

# MSD 2019 Final Project

A replication and extension of Chilling Effects: Online Surveillance and Wikipedia Use by  
Jonathon W. Penney, Berkeley Technology Law Journal

*Thanaspakorn Niyomkarn, Alex Li Kong, and Sang Won Lee (tn2381, alk2225, and sl4447)*

*2019-05-11 23:15:25*

## Contents

<b>1. Introduction</b>	<b>1</b>
<b>2. Reproducing the Original Study</b>	<b>1</b>
2.1 Data . . . . .	2
2.2 Methodology . . . . .	2
2.3 Criticism . . . . .	2
2.4 Replication Results . . . . .	2
2.4.1 Total views of terrorism-related keywords before and after the incident . . . . .	2
2.4.2 Linear model with interactions: Analysis and Plots . . . . .	4
<b>3. Extended Analysis</b>	<b>7</b>
3.1 Longer Trend Analysis . . . . .	7
3.2 Per-keyword Analysis . . . . .	17
3.3 Time-series Analysis . . . . .	21
3.4 Trend Recovery . . . . .	21

## 1. Introduction

This Rmd file attempts to replicate and extend the results in Chilling Effects: Online Surveillance and Wikipedia Use by Jonathon W. Penney in Berkeley Technology Law Journal. The author is a research fellow at University of Toronto. This single author paper has H5-index of 21. This paper is about the NSA/PRISM surveillance 2007, where United States National Security Agency (NSA) started collecting Internet communications from various US Internet companies. This information was made public in 2013 by Edward Snowden revelations. This paper deals with the NSA paranoia where the paper studies traffic to Wikipedia articles on topics that raise privacy concerns for Wikipedia users before and after the Edward Snowden revelations. The Wikipedia traffic was chosen because over 50% of Internet users use Wikipedia as a source of information. Over 1/3 of Americans annually access Wikipedia as a source of information and is in top 10 of most popular sites on the internet.

## 2. Reproducing the Original Study

```
load("data/terrorism_data.RData")
load("data/infra_data.RData")
load("data/popular_data.RData")

load("data/terrorism_data_long.RData")
```

## 2.1 Data

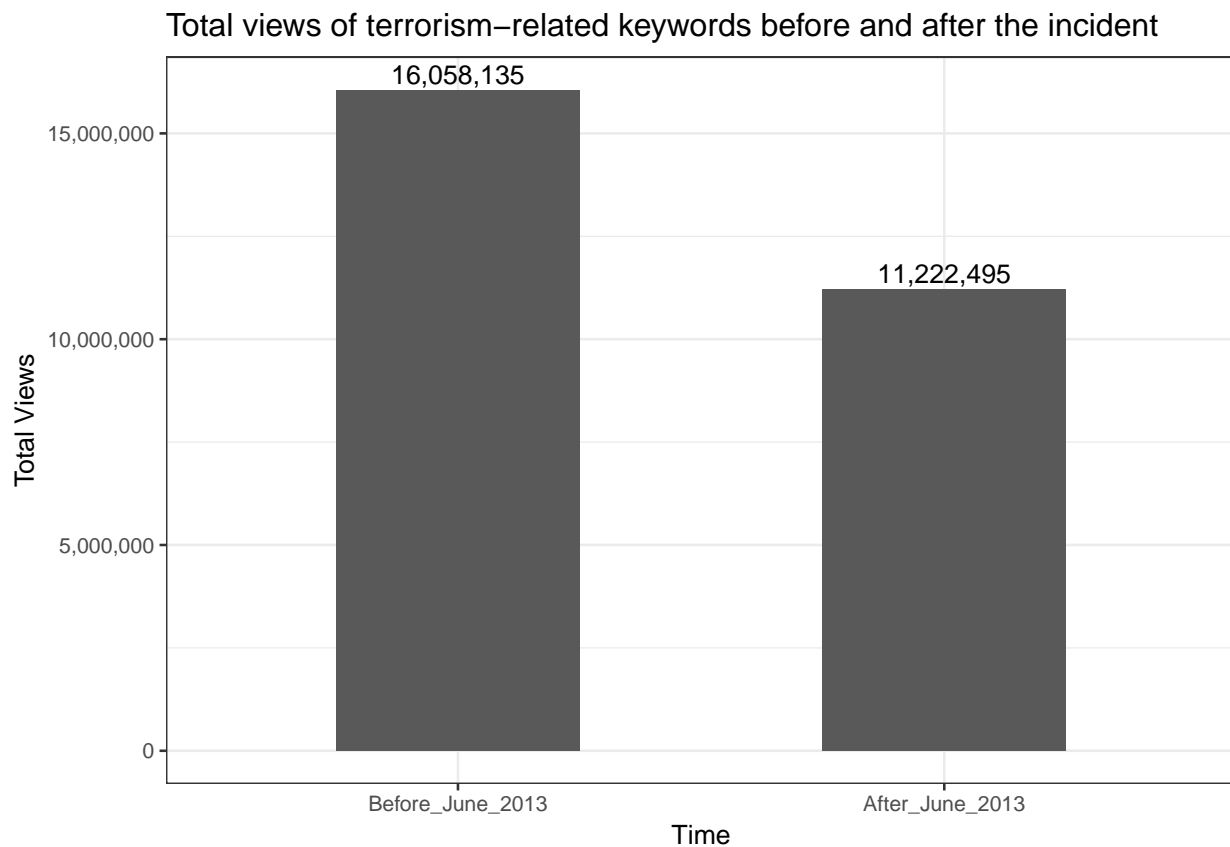
## 2.2 Methodology

## 2.3 Criticism

## 2.4 Replication Results

### 2.4.1 Total views of terrorism-related keywords before and after the incident

```
terrorism_data %>%  
  mutate(before_after = ifelse(date < '2013-06-01', "Before_June_2013", "After_June_2013")) %>%  
  group_by(before_after) %>%  
  summarise(total_views = sum(views)) %>%  
  ggplot(aes(x= factor(before_after, level = c("Before_June_2013", "After_June_2013")), y=total_views, v  
  scale_y_continuous(name="Total Views", labels = comma) +  
  xlab("Time") +  
  geom_text(aes(label=comma(total_views)), vjust=-0.3, color="black", size=3.5) +  
  theme_bw(base_size = 10) +  
  geom_bar(stat="identity") +  
  ggtitle("Total views of terrorism-related keywords before and after the incident")
```



### A. Terrorism Articles Study Group vs. Domestic Security Comparator Group

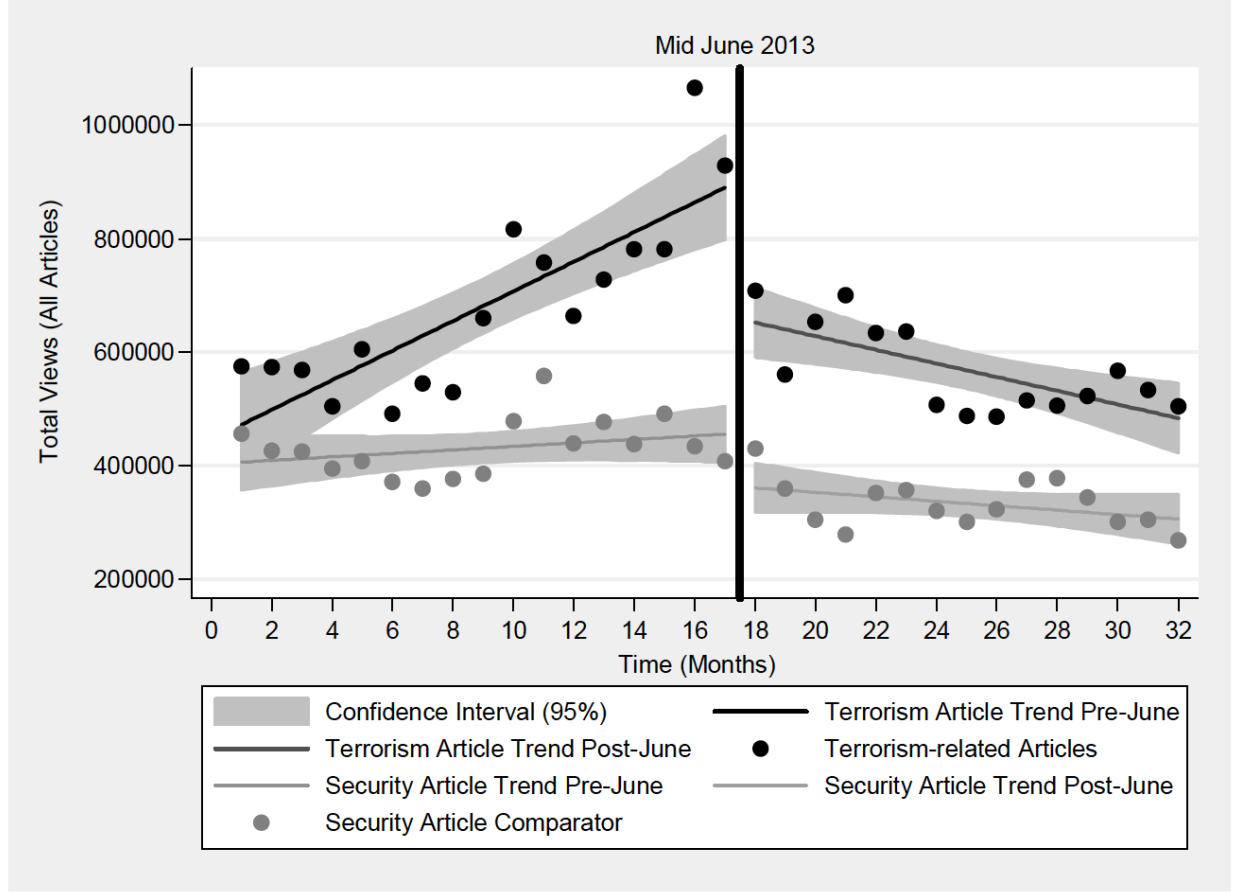


Figure 1: Original result for terrorism-related keywords

Table 2: Second Results, 47 Terrorism-related Articles ( Hamas Excluded)

