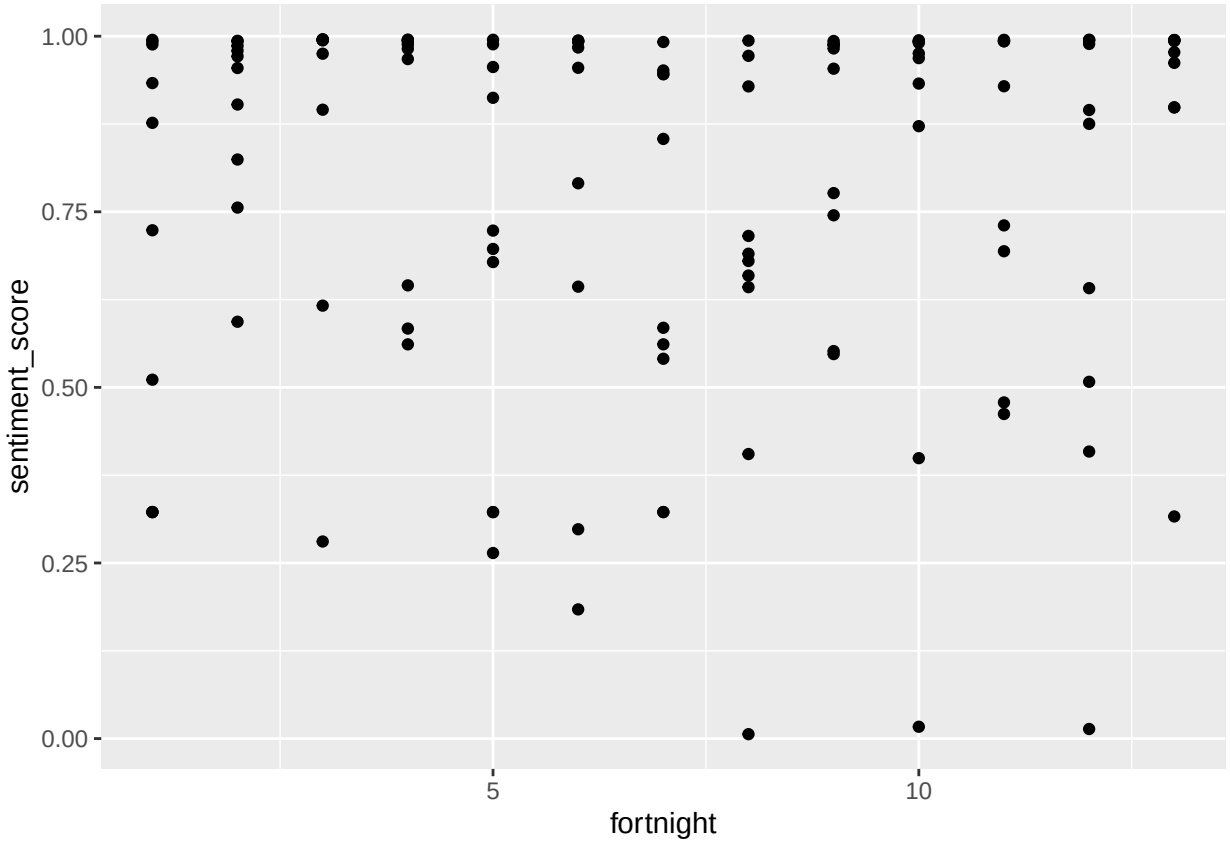
```r
#proportion of positive sentiment videos postoutbreak from sample
p_hat2 = 16/40

p_hat = (53+16)/(90+40)

sd <- sqrt((((p_hat)*(1-p_hat))/90)+(((p_hat)*(1-p_hat))/40))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -1.992) = P(Z <= -1.992) = 0.0232
##  P(X >  -1.992) = P(Z >  -1.992) = 0.9768
##
```

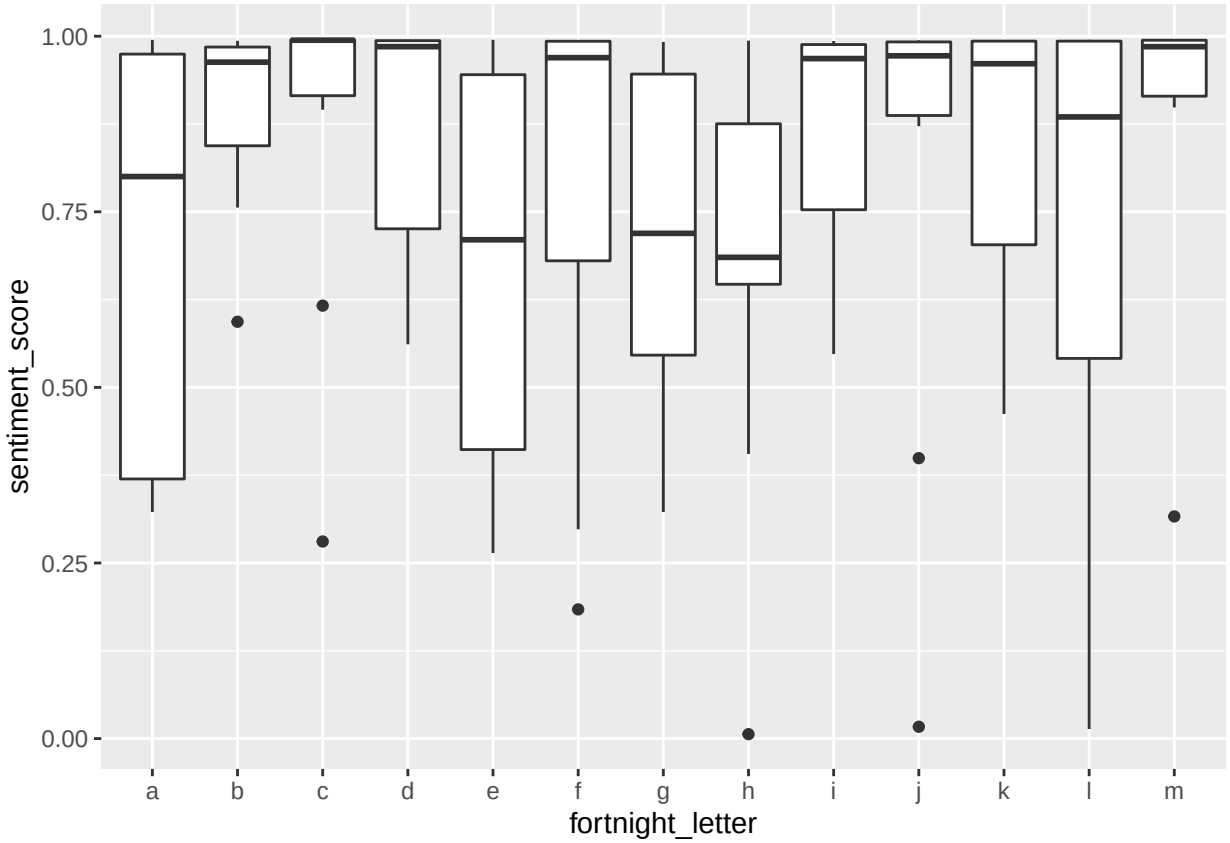



```
## [1] 0.04639595
```

```
#data summary education
ggplot(UK_analysis_education) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



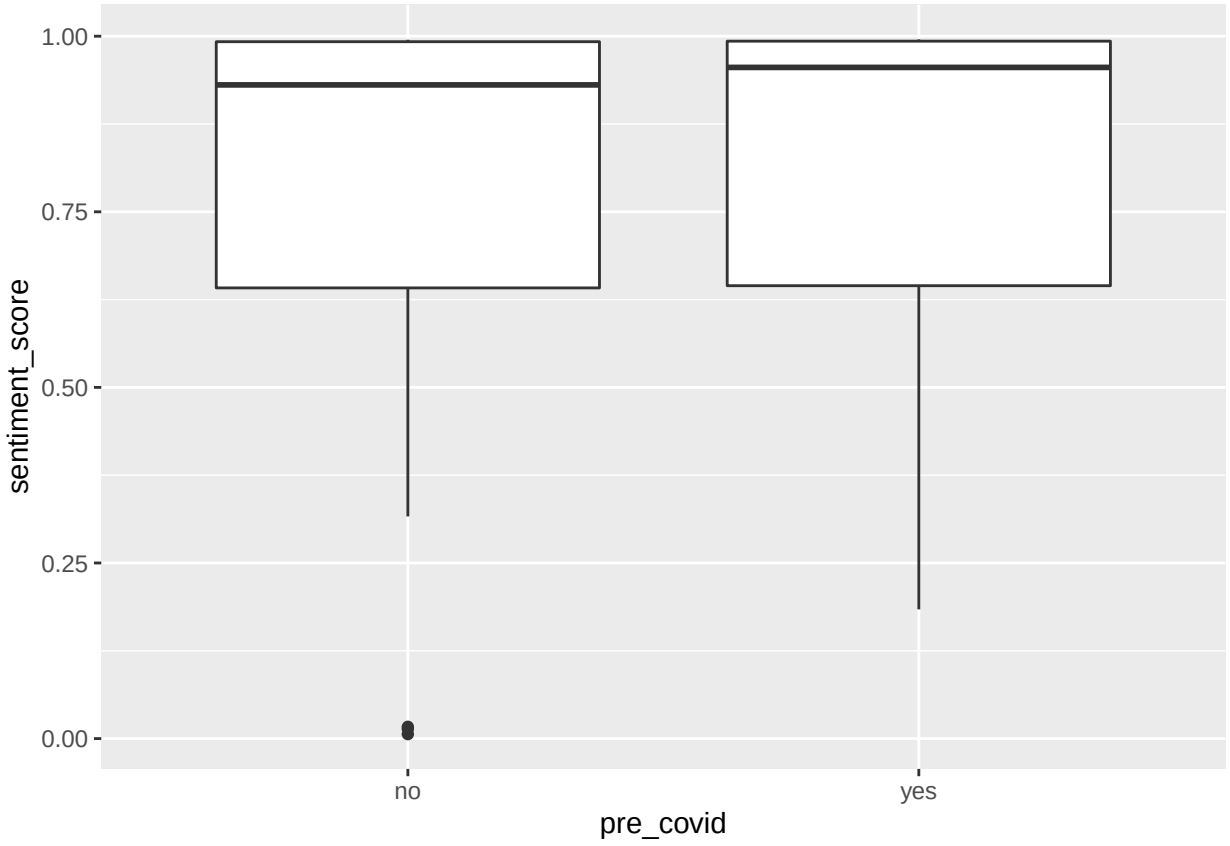

