

Germany Data Visualization

Taketo Muroya

2020/2/12

Loading data

```
# Germany hate crime data
devtools::install_github("davben/arvig")

## Skipping install of 'arvig' from a github remote, the SHA1 (18786b90) has not changed since last ins
##   Use `force = TRUE` to force installation

library(arvig)
data("arvig")
arvig = as.data.frame(arvig)

# Germany map
germany_shape_files = readShapePoly(
  "Germany/VG250_1Jan2011_WGS84/VG250_Bundeslaender.shp")

## Warning: readShapePoly is deprecated; use rgdal::readOGR or sf::st_read

# Crime Category and State
unique(arvig$category_en)

## [1] "miscellaneous attack"
## [2] "demonstration"
## [3] "assault"
## [4] "arson"
## [5] "other"
## [6] "demonstration & miscellaneous attack"
## [7] "miscellaneous attack & assault"
## [8] "arson & miscellaneous attack"
## [9] "suspicion"

unique(arvig$state)

## [1] "Berlin"                  "Sachsen"
## [3] "Rheinland-Pfalz"         "Thuringen"
## [5] "Niedersachsen"           "Nordrhein-Westfalen"
## [7] "Brandenburg"             "Bayern"
## [9] "Hessen"                  "Mecklenburg-Vorpommern"
## [11] "Baden-Wurttemberg"        "Sachsen-Anhalt"
## [13] "Schleswig-Holstein"       "Hamburg"
## [15] "Saarland"                "Bremen"
```

Comment: This data has 9 crime categories and 16 states.

Data manipulation

```

# Add monthly and annual variable
a <- mutate(arvig, ym = format(date, '%Y-%m'))
b <- mutate(arvig, y = format(date, '%Y'))

# Aggregate to time, category and state level
d1 = aggregate(a$ym, by=list(a$ym, a$category_en, a$state), FUN=length)
colnames(d1) = c("ym", "category", "state", "number")

# Aggregate to time and category level
d2 = aggregate(a$ym, by=list(a$ym, a$category_en), FUN=length)
colnames(d2) = c("ym", "category", "number")
d2$rate = 0
for (i in 2:nrow(d2)){
  d2$rate[i] = 100*(d2$number[i] - d2$number[i-1])/d2$number[i-1]
}
for (i in c(44,46,93,135,180,188,196)){
  d2$rate[i] = 0
}
d2 <- d2[c(-44,-45,-133,-134,-180:-187),]

# Aggregate to time level
d3 = aggregate(a$ym, by=list(a$ym), FUN=length)
colnames(d3) = c("ym", "number")
d3$rate = 0
for (i in 2:nrow(d3)){
  d3$rate[i] = 100*(d3$number[i] - d3$number[i-1])/d3$number[i-1]
}

# Aggregate to state and category level
d4 = aggregate(d1$number, by=list(d1$category, d1$state), FUN=sum)
colnames(d4) = c("category", "state", "number")
d4 <- d4[c(-11,-18,-22,-25,-27,-54,-61,-77,-84,-93),]

# Aggregate to time and state level
d5 = aggregate(b$y, by=list(b$y, b$state), FUN=length)
colnames(d5) = c("year", "state", "number")
d5$rate = 0
for (i in 2:nrow(d5)){
  d5$rate[i] = 100*(d5$number[i] - d5$number[i-1])/d5$number[i-1]
}
for (i in c(5,9,13,17,20,24,28,32,36,40,44,48,52,56,60)){
  d5$rate[i] = 0
}

# Aggregate to month and state level
d6 = aggregate(a$ym, by=list(a$ym, a$state), FUN=length)
colnames(d6) = c("ym", "state", "number")
d6$rate = 0
for (i in 2:nrow(d6)){
  d6$rate[i] = 100*(d6$number[i] - d6$number[i-1])/d6$number[i-1]
}
for (i in c(46,92,135,179,191,222,257,301,336,380,417,485,525,558)){
  d6$rate[i] = 0
}

```

```

}

# Aggregate to state level
d7 = aggregate(a$date, by=list(a$state), FUN=length)
colnames(d7) = c("state", "number")
others <- read.csv('others.csv')
d7 <- cbind(d7, others)
d7 <- d7[,-3]
d7$crime_pc <- 1000 * d7$number / d7$population

# Aggregate to monthly state level
d8 = aggregate(a$date, by=list(a$ym, a$state), FUN=length)
colnames(d8) = c("ym", "state", "number")

# Aggregate to time, category and state level
d9 = aggregate(arvig$date, by=list(arvig$date, arvig$state), FUN=length)
colnames(d9) = c("date", "state", "number")

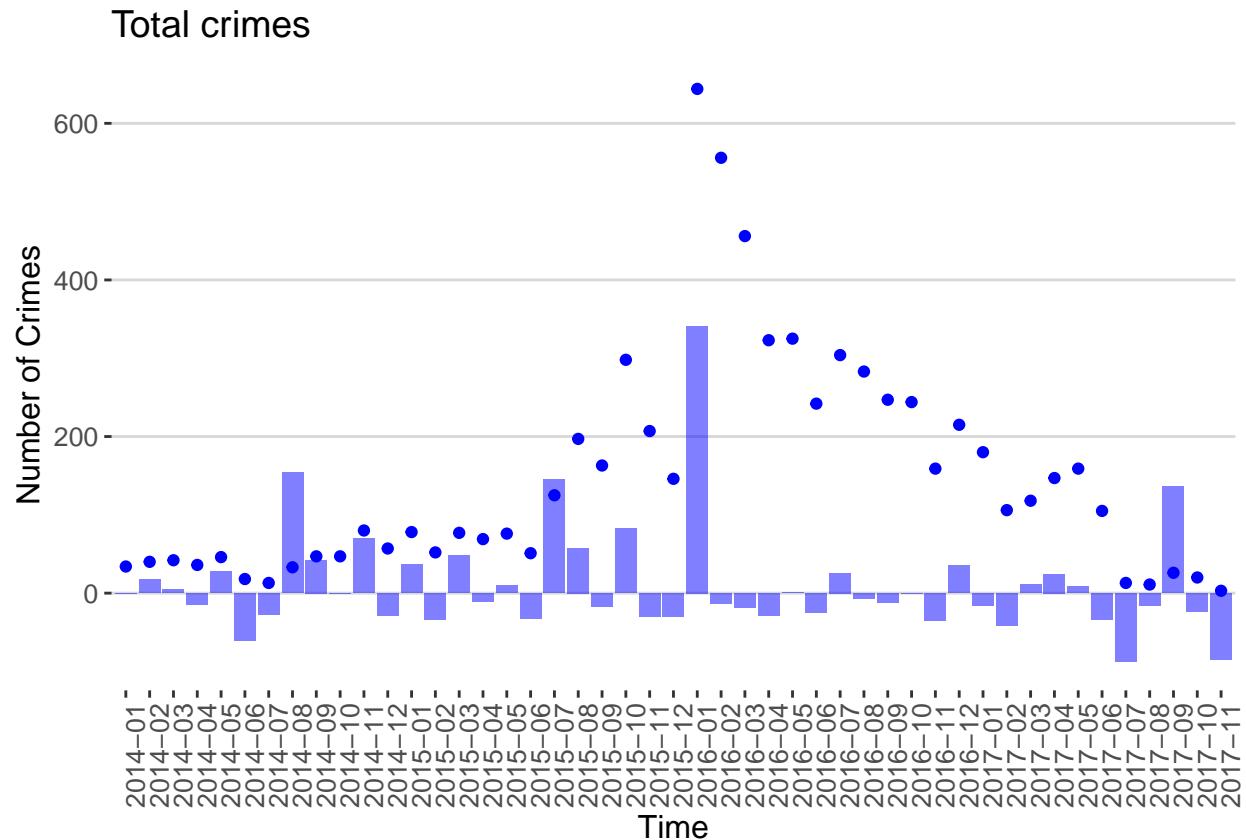
```

Total crime graphs

```

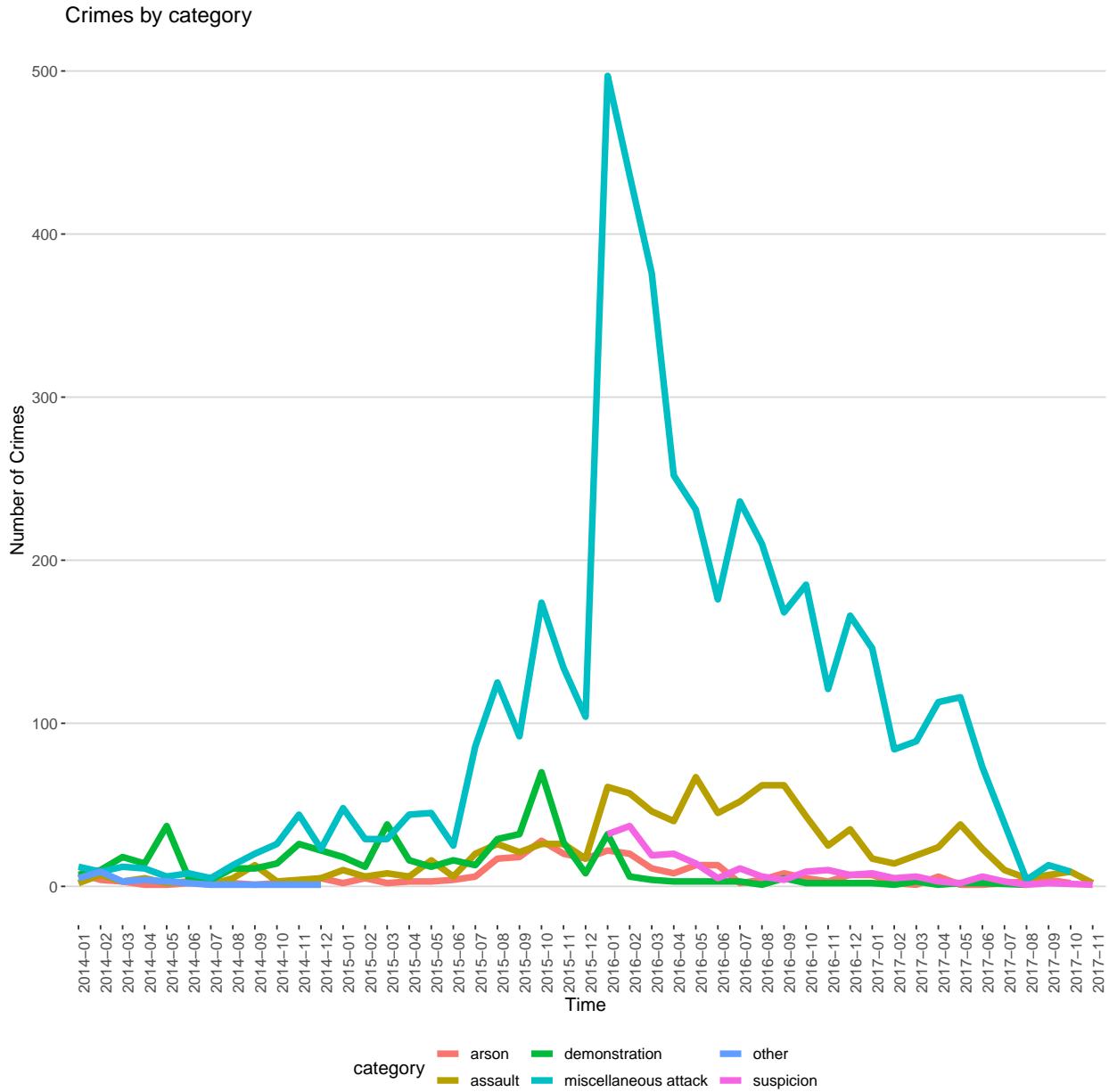
# Total
total1 <- ggplot(d3) +
  geom_bar(aes(x=ym, y=rate), fill="blue",
           stat="identity", position="identity", alpha=0.5) +
  geom_point(aes(x=ym, y=number),
             stat="identity", position="identity", colour="blue") +
  theme(legend.position = "top") +
  labs(x="Time", y="Number of Crimes") + ggtitle("Total crimes") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
total1