Independent Variable	Coefficients	Standard Error	P-value
<b>Coefficient (<math>\beta_0</math>)</b> Expected Total Views at Beginning of Study	2289153**	109751.5	0.000
<b>Secular trend in data (<math>\beta_1</math>)</b> Change in Views (Monthly) Before 6/2013	41420.51**	10710.65	0.001
<b>Change in level (<math>\beta_2</math>)</b> Change in Views Immediately After 6/2013	-693616.9**	154640.9	0.000
<b>Change in slope (<math>\beta_3</math>)</b> Change in Views (Monthly) After 6/2013	-67513.1**	16789.25	0.000

\* $p < 0.05$ , \*\* $p < 0.01$

Figure 2: Original regression summary for terrorism-related keywords

## 2.4.2 Linear model with interactions: Analysis and Plots

```
lm_plot_topic <- function(input_df, gg_title){

df <- data.frame(input_df)
df <- df %>%
  group_by(month=floor_date(date, "month")) %>%
  summarize(views=sum(views))
df$surveillance <- 'before'
df$surveillance[df$month >= '2013-06-01'] <- 'after'

model <- lm(views ~ month + surveillance + month*surveillance, data = df)
print(summary(model))

df$prediction <- predict(model, df)
df$se <- predict(model, df,
                  se.fit = TRUE)$se.fit
z.val <- qnorm(1 - (1 - 0.90)/2)
df$LoCI <- df$prediction - z.val * df$se
df$HiCI <- df$prediction + z.val * df$se

df$month <- ymd(df$month)

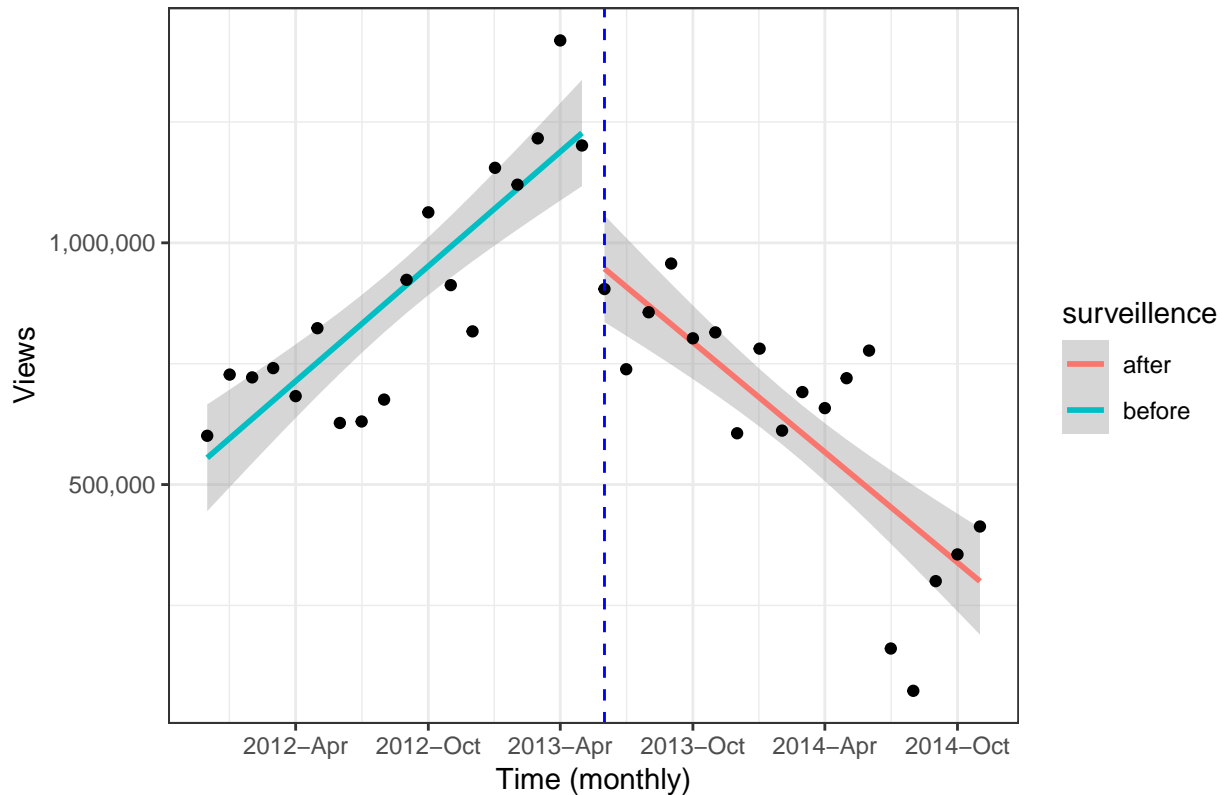
ggplot(df,
        aes(x = month,
            y = prediction)) +
  geom_smooth(aes(ymin = LoCI,
                  ymax = HiCI,
                  color = surveillance),
              stat = "identity") +
  geom_point(data = df, aes(x=month, y = views)) +
  geom_vline(xintercept = as.Date('2013-06-01'), linetype = 2, colour = 'blue') +
  ylab('Views') +
  xlab('Time (monthly)') +
  scale_x_date(date_breaks = "6 month", labels = date_format("%Y-%b")) +
  scale_y_continuous(labels = comma) +
  ggtitle(gg_title)
}

lm_plot_topic(terrorism_data, 'Terrorism-related keywords trend before and after June 2013')

##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -341385  -76768   13782   87116  286130
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.074e+07  3.568e+06   5.813 1.87e-06 ***
```

```
## month                -1.248e+03  2.214e+02  -5.638 3.10e-06 ***
## surveillancebefore    -4.008e+07  4.958e+06  -8.083 3.14e-09 ***
## month:surveillancebefore 2.548e+03  3.129e+02   8.142 2.68e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 148200 on 32 degrees of freedom
## Multiple R-squared:  0.7498, Adjusted R-squared:  0.7263
## F-statistic: 31.96 on 3 and 32 DF,  p-value: 9.546e-10
```

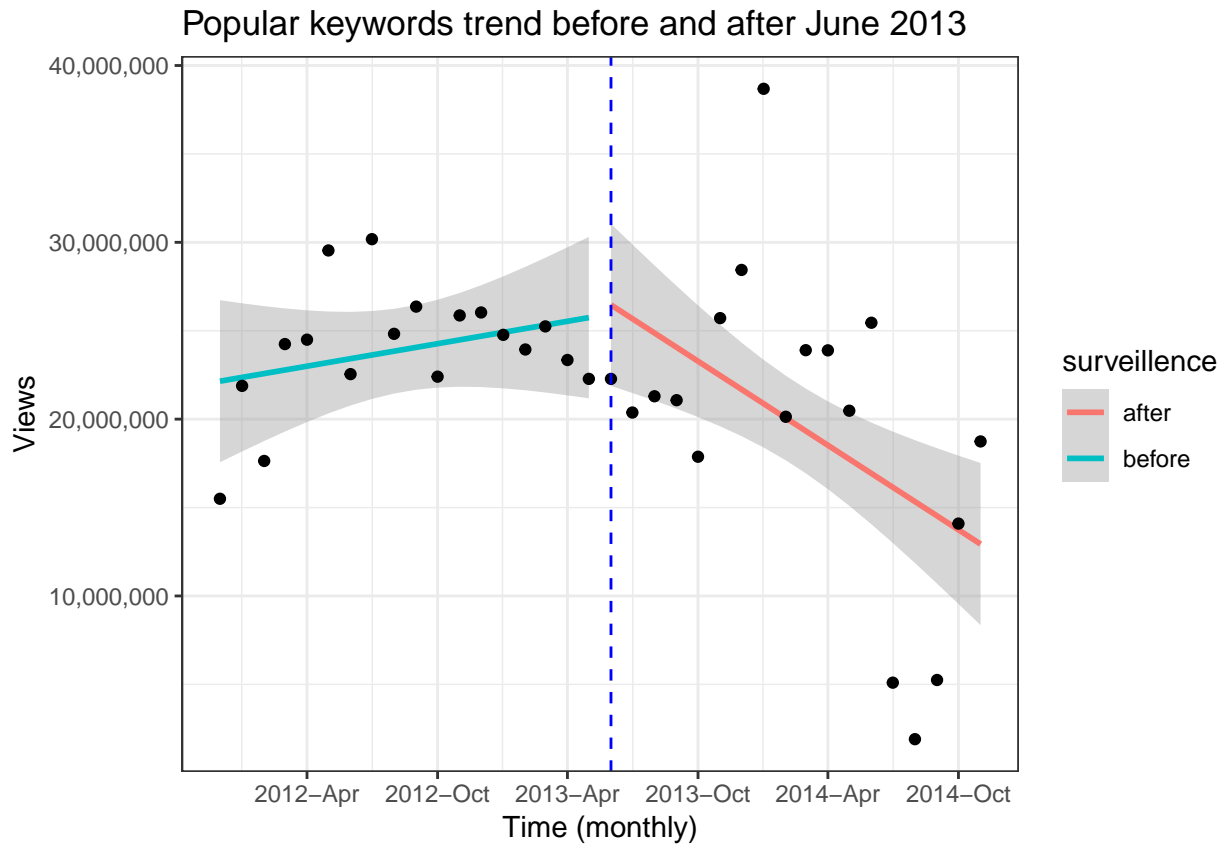
### Terrorism-related keywords trend before and after June 2013



```
lm_plot_topic(popular_data, 'Popular keywords trend before and after June 2013')
```

```
##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13436279  -3492306   -1164    2864260   17808433
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    440615805  148068036   2.976  0.00553 **
## month           -26118      9187   -2.843  0.00772 **
## surveillancebefore -524944327  205785253  -2.551  0.01573 *
## month:surveillancebefore  33073    12987   2.547  0.01589 *
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6152000 on 32 degrees of freedom
## Multiple R-squared:  0.288, Adjusted R-squared:  0.2212
## F-statistic: 4.314 on 3 and 32 DF,  p-value: 0.01155
```

