Meetup Activity Analysis v1

Augustina Ragwitz

May 26, 2017

```
##
## 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
```

Overview

Meetup.com provides a centralized source for community organization. While it is not the only source for tracking community activity, it is a very popular one. We want to better understand where meetup activity for different types of groups are happening to ultimately identify where we should target community outreach efforts.

We want to identify: * where the most activity is happening, to prioritize supporting groups and events in those areas * where less activity is happening for one type of meetups compared to the most activity for a related type of meetups, to identify opportunities to increase community influence

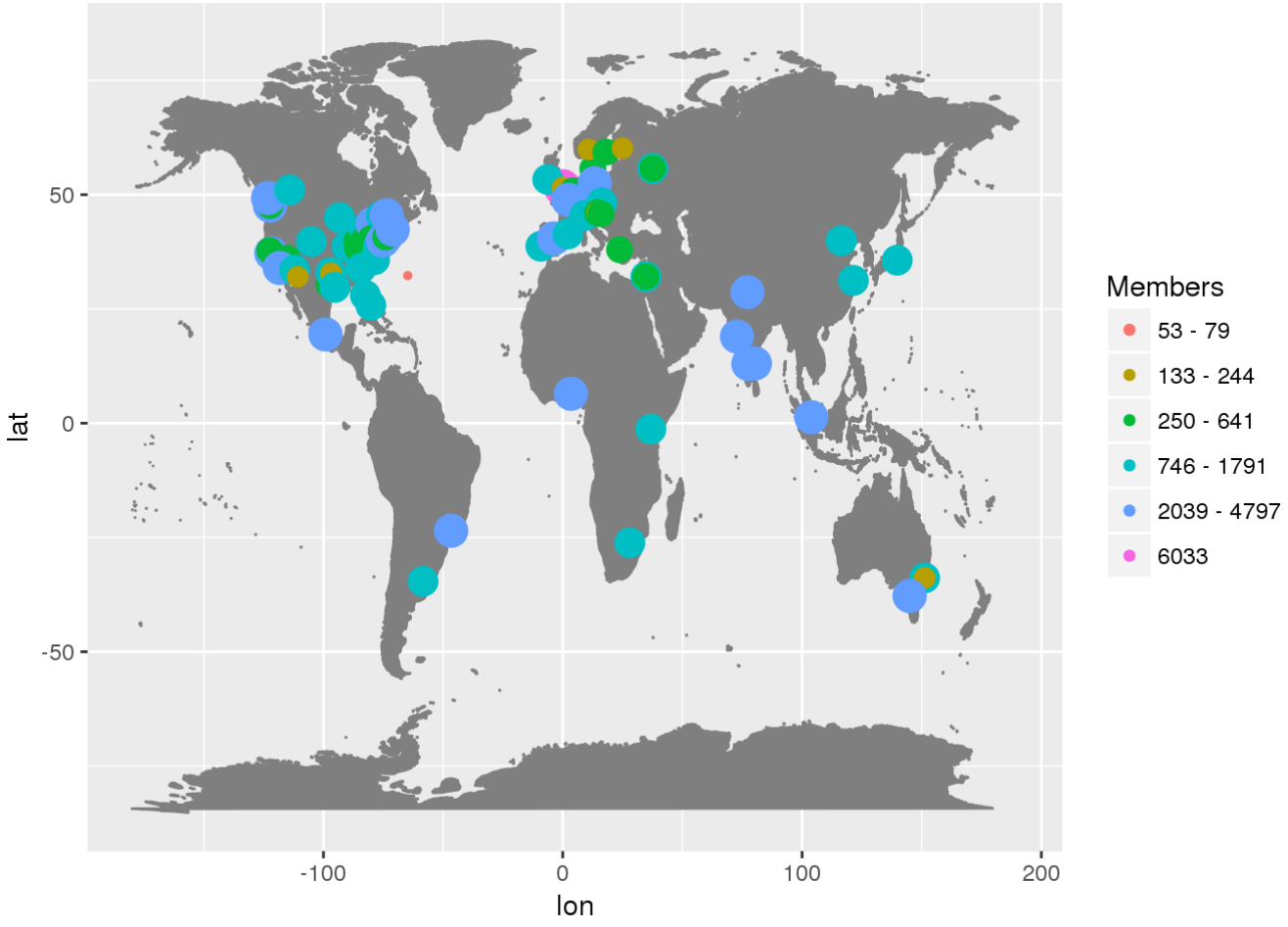
The goal of this study is to determine a method for measuring the activity for Meetup groups. In this study, data were pulled via the Meetup API for the IBM Big Data Developer groups that span the globe. We will look at the overall distribution of groups and calculate different metrics based on available data points to develop a way to categorize these groups by overall activity level.

Global Distribution of Groups

Where are the groups located and how many members are there per group?

```
bdd_groups <- read.csv('../data/groups_find_bdd.csv')
bdd_groups <- bdd_groups %>% filter(organizer_name == "Nancy Berlin")
```

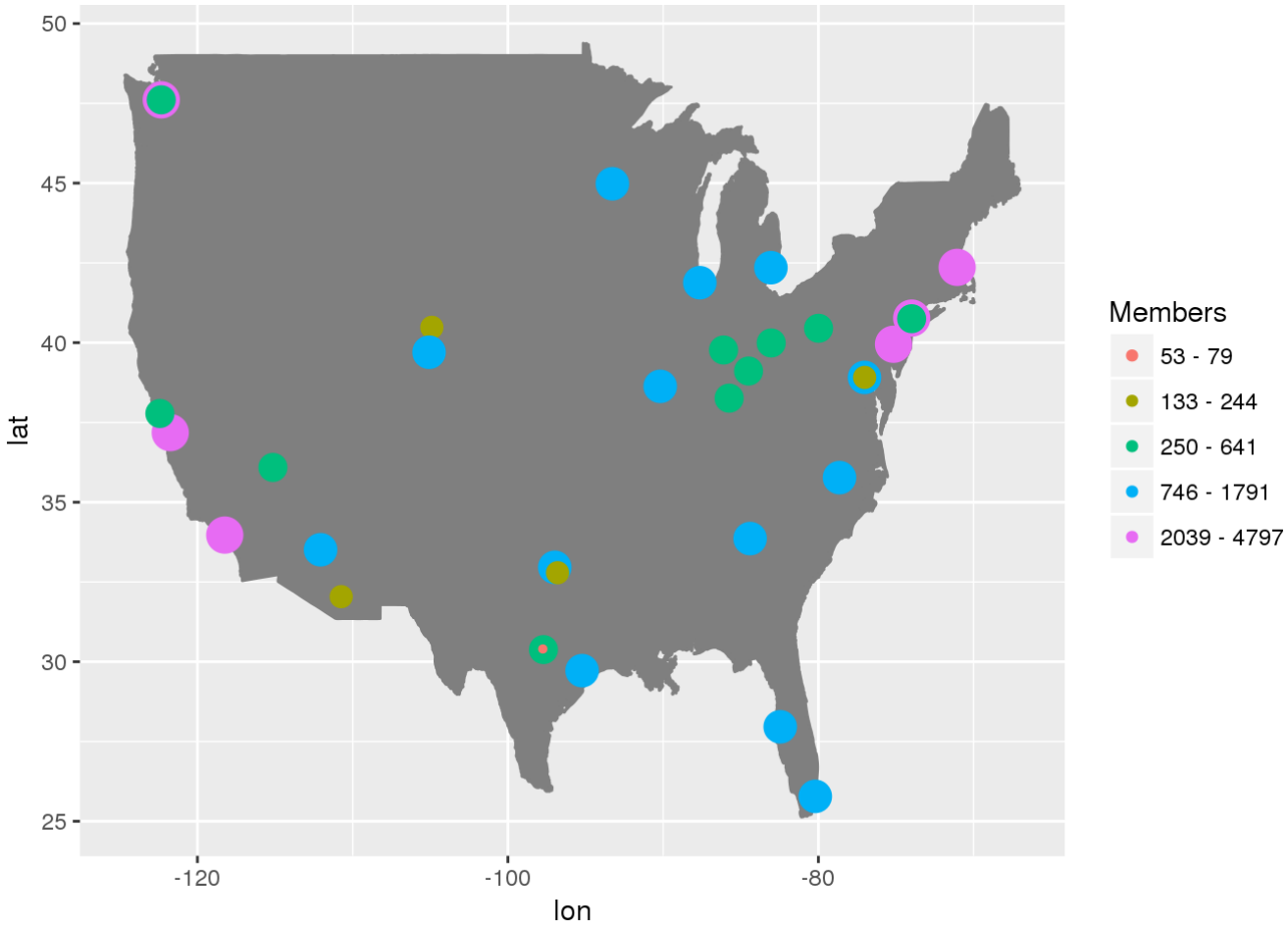
```
bdd_groups_members <- bdd_groups %>%
  mutate(members_log = round(log(members)))

bdd_groups_members <- bdd_groups_members %>%
  group_by(members_log) %>%
  mutate(members_min_max = ifelse(min(members) == max(members),
                                  paste(min(members)),
                                  paste(min(members), "-", max(members))))

ggplot(data=bdd_groups_members,
       aes(x=lon, y=lat, size=members_log, colour=reorder(members_min_max,
members_log))) +
  borders("world", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="Members") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_groups_members.png")
```

```
## Saving 7 x 5 in image
```

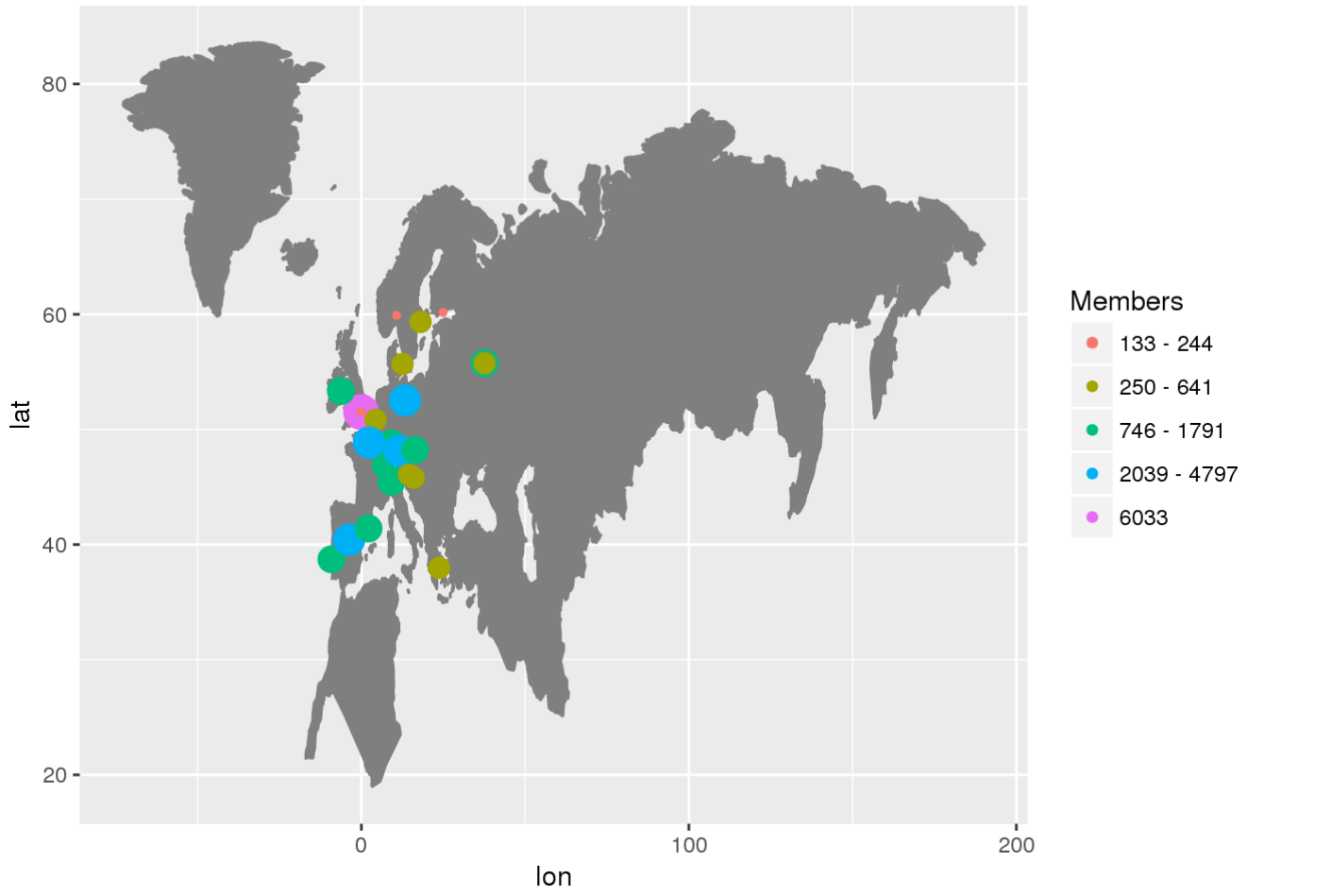
```
ggplot(data=bdd_groups_members %>% filter(country=="US"),
       aes(x=lon, y=lat, size=members_log, colour=reorder(members_min_max,
members_log))) +
  borders("usa", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="Members") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_groups_members_usa.png")
```

```
## Saving 7 x 5 in image
```