```



Comment: The trend totally changed before and after Jan-2016.

```
total2 <- ggplot(d2, aes(x = ym, y = number)) +
  geom_line(aes(group=category, color=category), size=2) +
  theme(legend.position = "top") +
  labs(x="Time", y="Number of Crimes") +
  ggtitle("Crimes by category") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
total2
```



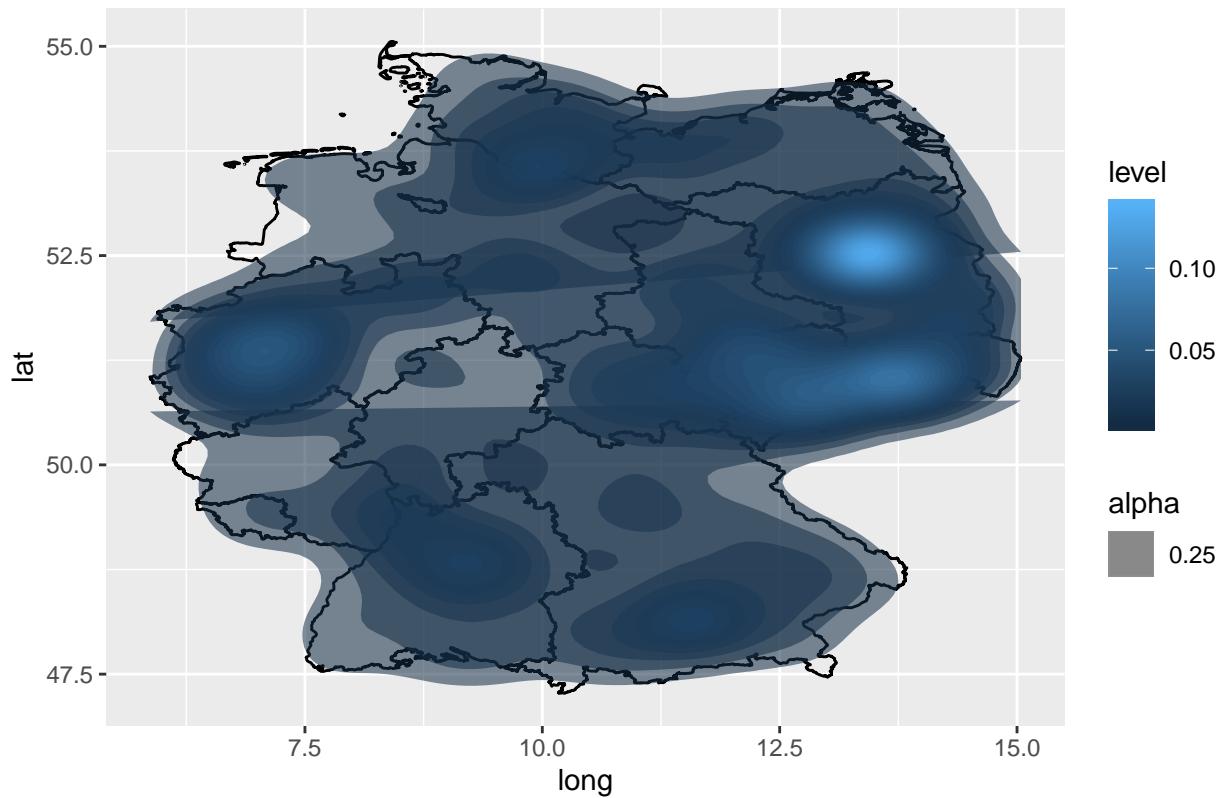
Comment: Most of data are from 'Miscellaneous Attack' category.

```
total2.5 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = b, geom = "polygon") +
  ggtitle("Total crimes: 2014-2017")

## Regions defined for each Polygons
```

```
total2.5
```

Total crimes: 2014–2017

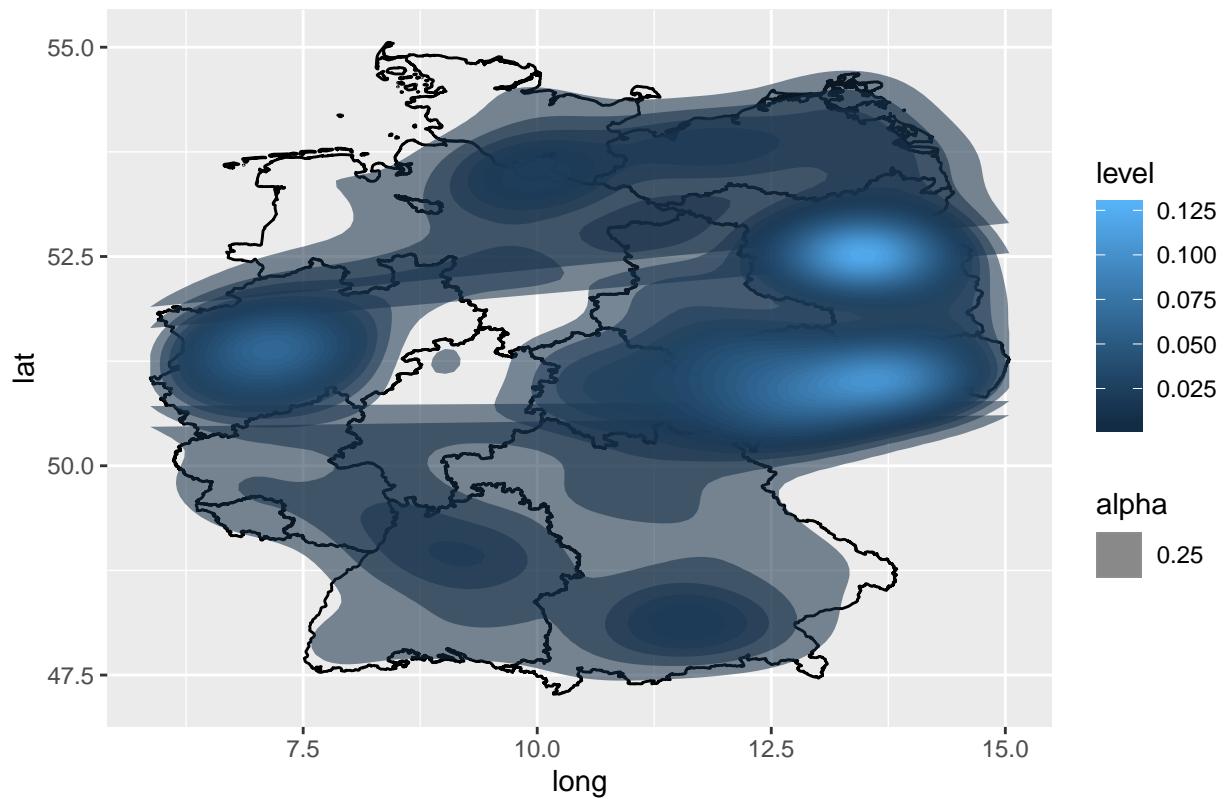


```
total3 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(b, y <= 2015),
    geom = "polygon") + ggttitle("Total crimes: 2014 & 2015")
```

```
## Regions defined for each Polygons
```

```
total3
```