```
ggplot(UK_analysis_education) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_education %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.699
##  2         2                   0.895
##  3         3                   0.874
##  4         4                   0.870
##  5         5                   0.686
##  6         6                   0.783
##  7         7                   0.702
##  8         8                   0.669
##  9         9                   0.851
## 10        10                   0.813
## 11        11                   0.826
## 12        12                   0.731
## 13        13                   0.902
```

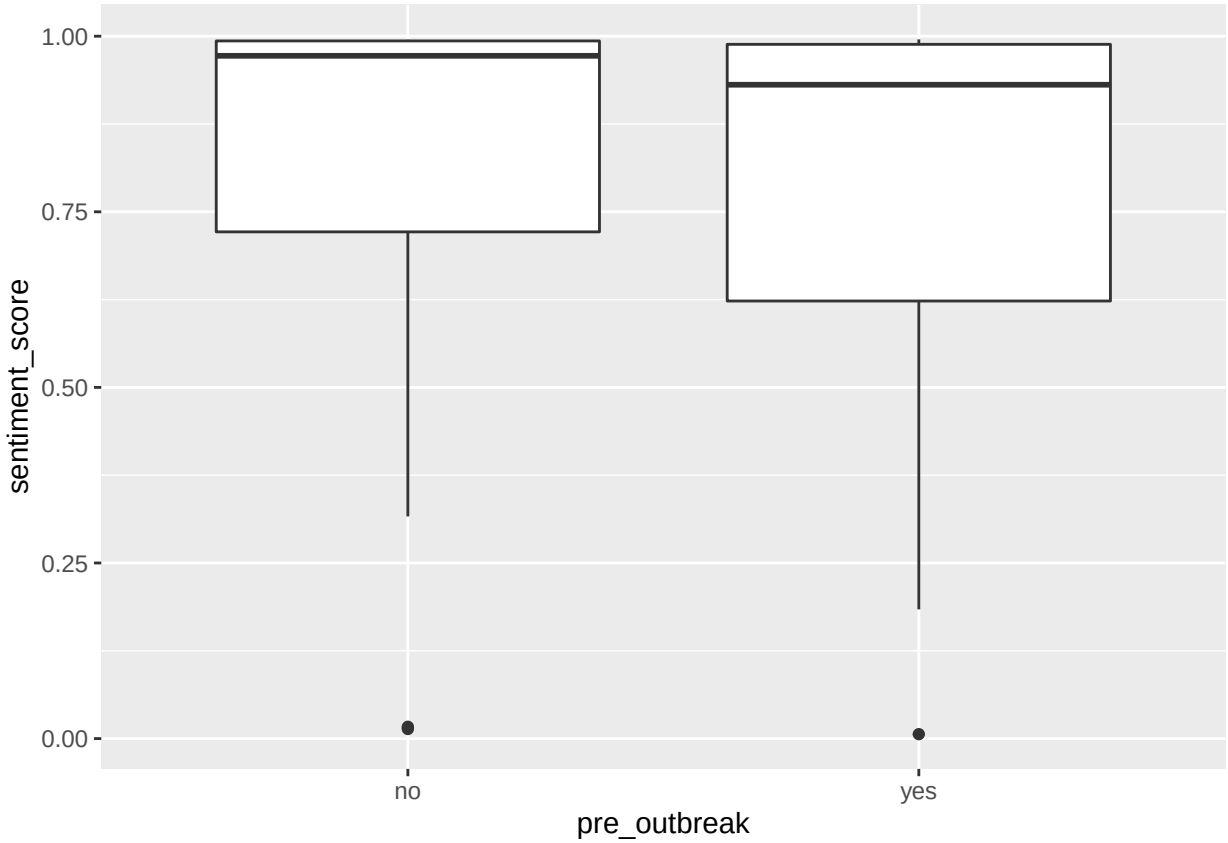
```
ggplot(UK_analysis_education) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_education %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.785
## 2 yes                         0.801
```

```
ggplot(UK_analysis_education) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_education %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.818
## 2 yes                            0.781
```

```
#precovid education
count(UK_analysis_education, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    60
```

```
num_precovid = 60
num_postcovid = 70
num = 130

UK_analysis_education %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
```

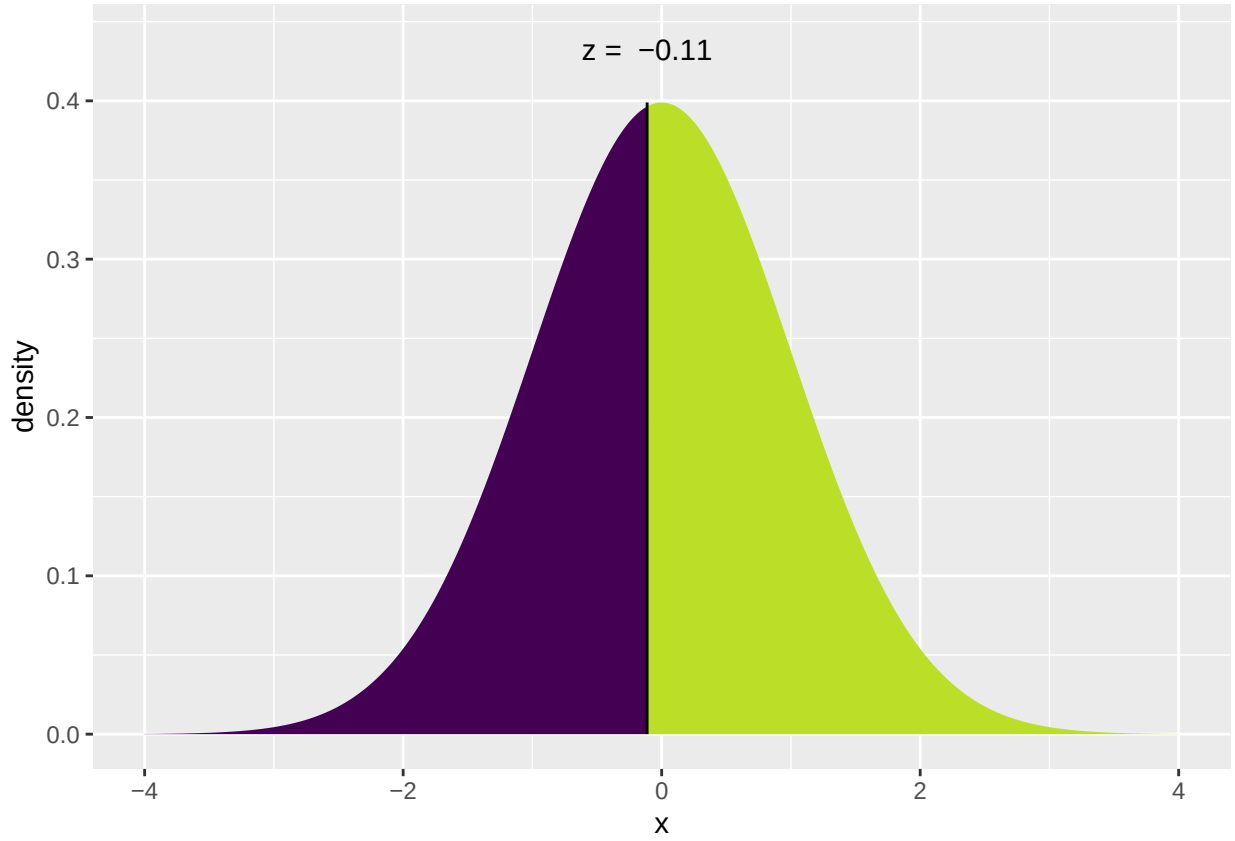
```
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                       9
## 2 TRUE                       51
```

```
#proportion of positive sentiment videos precovid from sample
p_hat1 = 51/60

UK_analysis_education %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      11
## 2 TRUE                       59
```

```
#proportion of positive sentiment videos postcovid from sample
p_hat2 = 59/70

p_hat = (51+59)/(60+70)

sd <- sqrt((((p_hat)*(1-p_hat))/60)+(((p_hat)*(1-p_hat))/70))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -0.1125) = P(Z <= -0.1125) = 0.4552
##  P(X >  -0.1125) = P(Z >  -0.1125) = 0.5448
##
```