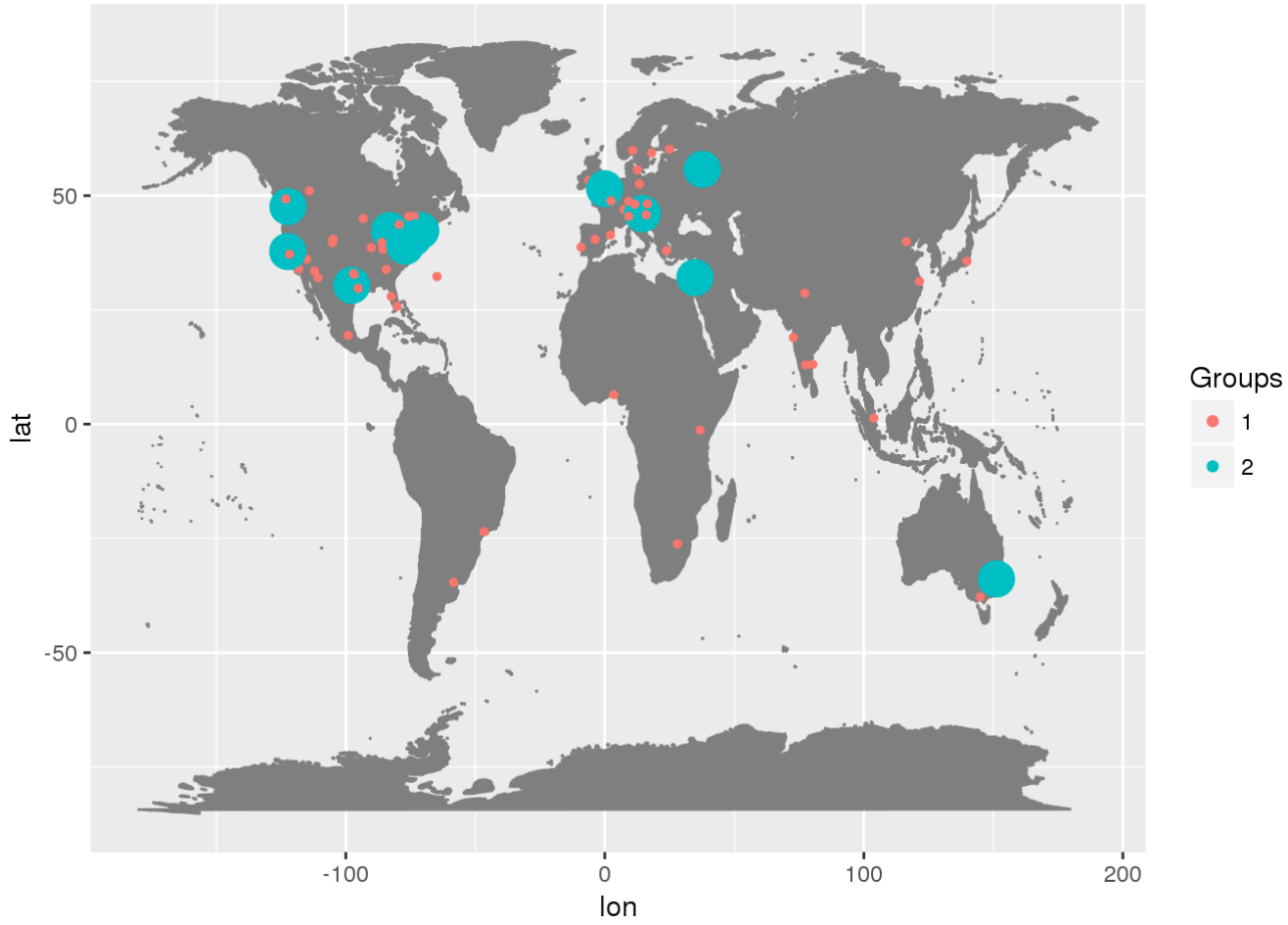
```
ggplot(data=bdd_groups_members %>% filter(grepl("^Europe", timezone)),
       aes(x=lon, y=lat, size=members_log, colour=reorder(members_min_max,
members_log))) +
  borders("world", colour="gray50", fill="gray50", xlim = c(-20, 59), ylim = c(35,
71)) +
  geom_point() +
  scale_color_discrete(name="Members") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_groups_members_eu.png")
```

```
## Saving 7 x 5 in image
```

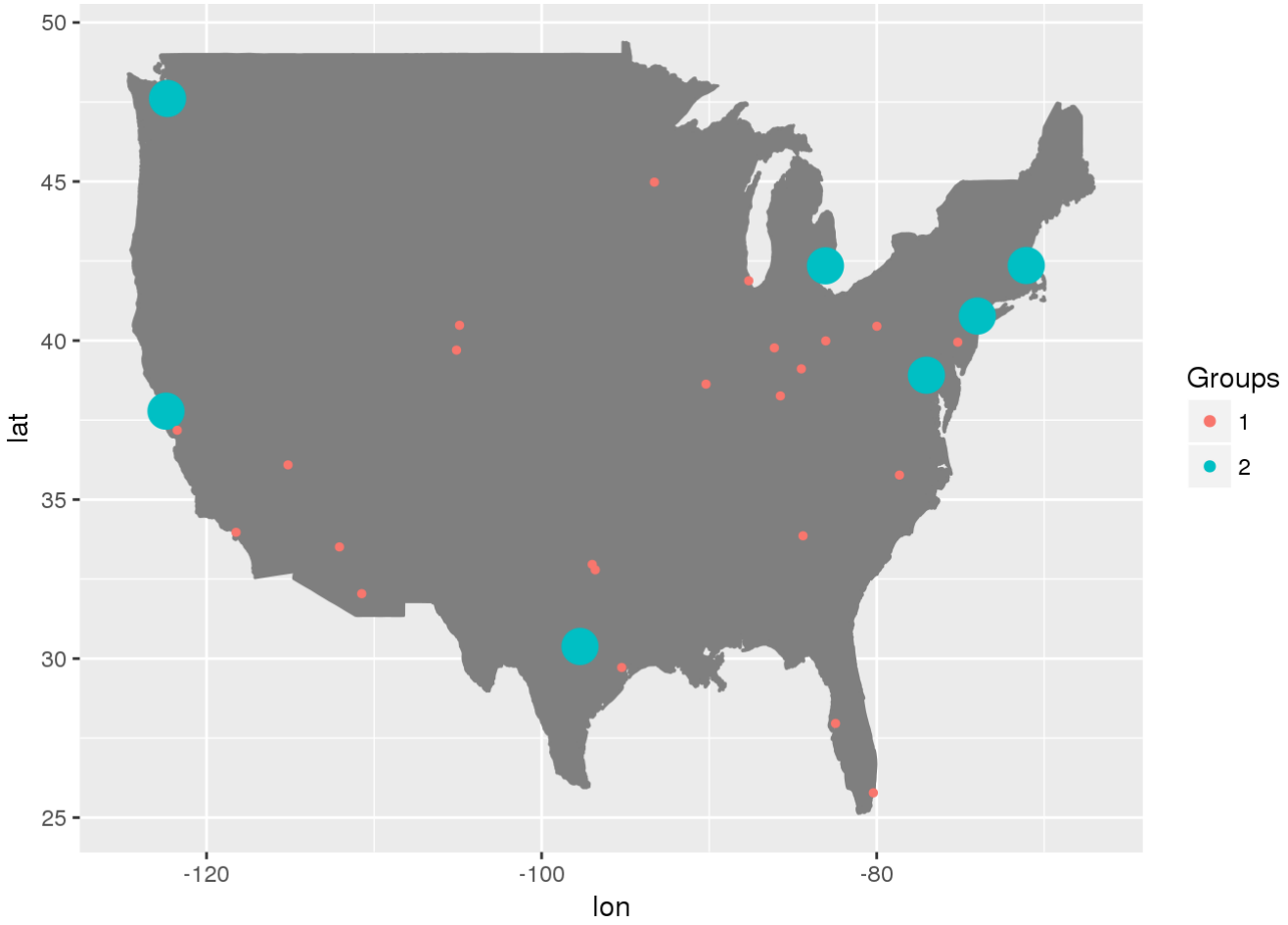
```
bdd_groups_per_city <- bdd_groups %>% group_by(city) %>%
  summarise(
    num_groups = n(),
    lon = first(lon),
    lat = first(lat),
    country = first(country),
    timezone = first(timezone)
  )

ggplot(data=bdd_groups_per_city, aes(x=lon, y=lat, size=num_groups, colour=factor(num_
groups))) +
  borders("world", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="Groups") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_groups_per_city.png")
```

```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_groups_per_city %>% filter(country=="US"),
       aes(x=lon, y=lat, size=num_groups, colour=factor(num_groups))) +
  borders("usa", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="Groups") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_groups_per_city_usa.png")
```

```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_groups_per_city %>% filter(grepl("^Europe", timezone)),
       aes(x=lon, y=lat, size=num_groups, colour=factor(num_groups))) +
  borders("world", colour="gray50", fill="gray50", xlim = c(-20, 59), ylim = c(35,
71)) +
  geom_point() +
  scale_color_discrete(name="Groups") +
  scale_size_continuous(guide=FALSE)
```


```
ggsave("bdd_groups_per_city_eu.png")
```