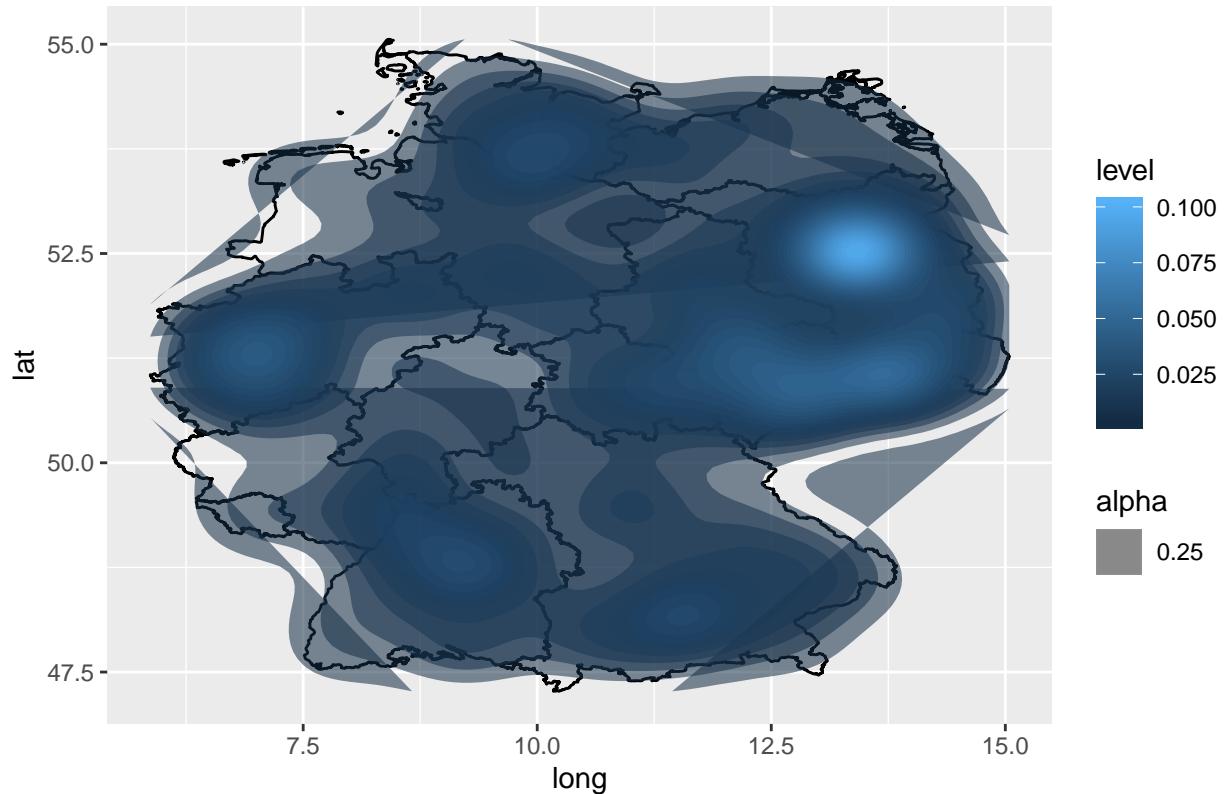
Total crimes: 2014 & 2015



```
total4 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(b, y >= 2016),
    geom = "polygon") + ggttitle("Total crimes: 2016 & 2017")

## Regions defined for each Polygons
total4
```

Total crimes: 2016 & 2017



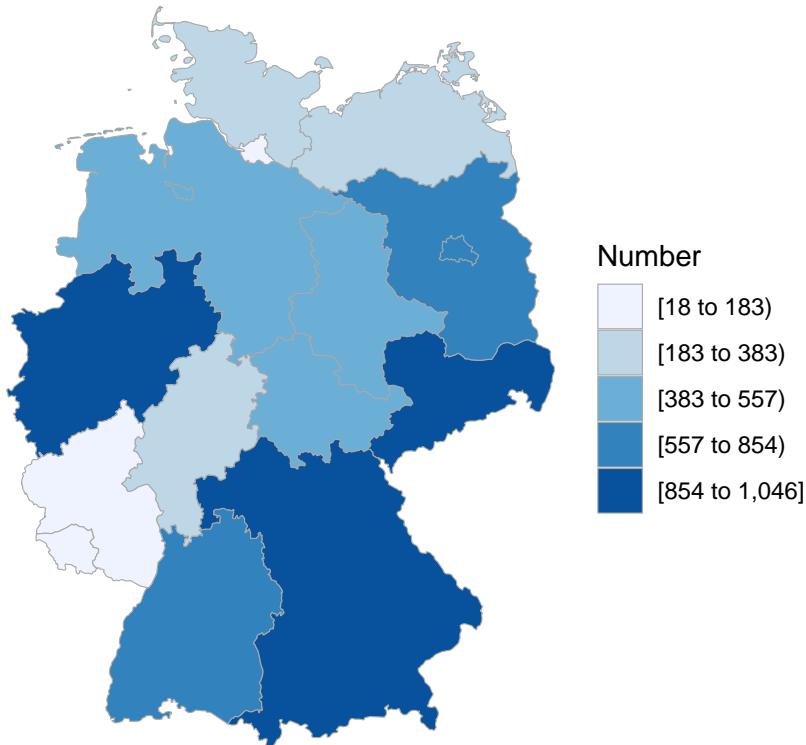
Comment: I divided heat mapping before and after Jan-2016. The crime may expand to broader area after Jan-2016. We can change the time period according to remarkable events.

```
state_trans <- read.csv('state_trans.csv')
d5$region <- state_trans$region

all = aggregate(d5$number, by=list(d5$region), FUN=sum)
colnames(all) = c("region", "number")
totalall <- data.frame(region=all$region, value=all$number)
total4.5 <-
  admin1_choropleth(country.name = "germany", df = totalall,
                     title="Total Number of Crimes by State",
                     legend = "Number", num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total4.5
```

Total Number of Crimes by State



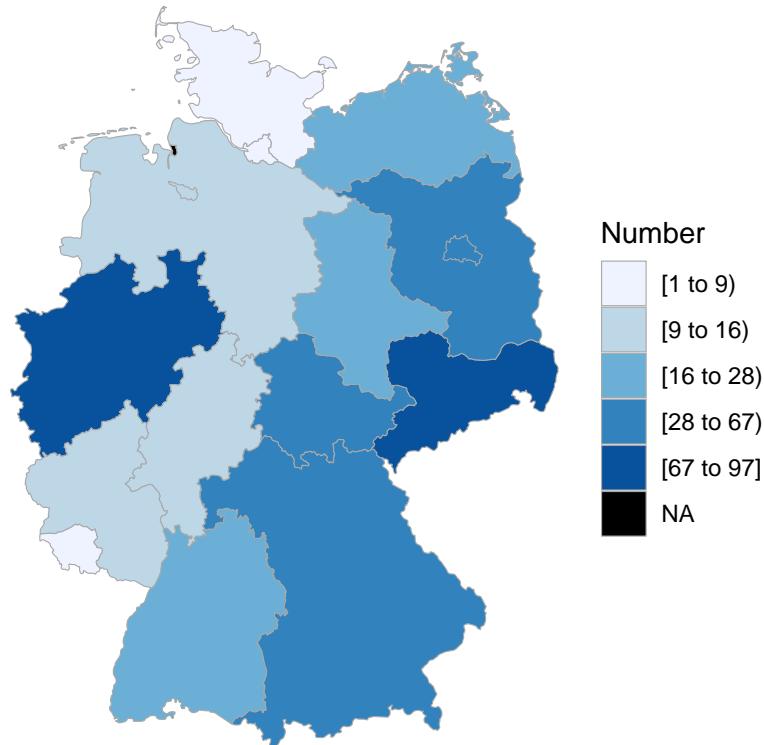
```
total2014 <- data.frame(region=subset(d5, year==2014)[,5],
                           value=subset(d5, year==2014)[,3])
total5 <-
  admin1_choropleth(country.name = "germany", df = total2014,
                     title="Total Number of Crimes by State (2014)",
                     legend = "Number", num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector

## Warning in self$bind(): The following regions were missing and are being
## set to NA: bremen

total5
```

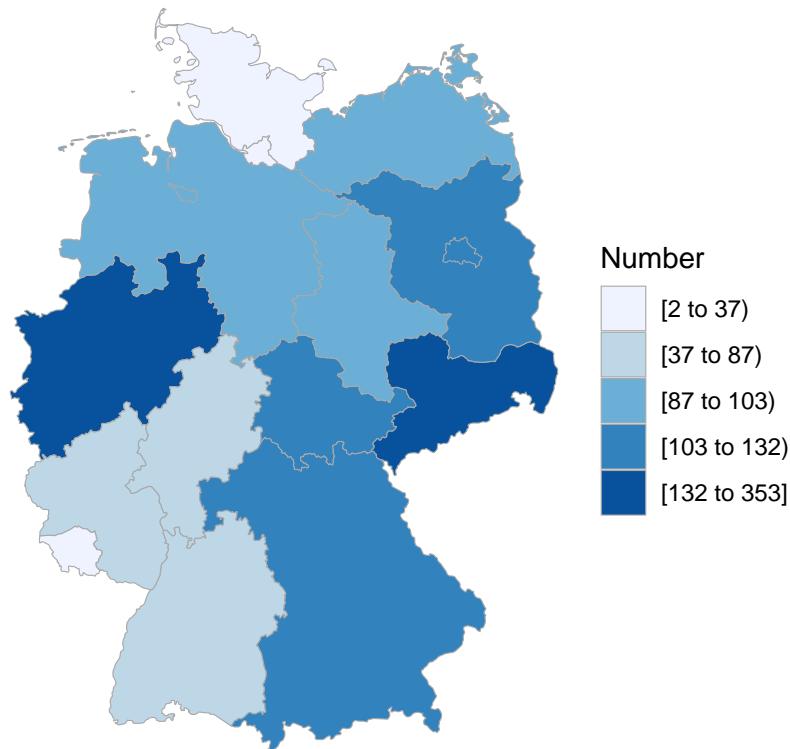
Total Number of Crimes by State (2014)



```
total2015 <- data.frame(region=subset(d5, year==2015)[,5],
                           value=subset(d5, year==2015)[,3])
total6 <-
  admin1_choropleth(country.name = "germany", df = total2015,
                     title="Total Number of Crimes by State (2015)",
                     legend = "Number", num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total6
```