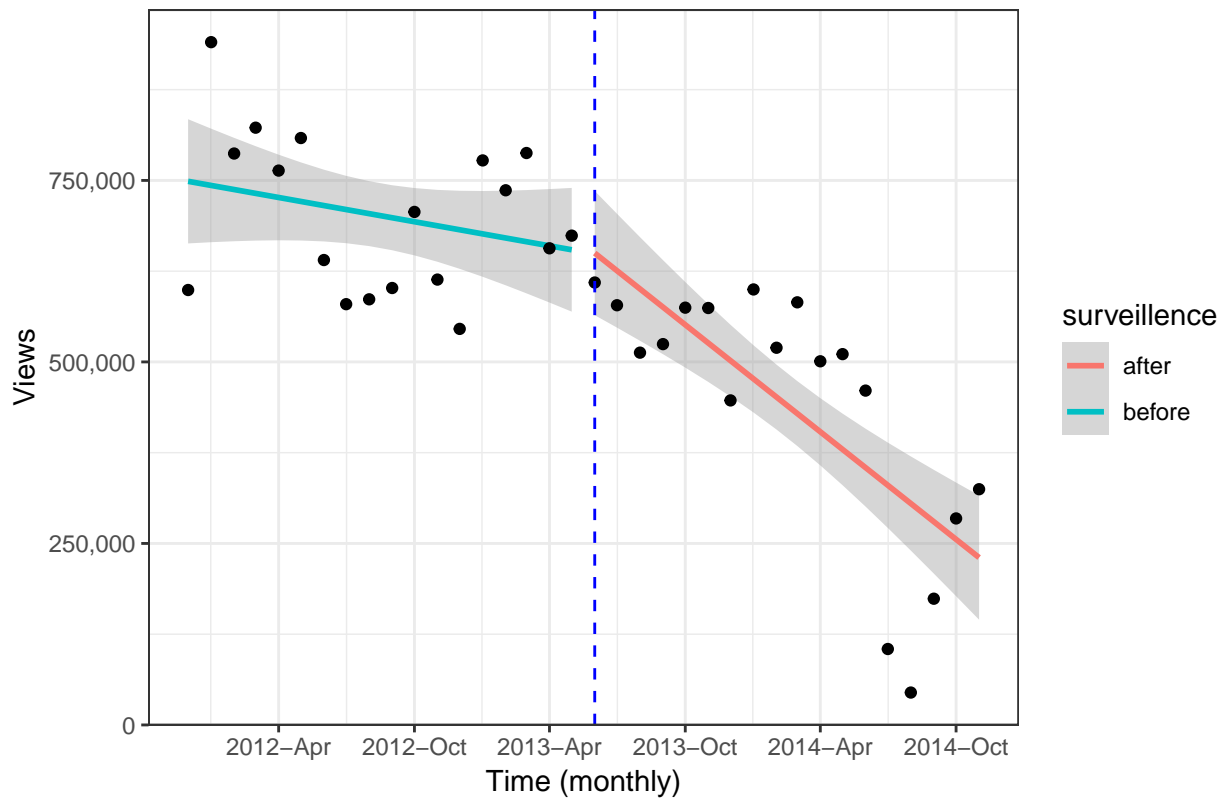


```
lm_plot_topic(infra_data, 'Infrastructure security-related keywords trend before and after June 2013')
```

```
##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -260393  -78202   21543   91386  197325
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    13488540.4   2763780.6    4.880 2.81e-05 ***
## month           -809.7       171.5   -4.721 4.46e-05 ***
## surveillancebefore -9948117.6  3841107.8   -2.590  0.0143 *
## month:surveillancebefore    627.3       242.4    2.588  0.0144 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 114800 on 32 degrees of freedom
## Multiple R-squared:  0.6862, Adjusted R-squared:  0.6567
## F-statistic: 23.32 on 3 and 32 DF,  p-value: 3.434e-08
```

### Infrastructure security-related keywords trend before and after June 2013



## 3. Extended Analysis

### 3.1 Longer Trend Analysis

The link to our data broke on Saturday May 11th, 2019. Our original plots consisted data from 2012 to 2019. Our group tried to reconstruct the graph from data that was saved on our laptops before the link broke, which is until 2015. Luckily we have original graphs saved as png file. After each reproduced graphs, we included what they used to look like before the link broke, and full data was available.

```
monthly_agg <- terrorism_data_long %>%
  group_by(month=floor_date(date, "month")) %>%
  summarize(views=sum(views))
monthly_agg$surveillance <- 'before'
monthly_agg$surveillance[monthly_agg$month >= '2013-06-01'] <- 'after'
model <- lm(views ~ month + surveillance + month*surveillance, data = monthly_agg)
summary(model)
```

```
##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = monthly_agg)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -376110  -77479   25547   74033  242163
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.829e+06  1.305e+06   3.701 0.000491 ***
## month          -2.708e+02  8.004e+01  -3.383 0.001316 **
## surveillancebefore -3.686e+06  1.793e+06  -2.056 0.044497 *
## month:surveillancebefore 2.417e+02  1.131e+02   2.137 0.036990 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 115400 on 56 degrees of freedom
## Multiple R-squared:  0.6412, Adjusted R-squared:  0.6219
## F-statistic: 33.35 on 3 and 56 DF,  p-value: 1.684e-12

monthly_agg$prediction <- predict(model, monthly_agg)
monthly_agg$se <- predict(model, monthly_agg,
                          se.fit = TRUE)$se.fit
z.val <- qnorm(1 - (1 - 0.90)/2)
monthly_agg$LoCI <- monthly_agg$prediction - z.val * monthly_agg$se
monthly_agg$HiCI <- monthly_agg$prediction + z.val * monthly_agg$se
monthly_agg$month <- ymd(monthly_agg$month)
ggplot(monthly_agg,
       aes(x = month,
           y = views)) +
  geom_point(data = monthly_agg, aes(x=month, y = views)) +
  geom_vline(xintercept = as.Date('2013-06-01'), linetype = 2, colour = 'blue') +
  geom_vline(xintercept = as.Date('2014-12-31'), linetype = 2, colour = 'blue') +
  ylab('Views') +
  xlab('Time (monthly)') +
  scale_x_date(date_breaks = "6 month", labels = date_format("%Y-%b")) +
  theme_bw(base_size = 5) +
  scale_y_continuous(labels = comma)
```



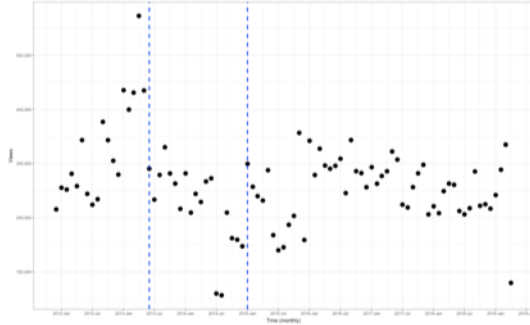
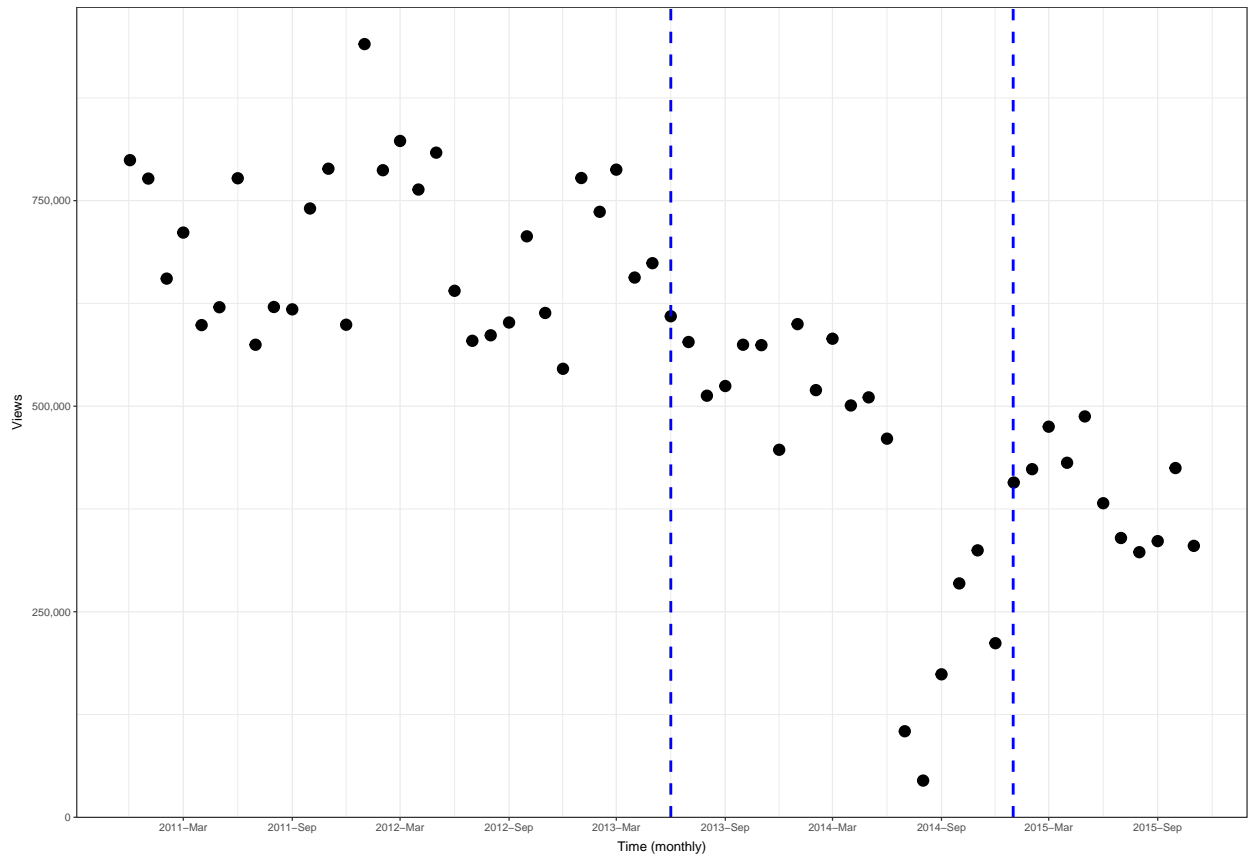