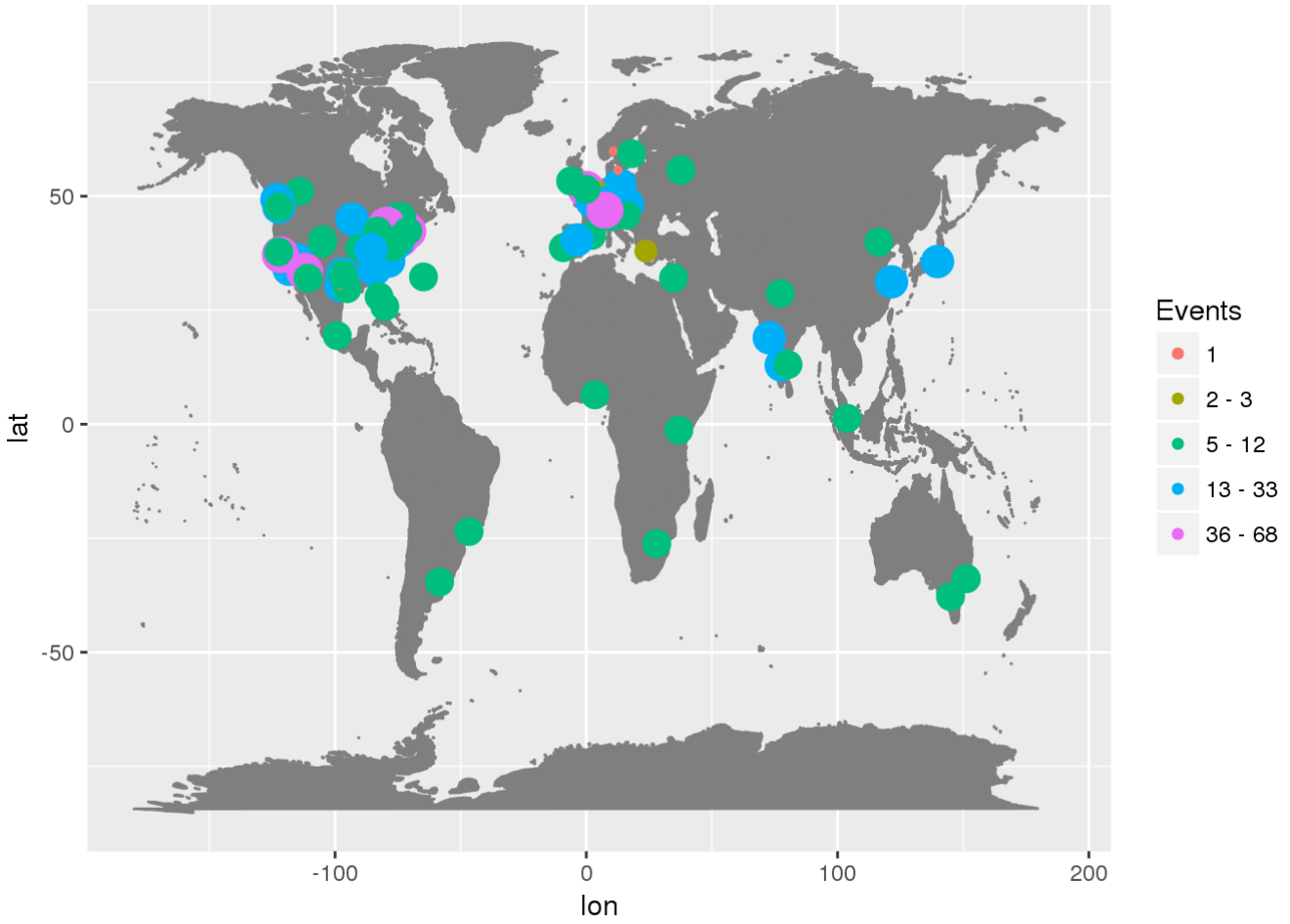
```
## Saving 7 x 5 in image
```

Past Events

```
bdd_past_events <- read.csv('../data/big_data_developers_events_past.csv')

bdd_past_events_merge <- merge(bdd_groups %>% select(urlname, lat, lon, city, country,
 timezone, members),
                          bdd_past_events, by.x="urlname", by.y="group_urlname")
```

How many events has each meetup group had?

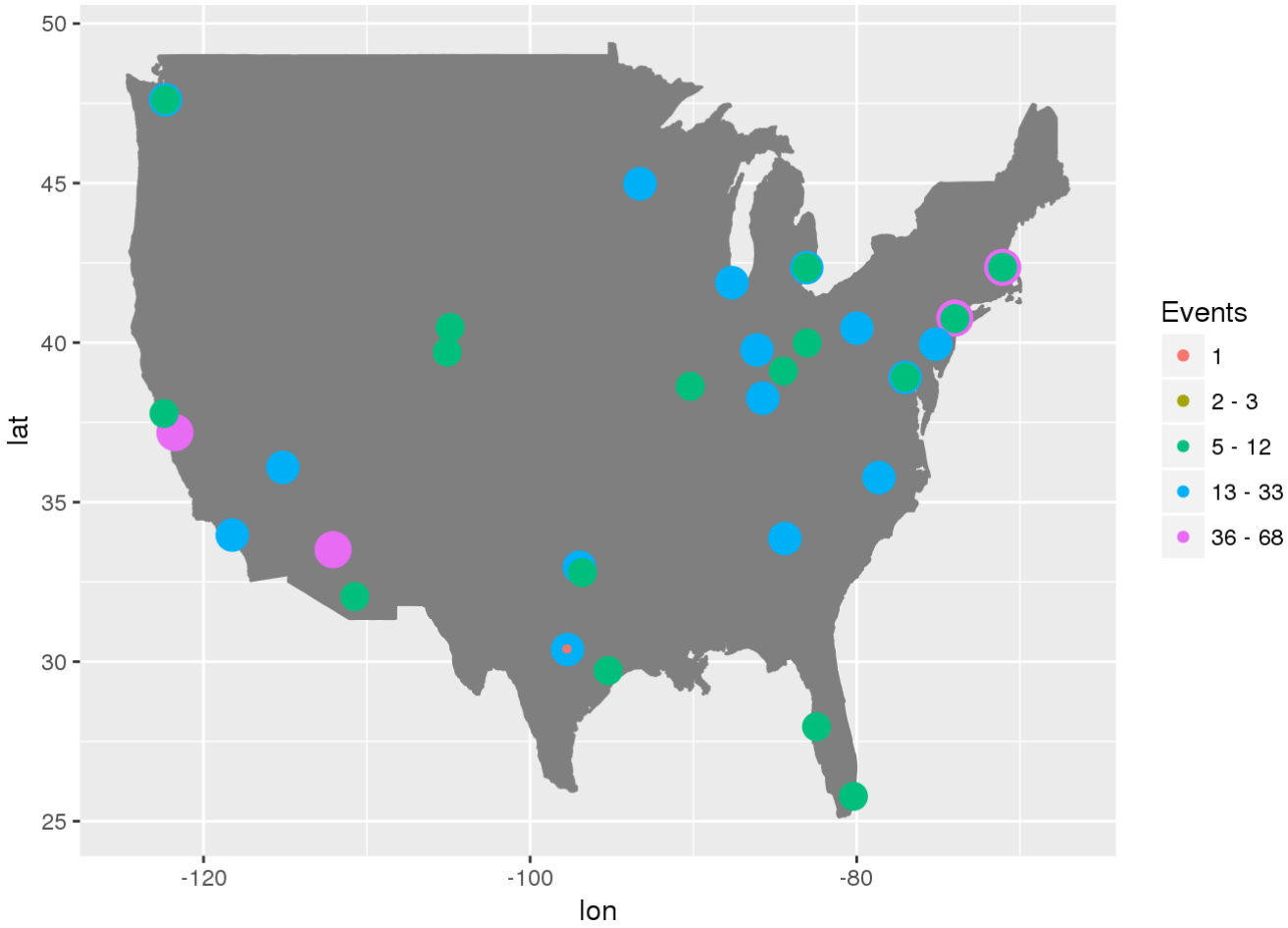
```
bdd_past_events_frequency <- bdd_past_events_merge %>%
  group_by(urlname) %>%
  summarise(num_events=n(),
            lat=first(lat), lon=first(lon),
            city=first(city), country=first(country), timezone=first(timezone)) %>%
  mutate(num_events_log = round(log(num_events)))

bdd_past_events_frequency <- bdd_past_events_frequency %>%
  group_by(num_events_log) %>%
  mutate(events_min_max = ifelse(min(num_events) == max(num_events),
                                 paste(min(num_events)),
                                 paste(min(num_events), "-", max(num_events))))

ggplot(data=bdd_past_events_frequency,
       aes(x=lon, y=lat, size=num_events_log, colour=reorder(events_min_max, num_event
s_log))) +
  borders("world", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="Events") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_past_events.png")
```

```
## Saving 7 x 5 in image
```

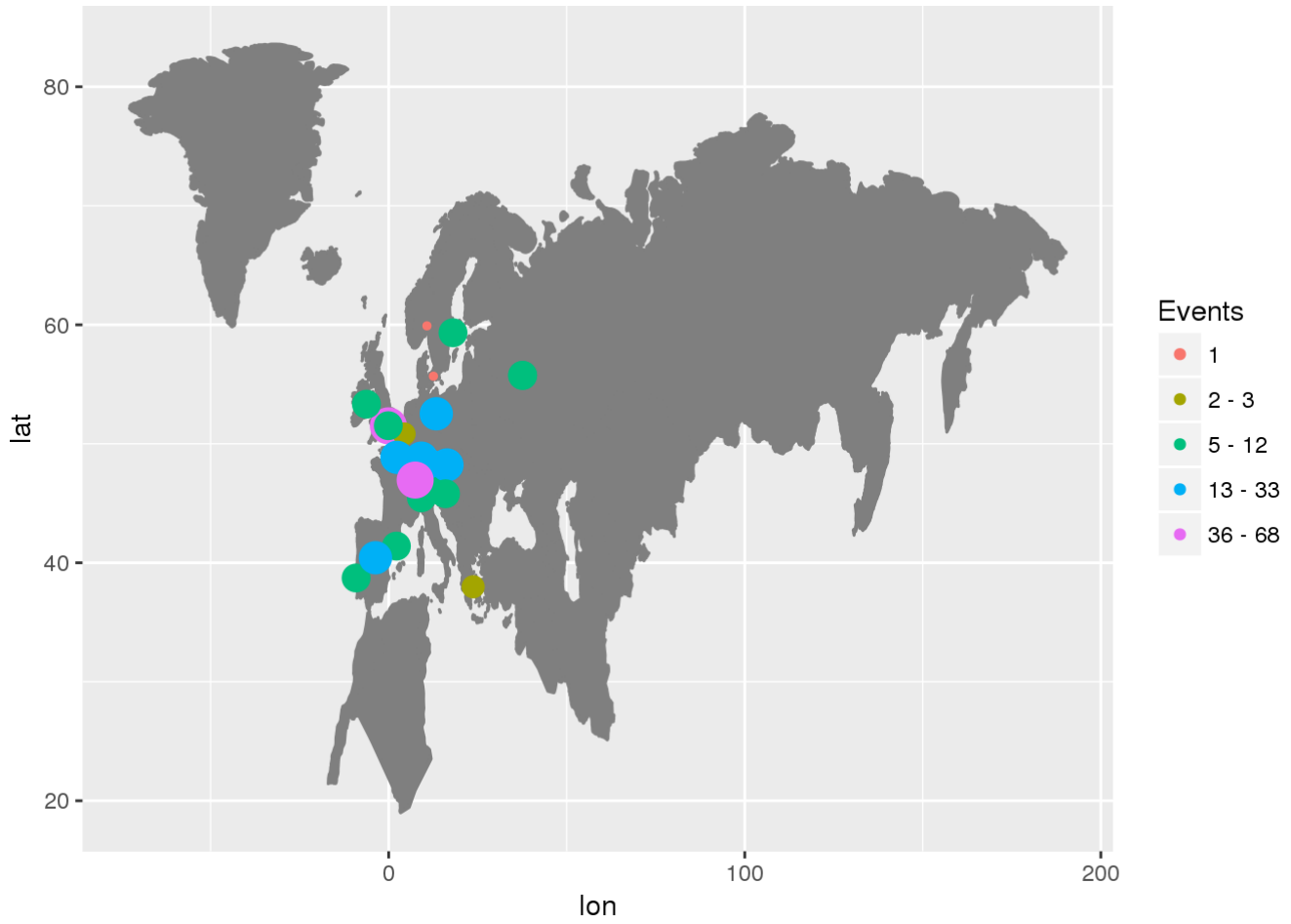
```
ggplot(data=bdd_past_events_frequency %>% filter(country=="US"),
       aes(x=lon, y=lat, size=num_events_log, colour=reorder(events_min_max, num_event
s_log))) +
  borders("usa", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="Events") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_past_events_usa.png")
```

```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_past_events_frequency %>% filter(grepl("^Europe", timezone)),
       aes(x=lon, y=lat, size=num_events_log, colour=reorder(events_min_max, num_event
s_log))) +
  borders("world", colour="gray50", fill="gray50", xlim = c(-20, 59), ylim = c(35,
71)) +
  geom_point() +
  scale_color_discrete(name="Events") +
  scale_size_continuous(guide=FALSE)
```