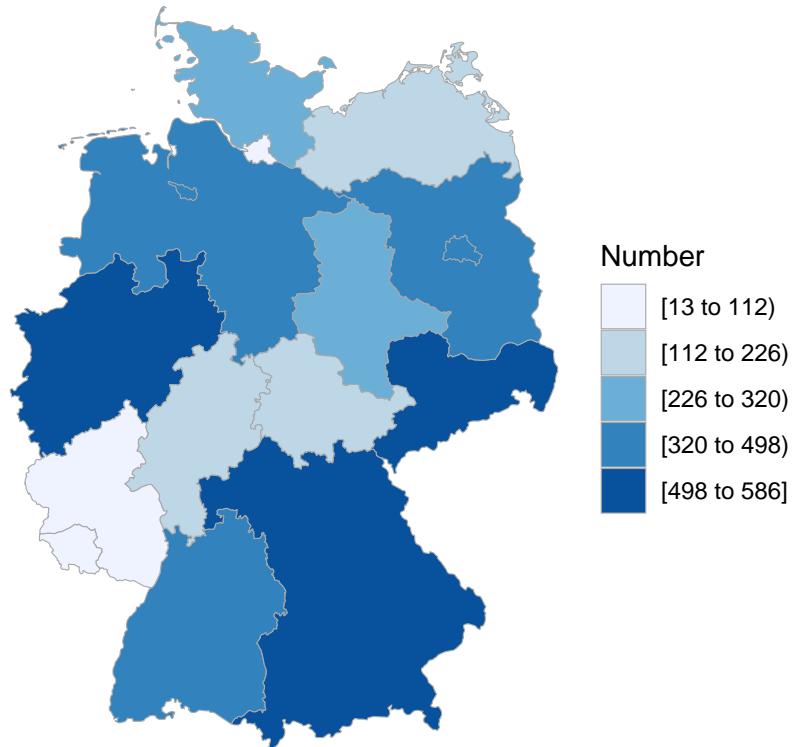
Total Number of Crimes by State (2015)



```
total2016 <- data.frame(region=subset(d5, year==2016)[,5],
                           value=subset(d5, year==2016)[,3])
total7 <-
  admin1_choropleth(country.name = "germany", df = total2016,
                     title="Total Number of Crimes by State (2016)",
                     legend = "Number", num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total7
```

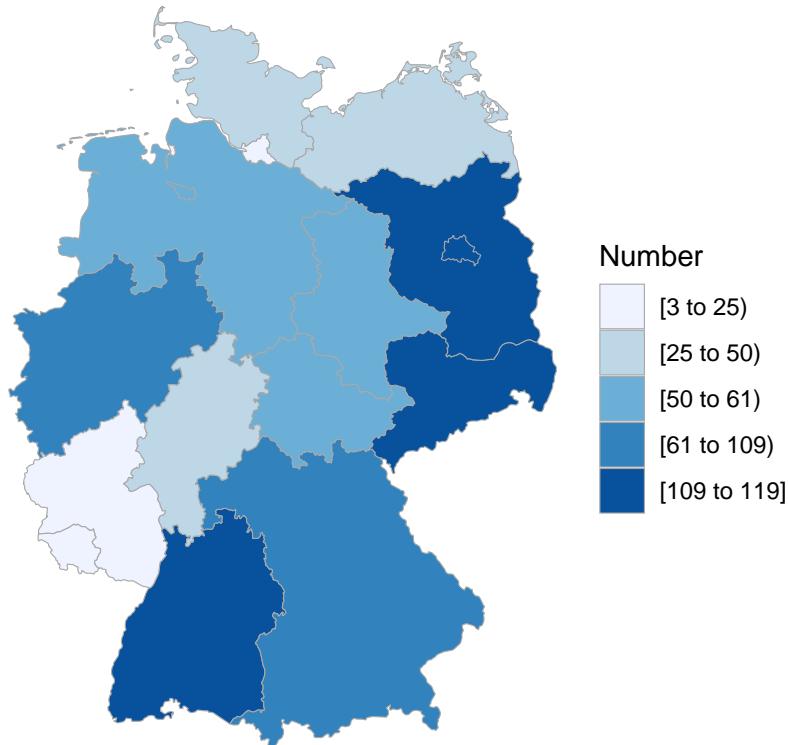
Total Number of Crimes by State (2016)



```
total2017 <- data.frame(region=subset(d5, year==2017)[,5],
                           value=subset(d5, year==2017)[,3])
total8 <-
  admin1_choropleth(country.name = "germany", df = total2017,
                     title="Total Number of Crimes by State (2017)",
                     legend = "Number", num_colors = 5)

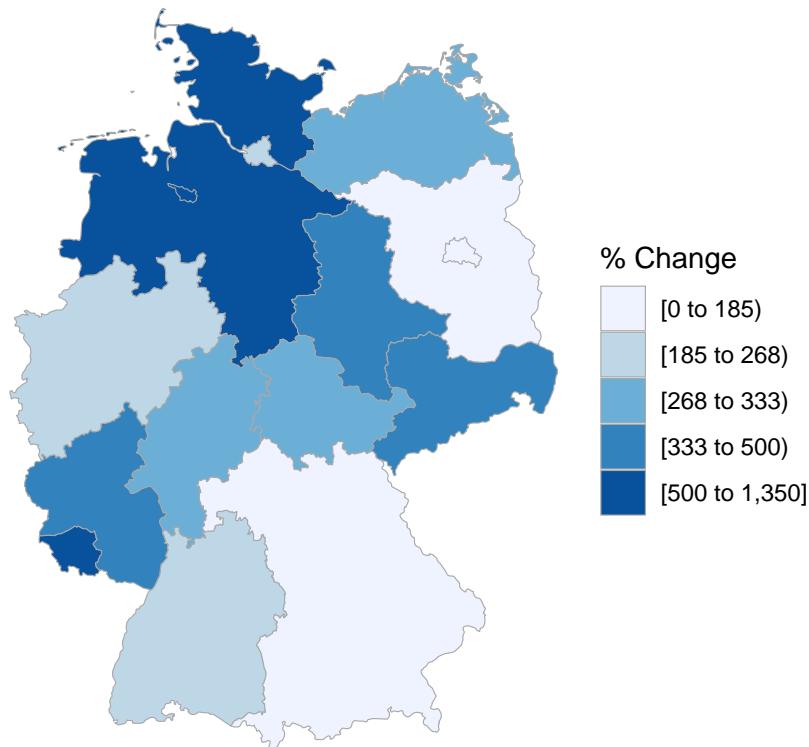
## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total8
```

Total Number of Crimes by State (2017)



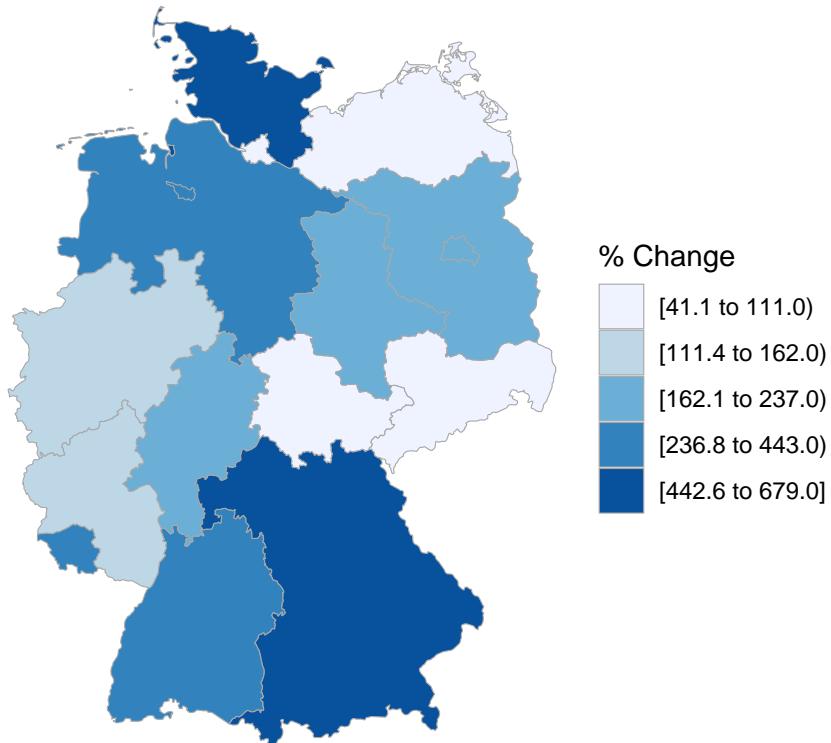
```
total2015r <- data.frame(region=subset(d5, year==2015)[,5],  
                           value=subset(d5, year==2015)[,4])  
total9 <-  
  admin1_choropleth(country.name = "germany", df = total2015r,  
                     title="Total Change of Crimes by State (2015)",  
                     legend = "% Change", num_colors = 5)  
  
## Warning: Column `region` joining character vector and factor, coercing into  
## character vector  
total9
```

Total Change of Crimes by State (2015)



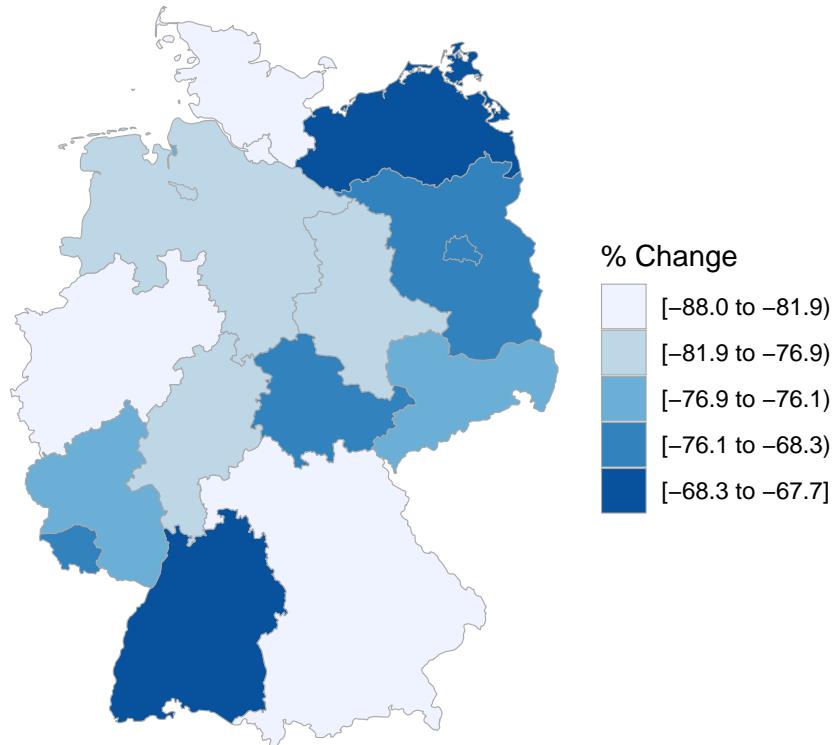
```
total2016r <- data.frame(region=subset(d5, year==2016)[,5],  
                           value=subset(d5, year==2016)[,4])  
total10 <-  
  admin1_choropleth(country.name = "germany", df = total2016r,  
                     title="Total Change of Crimes by State (2016)",  
                     legend = "% Change", num_colors = 5)  
  
## Warning: Column `region` joining character vector and factor, coercing into  
## character vector  
total10
```