Figure 3: Original Scatter Plot



First, we take out all the line graphs, and plot scattered graph according to data. The original graph with full data is shown above.

```
monthly_agg <- terrorism_data_long %>%
  group_by(month=floor_date(date, "month")) %>%
  summarize(views=sum(views))
monthly_agg$surveillance <- 'before'
monthly_agg$surveillance[monthly_agg$month >= '2013-06-01'] <- 'after'

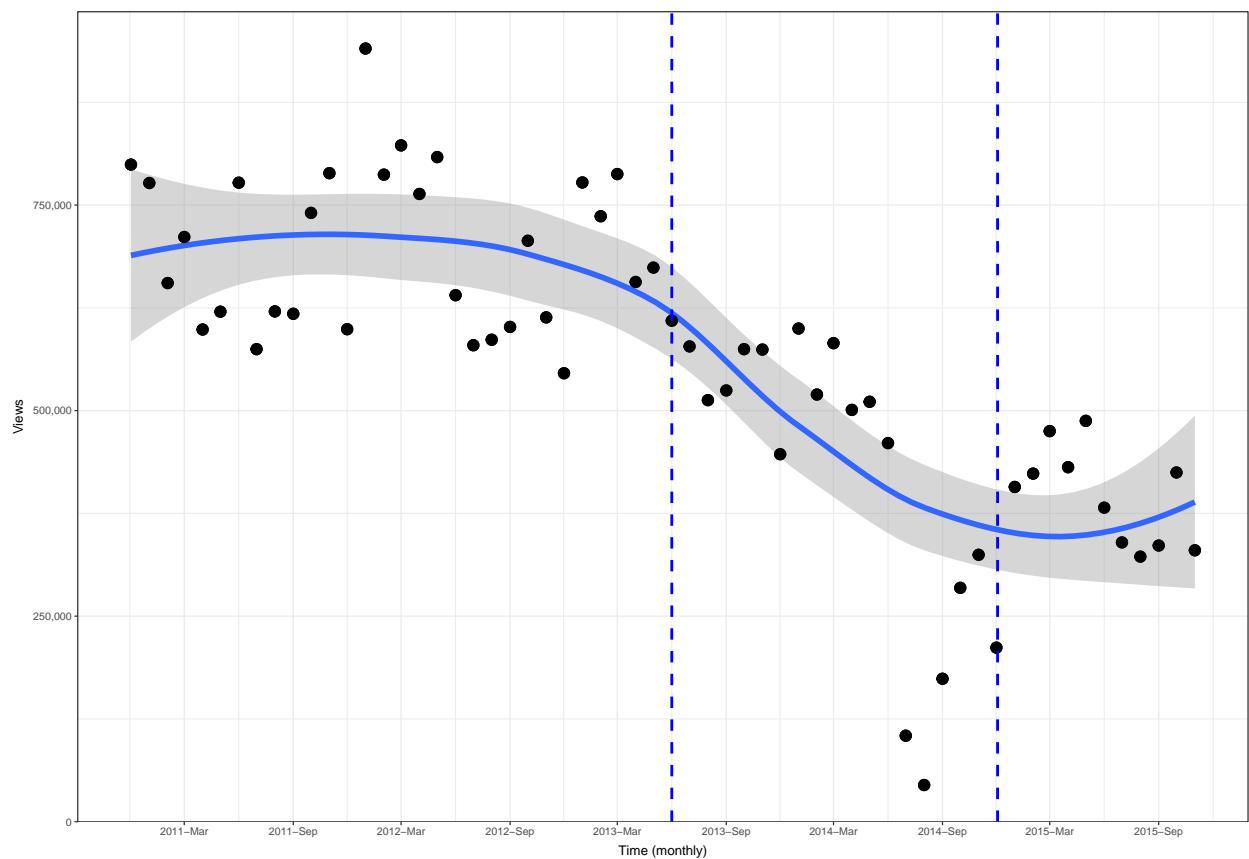
model <- lm(views ~ month + surveillance + month*surveillance, data = monthly_agg)
monthly_agg$prediction <- predict(model, monthly_agg)
monthly_agg$se <- predict(model, monthly_agg,
  se.fit = TRUE)$se.fit
```

```

z.val <- qnorm(1 - (1 - 0.90)/2)
monthly_agg$LoCI <- monthly_agg$prediction - z.val * monthly_agg$se
monthly_agg$HiCI <- monthly_agg$prediction + z.val * monthly_agg$se
monthly_agg$month <- ymd(monthly_agg$month)
ggplot(monthly_agg,
  aes(x = month,
      y = views)) + geom_point()+stat_smooth( se=T)+
  geom_point(data = monthly_agg, aes(x=month, y = views)) +
  geom_vline(xintercept = as.Date('2013-06-01'), linetype = 2, colour = 'blue') +
  geom_vline(xintercept = as.Date('2014-12-3'), linetype = 2, colour = 'blue') +
  ylab('Views') +
  xlab('Time (monthly)') +
  scale_x_date(date_breaks = "6 month", labels = date_format("%Y-%b")) +
  theme_bw(base_size = 5) +
  scale_y_continuous(labels = comma)

```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Then, our group had fit a polynomial surface determined by one or more numerical predictors, using local fitting. The graph also displays confidence interval around as gray. The graph below shows rise in trend from 2015 January to July 2016. The graph trend again drops from July 2016 to Jan 2019. The views counts are similar at the beginning of 2016 and end of plot at 2019. This shows that there has been a “trend reovery,” but the trend again drops without second “Snowden Revelation.” This may mean that the decrease in trend from 2013 to 2015 may be due to other factors rather than due to NSA paranoia.

The original graph with full data is shown below.

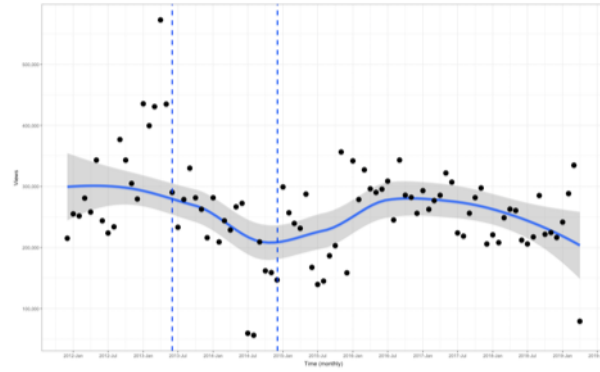


Figure 4: Original\_Stat\_Smooth\_Curve

```
monthly_agg <- terrorism_data_long %>%
  group_by(month=floor_date(date, "month")) %>%
  summarize(views=sum(views))
monthly_agg$surveillance <- 'before'
monthly_agg$surveillance[monthly_agg$month >= '2013-06-01'] <- 'after'
model <- lm(views ~ month + surveillance + month*surveillance, data = monthly_agg)
monthly_agg$prediction <- predict(model, monthly_agg)
monthly_agg$se <- predict(model, monthly_agg,
  se.fit = TRUE)$se.fit
z.val <- qnorm(1 - (1 - 0.90)/2)
monthly_agg$LoCI <- monthly_agg$prediction - z.val * monthly_agg$se
monthly_agg$HiCI <- monthly_agg$prediction + z.val * monthly_agg$se
monthly_agg$month <- ymd(monthly_agg$month)
monthly_agg$month <- as.numeric(monthly_agg$month)
fit <- lm(views~poly(month,4,row=TRUE),monthly_agg)
plot(views~month,monthly_agg)
curve(predict(fit,newdata=data.frame(month=x)),add=T)
```