```
## [1] 0.9104055
```

```
#outbreak education
count(UK_analysis_education, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       90
```

```
num_preoutbreak = 90
num_postoutbreak = 40
num = 130

UK_analysis_education %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      13
## 2 TRUE                       77
```

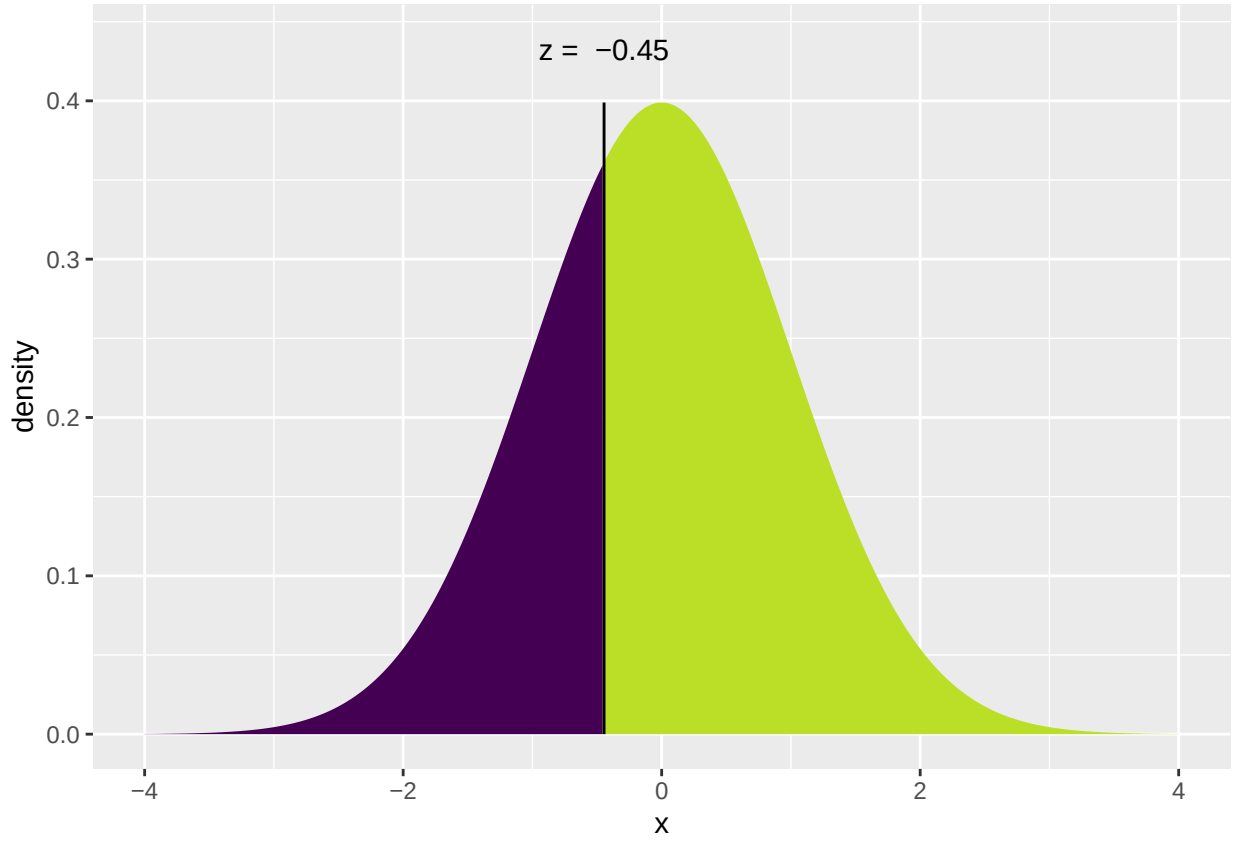
```
#proportion of positive sentiment videos preoutbreak from sample
p_hat1 = 77/90
```

```r
UK_analysis_education %>%
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                       7
## 2 TRUE                       33
```

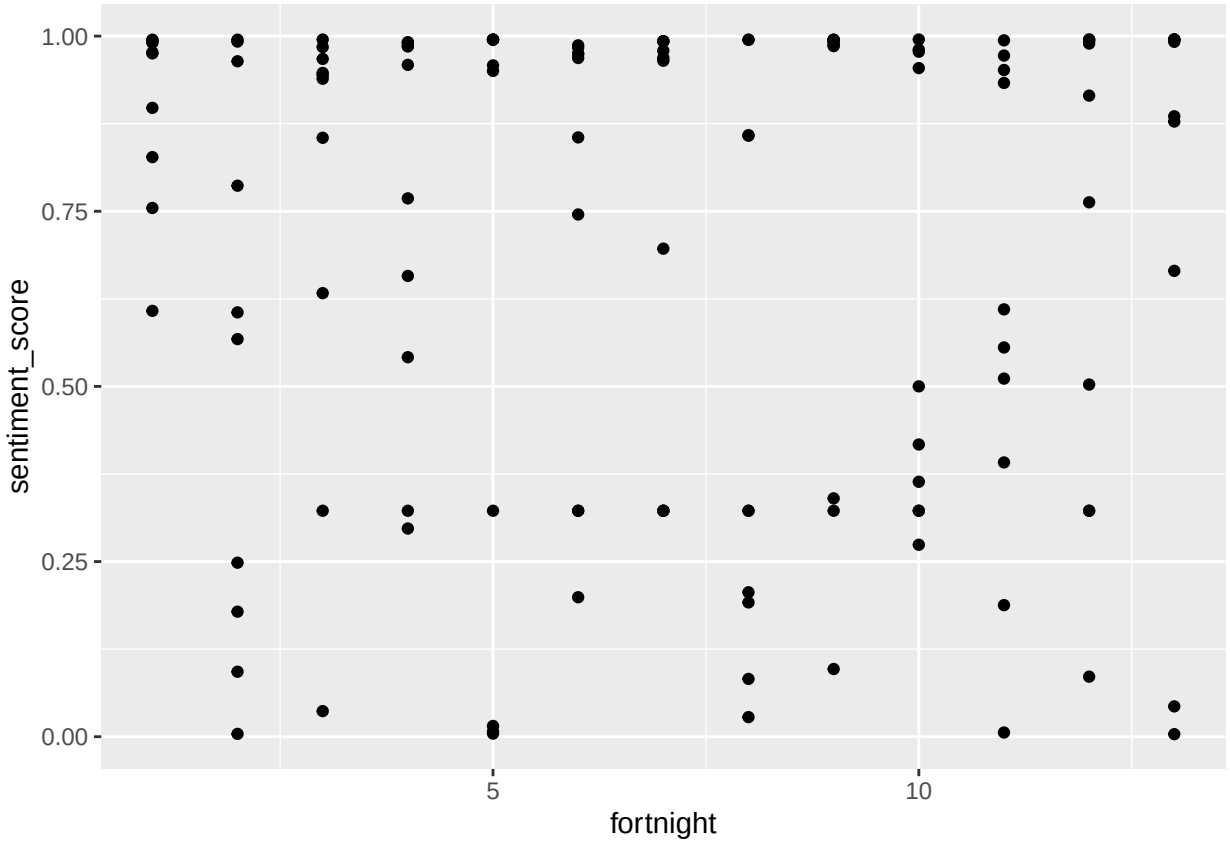
```r
#proportion of positive sentiment videos postoutbreak from sample
p_hat2 = 33/40

p_hat = (77+33)/(90+40)

sd <- sqrt((((p_hat)*(1-p_hat))/90)+(((p_hat)*(1-p_hat))/40))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -0.4457) = P(Z <= -0.4457) = 0.3279
##  P(X >  -0.4457) = P(Z >  -0.4457) = 0.6721
##
```

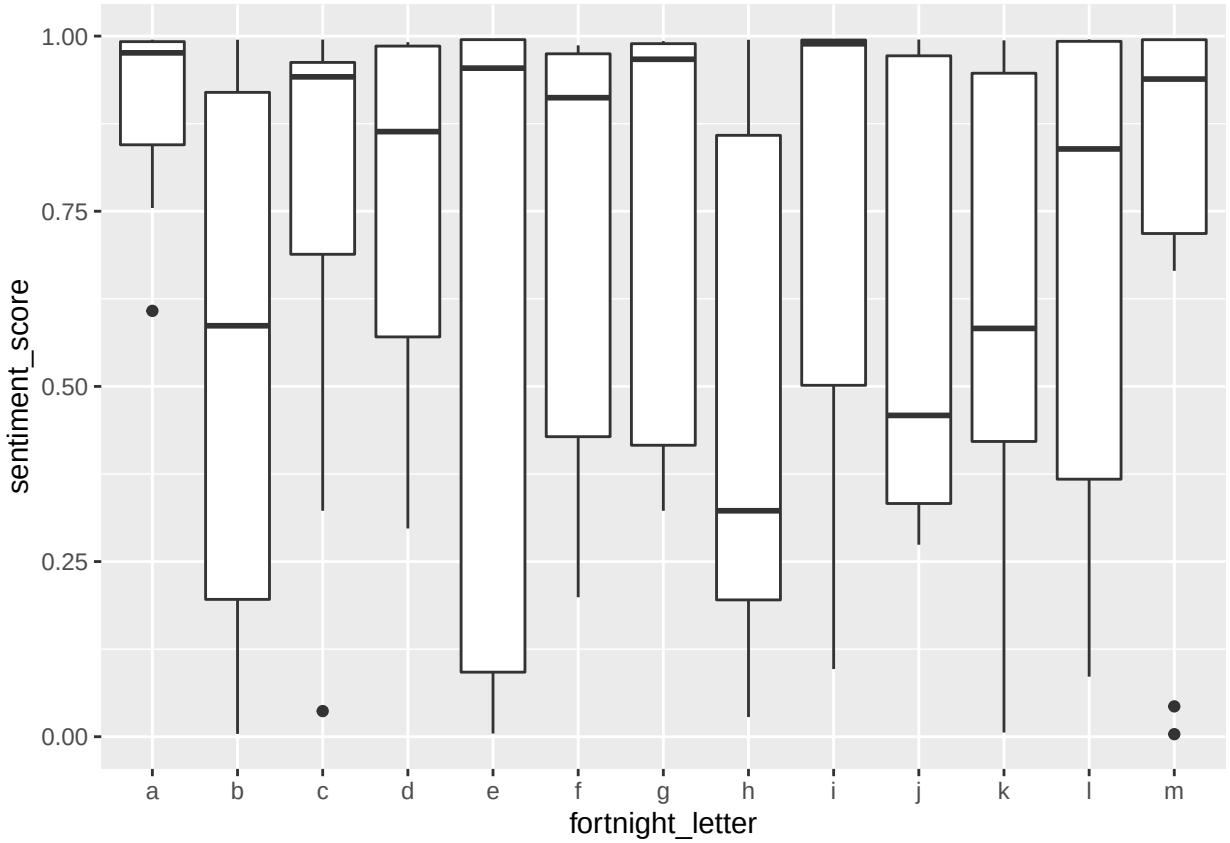