Total Change of Crimes by State (2016)



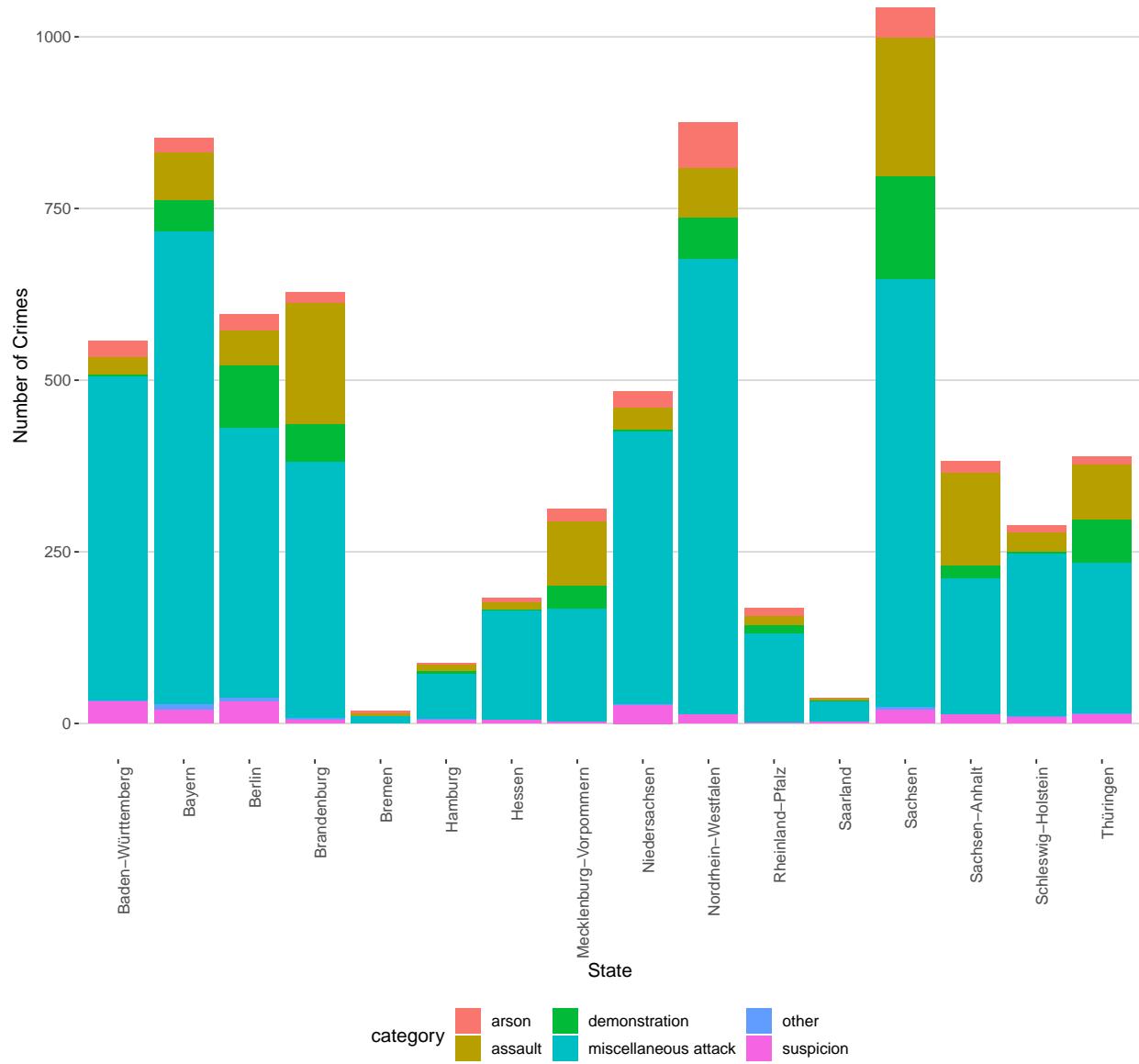
```
total2017r <- data.frame(region=subset(d5, year==2017)[,5],  
                           value=subset(d5, year==2017)[,4])  
total11 <-  
  admin1_choropleth(country.name = "germany", df = total2017r,  
                     title="Total Change of Crimes by State (2017)",  
                     legend = "% Change", num_colors = 5)  
  
## Warning: Column `region` joining character vector and factor, coercing into  
## character vector  
total11
```

Total Change of Crimes by State (2017)



```
total12 <- ggplot(d4) +  
  geom_bar(aes(x=state, y=number, group=category, fill=category),  
           stat="identity") + theme(legend.position = "top") +  
  labs(x="State", y="Number of Crimes") +  
  ggtitle("Crimes by State") +  
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))  
total12
```

Crimes by State



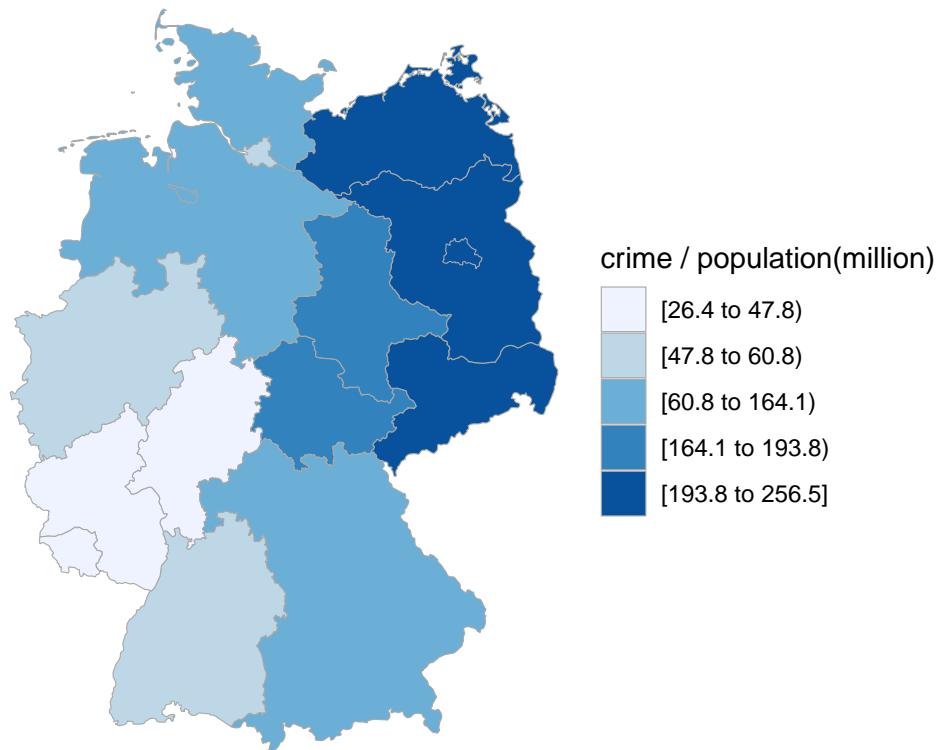
Crime per capita graphs

```
state_trans <- read.csv('state_trans2.csv')
d7$region <- state_trans$region

crime_per_capita <- data.frame(region=d7[,7], value=d7[,6])
total13 <-
  admin1_choropleth(country.name = "germany", df = crime_per_capita,
                     title="Refugee Crimes per population by State",
                     legend = "crime / population(million)",
                     num_colors = 5)
```

```
## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total13
```

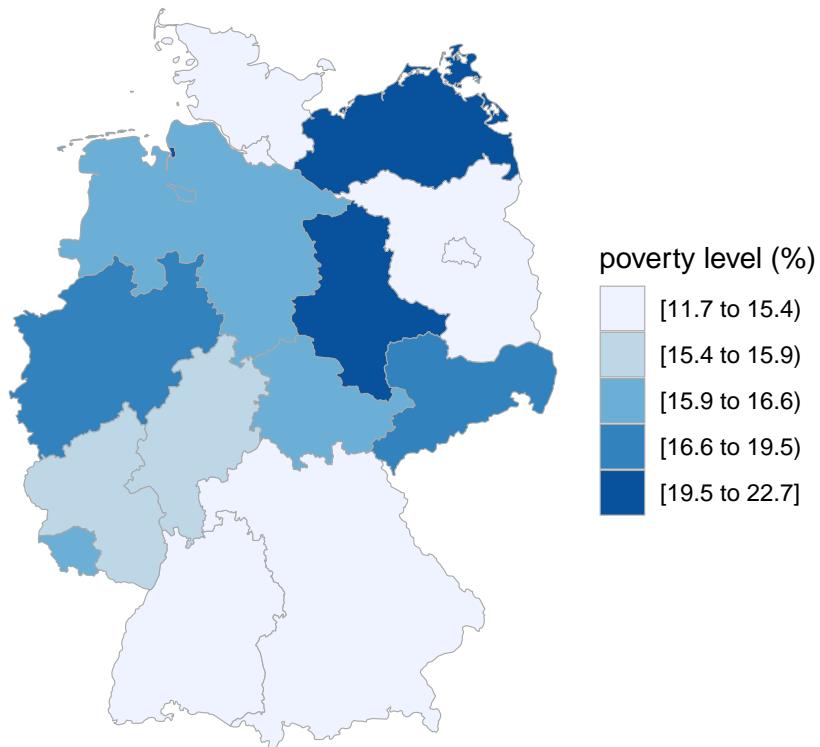
Refugee Crimes per population by State



```
poverty <- data.frame(region=d7[,7], value=d7[,4])
total14 <-
  admin1_choropleth(country.name = "germany", df = poverty,
                     title="Poverty level by State",
                     legend = "poverty level (%)",
                     num_colors = 5)
```

```
## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total14
```