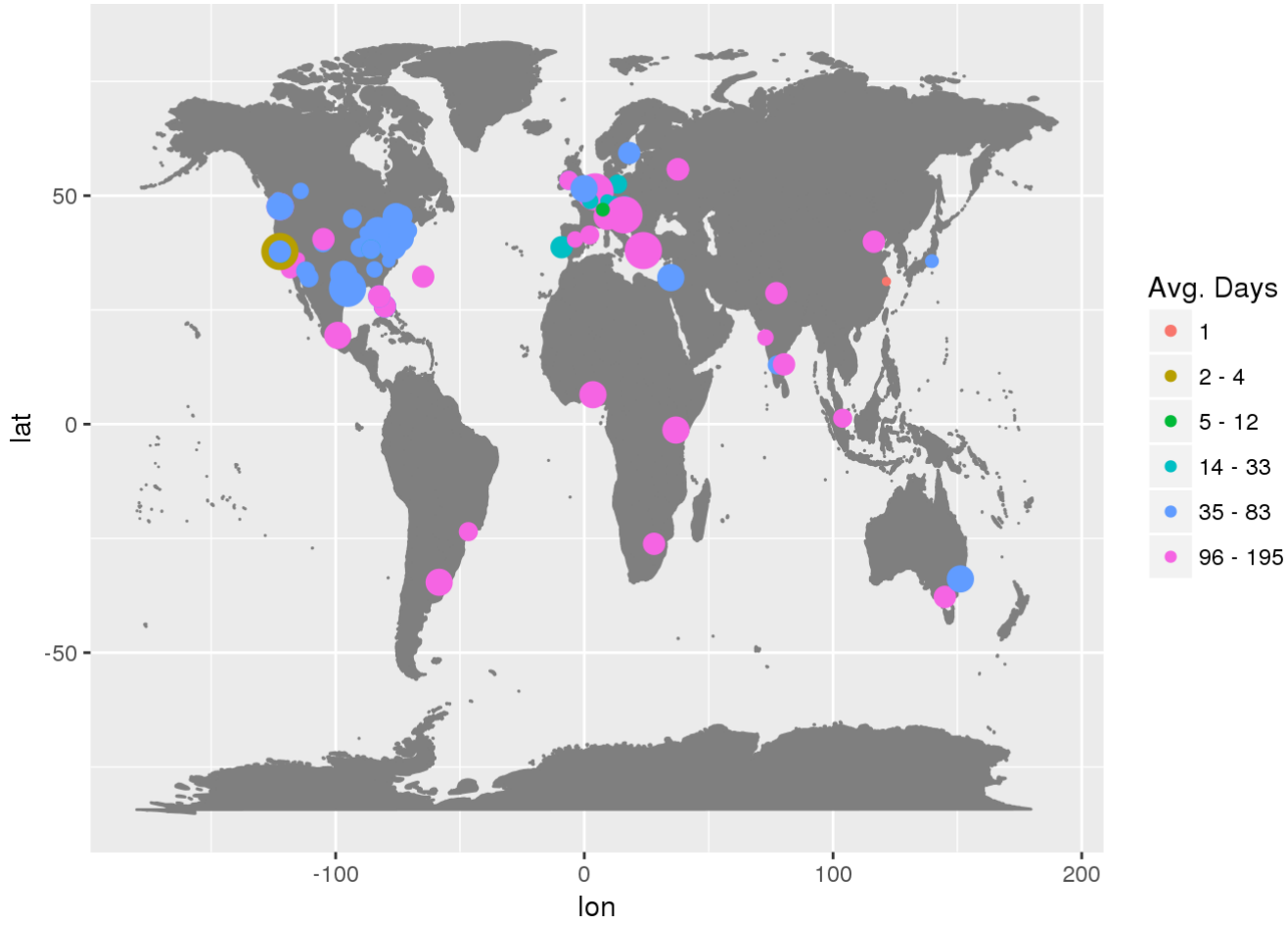

```
ggsave("bdd_past_events_eu.png")
```

```
## Saving 7 x 5 in image
```

How often do groups have events? Is there a regular interval?

Size indicates how much variability there was in the overall times between events. A group with a lot of variability in event times has a smaller dot, whereas a group with less variability will have a larger dot indicating it's more likely their events are happening at a more regular time. Color indicates the bucket with the range of average times between listed in the legend.

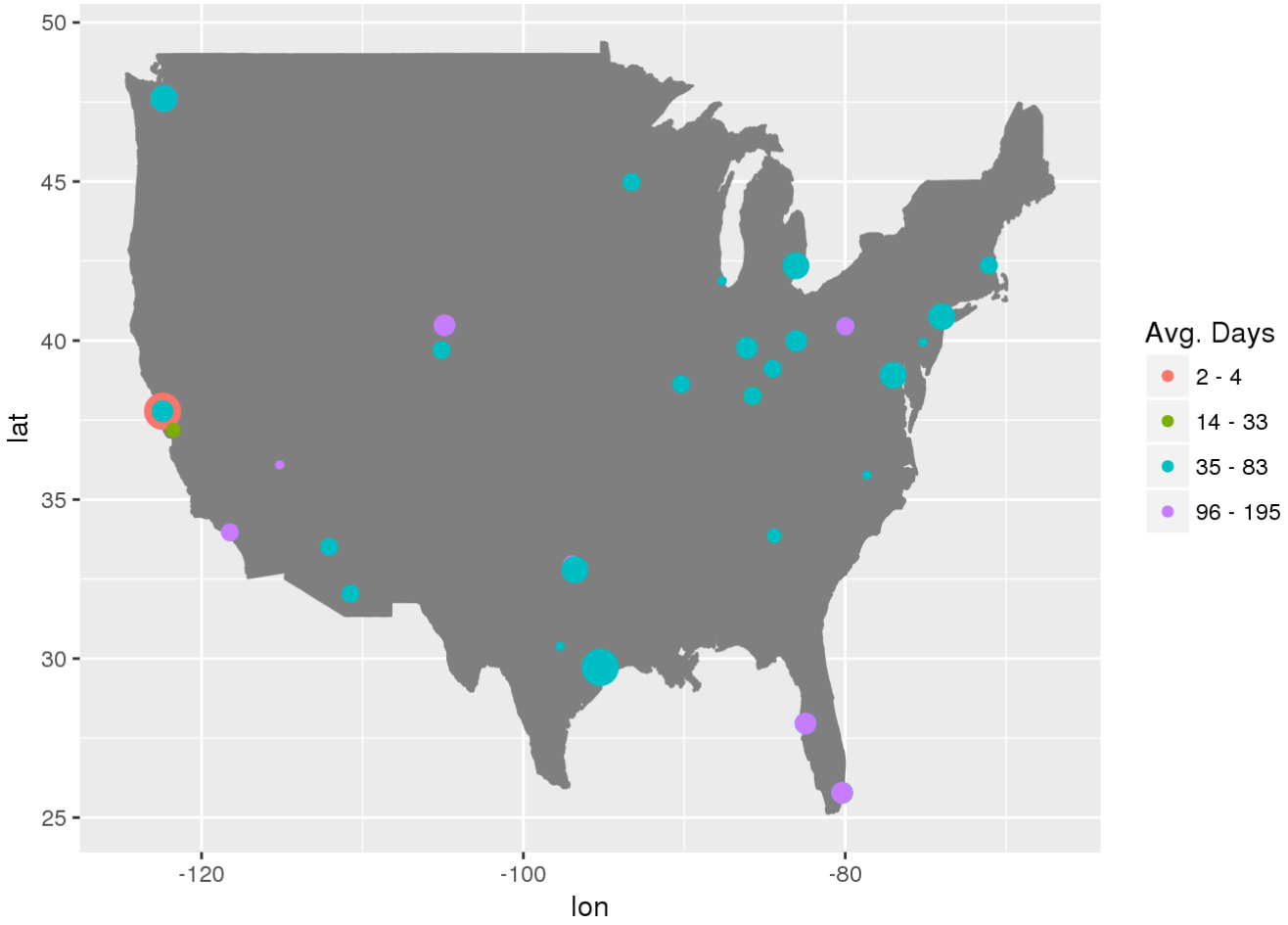
```
# compute time between events for each group
bdd_past_events_change <- bdd_past_events_merge %>%
  group_by(urlname) %>%
  arrange(urlname, time) %>%
  mutate(
    event_time = as.POSIXct(time/1000, origin="1970-01-01"),
    prev_event_time = lag(event_time),
    event_interval = ceiling(as.numeric(difftime(event_time, prev_event_time, units="d
ays")))
  )

bdd_past_events_time_between <- bdd_past_events_change %>%
  filter(!is.na(event_interval)) %>%
  mutate(event_interval_log = round(log(event_interval))) %>%
  group_by(urlname, event_interval_log) %>%
  summarise(
    num_events = n(),
    event_interval_mean = ceiling(mean(event_interval, na.rm=TRUE)),
    lat=first(lat),
    lon=first(lon),
    city=first(city),
    country=first(country),
    timezone=first(timezone)
  ) %>% mutate(
    event_interval_mean = ifelse(is.na(event_interval_mean), 0, event_interval_mean)
  ) %>% group_by(urlname) %>% mutate(
    num_events_max = max(num_events),
    buckets = n() # how much variation between events
  ) %>% ungroup() %>%
  group_by(event_interval_log) %>%
  mutate(
    event_interval_mean_min_max = ifelse(
      min(event_interval_mean) == max(event_interval_mean),
      paste(min(event_interval_mean)),
      paste(min(event_interval_mean), "-", max(event_interval_mean)))
  )

bdd_past_events_time_between_top <- bdd_past_events_time_between %>%
  filter(num_events == num_events_max)

ggplot(data=bdd_past_events_time_between_top,
       aes(x=lon, y=lat, size=1/buckets, colour=reorder(event_interval_mean_min_max, e
vent_interval_log))) +
  borders("world", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="Avg. Days") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_event_time.png")
```

```
## Saving 7 x 5 in image
```

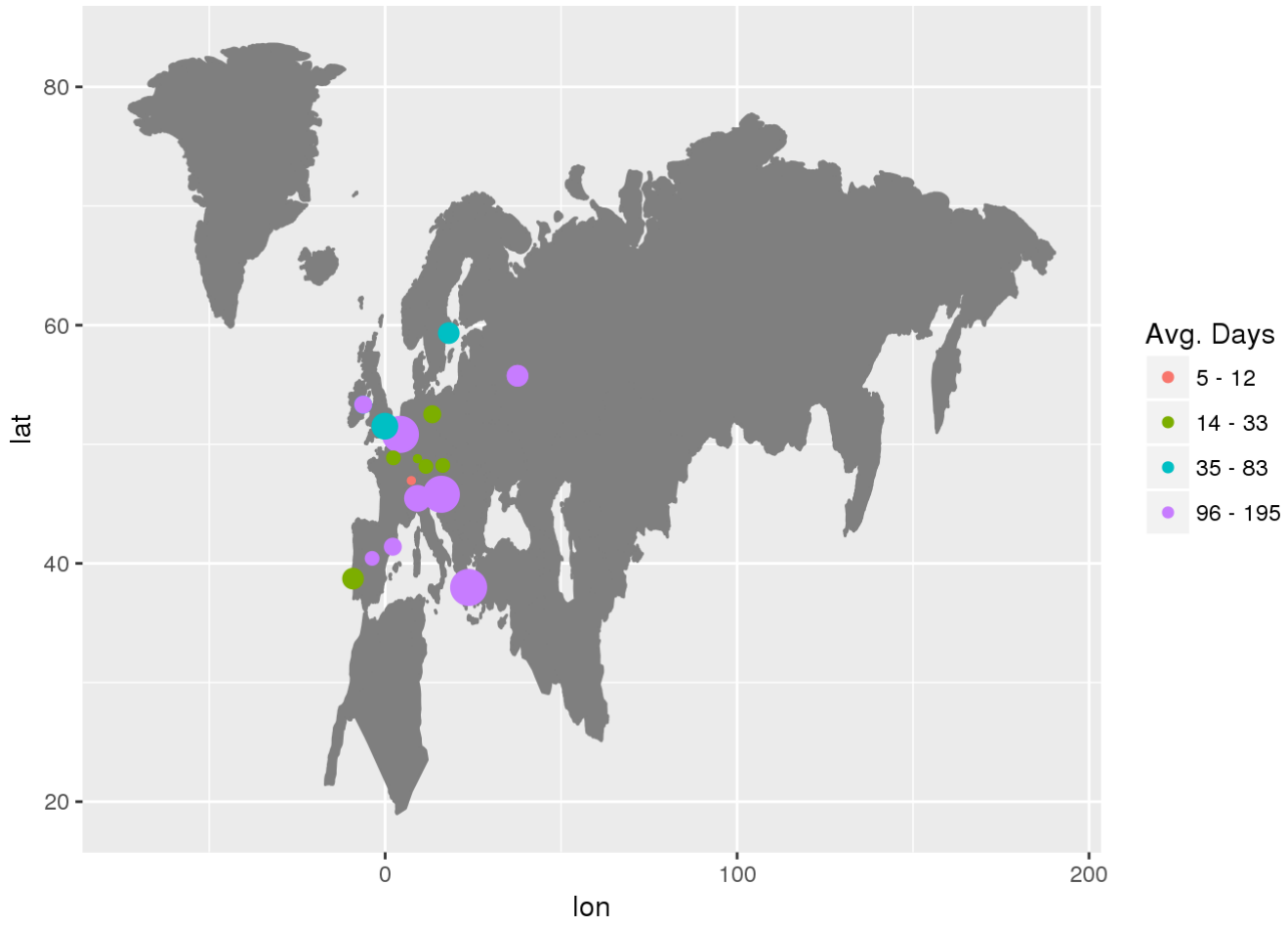
```
ggplot(data=bdd_past_events_time_between_top %>% filter(country=="US"),
       aes(x=lon, y=lat, size=1/buckets, colour=reorder(event_interval_mean_min_max, e
vent_interval_log))) +
  borders("usa", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="Avg. Days") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_event_time_usa.png")
```

```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_past_events_time_between_top %>% filter(grepl("^Europe", timezone)),
       aes(x=lon, y=lat, size=1/buckets, colour=reorder(event_interval_mean_min_max, e
vent_interval_log))) +
  borders("world", colour="gray50", fill="gray50", xlim = c(-20, 59), ylim = c(35,
71)) +
  geom_point() +
  scale_color_discrete(name="Avg. Days") +
  scale_size_continuous(guide=FALSE)
```