```
## [1] 0.6558443
```

```
#data summary science and technology
ggplot(UK_analysis_science) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



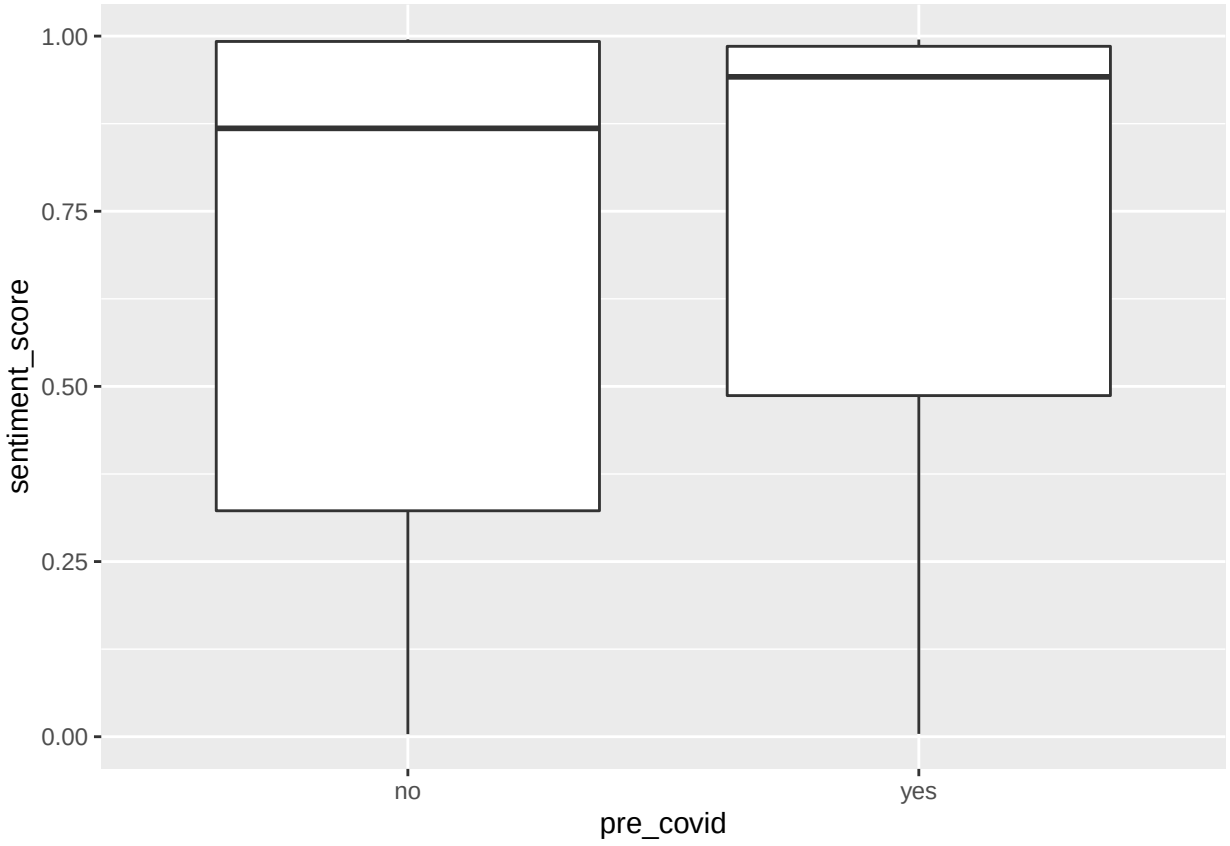

```
ggplot(UK_analysis_science) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_science %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.901
##  2         2                   0.543
##  3         3                   0.763
##  4         4                   0.750
##  5         5                   0.624
##  6         6                   0.733
##  7         7                   0.755
##  8         8                   0.486
##  9         9                   0.770
## 10        10                   0.611
## 11        11                   0.611
## 12        12                   0.688
## 13        13                   0.745
```

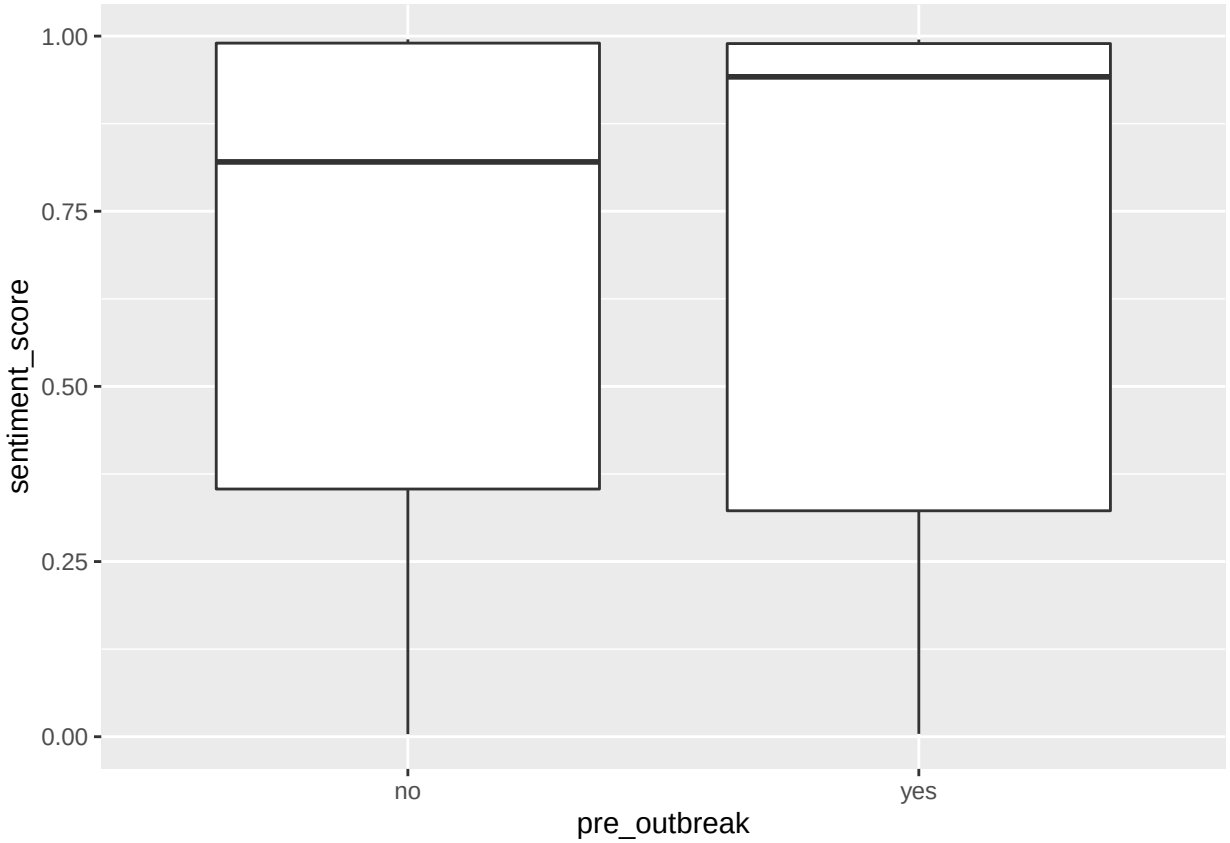
```
ggplot(UK_analysis_science) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_science %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.667
## 2 yes                         0.719
```

```
ggplot(UK_analysis_science) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_science %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.664
## 2 yes                            0.703
```

```
#precovid scitech
count(UK_analysis_science, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    60
```

```
num_precovid = 60
num_postcovid = 70
num = 130

UK_analysis_science %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
```