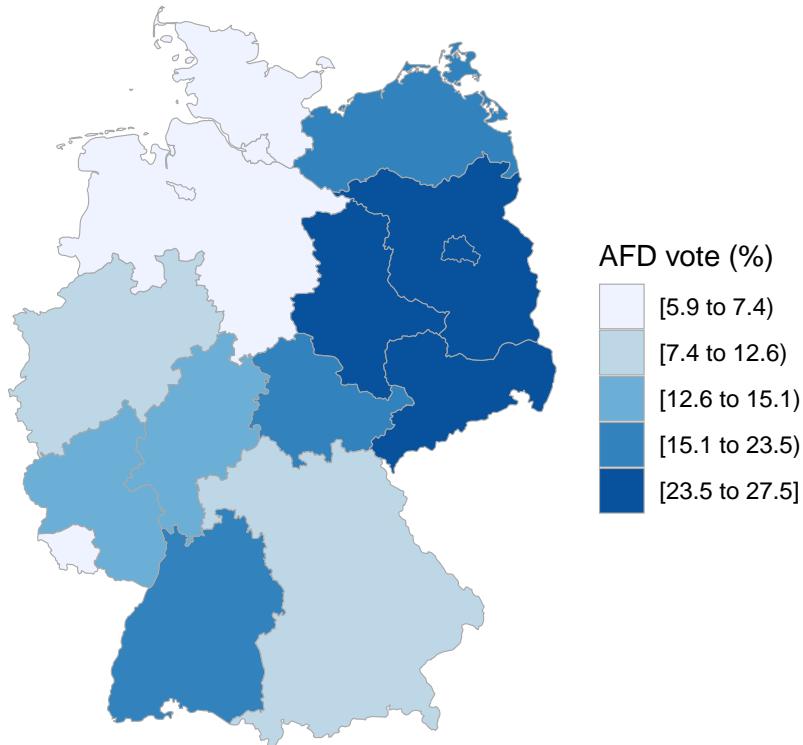
Poverty level by State



```
AFD <- data.frame(region=d7[,7], value=d7[,5])
total15 <-
  admin1_choropleth(country.name = "germany", df = AFD,
                     title="AFD vote by State",
                     legend = "AFD vote (%)",
                     num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total15
```

AFD vote by State

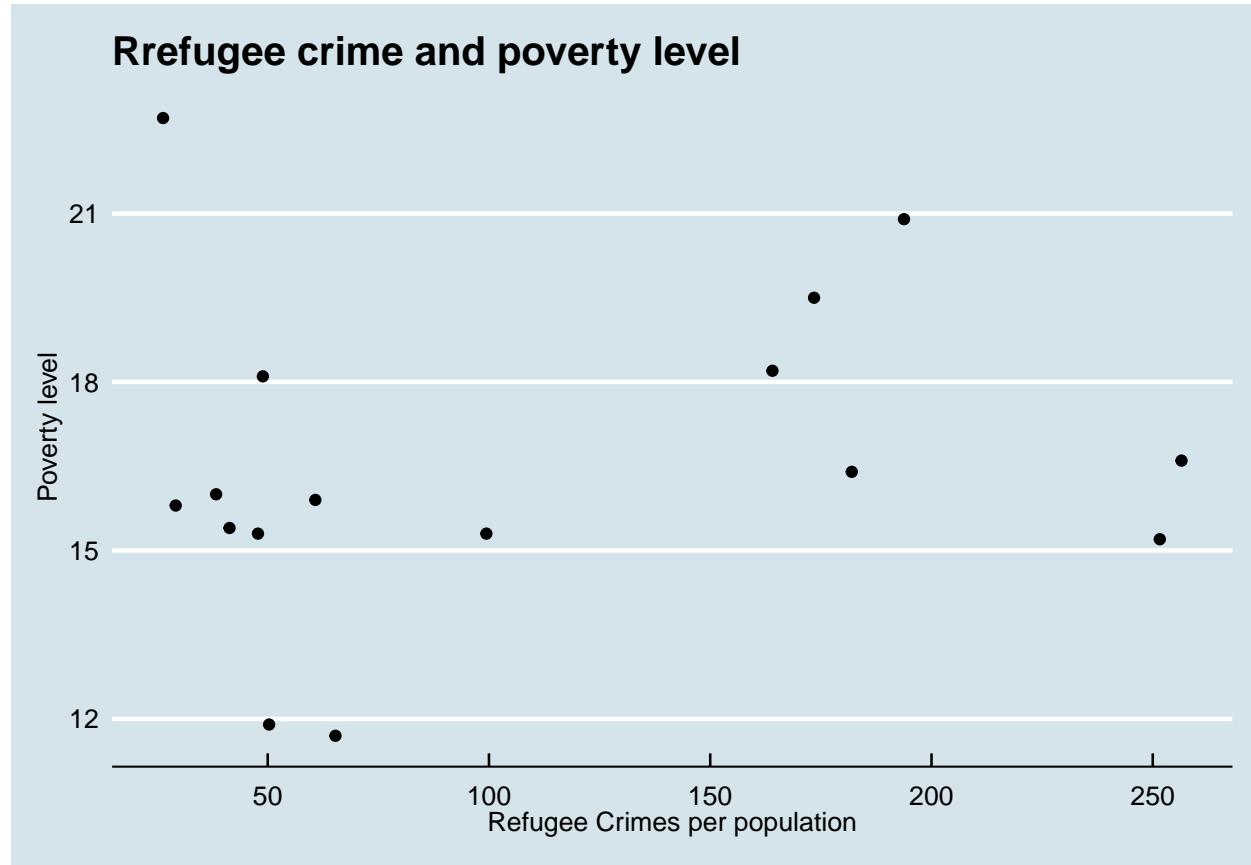


```
result <- lm(crime_pc ~ poverty + AFD, data=d7)
summary(result)

##
## Call:
## lm(formula = crime_pc ~ poverty + AFD, data = d7)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -69.507 -30.018   2.721  24.448  67.086 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -63.043    67.725  -0.931   0.369    
## poverty      2.751     3.965   0.694   0.500    
## AFD          9.029     1.480   6.101 3.77e-05 *** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 43.81 on 13 degrees of freedom
## Multiple R-squared:  0.7496, Adjusted R-squared:  0.7111 
## F-statistic: 19.46 on 2 and 13 DF,  p-value: 0.0001233

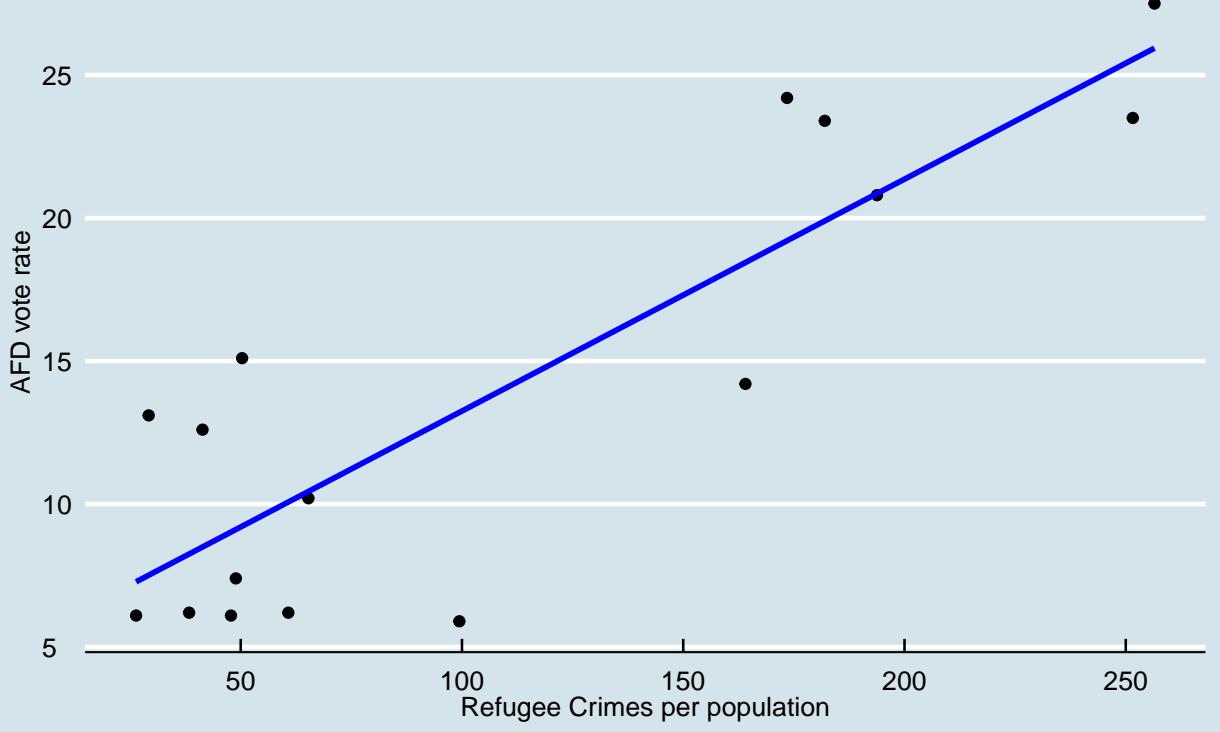
total16 <- ggplot(data = d7, aes(x = crime_pc, y = poverty)) +
  geom_point() +
  xlab("Refugee Crimes per population") + ylab("Poverty level") +
  theme_economist() + theme(legend.position="none") +
```

```
ggtitle("Rrefugee crime and poverty level")
total16
```



```
total17 <- ggplot(data = d7, aes(x = crime_pc, y = AFD)) +
  geom_point() + geom_smooth(color="blue", lwd=1, se=FALSE, method=lm) +
  xlab("Refugee Crimes per population") + ylab("AFD vote rate") +
  theme_economist() + theme(legend.position="none") +
  ggtitle("Rrefugee crime and AFD vote")
total17
```