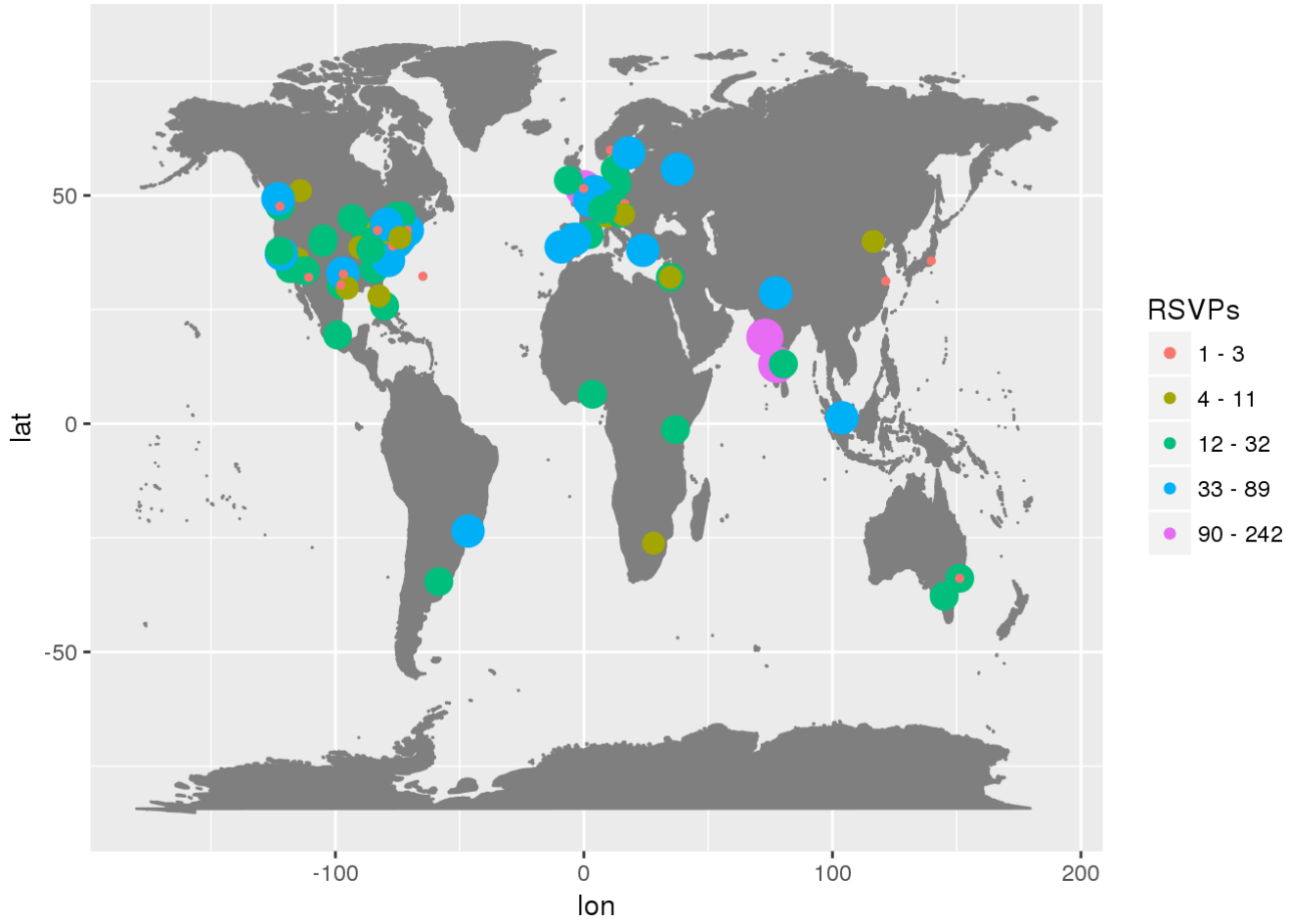



```
ggsave("bdd_event_time_eu.png")
```

```
## Saving 7 x 5 in image
```

How many RSVPs did each meetup event have?

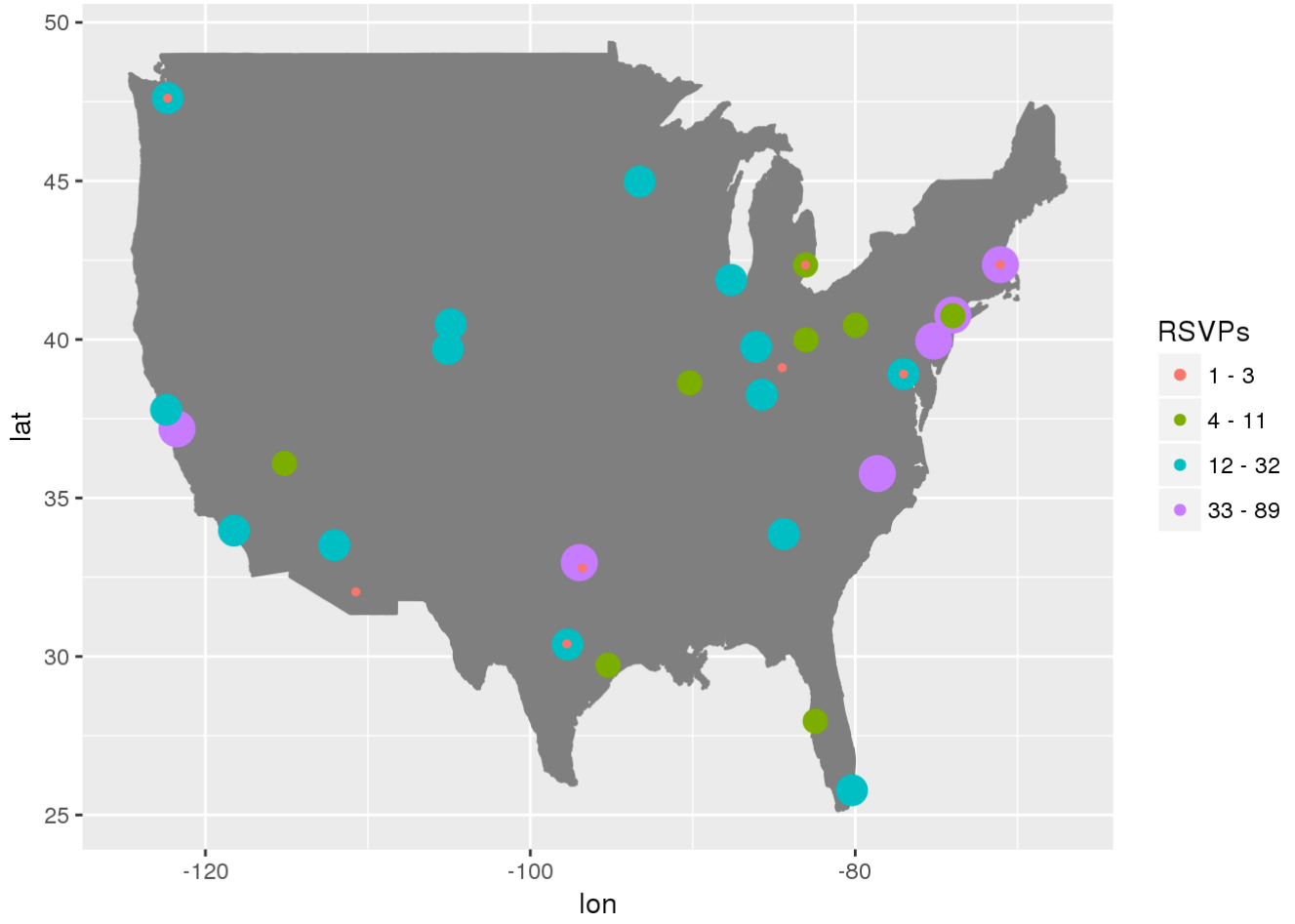
```
# num rsvps rounded log, group by city, count rsvp logs per city, histogram showing to
tals (or just top?)
# rsvp's as a proportion of total membership ??

bdd_past_events_merge <- bdd_past_events_merge %>%
  mutate(rsvp_log = round(log(yes_rsvp_count + 1)), # some have 0
         rsvp_members = round((yes_rsvp_count + 1)/members, 2),
         rsvp_members_log = round(log((yes_rsvp_count + 1)/members)))

bdd_past_events_rsvp <- bdd_past_events_merge %>%
  group_by(urlname, rsvp_log) %>%
  summarise(
    num_events = n(),
    rsvp_min = min(yes_rsvp_count),
    rsvp_max = max(yes_rsvp_count),
    lon = first(lon),
    lat = first(lat),
    city = first(city),
    country = first(country),
    timezone = first(timezone)
  )

bdd_past_events_rsvp <- bdd_past_events_rsvp %>%
  group_by(rsvp_log) %>%
  mutate(rsvp_min_max = ifelse(min(rsvp_min) == max(rsvp_max),
                               paste(min(rsvp_min)),
                               paste(min(rsvp_min), "-", max(rsvp_max))))

bdd_past_events_rsvp <- bdd_past_events_rsvp %>%
  group_by(urlname) %>%
  mutate(num_events_max = max(num_events))

# How many rsvp's did the majority of a group's events have?
bdd_past_events_rsvp_top <- bdd_past_events_rsvp %>%
  filter(num_events_max == num_events)

ggplot(data=bdd_past_events_rsvp_top,
       aes(x=lon, y=lat,
           size=rsvp_log,
           colour=reorder(rsvp_min_max, rsvp_log))) +
  borders("world", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="RSVPs") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_rsvps.png")
```

```
## Saving 7 x 5 in image
```