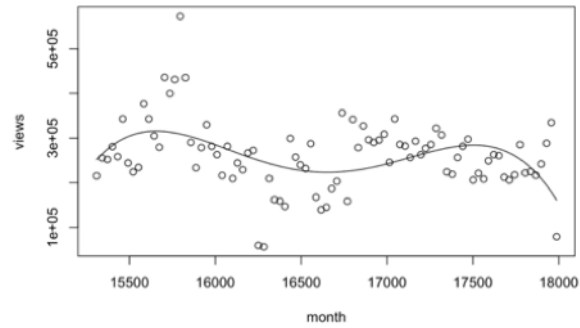
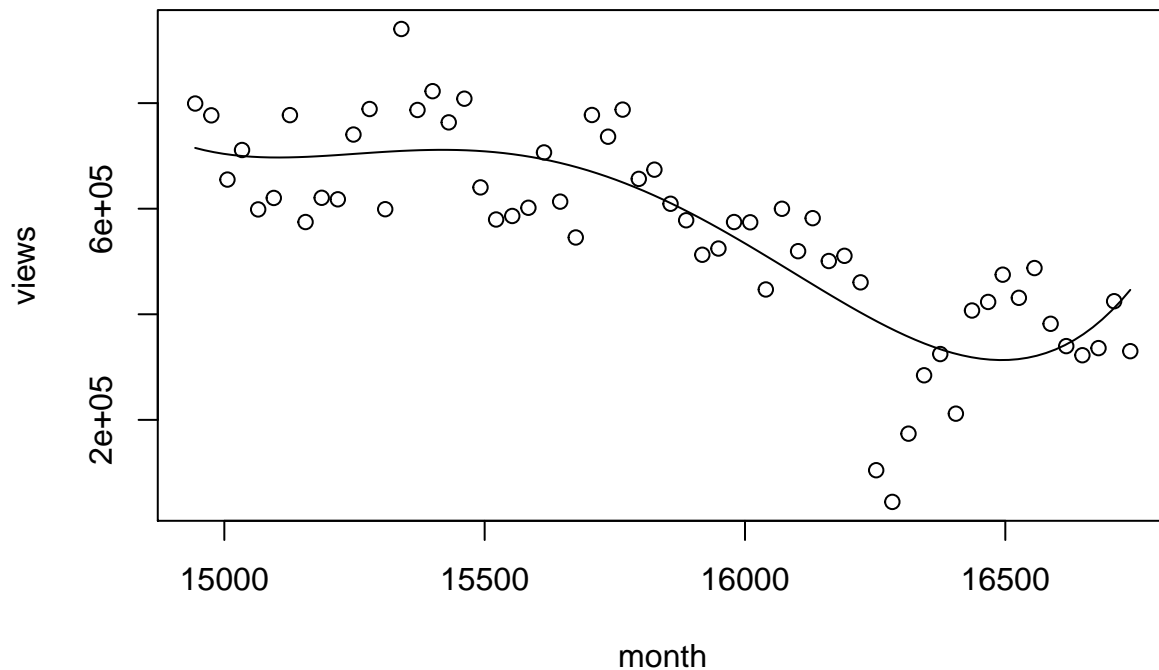


Figure 5: Polynomial\_crve\_n\_4



Then, our group had went further and fit a polynomial graph of degree 4. The curve above shows that there is sharper decrease in trend at the end of graph, far from 2013 region. This again backs our claim that there is a decrease in 2018 to 2019 without another “Snowden Revelation.” This means the decrease in trend after 2013 can be attributed to another resason than Snowden Revelation.

```
monthly_agg <- terrorism_data_long %>%
  group_by(month=floor_date(date, "month")) %>%
  summarize(views=sum(views))
monthly_agg$surveillance <- 'before'
monthly_agg$surveillance[monthly_agg$month >= '2013-06-01'] <- 'after'

model <- lm(views ~ month + surveillance + month*surveillance, data = monthly_agg)
monthly_agg$prediction <- predict(model, monthly_agg)
monthly_agg$se <- predict(model, monthly_agg,
  se.fit = TRUE)$se.fit
z.val <- qnorm(1 - (1 - 0.90)/2)
monthly_agg$LoCI <- monthly_agg$prediction - z.val * monthly_agg$se
monthly_agg$HiCI <- monthly_agg$prediction + z.val * monthly_agg$se
```

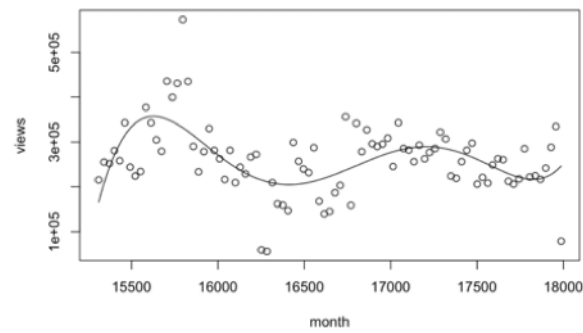
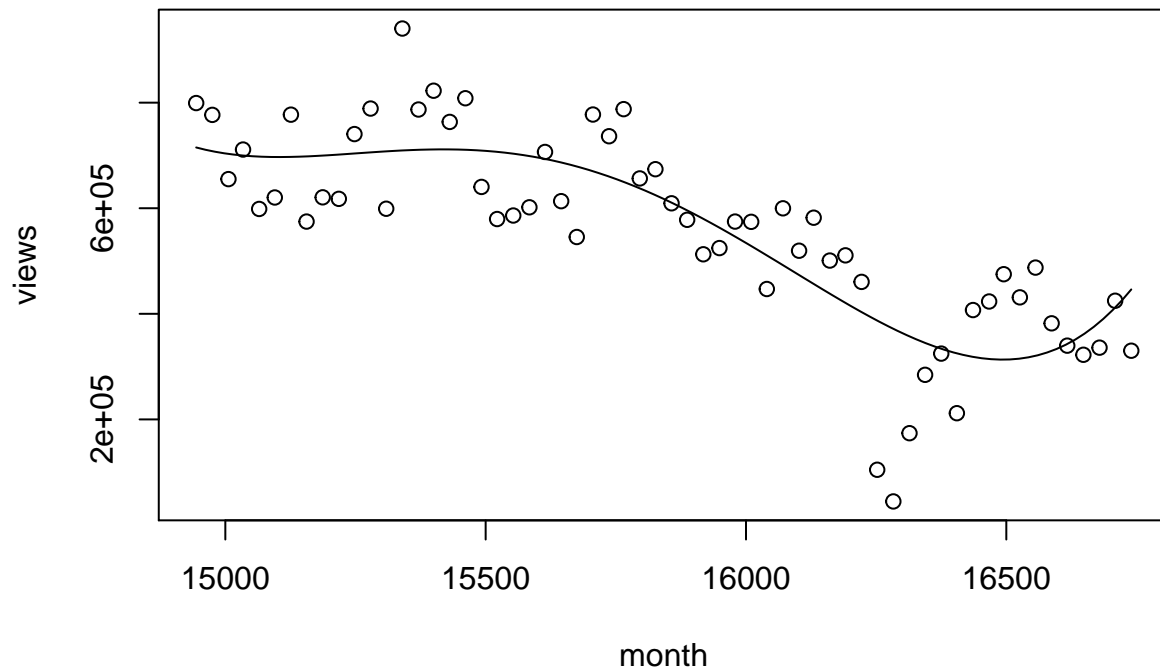


Figure 6: Polynomial\_crve\_n\_5

```
monthly_agg$month <- ymd(monthly_agg$month)
monthly_agg$month <- as.numeric(monthly_agg$month)
fit <- lm(views~poly(month,5,row=TRUE),monthly_agg)
plot(views~month,monthly_agg)
curve(predict(fit,newdata=data.frame(month=x)),add=T)
```

```
## Warning in predict.lm(fit, newdata = data.frame(month = x)): prediction
## from a rank-deficient fit may be misleading
```



Then, our group had went further and fit a polynomial graph of degree 5 instead of 4. This may be overfitting, but wanted to analyze the graph to full extent. This graph also shows that there has been decrease in trend from 2018 to 2019, but end of the trend is actually increasing. This fluctuating trend shows that author of original paper's claim that chilling effect is not caused due to NSA paranoia.

```
monthly_agg <- terrorism_data_long %>%
  group_by(month=floor_date(date, "month")) %>%
  summarize(views=sum(views))
monthly_agg$surveillance <- 'before'
```

```

monthly_agg$surveillance[monthly_agg$month >= '2013-06-01'] <- 'after'
monthly_agg$surveillance[monthly_agg$month >= '2014-12-3'] <- 'after_after'
model <- lm(views ~ month + surveillance + month*surveillance, data = monthly_agg)
summary(model)

##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = monthly_agg)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -261116  -60090   -3068    74682   242163
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.344e+07  2.212e+06   6.078 1.28e-07 ***
## month          -8.068e+02  1.371e+02  -5.884 2.61e-07 ***
## surveillanceafter_after -7.257e+06  5.632e+06  -1.289   0.203
## surveillancebefore    -1.230e+07  2.453e+06  -5.014 6.11e-06 ***
## month:surveillanceafter_after  4.577e+02  3.410e+02   1.342   0.185
## month:surveillancebefore    7.778e+02  1.535e+02   5.068 5.04e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 99590 on 54 degrees of freedom
## Multiple R-squared:  0.7424, Adjusted R-squared:  0.7186
## F-statistic: 31.13 on 5 and 54 DF,  p-value: 9.146e-15

monthly_agg$prediction <- predict(model, monthly_agg)
monthly_agg$se <- predict(model, monthly_agg,
                          se.fit = TRUE)$se.fit
z.val <- qnorm(1 - (1 - 0.90)/2)
monthly_agg$LoCI <- monthly_agg$prediction - z.val * monthly_agg$se
monthly_agg$HiCI <- monthly_agg$prediction + z.val * monthly_agg$se
monthly_agg$month <- ymd(monthly_agg$month)
ggplot(monthly_agg,
       aes(x = month,
           y = prediction)) +
  geom_smooth(aes(ymin = LoCI,
                 ymax = HiCI,
                 color = surveillance),
             stat = "identity") +
  geom_point(data = monthly_agg, aes(x=month, y = views)) +
  geom_vline(xintercept = as.Date('2013-06-01'), linetype = 2, colour = 'blue') +
  geom_vline(xintercept = as.Date('2014-12-3'), linetype = 2, colour = 'blue') +
  ylab('Views') +
  xlab('Time (monthly)') +
  scale_x_date(date_breaks = "6 month", labels = date_format("%Y-%b")) +
  theme_bw(base_size = 5) +
  scale_y_continuous(labels = comma)