Rrefugee crime and AFD vote



1. December 2015 VS January 2016

```
data201511 = subset(d8, ym=="2015-11")
others <- read.csv('others.csv')
others <- others[-5,]
others <- others[-11,]
data201511 <- cbind(data201511, others)
data201511 <- data201511[,-4]
data201511$crime_pc <- 1000 * data201511$number / data201511$population
state_trans <- read.csv('state_trans2.csv')
state_trans <- state_trans[-5,]
state_trans <- state_trans[-11,]
data201511$region <- state_trans$region

data201512 = subset(d8, ym=="2015-12")
others <- read.csv('others.csv')
others <- others[-5,]
data201512 <- cbind(data201512, others)
data201512 <- data201512[,-4]
data201512$crime_pc <- 1000 * data201512$number / data201512$population
state_trans <- read.csv('state_trans2.csv')
state_trans <- state_trans[-5,]
data201512$region <- state_trans$region
```

```

data201601 = subset(d8, ym=="2016-01")
others <- read.csv('others.csv')
data201601 <- cbind(data201601, others)
data201601 <- data201601[,-4]
data201601$crime_pc <- 1000 * data201601$number / data201601$population
state_trans <- read.csv('state_trans2.csv')
data201601$region <- state_trans$region

mix1 <- rbind(data201511, data201512, data201601)
mix1 <- mix1[order(mix1$state),]
mix1$rate = 0
for (i in 2:nrow(mix1)){
  mix1$rate[i] =
    100*(mix1$crime_pc[i] - mix1$crime_pc[i-1])/mix1$crime_pc[i-1]
}

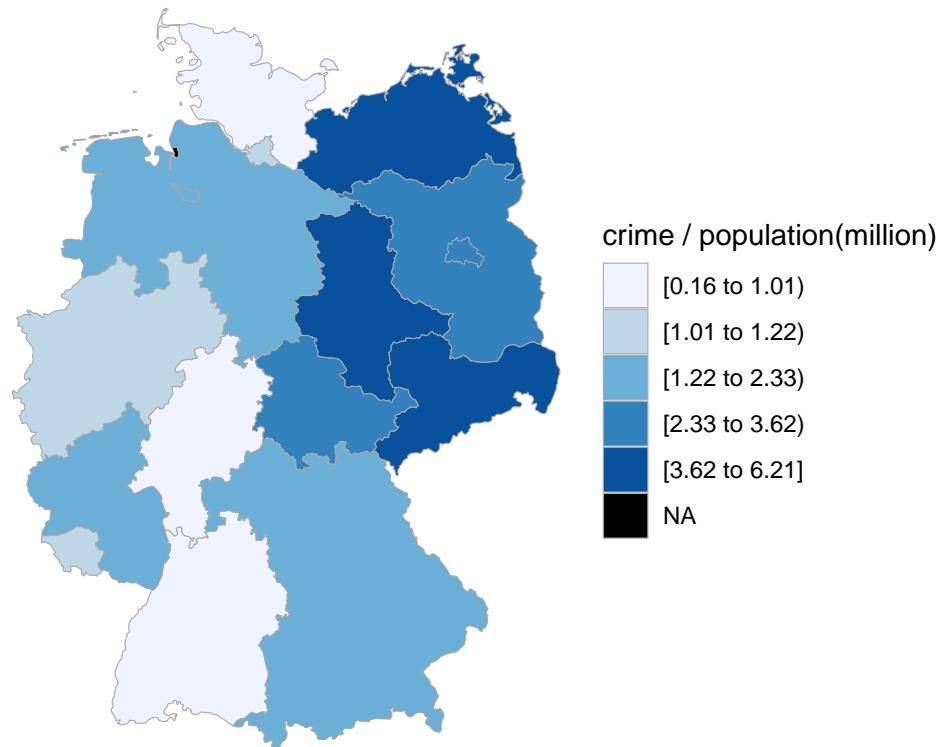
crime201512 <- data.frame(
  region=subset(mix1, ym=="2015-12")[,8],
  value=subset(mix1, ym=="2015-12")[,7])
total18 <-
  admin1_choropleth(country.name = "germany", df = crime201512,
                     title="Crimes per population by State (Dec.2015)",
                     legend = "crime / population(million)",
                     num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector

## Warning in self$bind(): The following regions were missing and are being
## set to NA: bremen
total18

```

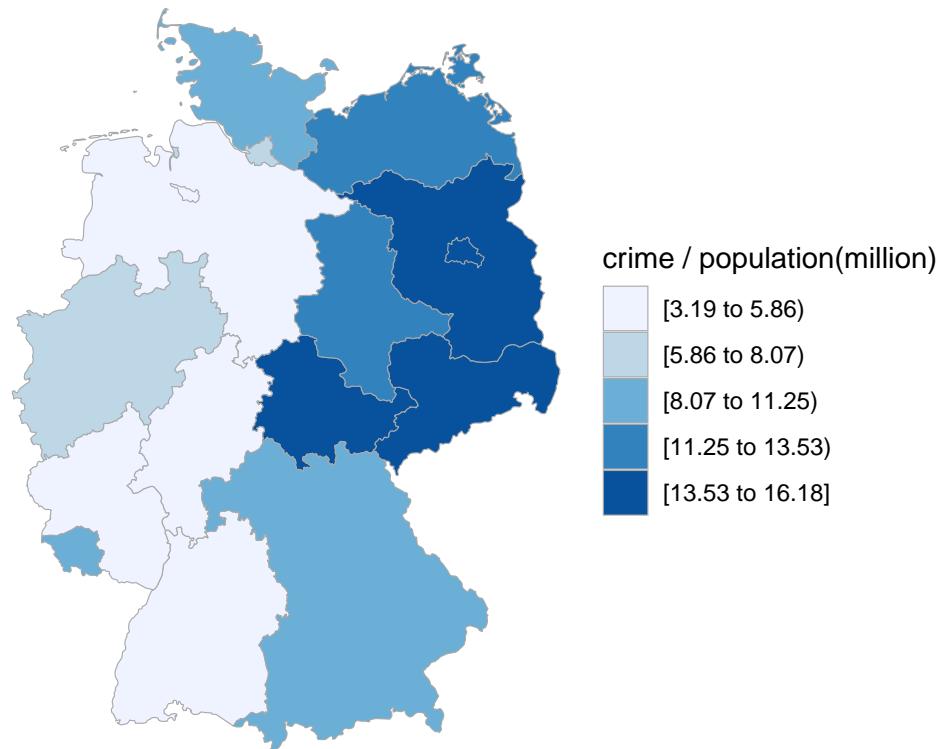
Crimes per population by State (Dec.2015)



```
crime201601 <- data.frame(
  region=subset(mix1, ym=="2016-01")[,8],
  value=subset(mix1, ym=="2016-01")[,7])
total19 <-
  admin1_choropleth(country.name = "germany", df = crime201601,
                     title="Crimes per population by State (Jan.2016)",
                     legend = "crime / population(million)",
                     num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total19
```

Crimes per population by State (Jan.2016)

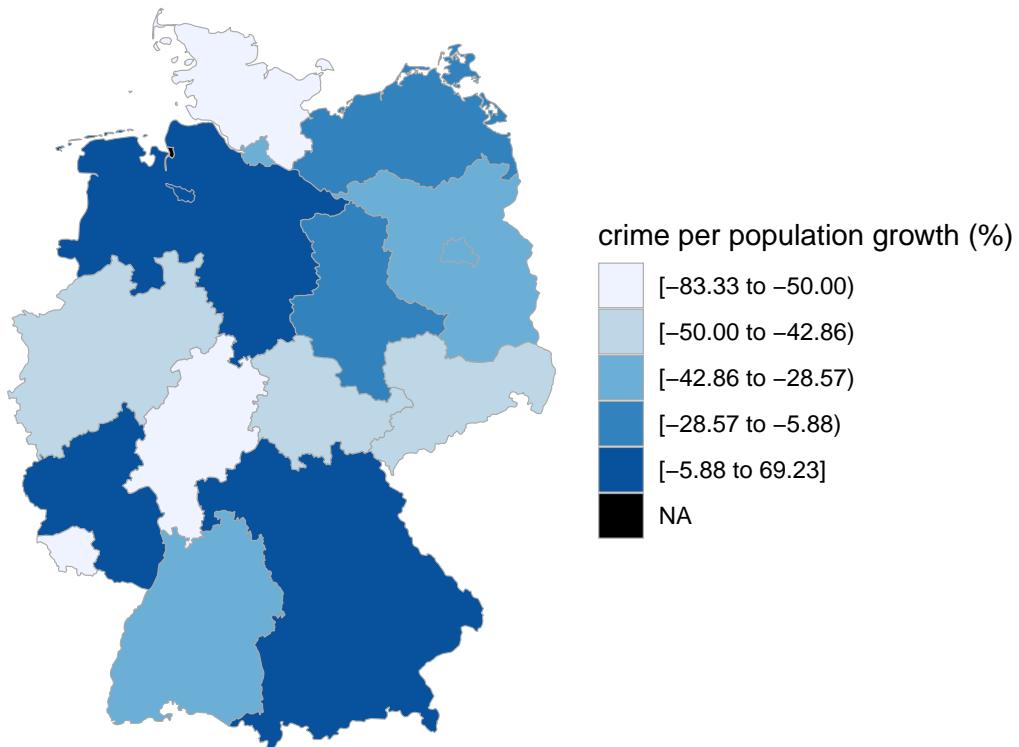


```
rate201512 <- data.frame(
  region=subset(mix1, ym=="2015-12")[,8],
  value=subset(mix1, ym=="2015-12")[,9])
total20 <-
  admin1_choropleth(country.name = "germany", df = rate201512,
                     title="Crimes per population growth by State (Dec.2015)",
                     legend = "crime per population growth (%)",
                     num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector

## Warning: The following regions were missing and are being set to NA: bremen
total20
```