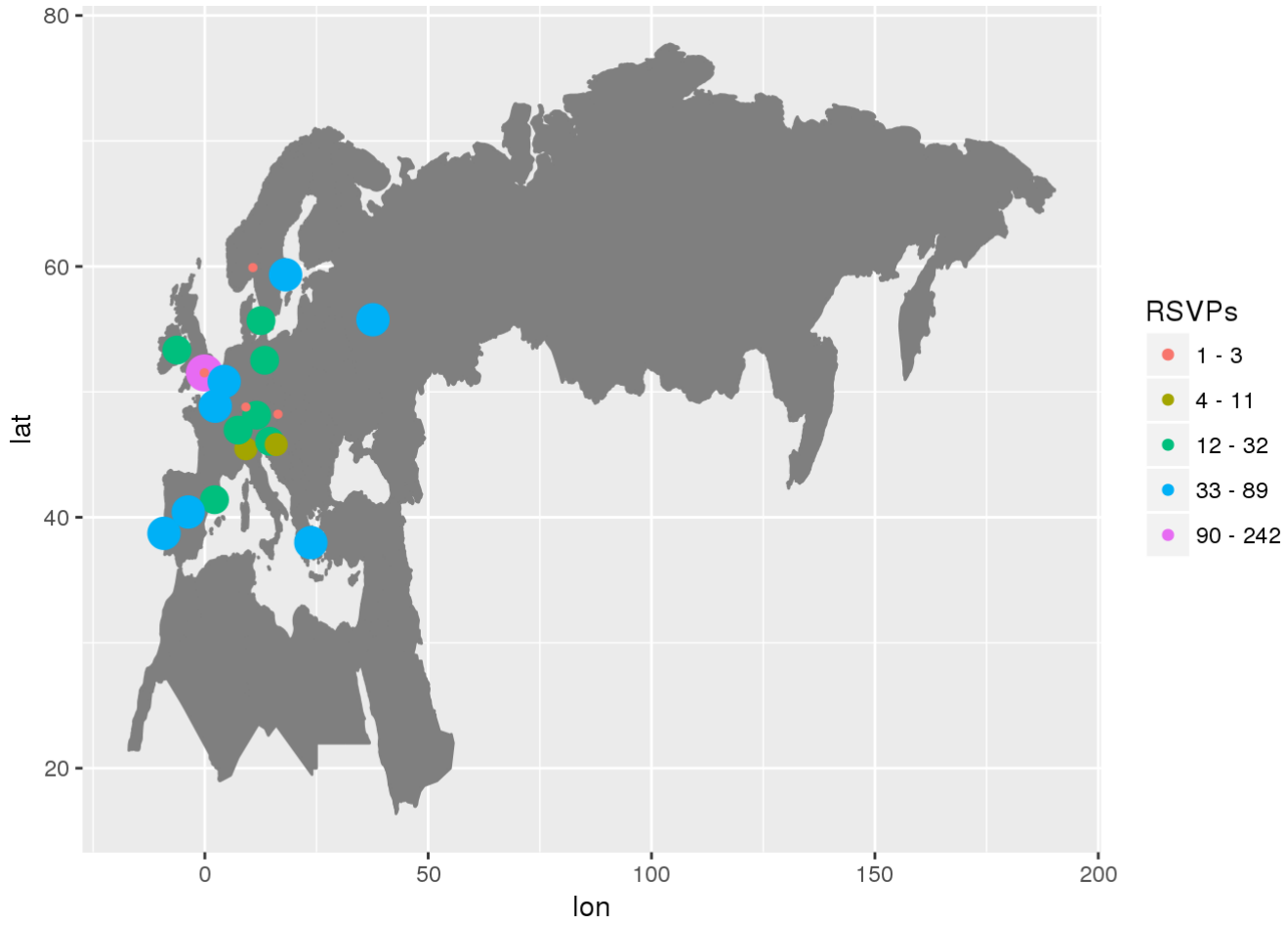
```
ggplot(data=bdd_past_events_rsvp_top %>% filter(country=="US"),
       aes(x=lon, y=lat, size=rsvp_log, colour=reorder(rsvp_min_max, rsvp_log))) +
  borders("usa", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="RSVPs") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_rsvps_us.png")
```

```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_past_events_rsvp_top %>% filter(grepl("^Europe", timezone)),
       aes(x=lon, y=lat, size=rsvp_log, color=reorder(rsvp_min_max, rsvp_log))) +
  borders("world", colour="gray50", fill="gray50", xlim = c(-10, 40), ylim = c(30,
60)) +
  geom_point() +
  scale_color_discrete(name="RSVPs") +
  scale_size_continuous(guide=FALSE)
```

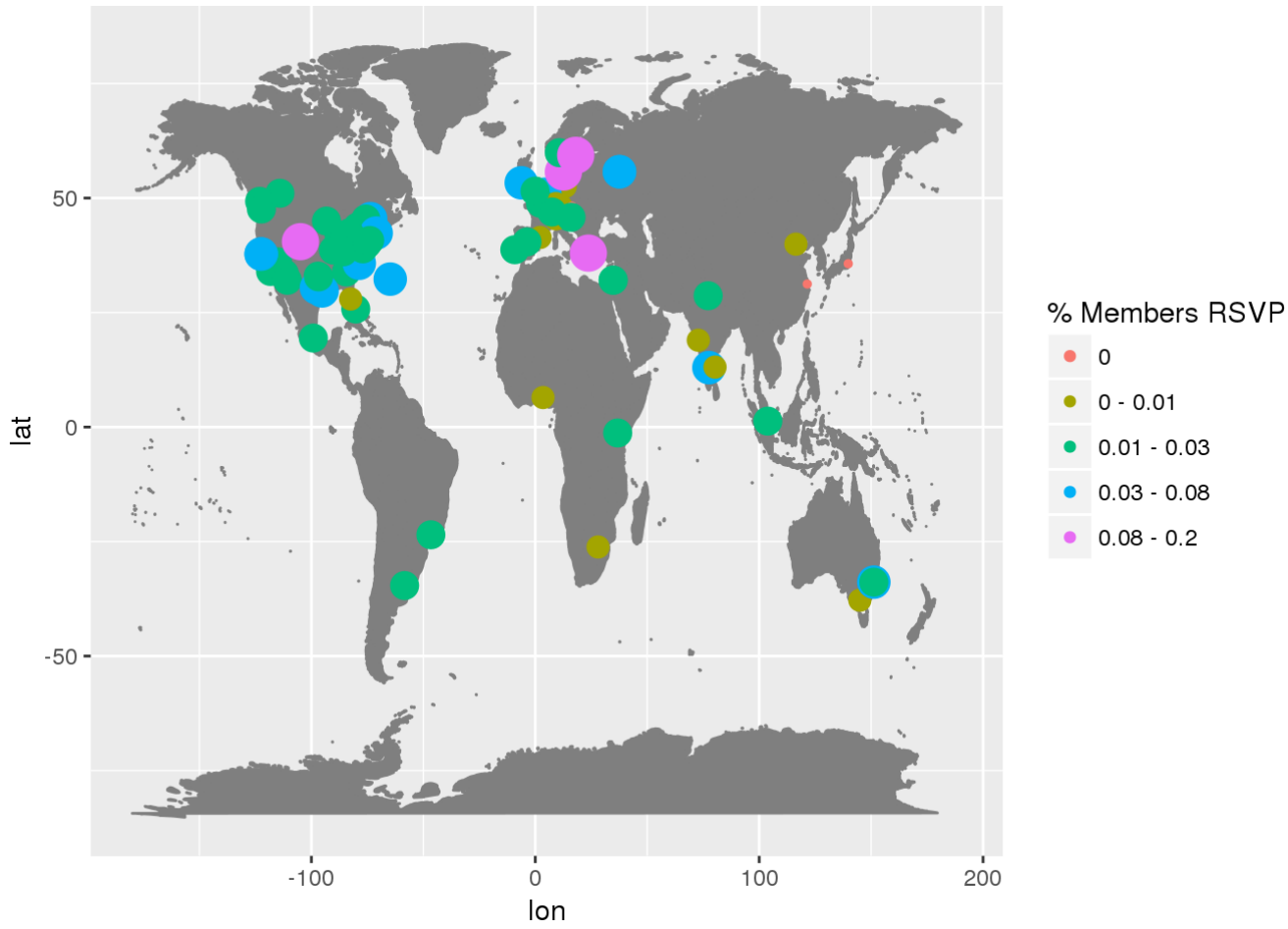



```
ggsave("bdd_rsvps_eu.png")
```

```
## Saving 7 x 5 in image
```

What proportion of members RSVP’ed?

```
bdd_past_events_rsvp_members <- bdd_past_events_merge %>%
  group_by(urlname, rsvp_members_log) %>%
  summarise(
    num_events = n(),
    rsvp_members_min = min(rsvp_members),
    rsvp_members_max = max(rsvp_members),
    lon = first(lon),
    lat = first(lat),
    city = first(city),
    country = first(country),
    timezone = first(timezone)
  )

bdd_past_events_rsvp_members <- bdd_past_events_rsvp_members %>%
  group_by(rsvp_members_log) %>%
  mutate(rsvp_members_min_max = ifelse(min(rsvp_members_min) == max(rsvp_members_max),

                                paste(min(rsvp_members_min)),
                                paste(min(rsvp_members_min), "-",
max(rsvp_members_max))))

bdd_past_events_rsvp_members <- bdd_past_events_rsvp_members %>%
  group_by(urlname) %>%
  mutate(num_events_max = max(num_events))

# How many rsvp's did the majority of a group's events have?
bdd_past_events_rsvp_members_top <- bdd_past_events_rsvp_members %>%
  filter(num_events_max == num_events)

ggplot(data=bdd_past_events_rsvp_members_top,
       aes(x=lon, y=lat,
           size=rsvp_members_log,
           colour=reorder(rsvp_members_min_max, rsvp_members_log))) +
  borders("world", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="% Members RSVP") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_rsvps_members.png")
```

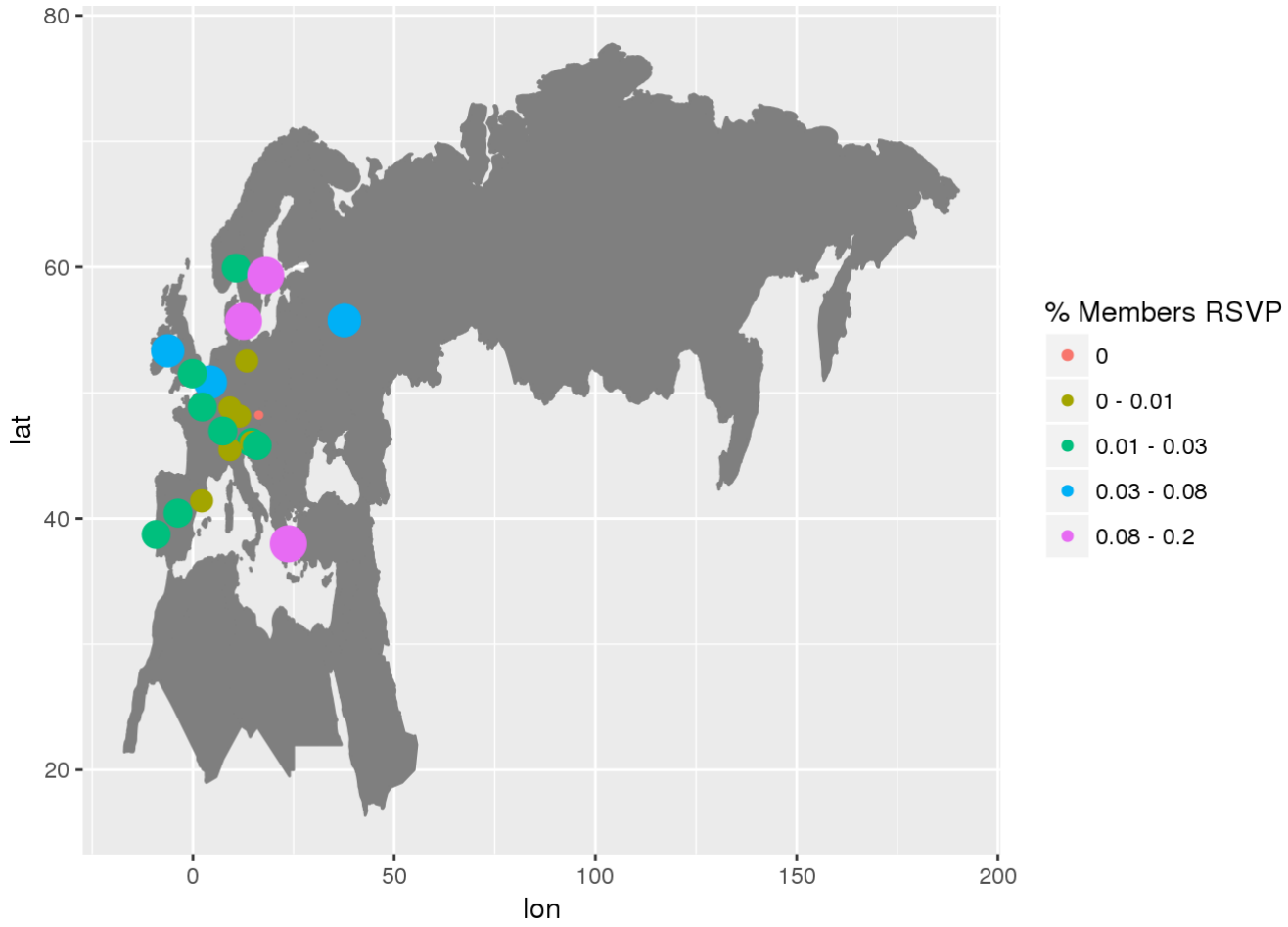
```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_past_events_rsvp_members_top %>% filter(country=="US"),
       aes(x=lon, y=lat, size=rsvp_members_log, colour=reorder(rsvp_members_min_max, r
svp_members_log))) +
  borders("usa", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="% Members RSVP") +
  scale_size_continuous(guide=FALSE)
```


```
ggsave("bdd_rsvps_members_us.png")
```

```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_past_events_rsvp_members_top %>% filter(grepl("^Europe", timezone)),
       aes(x=lon, y=lat, size=rsvp_members_log, color=reorder(rsvp_members_min_max, rs
vp_members_log))) +
  borders("world", colour="gray50", fill="gray50", xlim = c(-10, 40), ylim = c(30,
60)) +
  geom_point() +
  scale_color_discrete(name="% Members RSVP") +
  scale_size_continuous(guide=FALSE)
```