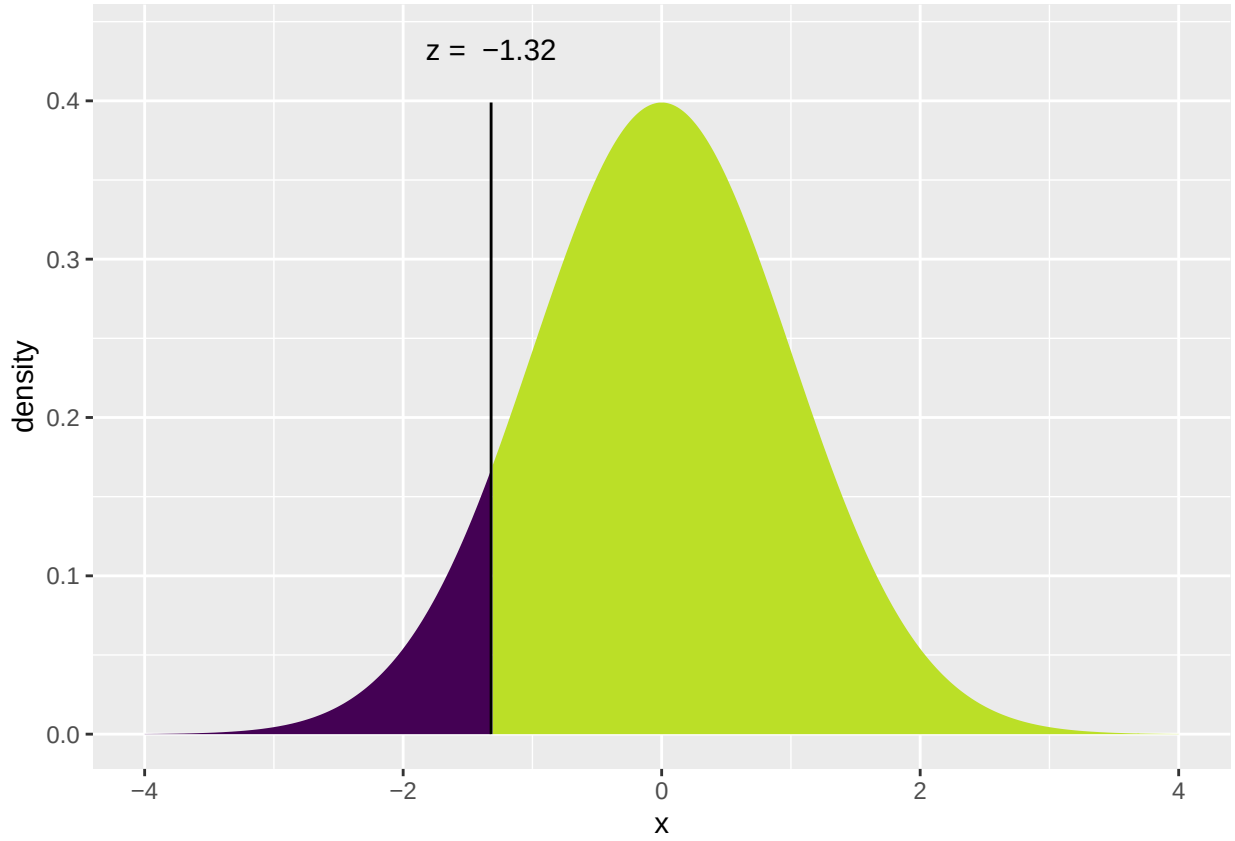
```
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      15
## 2 TRUE                       45
```

```
#proportion of positive sentiment videos precovid from sample
p_hat1 = 45/60

UK_analysis_science %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      25
## 2 TRUE                       45
```

```
#proportion of positive sentiment videos postcovid from sample
p_hat2 = 45/70

p_hat = (45+45)/(60+70)

sd <- sqrt((((p_hat)*(1-p_hat))/60)+(((p_hat)*(1-p_hat))/70))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -1.319) = P(Z <= -1.319) = 0.0935
##  P(X >  -1.319) = P(Z >  -1.319) = 0.9065
##
```




```
## [1] 0.1870031
```

```
#outbreak scitech
count(UK_analysis_science, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       90
```

```
num_preoutbreak = 90
num_postoutbreak = 40
num = 130

UK_analysis_science %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      27
## 2 TRUE                       63
```

