eda_v2.R

esteban

2025-04-24

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(evir)
```

```
##
## Attaching package: 'evir'
```

```
## The following object is masked from 'package:ggplot2':
##
##     qplot
```

```
library(qrmtools)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(purrr)
df=read.csv('~/rug/thesis/data/influencer_sample160325.csv') %>% rename(channel_uid=channelId) %>%
  mutate(return=engagements_rate) %>% filter(!is.na(engagements))

library(evd)
```

```
##
## Attaching package: 'evd'
```

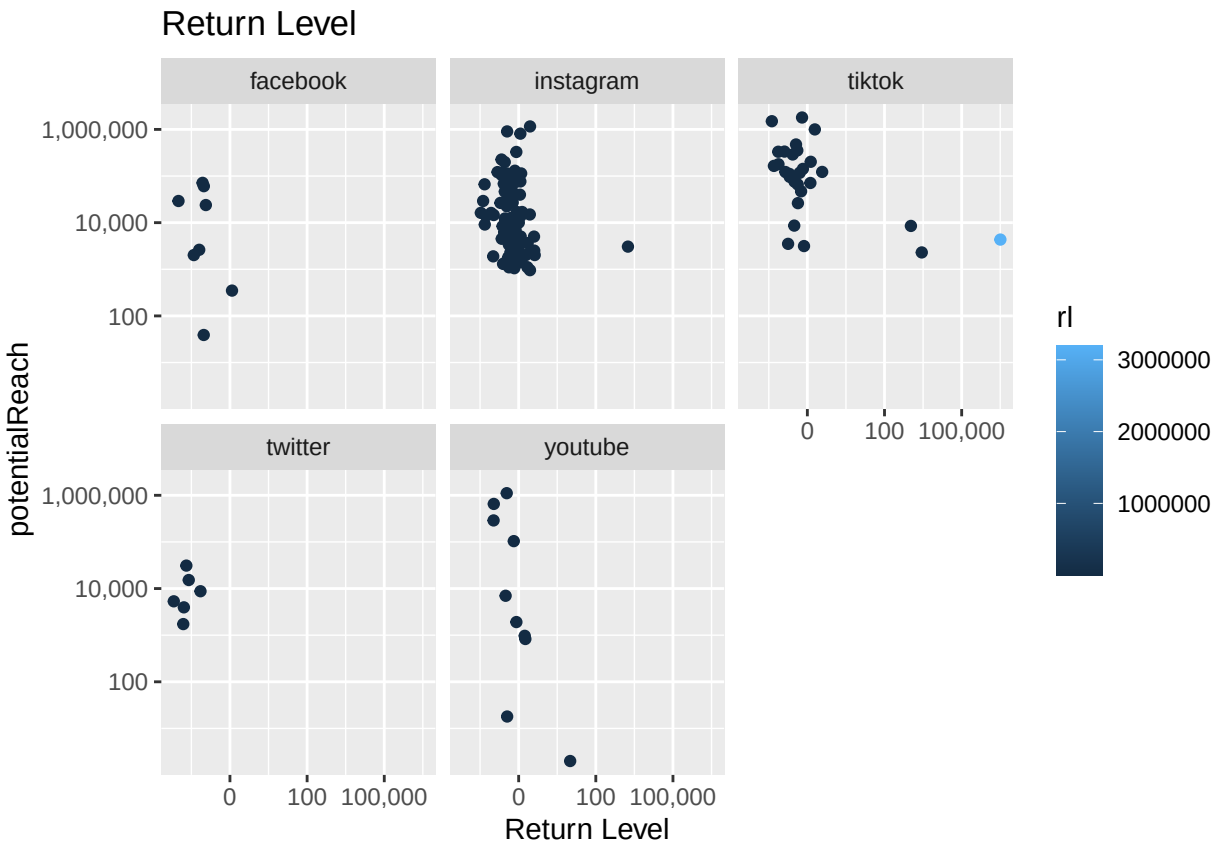
```
## The following objects are masked from 'package:evir':
##
##     dgev, dgpd, pgev, pgpd, qgev, qgpd, rgev, rgpd
```

```
#plot(M1)
options(scipen = 999)