```

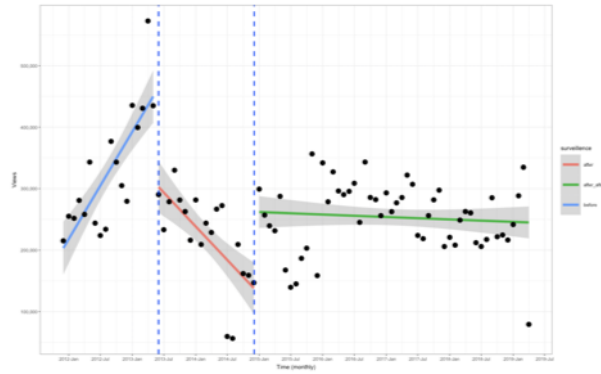
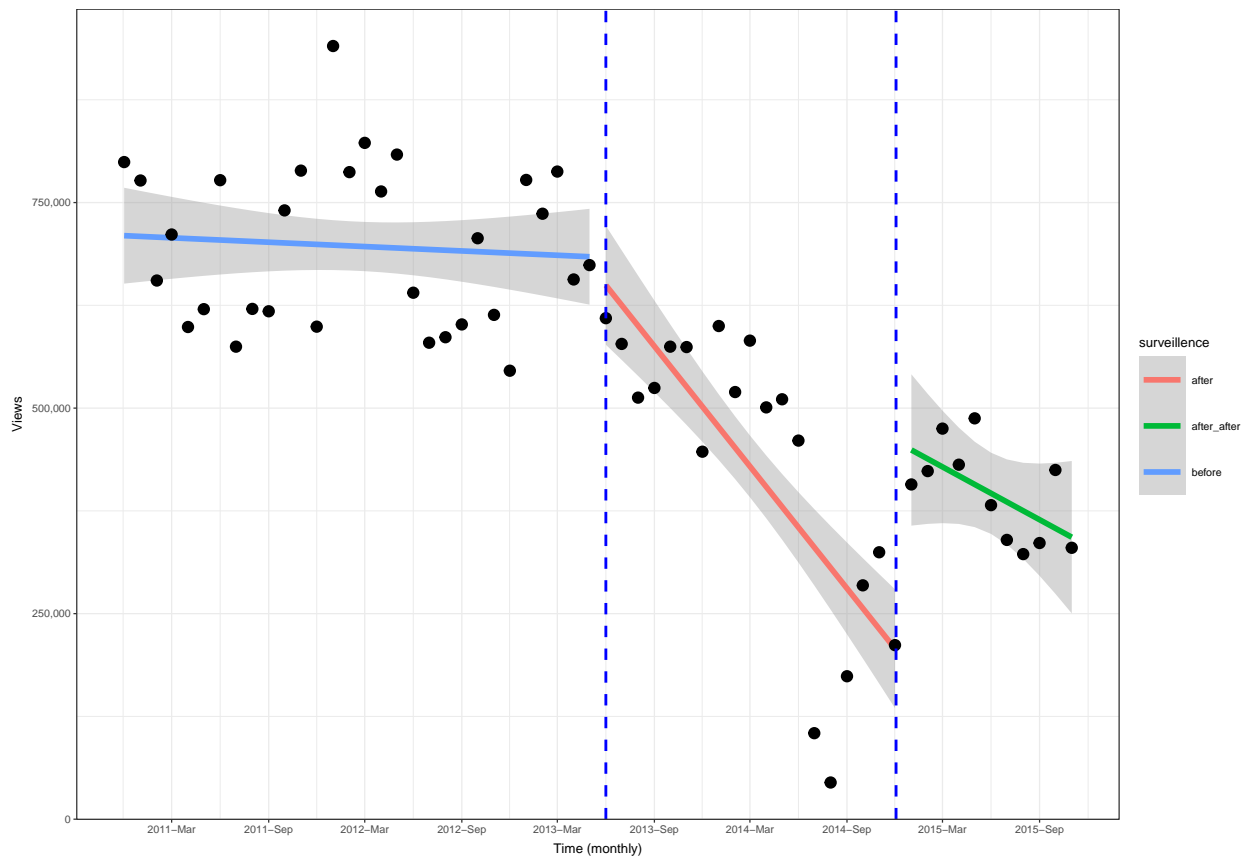


Figure 7: After



The graph below shows that there is stable trend after January 2015. This graph shows that view after 2015 is higher than from 2013 to 2015. This is only graph that may support author's claim that there had been a trend recovery, and trend stays constant. However, the data is not segmented in equal time bins. This made our group explore further in equal time segments of trends after 2015, in the following works below.

```
monthly_agg <- terrorism_data_long %>%
  group_by(month=floor_date(date, "month")) %>%
  summarize(views=sum(views))
monthly_agg$surveillance <- 'before'
monthly_agg$surveillance[monthly_agg$month >= '2013-06-01'] <- 'after'
monthly_agg$surveillance[monthly_agg$month >= '2014-12-3'] <- 'after_after'
```

```
monthly_agg$surveillance[monthly_agg$month >= '2016-06-3'] <- 'after_after_after'
monthly_agg$surveillance[monthly_agg$month >= '2016-06-3'] <- 'after_after_after_after'
model <- lm.views ~ month + surveillance + month*surveillance, data = monthly_agg)
summary(model)
```

```
##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = monthly_agg)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -261116  -60090   -3068    74682   242163
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.344e+07  2.212e+06   6.078 1.28e-07 ***
## month          -8.068e+02  1.371e+02  -5.884 2.61e-07 ***
## surveillanceafter_after -7.257e+06  5.632e+06  -1.289   0.203
## surveillancebefore    -1.230e+07  2.453e+06  -5.014 6.11e-06 ***
## month:surveillanceafter_after  4.577e+02  3.410e+02   1.342   0.185
## month:surveillancebefore    7.778e+02  1.535e+02   5.068 5.04e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 99590 on 54 degrees of freedom
## Multiple R-squared:  0.7424, Adjusted R-squared:  0.7186
## F-statistic: 31.13 on 5 and 54 DF,  p-value: 9.146e-15
```

```
monthly_agg$prediction <- predict(model, monthly_agg)
monthly_agg$se <- predict(model, monthly_agg,
                          se.fit = TRUE)$se.fit
z.val <- qnorm(1 - (1 - 0.90)/2)
monthly_agg$LoCI <- monthly_agg$prediction - z.val * monthly_agg$se
monthly_agg$HiCI <- monthly_agg$prediction + z.val * monthly_agg$se
monthly_agg$month <- ymd(monthly_agg$month)
ggplot(monthly_agg,
       aes(x = month,
           y = prediction)) +
  geom_smooth(aes(ymin = LoCI,
                 ymax = HiCI,
                 color = surveillance),
             stat = "identity") +
  geom_point(data = monthly_agg, aes(x=month, y = views)) +
  geom_vline(xintercept = as.Date('2013-06-01'), linetype = 2, colour = 'blue') +
  geom_vline(xintercept = as.Date('2014-12-3'), linetype = 2, colour = 'blue') +
  geom_vline(xintercept = as.Date('2016-06-3'), linetype = 2, colour = 'blue') +
  geom_vline(xintercept = as.Date('2016-06-3'), linetype = 2, colour = 'blue') +
  ylab('Views') +
  xlab('Time (monthly)') +
  scale_x_date(date_breaks = "6 month", labels = date_format("%Y-%b")) +
  theme_bw(base_size = 5) +
  scale_y_continuous(labels = comma)
```