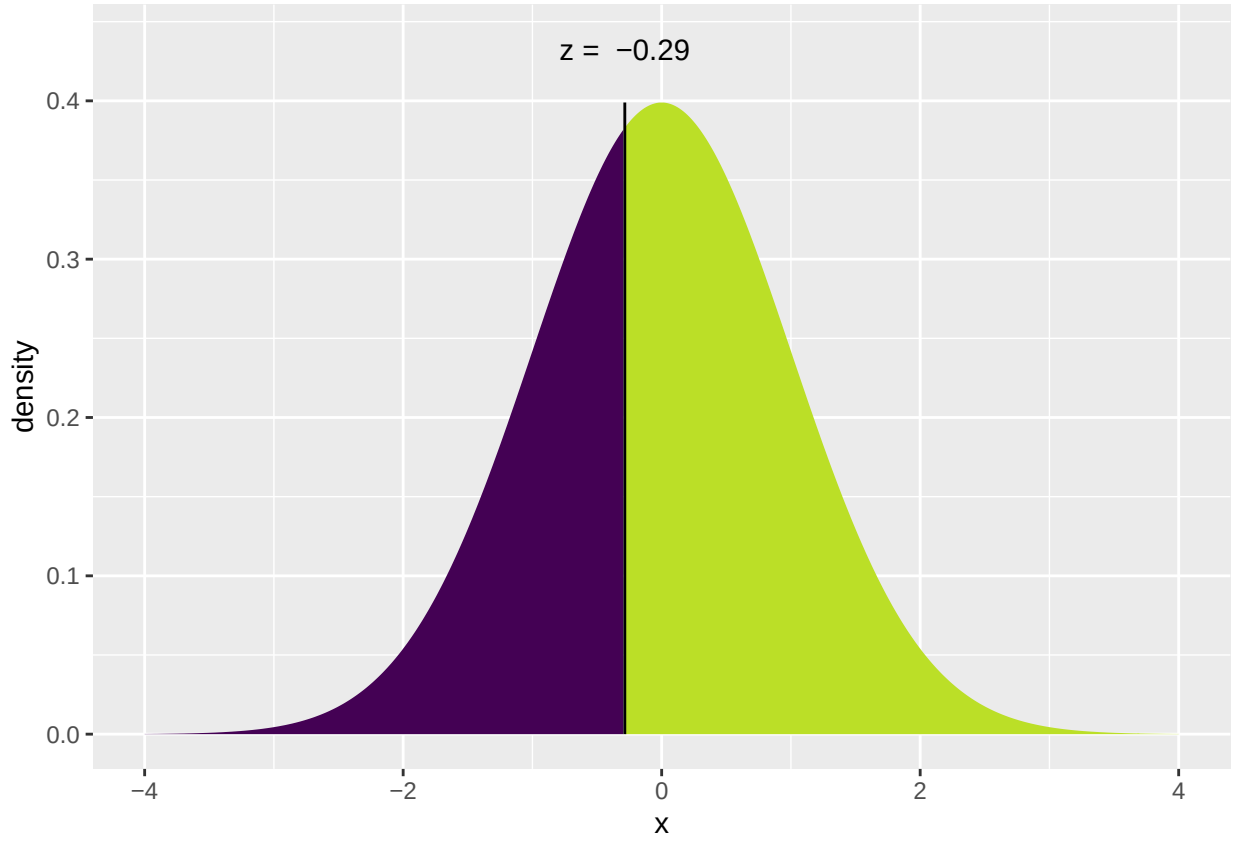
```
#proportion of positive sentiment videos preoutbreak from sample
p_hat1 = 63/90
```

```r
UK_analysis_science %>%
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      13
## 2 TRUE                       27
```

```r
#proportion of positive sentiment videos postoutbreak from sample
p_hat2 = 27/40

p_hat = (63+27)/(90+40)

sd <- sqrt((((p_hat)*(1-p_hat))/90)+(((p_hat)*(1-p_hat))/40))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

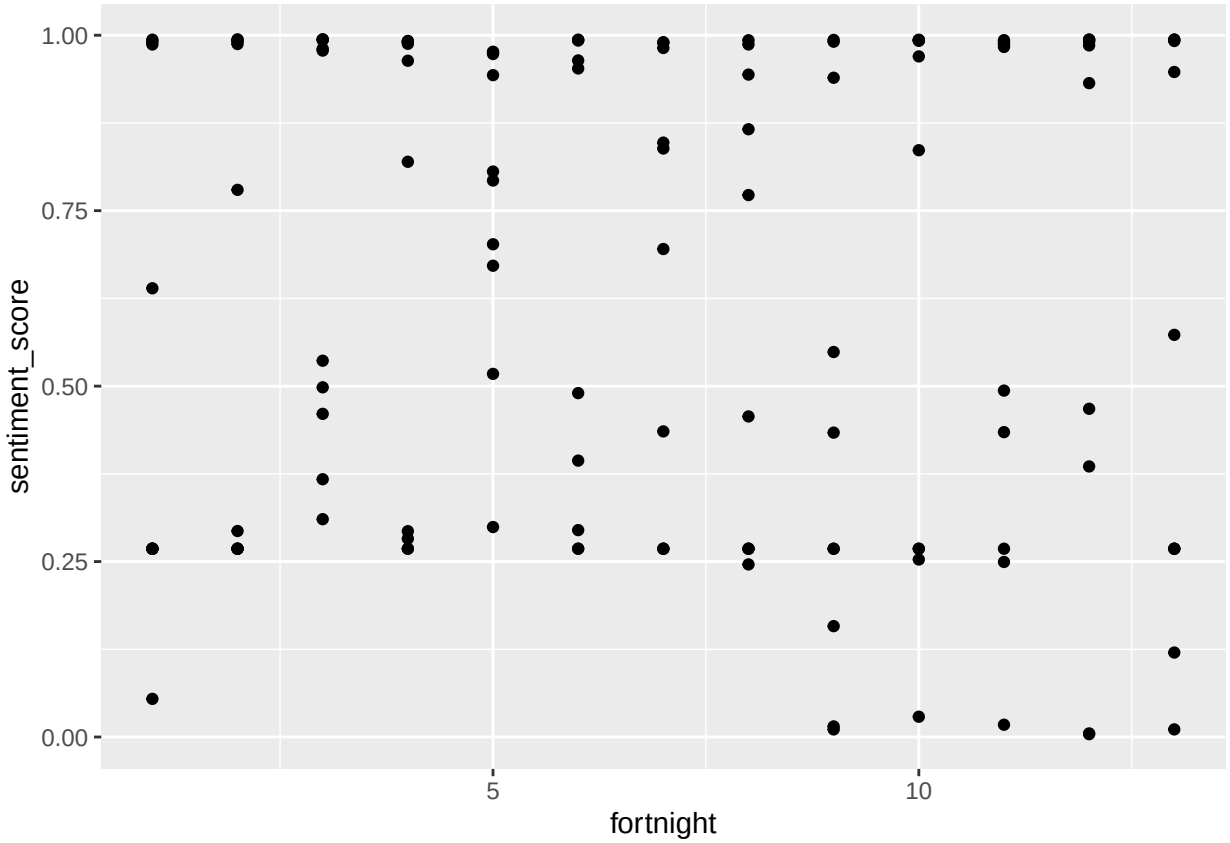
```
##
## If X ~ N(0, 1), then
##  P(X <= -0.285) = P(Z <= -0.285) = 0.3878
##  P(X >  -0.285) = P(Z >  -0.285) = 0.6122
##
```



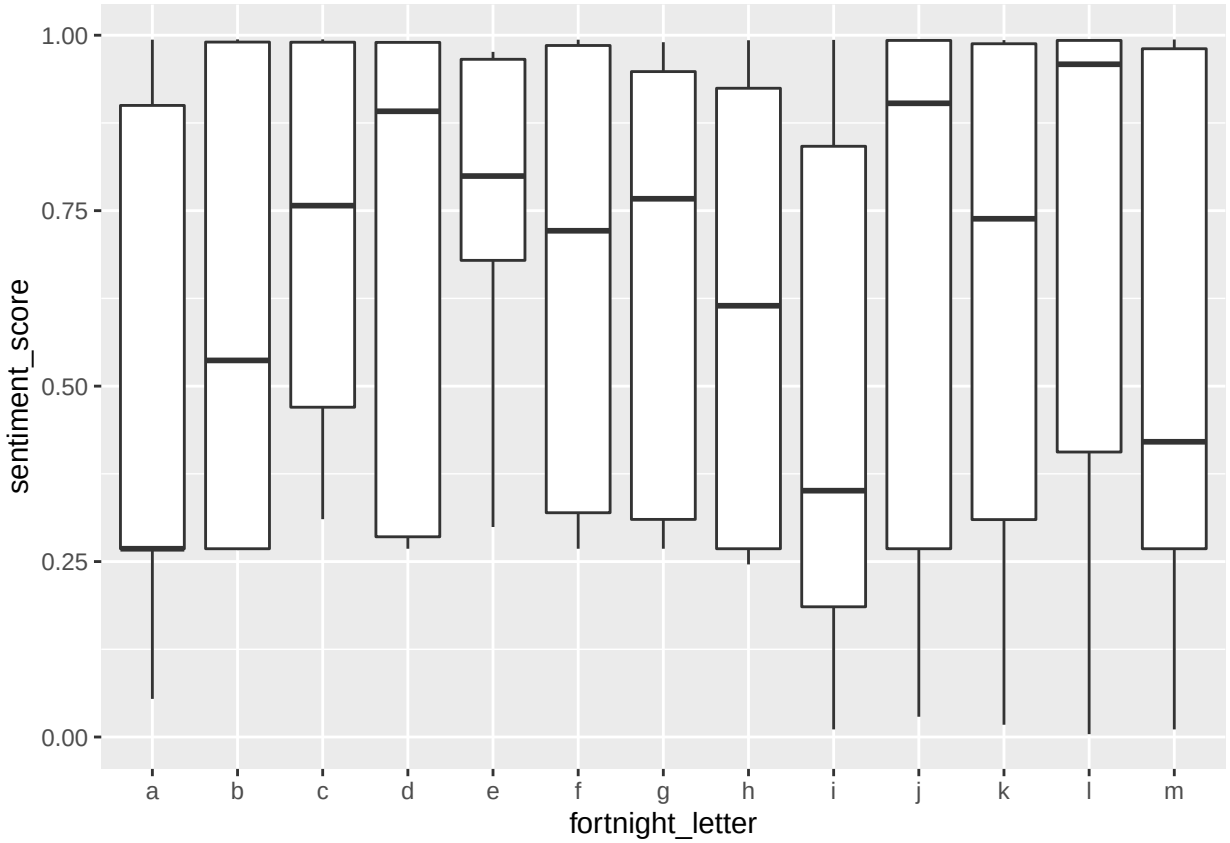

```
## [1] 0.7756105
```

```
#Youtube API All Categories
UK_analysis_all <- UK_analysis %>%
  filter(video_category == "All")
```

```
#data summary all categories
ggplot(UK_analysis_all) +
  geom_point(aes(x = fortnight, y = sentiment_score))
```



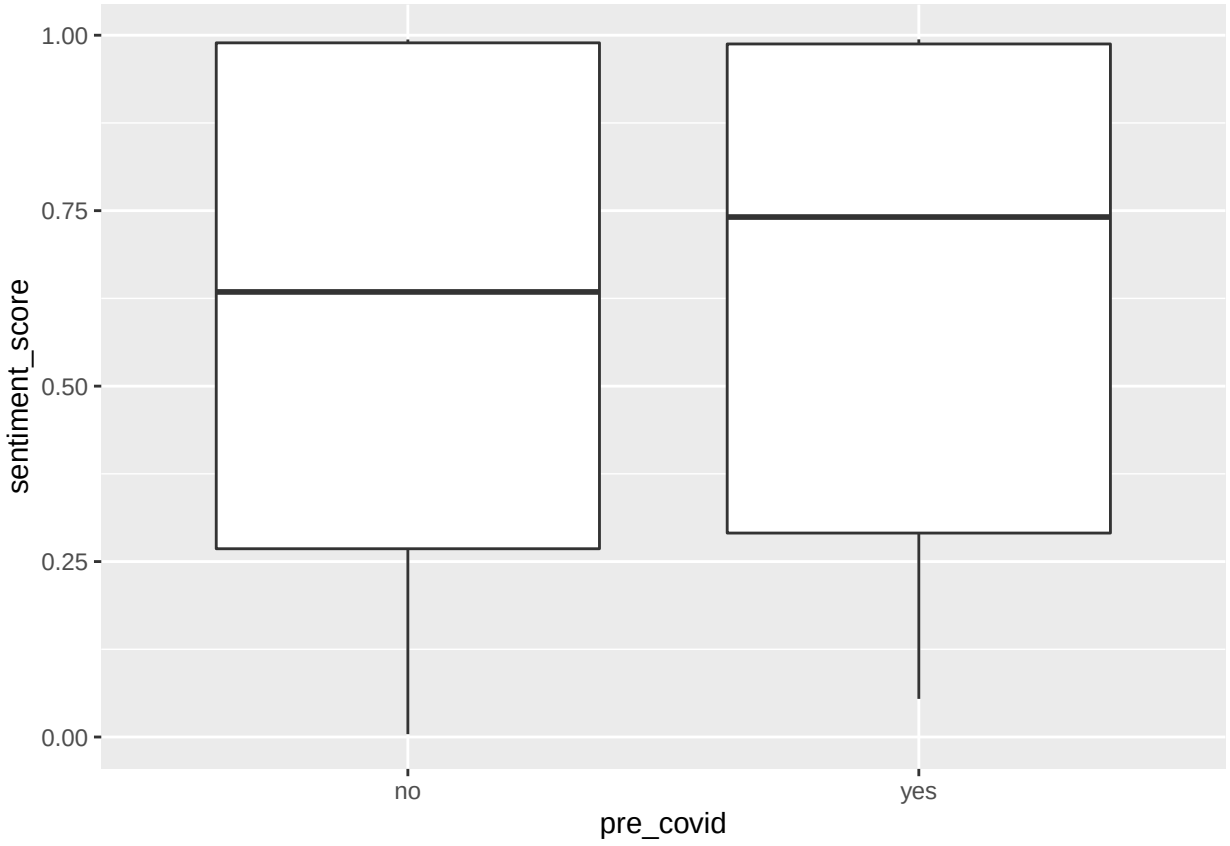

```
ggplot(UK_analysis_all) +
  geom_boxplot(aes(x = fortnight_letter, y = sentiment_score))
```