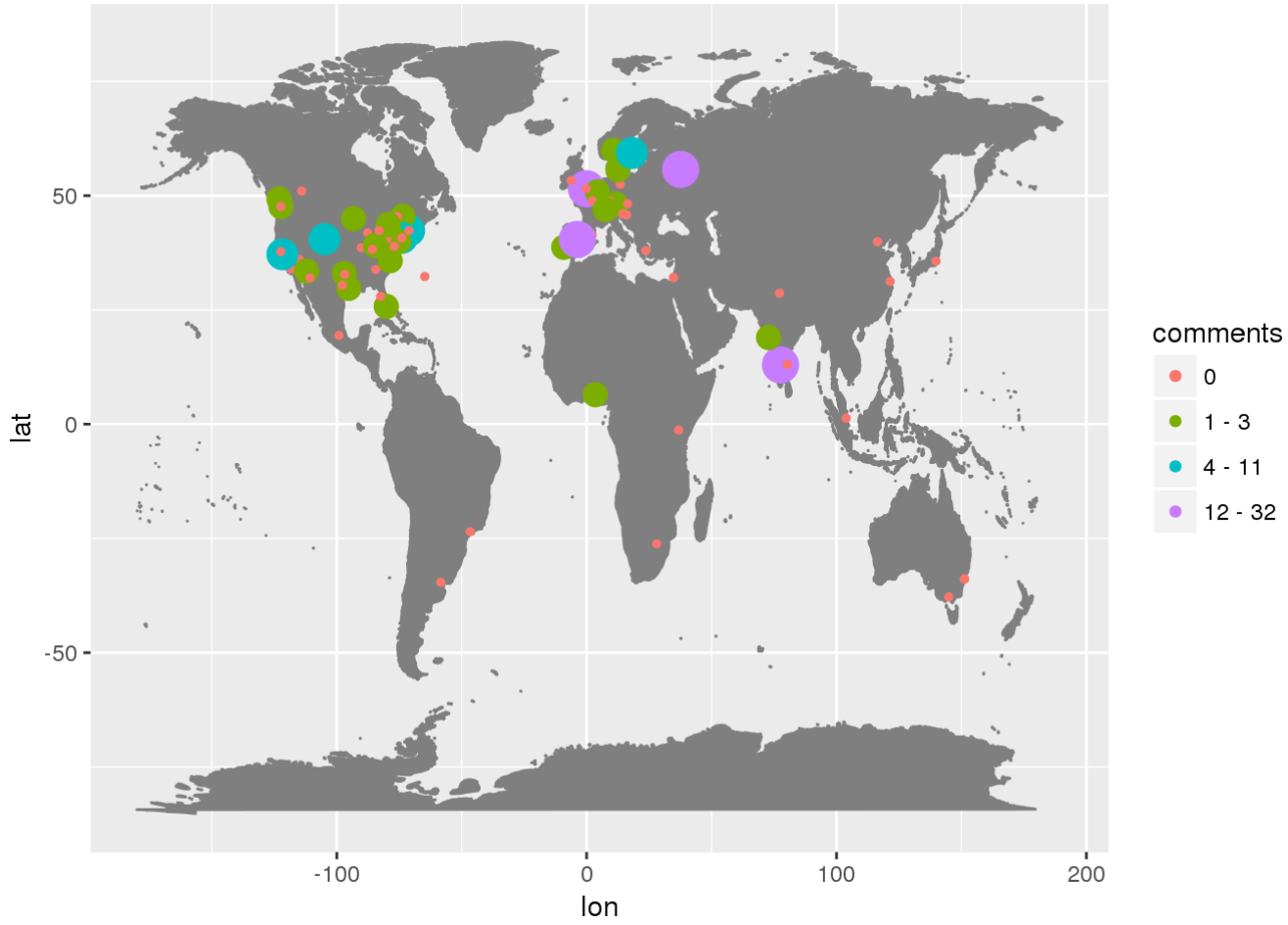



```
ggsave("bdd_rsvps_members_eu.png")
```

```
## Saving 7 x 5 in image
```

How many comments did each event have?

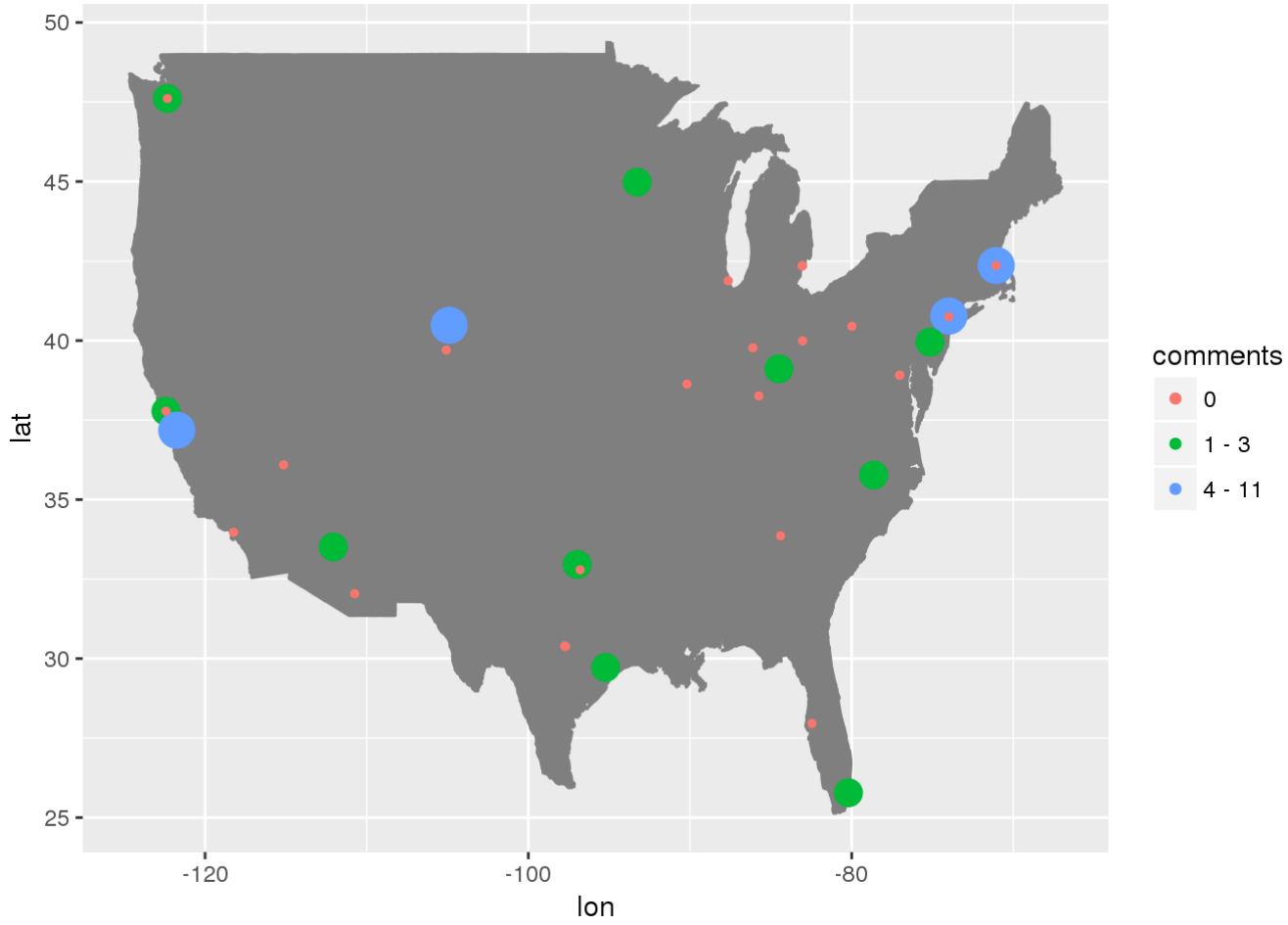
```
# Number of event comments

# TODO: as a proportion of members

bdd_past_events_merge <- bdd_past_events_merge %>%
  mutate(comment_log = round(log(comment_count + 1)), # some have 0
         comment_members = round((comment_count + 1)/members, 2),
         comment_members_log = round(log((comment_count + 1)/members)))

bdd_past_events_comment <- bdd_past_events_merge %>%
  group_by(urlname, comment_log) %>%
  summarise(
    num_events = n(),
    comment_min = min(comment_count),
    comment_max = max(comment_count),
    lon = first(lon),
    lat = first(lat),
    city = first(city),
    country = first(country),
    timezone = first(timezone)
  )

bdd_past_events_comment <- bdd_past_events_comment %>%
  group_by(comment_log) %>%
  mutate(comment_min_max = ifelse(min(comment_min) == max(comment_max),
                                  paste(min(comment_min)),
                                  paste(min(comment_min), "-", max(comment_max))))

bdd_past_events_comment <- bdd_past_events_comment %>%
  group_by(urlname) %>%
  mutate(num_events_max = max(num_events))

# How many comment's did the majority of a group's events have?
bdd_past_events_comment_top <- bdd_past_events_comment %>%
  filter(num_events_max == num_events)

ggplot(data=bdd_past_events_comment_top,
       aes(x=lon, y=lat,
           size=comment_log,
           colour=reorder(comment_min_max, comment_log))) +
  borders("world", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="comments") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_comments.png")
```

```
## Saving 7 x 5 in image
```

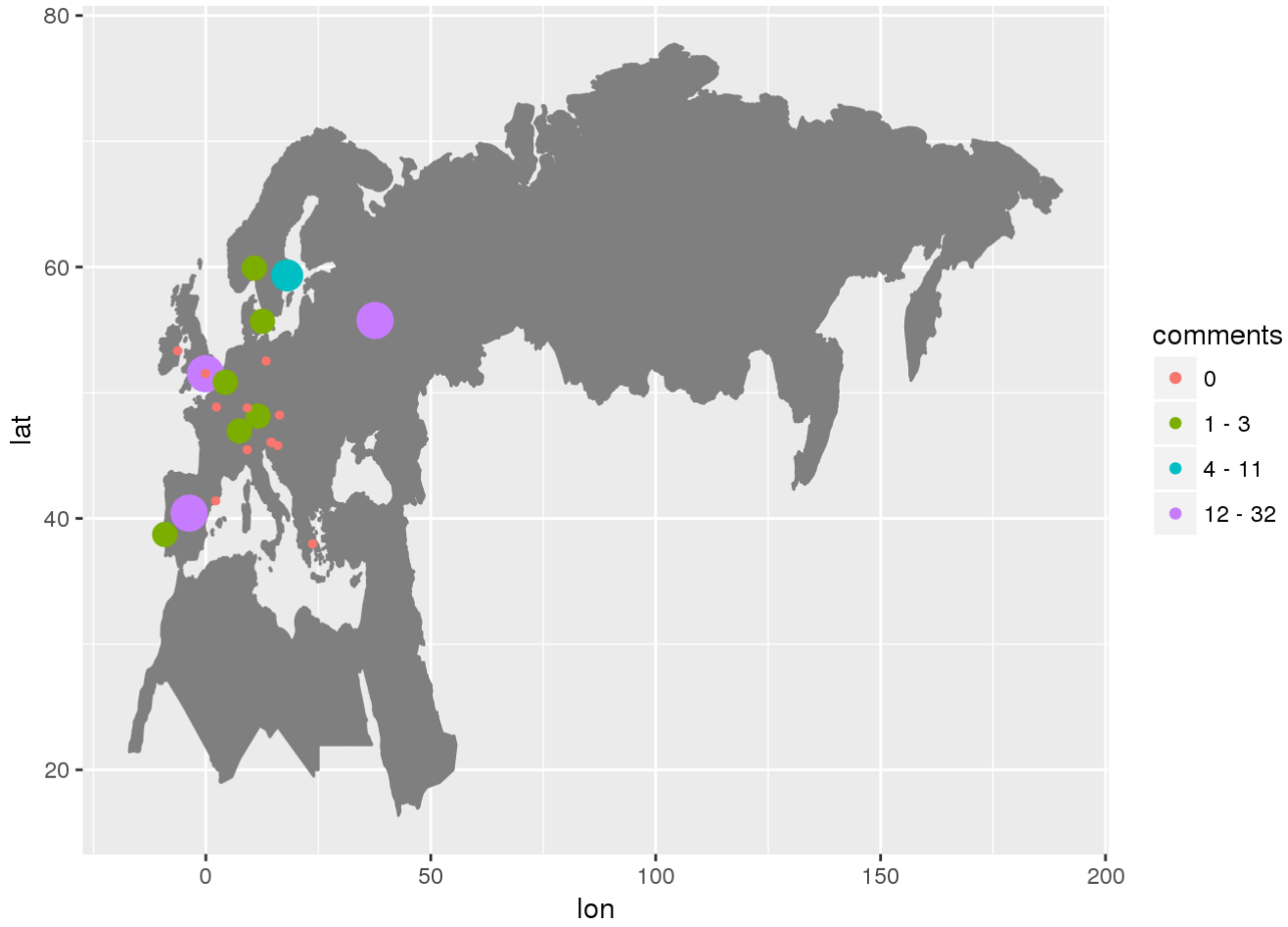
```
ggplot(data=bdd_past_events_comment_top %>% filter(country=="US"),
       aes(x=lon, y=lat, size=comment_log, colour=reorder(comment_min_max,
comment_log))) +
  borders("usa", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="comments") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_comments_us.png")
```

```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_past_events_comment_top %>% filter(grepl("^Europe", timezone)),
       aes(x=lon, y=lat, size=comment_log, color=reorder(comment_min_max,
comment_log))) +
  borders("world", colour="gray50", fill="gray50", xlim = c(-10, 40), ylim = c(30,
60)) +
  geom_point() +
  scale_color_discrete(name="comments") +
  scale_size_continuous(guide=FALSE)
```