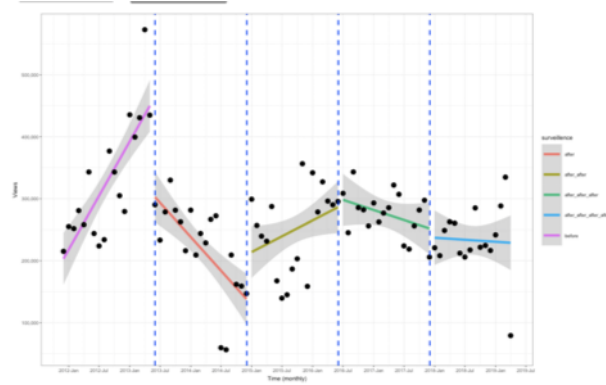
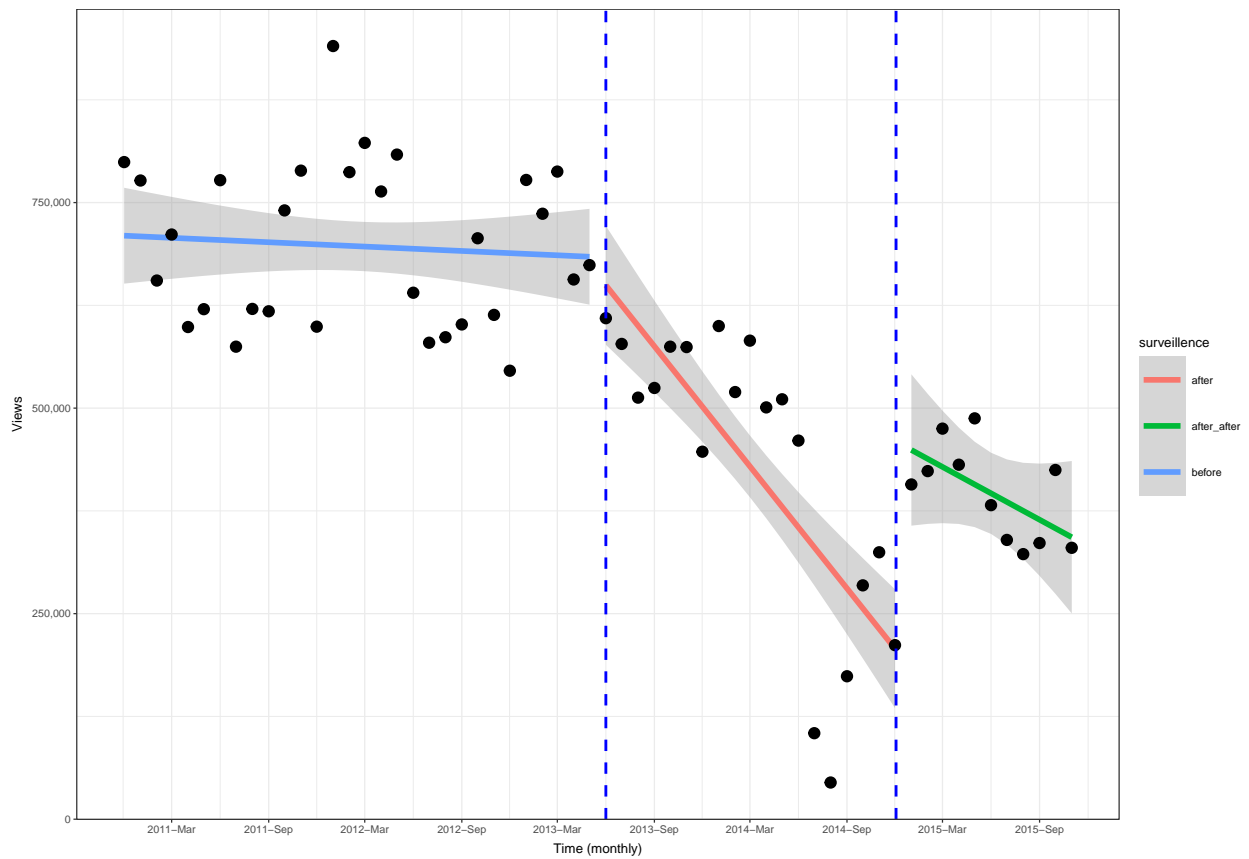


Figure 8: After\_broken



Now, our group had separated the data into equal segments. The graph above shows trend recovery from Jan 2015 to July 2016, trend decrease from July 2016 to Jan 2018, then a stable trend from Jan 2018 to July 2019. This fluctuations may again raise question to paper's author's claim that there exists chilling effect due to Snowden Revelation.

### 3.2 Per-keyword Analysis

```
lm_plot_keyword <- function(input_df, article_name, gg_title){
df <- data.frame(input_df)
```

```

df <- df %>%
  group_by(article ,month=floor_date(date, "month")) %>%
  summarize(views=sum(views)) %>%
  filter(article == article_name)

df$surveillance <- 'before'
df$surveillance[df$month >= '2013-06-01'] <- 'after'

model <- lm(views ~ month + surveillance + month*surveillance, data = df)
print(summary(model))

df$prediction <- predict(model, df)
df$se <- predict(model, df,
                  se.fit = TRUE)$se.fit
z.val <- qnorm(1 - (1 - 0.90)/2)
df$LoCI <- df$prediction - z.val * df$se
df$HiCI <- df$prediction + z.val * df$se

df$month <- ymd(df$month)

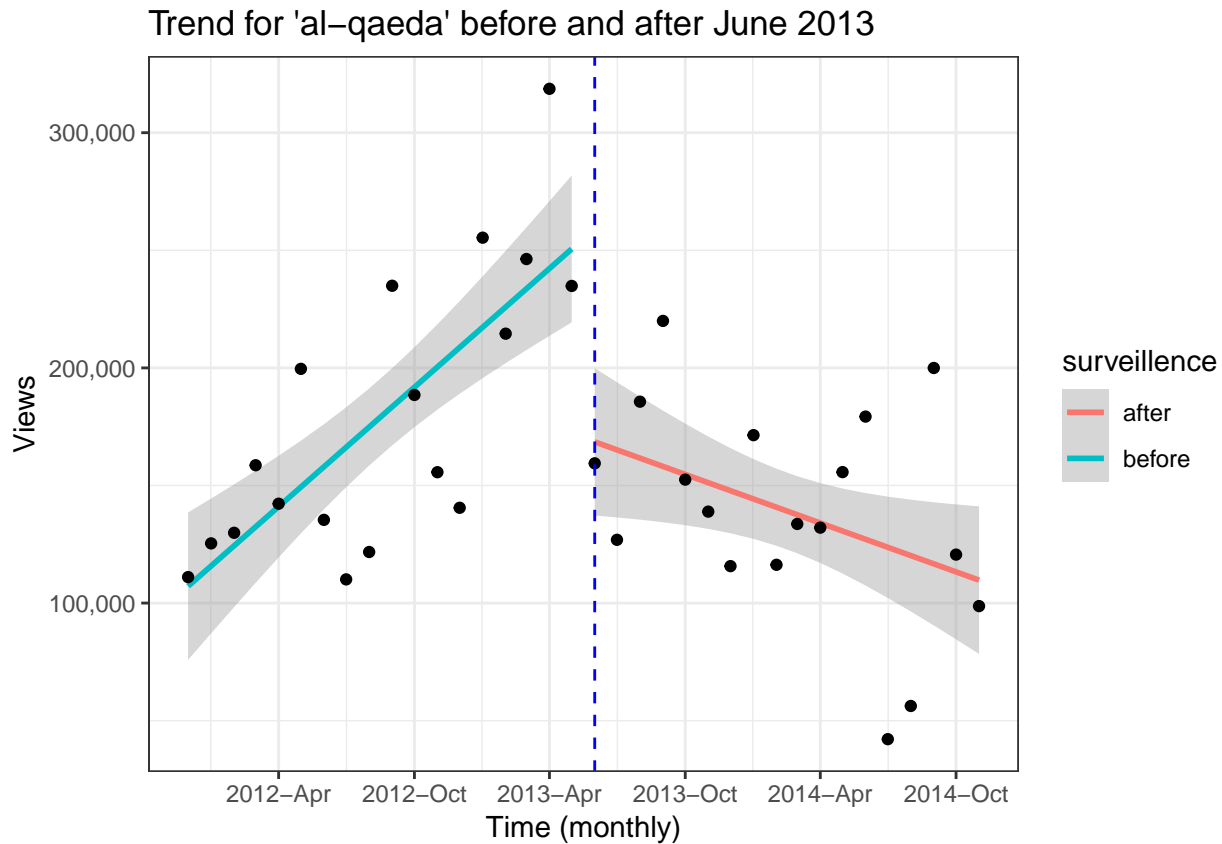
ggplot(df,
  aes(x = month,
      y = prediction)) +
  geom_smooth(aes(ymin = LoCI,
                  ymax = HiCI,
                  color = surveillance),
              stat = "identity") +
  geom_point(data = df, aes(x=month, y = views)) +
  geom_vline(xintercept = as.Date('2013-06-01'), linetype = 2, colour = 'blue') +
  ylab('Views') +
  xlab('Time (monthly)') +
  scale_x_date(date_breaks = "6 month", labels = date_format("%Y-%b")) +
  scale_y_continuous(labels = comma) +
  ggtitle(gg_title)
}

lm_plot_keyword(terrorism_data, 'al-qaeda', 'Trend for \'al-qaeda\' before and after June 2013')

##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -81582 -23037  -2093   25332   83296
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.969e+06  1.013e+06   1.943 0.060805 .
## month          -1.135e+02  6.285e+01  -1.806 0.080314 .
## surveillancebefore -6.107e+06  1.408e+06  -4.338 0.000134 ***
## month:surveillancebefore  3.909e+02  8.884e+01   4.400 0.000113 ***