```
UK_analysis_all %>%
  group_by(fortnight) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 13 x 2
##    fortnight `mean(sentiment_score)`
##        <dbl>                   <dbl>
##  1         1                   0.501
##  2         2                   0.611
##  3         3                   0.711
##  4         4                   0.686
##  5         5                   0.766
##  6         6                   0.661
##  7         7                   0.658
##  8         8                   0.607
##  9         9                   0.463
## 10        10                   0.660
## 11        11                   0.641
## 12        12                   0.675
## 13        13                   0.544
```

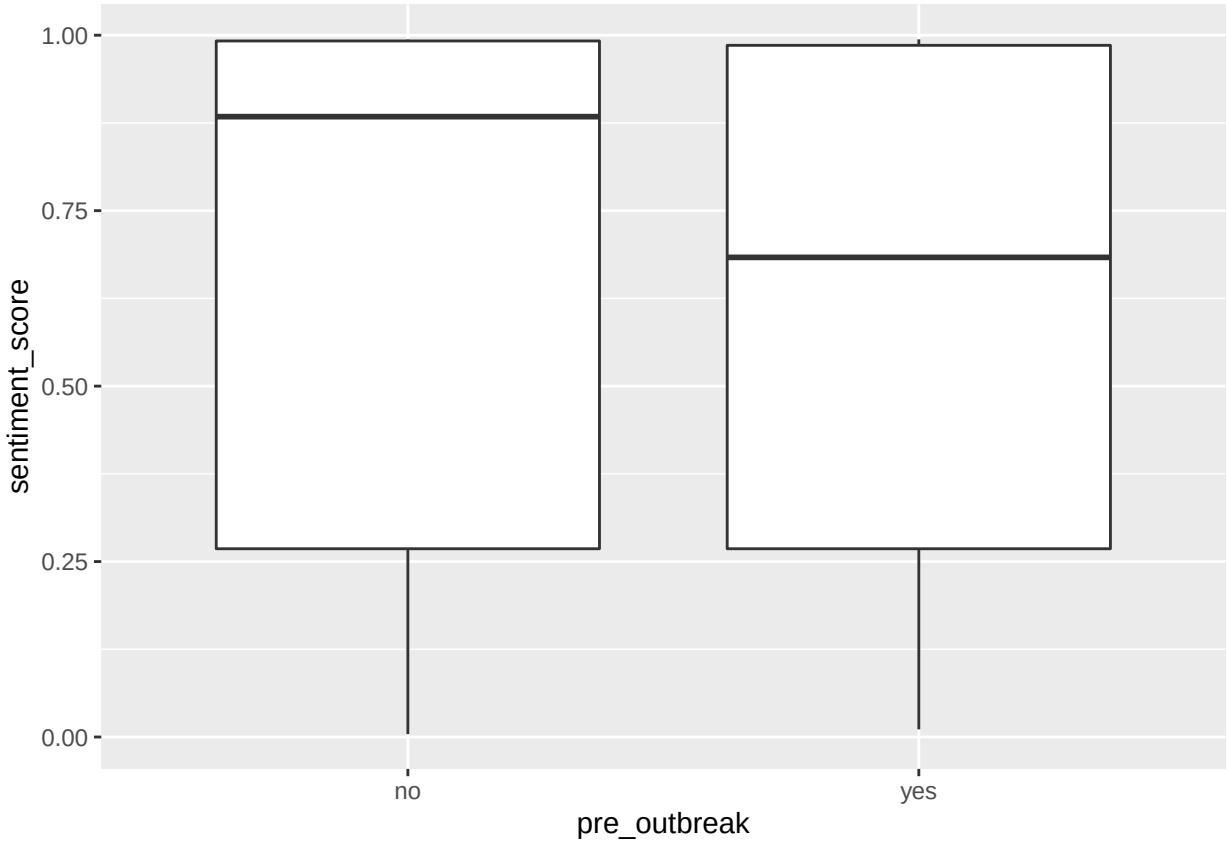
```
ggplot(UK_analysis_all) +
  geom_boxplot(aes(x = pre_covid, y = sentiment_score))
```




```
UK_analysis_all %>%
  group_by(pre_covid) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_covid `mean(sentiment_score)`
##   <chr>                       <dbl>
## 1 no                          0.607
## 2 yes                         0.656
```

```
ggplot(UK_analysis_all) +
  geom_boxplot(aes(x = pre_outbreak, y = sentiment_score))
```




```
UK_analysis_all %>%
  group_by(pre_outbreak) %>%
  summarize(mean(sentiment_score))
```

```
## # A tibble: 2 x 2
##   pre_outbreak `mean(sentiment_score)`
##   <chr>                          <dbl>
## 1 no                             0.630
## 2 yes                            0.629
```

```
#precovid all categories
count(UK_analysis_all, pre_covid == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_covid == "yes"`     n
##   <lgl>                <int>
## 1 FALSE                   70
## 2 TRUE                    60
```

```
num_precovid = 60
num_postcovid = 70
num = 130

UK_analysis_all %>%
  filter(pre_covid == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
```

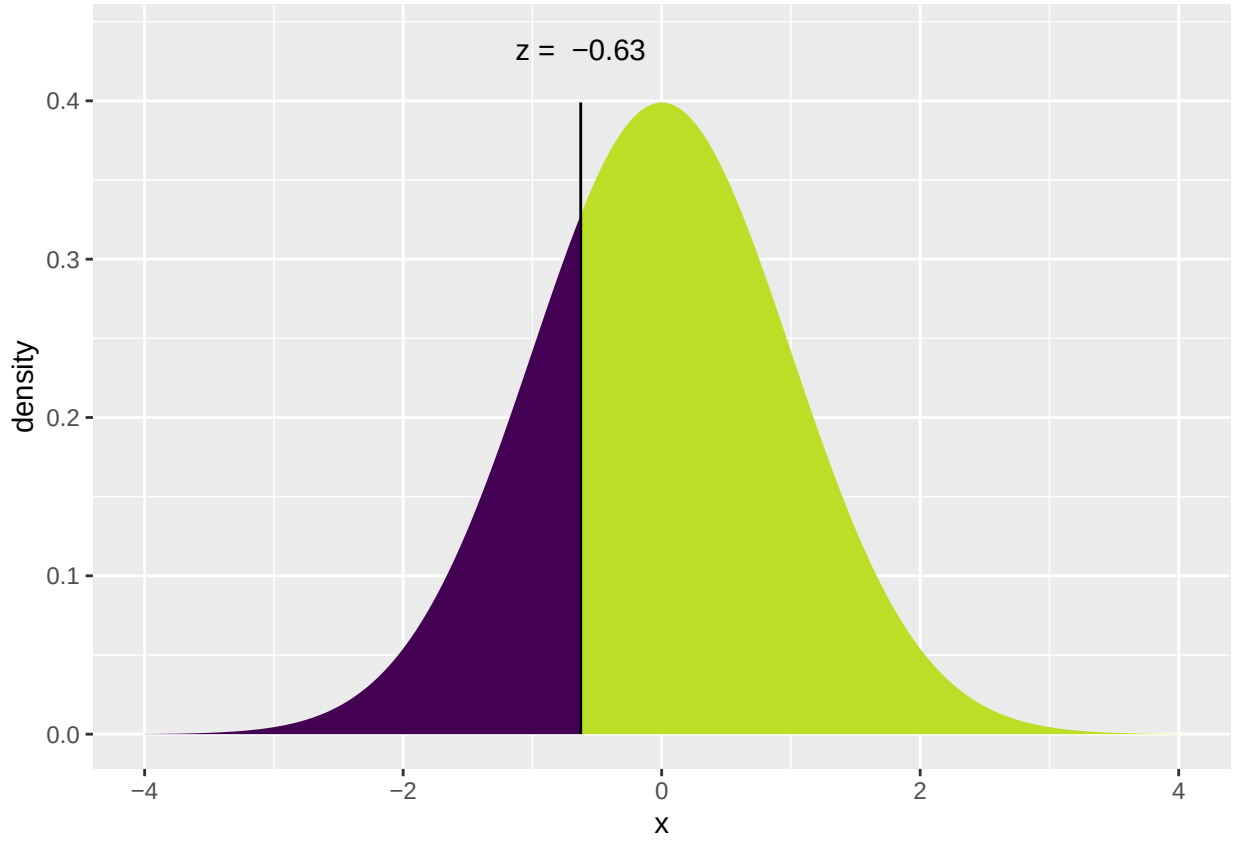
```
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      25
## 2 TRUE                       35
```

```
#proportion of positive sentiment videos precovid from sample
p_hat1 = 35/60

UK_analysis_all %>%
  filter(pre_covid == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      33
## 2 TRUE                       37
```

```
#proportion of positive sentiment videos postcovid from sample
p_hat2 = 37/70

p_hat = (35+37)/(60+70)

sd <- sqrt((((p_hat)*(1-p_hat))/60)+(((p_hat)*(1-p_hat))/70))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -0.6262) = P(Z <= -0.6262) = 0.2656
##  P(X >  -0.6262) = P(Z >  -0.6262) = 0.7344
##
```




```
## [1] 0.5312012
```

```
#outbreak all categories
count(UK_analysis_all, pre_outbreak == "yes")
```

```
## # A tibble: 2 x 2
##   `pre_outbreak == "yes"`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       90
```