df_temp=df %>% filter(channel_uid=='b8f9c1d87da534d29cdf67ebd1525fdc')
```

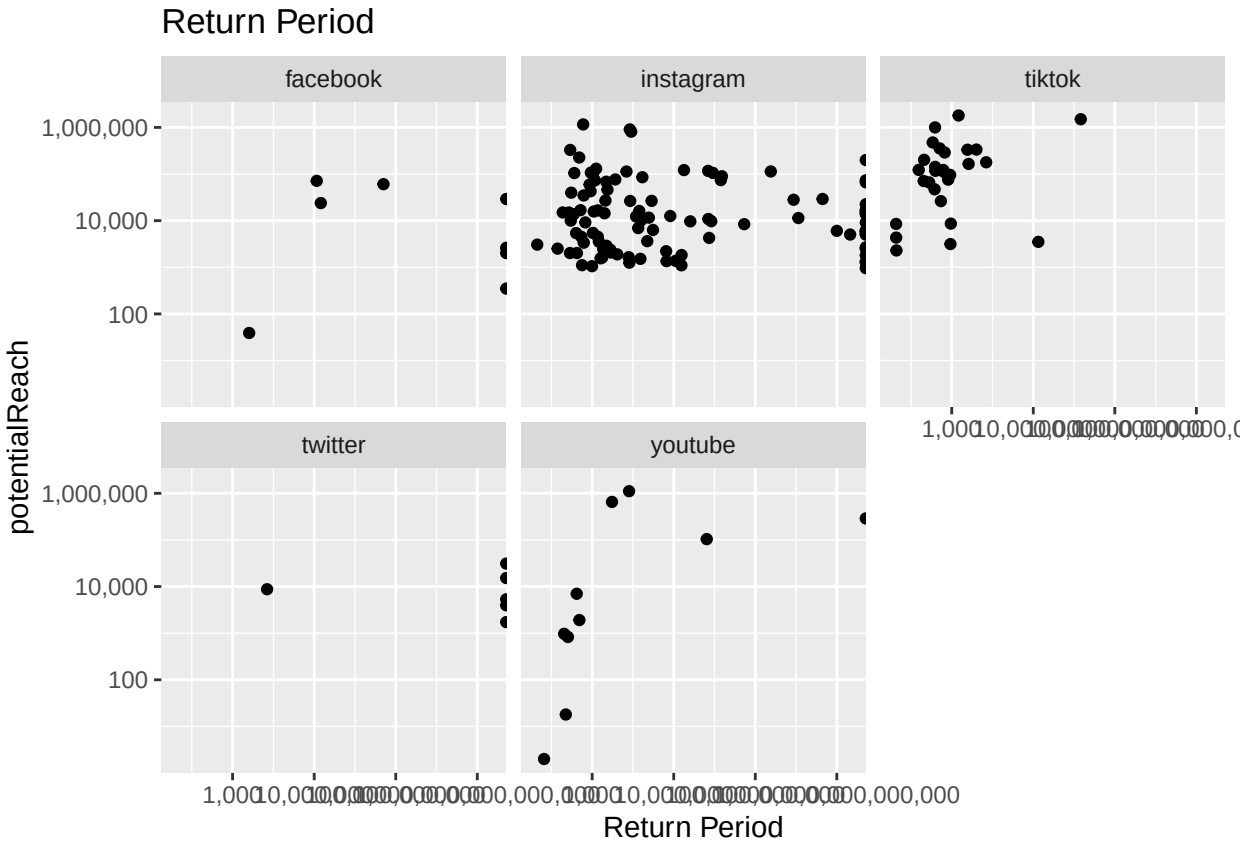
```r
m1=fgev(df_temp$engagements_rate,std.err = FALSE)
m1
```

```
##
## Call: fgev(x = df_temp$engagements_rate, std.err = FALSE)
## Deviance: -1668.771
##
## Estimates
##      loc     scale     shape
## 0.004623  0.005762  1.045781
##
## Optimization Information
##   Convergence: successful
##   Function Evaluations: 121
##   Gradient Evaluations: 18
```

```r
##Return level
rl=qgev(1-1/10,loc = m1$param[1],scale = m1$param[2],shape = m1$param[3])
##Return period
rp=as.double(1/(1-pgev(1,loc = m1$param[1],scale = m1$param[2],shape = m1$param[3])))


# Function to compute rl and rp for a given channel
evaluate_channel <- function(data) {
  tryCatch({
    m1 <- fgev(data$engagements_rate,std.err = FALSE)
    n=length(data$engagements_rate )
    rl <- qgev(1 - 1/10, loc = m1$param[1], scale = m1$param[2], shape = m1$param[3])
    rp <- as.double(1 / (1 - pgev(1, loc = m1$param[1], scale = m1$param[2], shape = m1$param[3])))

    tibble(channel_uid = unique(data$channel_uid), rl = rl, rp = rp,n=n,mean=mean(data$engagements_rate
  }, error = function(e) {
    tibble(channel_uid = unique(data$channel_uid), rl = NA, rp = NA)
  })
}
df_metadata=df %>% group_by(channel_uid) %>% summarise(platform=first(platform),potentialReach=median(p
# Apply function to each channel
df_results <- df %>%
  group_split(channel_uid) %>%
  map_dfr(evaluate_channel) %>%
  left_join(df_metadata,by=c('channel_uid'))%>%
  mutate(el=rl*potentialReach,er=mean*potentialReach)%>%
  mutate_at(c('rl','mean','el','el'),list(rank=function(x)min_rank(-x))) %>%
  mutate_at(c('rp','n'),list(rank=function(x)min_rank(x)))


df_results %>% ggplot(aes(x=rl,y=potentialReach,color=rl))+
  geom_point()+
  scale_y_log10(labels= scales::comma_format())+
  scale_x_log10(labels= scales::comma_format())+facet_wrap(platform~.)+
  scale_color_continuous(type='gradient')+
  labs(title='Return Level',x='Return Level')
```




```
df_results %>% ggplot(aes(x=rp,y=potentialReach))+
  geom_point()+
  scale_y_log10(labels= scales::comma_format())+
  scale_x_log10(labels= scales::comma_format())+facet_wrap(platform~.)+
  labs(title='Return Period',x='Return Period')
```


Return Period
facebook
instagram
tiktok
twitter
youtube
1,000,000
10,000
100
potentialReach
Return Period