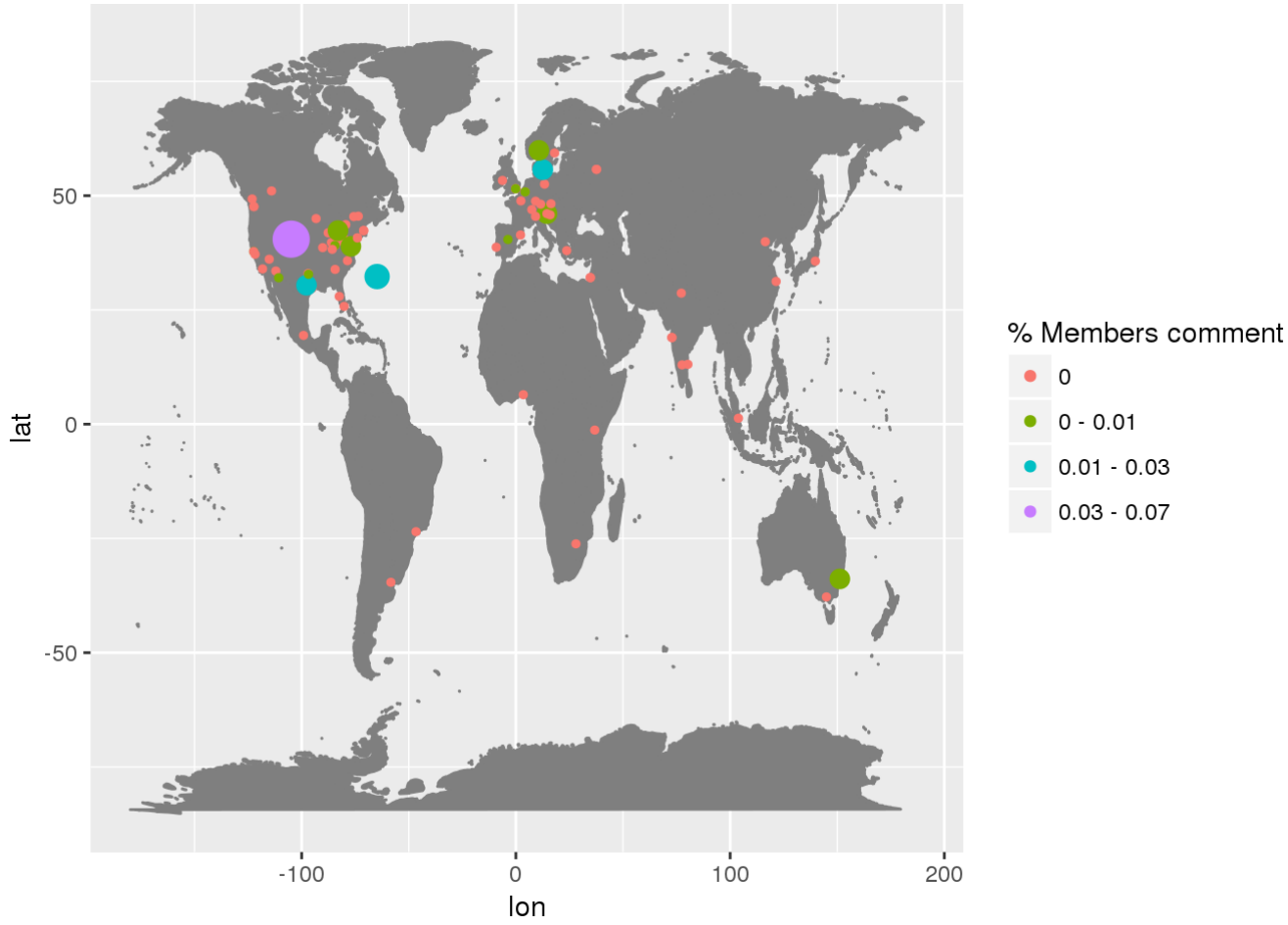

```
ggsave("bdd_comments_eu.png")
```

```
## Saving 7 x 5 in image
```

What proportion of members were represented by the comments?

Right now this only looks at the comment count as if each comment came from a distinct member. Additional work should be done to get the additional comments data and actually compute how many members made comments. Then the members proportion will actually make sense.

```
bdd_past_events_comment_members <- bdd_past_events_merge %>%
  group_by(urlname, comment_members_log) %>%
  summarise(
    num_events = n(),
    comment_members_min = min(comment_members),
    comment_members_max = max(comment_members),
    lon = first(lon),
    lat = first(lat),
    city = first(city),
    country = first(country),
    timezone = first(timezone)
  )

bdd_past_events_comment_members <- bdd_past_events_comment_members %>%
  group_by(comment_members_log) %>%
  mutate(comment_members_min_max = ifelse(min(comment_members_min) == max(comment_memb
ers_max),
                                  paste(min(comment_members_min)),
                                  paste(min(comment_members_min), "-", max(comment_member
s_max))))

bdd_past_events_comment_members <- bdd_past_events_comment_members %>%
  group_by(urlname) %>%
  mutate(num_events_max = max(num_events))

# How many comment's did the majority of a group's events have?
bdd_past_events_comment_members_top <- bdd_past_events_comment_members %>%
  filter(num_events_max == num_events)

ggplot(data=bdd_past_events_comment_members_top,
       aes(x=lon, y=lat,
           size=comment_members_min,
           colour=reorder(comment_members_min_max, comment_members_log))) +
  borders("world", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="% Members comment") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_comments_members.png")
```

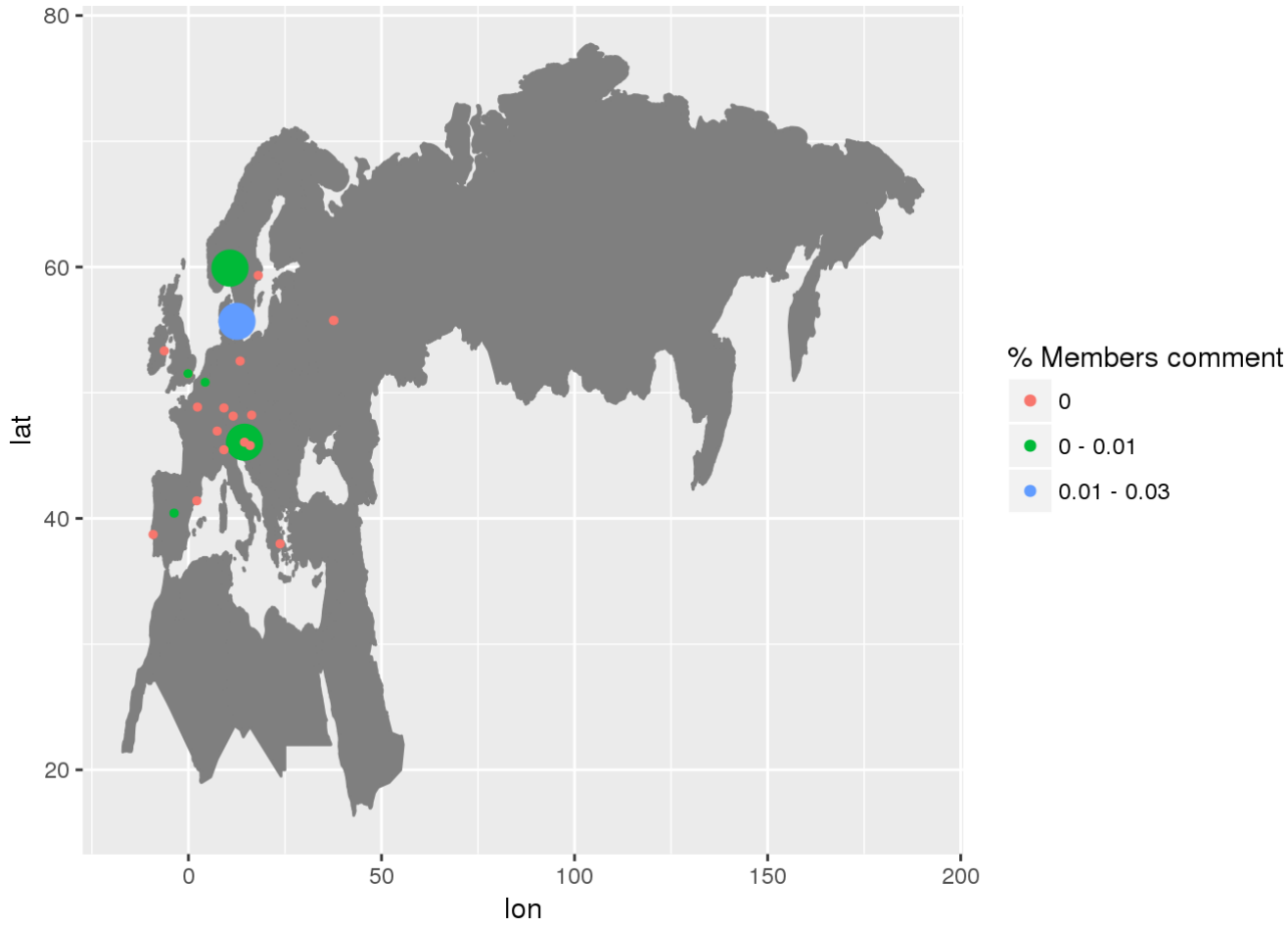
```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_past_events_comment_members_top %>% filter(country=="US"),
       aes(x=lon, y=lat,
           size=comment_members_min,
           colour=reorder(comment_members_min_max, comment_members_log))) +
  borders("usa", colour="gray50", fill="gray50") +
  geom_point() +
  scale_color_discrete(name="% Members comment") +
  scale_size_continuous(guide=FALSE)
```


```
ggsave("bdd_comments_members_us.png")
```

```
## Saving 7 x 5 in image
```

```
ggplot(data=bdd_past_events_comment_members_top %>% filter(grepl("^Europe",
timezone)),
       aes(x=lon, y=lat,
           size=comment_members_min,
           color=reorder(comment_members_min_max, comment_members_log))) +
  borders("world", colour="gray50", fill="gray50", xlim = c(-10, 40), ylim = c(30,
60)) +
  geom_point() +
  scale_color_discrete(name="% Members comment") +
  scale_size_continuous(guide=FALSE)
```




```
ggsave("bdd_comments_members_eu.png")
```

```
## Saving 7 x 5 in image
```