```
num_preoutbreak = 90
num_postoutbreak = 40
num = 130

UK_analysis_all %>%
  filter(pre_outbreak == "yes") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      40
## 2 TRUE                       50
```

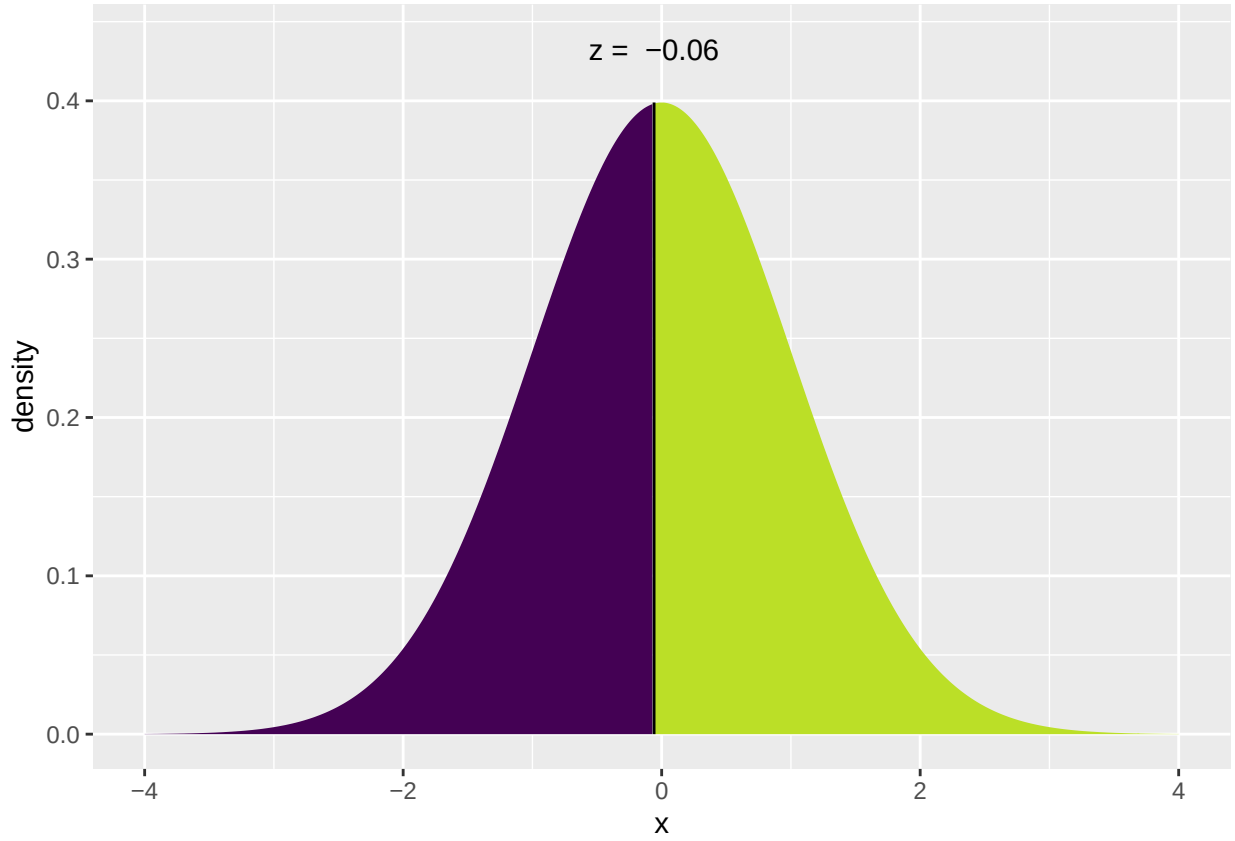
```
#proportion of positive sentiment videos preoutbreak from sample
p_hat1 = 50/90
```

```r
UK_analysis_all %>%
  filter(pre_outbreak == "no") %>%
  count(sentiment_score > 0.5)
```

```
## # A tibble: 2 x 2
##   `sentiment_score > 0.5`     n
##   <lgl>                   <int>
## 1 FALSE                      18
## 2 TRUE                       22
```

```r
#proportion of positive sentiment videos postoutbreak from sample
p_hat2 = 22/40

p_hat = (50+22)/(90+40)

sd <- sqrt((((p_hat)*(1-p_hat))/90)+(((p_hat)*(1-p_hat))/40))
z_score <- ((p_hat2-p_hat1)-0)/sd

#p-value
2* (xpnorm(z_score, 0, 1))
```

```
##
## If X ~ N(0, 1), then
##  P(X <= -0.05881) = P(Z <= -0.05881) = 0.4766
##  P(X >  -0.05881) = P(Z >  -0.05881) = 0.5234
##
```




```
## [1] 0.9531014
```

```
#Two independent samples t-tests; Comparing two independent means

#pre_covid music
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_music)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = -1.3513, df = 124.03, p-value = 0.1791
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2054222  0.0387348
## sample estimates:
##  mean in group no mean in group yes
##         0.4174173         0.5007610
```

```
#pre_outbreak music
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_music)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_outbreak
## t = -1.8243, df = 80.834, p-value = 0.0718
## alternative hypothesis: true difference in means is not equal to 0
```

```
## 95 percent confidence interval:
##  -0.24449090  0.01060667
## sample estimates:
##  mean in group no mean in group yes
##         0.3749237         0.4918658
```

```
#pre_covid travel and events
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_travel)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = 0.35455, df = 121.07, p-value = 0.7235
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.09273624  0.13319842
## sample estimates:
##  mean in group no mean in group yes
##         0.7087728         0.6885417
```

```
#pre_outbreak travel and events
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_travel)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_outbreak
## t = 1.2463, df = 77.965, p-value = 0.2164
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.04458545  0.19385740
## sample estimates:
##  mean in group no mean in group yes
##         0.7511065         0.6764705
```

```
#pre_covid people and blogs
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_people)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = 0.72944, df = 125.74, p-value = 0.4671
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.07240725  0.15694222
## sample estimates:
##  mean in group no mean in group yes
##         0.6206119         0.5783444
```

```
#pre_outbreak people and blogs
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_people)
```

```
##
##  Welch Two Sample t-test
##
```

```
## data:  sentiment_score by pre_outbreak
## t = 0.28803, df = 79.529, p-value = 0.7741
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1046391  0.1400507
## sample estimates:
##  mean in group no mean in group yes
##         0.6134959         0.5957901
```

```
#pre_covid entertainment
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_entertainment)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = -3.2196, df = 119.56, p-value = 0.001653
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.28687516 -0.06839027
## sample estimates:
##  mean in group no mean in group yes
##         0.4561732         0.6338059
```

```
#pre_outbreak entertainment
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_entertainment)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_outbreak
## t = -2.4565, df = 87.877, p-value = 0.01599
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.25108396 -0.02651152
## sample estimates:
##  mean in group no mean in group yes
##         0.4416564         0.5804541
```

```
#pre_covid news and politics
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_news)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = 1.0093, df = 122.72, p-value = 0.3148
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.05914756  0.18220701
## sample estimates:
##  mean in group no mean in group yes
##         0.6561818         0.5946521
```

```
#pre_outbreak news and politics
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_news)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_outbreak
## t = 0.88301, df = 73.463, p-value = 0.3801
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.07344867  0.19032887
## sample estimates:
##  mean in group no mean in group yes
##         0.6682420         0.6098019
```

```
#pre_covid how-to and style
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_how_to)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = -1.1205, df = 123.64, p-value = 0.2647
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.18050624  0.05001265
## sample estimates:
##  mean in group no mean in group yes
##         0.5637391         0.6289859
```

```
#pre_outbreak how-to and style
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_how_to)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_outbreak
## t = -2.3773, df = 86.122, p-value = 0.01966
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.25493140 -0.02273949
## sample estimates:
##  mean in group no mean in group yes
##         0.4977362         0.6365717
```

```
#pre_covid education
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_education)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = -0.34695, df = 126.6, p-value = 0.7292
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.10706944  0.07512572
## sample estimates:
##  mean in group no mean in group yes
##         0.7851922         0.8011640
```

```r
#pre_outbreak education
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_education)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_outbreak
## t = 0.72624, df = 69.988, p-value = 0.4701
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.06508533  0.13962780
## sample estimates:
##  mean in group no mean in group yes
##         0.8183670         0.7810957
```

```r
#pre_covid science and technology
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_science)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = -0.85681, df = 126.06, p-value = 0.3932
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.17326427  0.06856293
## sample estimates:
##  mean in group no mean in group yes
##         0.6666732         0.7190238
```

```r
#pre_outbreak science and technology
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_science)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_outbreak
## t = -0.59277, df = 75.859, p-value = 0.5551
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.17030291  0.09218303
## sample estimates:
##  mean in group no mean in group yes
##         0.6637935         0.7028535
```

```r
#pre_covid all categories
t.test(sentiment_score ~ pre_covid, alternative = "two.sided", data = UK_analysis_all)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_covid
## t = -0.80752, df = 127.92, p-value = 0.4209
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.17006484  0.07148561
```

```
## sample estimates:
##  mean in group no mean in group yes
##         0.6066743         0.6559639
```

```
#pre_outbreak categories
t.test(sentiment_score ~ pre_outbreak, alternative = "two.sided", data = UK_analysis_all)
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_score by pre_outbreak
## t = 0.005918, df = 65.24, p-value = 0.9953
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1413810  0.1422215
## sample estimates:
##  mean in group no mean in group yes
##         0.6297143         0.6292941
```