```

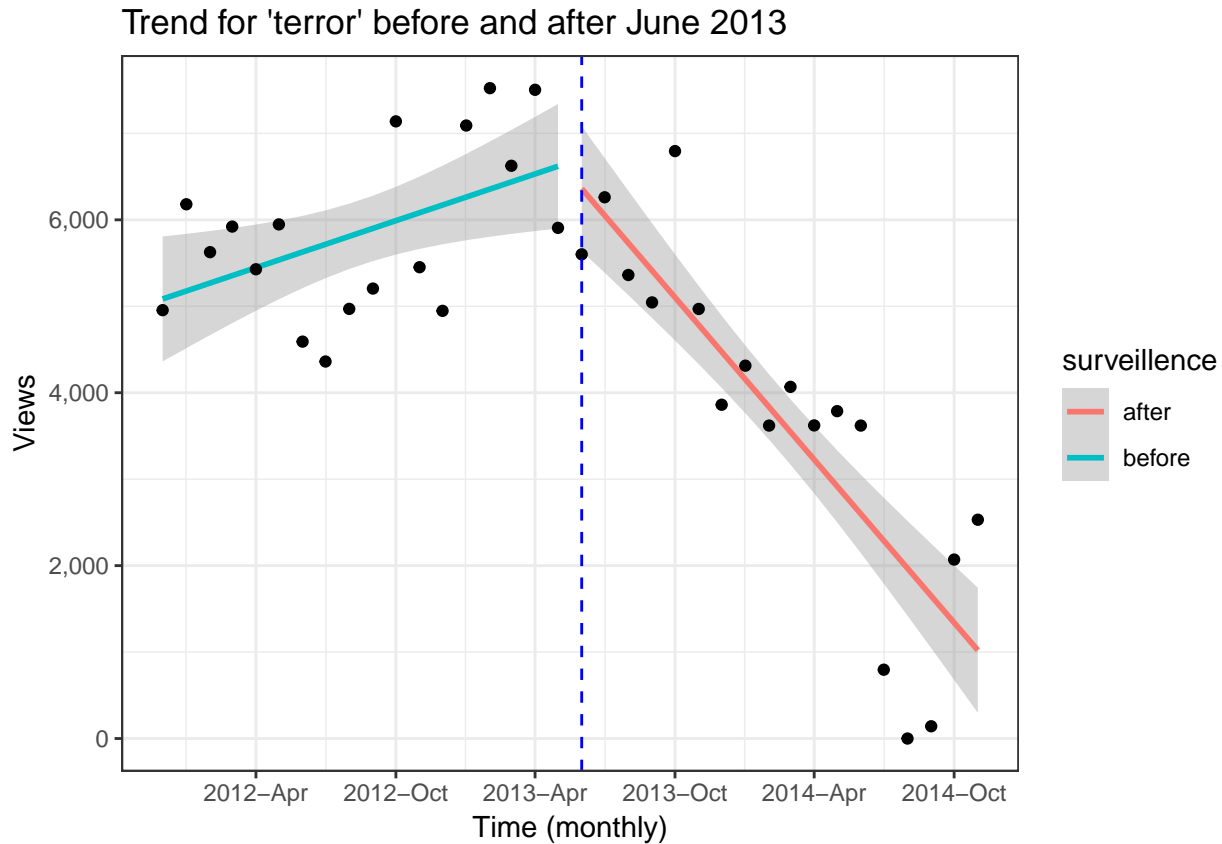
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42080 on 32 degrees of freedom
## Multiple R-squared:  0.4909, Adjusted R-squared:  0.4432
## F-statistic: 10.29 on 3 and 32 DF,  p-value: 6.776e-05
```



```
lm_plot_keyword(terrorism_data, 'terror', 'Trend for \'terror\' before and after June 2013')
```

```
##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1969.2  -700.2   172.0   754.2  1692.1
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.698e+05  2.337e+04   7.268 2.94e-08 ***
## month          -1.031e+01  1.450e+00  -7.110 4.56e-08 ***
## surveillancebefore -2.102e+05  3.247e+04  -6.473 2.78e-07 ***
## month:surveillancebefore  1.328e+01  2.049e+00   6.479 2.73e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

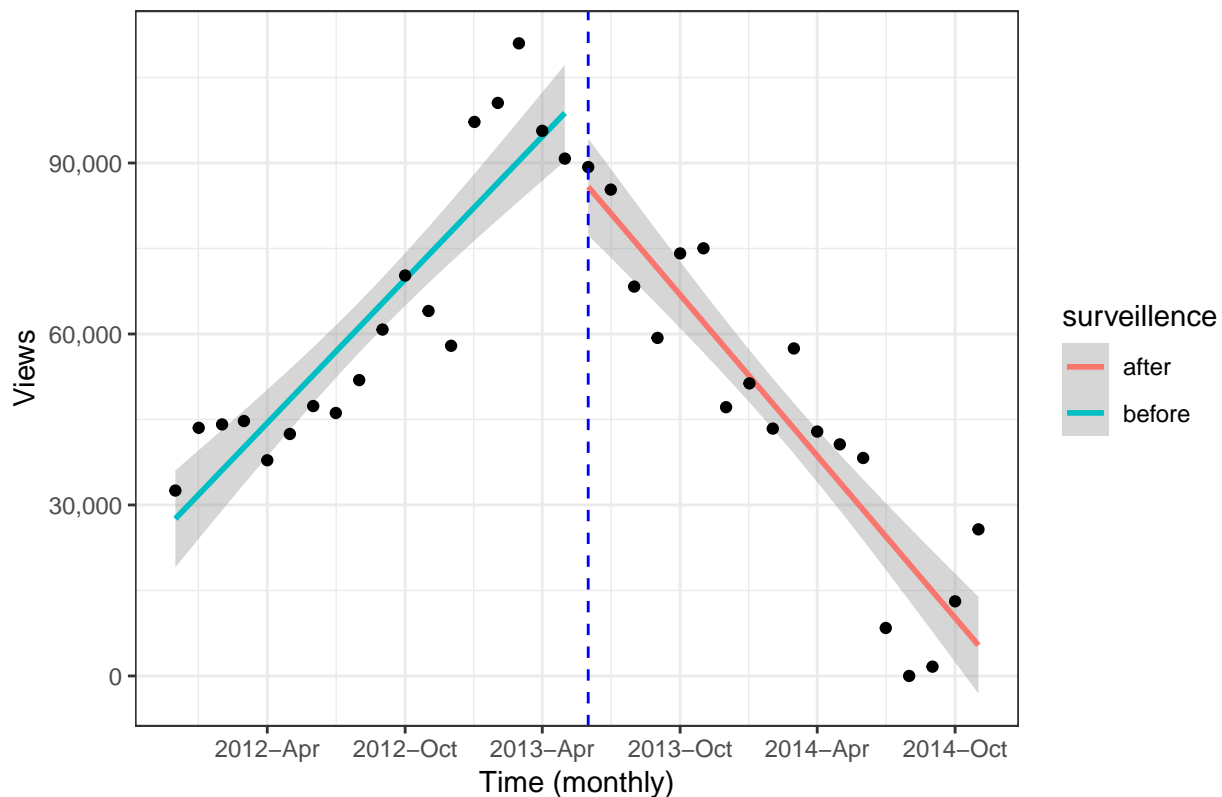
```
## Residual standard error: 970.8 on 32 degrees of freedom
## Multiple R-squared:  0.7564, Adjusted R-squared:  0.7336
## F-statistic: 33.13 on 3 and 32 DF,  p-value: 6.22e-10
```



```
lm_plot_keyword(terrorism_data, 'recruitment', 'Trend for \'recruitment\' before and after June 2013')
```

```
##
## Call:
## lm(formula = views ~ month + surveillance + month * surveillance,
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20046.9  -8343.9   843.7   7445.4  20621.8
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.548e+06  2.745e+05   9.282 1.36e-10 ***
## month          -1.553e+02  1.703e+01  -9.116 2.07e-10 ***
## surveillancebefore -4.629e+06  3.815e+05 -12.132 1.64e-13 ***
## month:surveillancebefore  2.930e+02  2.408e+01  12.169 1.52e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11410 on 32 degrees of freedom
## Multiple R-squared:  0.8417, Adjusted R-squared:  0.8268
## F-statistic: 56.7 on 3 and 32 DF,  p-value: 6.647e-13
```

### Trend for 'recruitment' before and after June 2013



## 3.3 Time-series Analysis

## 3.4 Trend Recovery

The following is a list of all packages used to generate these results. (Leave at very end of file.)

```
sessionInfo()
```

```
## R version 3.5.3 (2019-03-11)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.3
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] igraph_1.2.4      here_0.1          wikipediatrend_2.1.1
## [4] lubridate_1.7.4   forcats_0.3.0     stringr_1.4.0
## [7] dplyr_0.8.0.1     purrr_0.3.0       readr_1.3.1
```

```

## [10] tidyr_0.8.2          tibble_2.1.1          ggplot2_3.1.1
## [13] tidyverse_1.2.1      scales_1.0.0
##
## loaded via a namespace (and not attached):
## [1] tidyselect_0.2.5 xfun_0.4             haven_2.0.0          lattice_0.20-38
## [5] colorspace_1.4-1 generics_0.0.2      htmltools_0.3.6     yaml_2.2.0
## [9] rlang_0.3.4       pillar_1.3.1         glue_1.3.1           withr_2.1.2
## [13] hellno_0.0.1      modelr_0.1.3         readxl_1.3.0         plyr_1.8.4
## [17] munsell_0.5.0     gtable_0.3.0         cellranger_1.1.0    rvest_0.3.3
## [21] evaluate_0.13     labeling_0.3         knitr_1.21           broom_0.5.1
## [25] Rcpp_1.0.1        backports_1.1.3      jsonlite_1.6         hms_0.4.2
## [29] digest_0.6.18     stringi_1.4.3        grid_3.5.3           rprojroot_1.3-2
## [33] cli_1.1.0         tools_3.5.3          magrittr_1.5         lazyeval_0.2.2
## [37] crayon_1.3.4      pkgconfig_2.0.2      xml2_1.2.0           assertthat_0.2.1
## [41] rmarkdown_1.11    httr_1.4.0           rstudioapi_0.9.0    R6_2.4.0
## [45] nlme_3.1-137      compiler_3.5.3

```