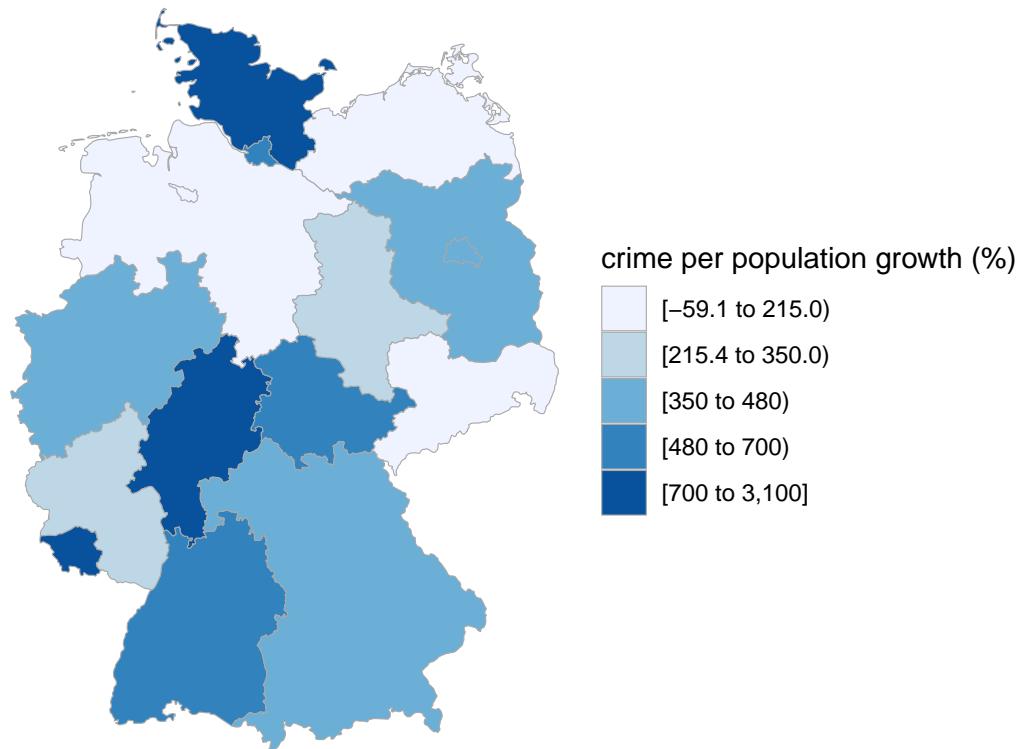
Crimes per population growth by State (Dec.2015)



```
rate201601 <- data.frame(
  region=subset(mix1, ym=="2016-01")[,8],
  value=subset(mix1, ym=="2016-01")[,9])
total21 <-
  admin1_choropleth(country.name = "germany", df = rate201601,
                     title="Crimes per population growth by State (Jan.2016)",
                     legend = "crime per population growth (%)",
                     num_colors = 5)

## Warning: Column `region` joining character vector and factor, coercing into
## character vector
total21
```

Crimes per population growth by State (Jan.2016)



2. July 5, 2016 - July 18th, 2016 VS Juli, 19 2016 - August 2, 2016

```

aaa = subset(d9, date >= "2016-07-05")
bbb = subset(aaa, date <= "2016-07-18")
before2 = aggregate(bbb$number, by=list(bbb$state), FUN=sum)
colnames(before2) = c("state", "number")

others <- read.csv('others.csv')
others <- others[-5,]
before2 <- cbind(before2, others)
before2 <- before2[,-3]
before2$crime_pc <- 1000 * before2$number / before2$population
state_trans <- read.csv('state_trans2.csv')
state_trans <- state_trans[-5,]
before2$region <- state_trans$region

before22 <- data.frame(region=before2$region, value=before2$crime_pc)
total22 <- admin1_choropleth(country.name = "germany", df = before22, title="Crimes per population by S

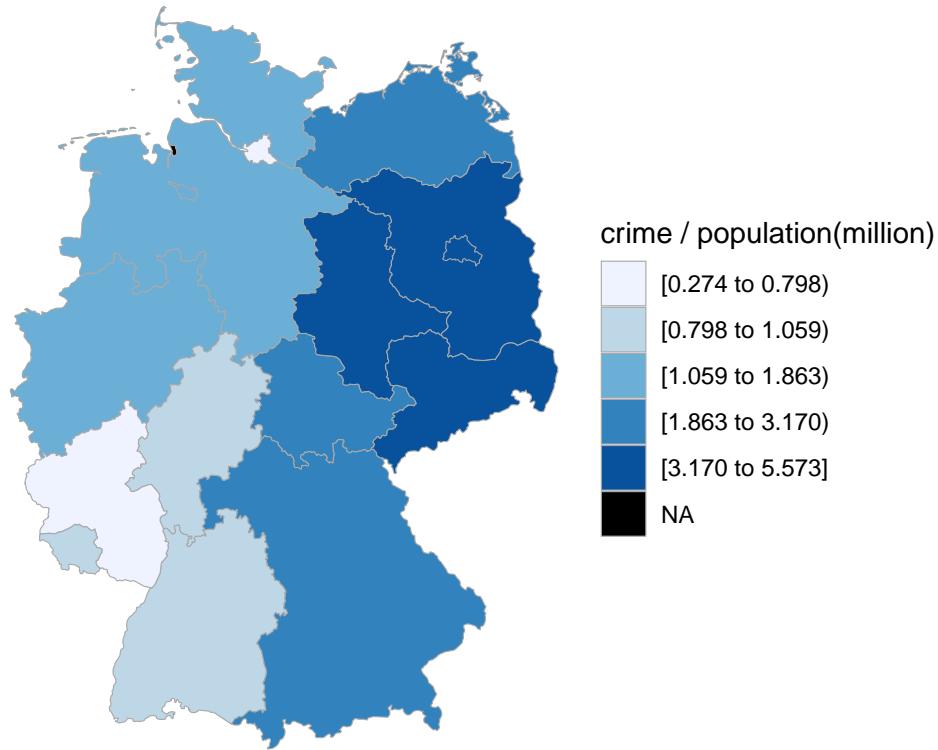
## Warning: Column `region` joining character vector and factor, coercing into
## character vector

## Warning in self$bind(): The following regions were missing and are being
## set to NA: bremen

```

```
total22
```

Crimes per population by State (Jul 5th – 18th, 2016)



```
ccc = subset(d9, date >= "2016-07-19")
ddd = subset(ccc, date <= "2016-08-02")
after2 = aggregate(ddd$number, by=list(ddd$state), FUN=sum)
colnames(after2) = c("state", "number")

others <- read.csv('others.csv')
others <- others[-5,]
after2 <- cbind(after2, others)
after2 <- after2[,-3]
after2$crime_pc <- 1000 * after2$number / after2$population
state_trans <- read.csv('state_trans2.csv')
state_trans <- state_trans[-5,]
after2$region <- state_trans$region

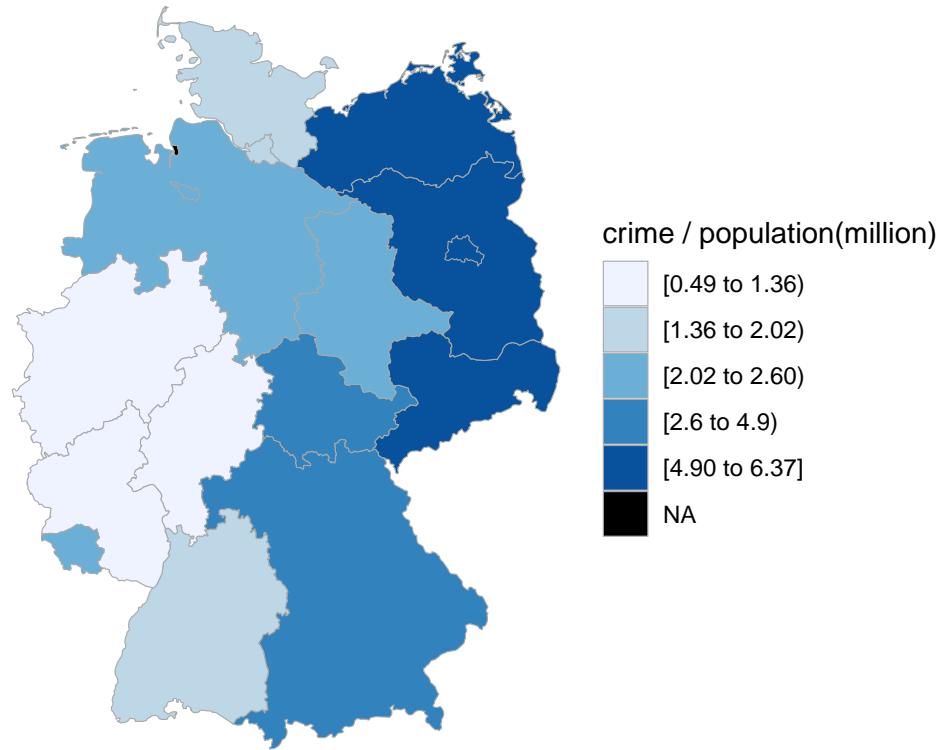
after22 <- data.frame(region=after2$region, value=after2$crime_pc)
total23 <- admin1_choropleth(country.name = "germany", df = after22, title="Crimes per population by State")

## Warning: Column `region` joining character vector and factor, coercing into
## character vector

## Warning: The following regions were missing and are being set to NA: bremen
```

```
total23
```

Crimes per population by State (Jul 19th – Aug 2nd, 2016)



3. Dec 5, 2016 - Dec 18, 2016 VS Dec, 19 2016 - Jan 2, 2017

```
aaa = subset(d9, date >= "2016-12-05")
bbb = subset(aaa, date <= "2016-12-18")
before2 = aggregate(bbb$number, by=list(bbb$state), FUN=sum)
colnames(before2) = c("state", "number")

others <- read.csv('others.csv')
others <- others[-5,]
others <- others[-5,]
others <- others[-9,]
others <- others[-9,]
before2 <- cbind(before2, others)
before2 <- before2[,-3]
before2$crime_pc <- 1000 * before2$number / before2$population
state_trans <- read.csv('state_trans2.csv')
state_trans <- state_trans[-5,]
state_trans <- state_trans[-5,]
state_trans <- state_trans[-9,]
state_trans <- state_trans[-9,]
before2$region <- state_trans$region

before22 <- data.frame(region=before2$region, value=before2$crime_pc)
```

```

total24 <- admin1_choropleth(country.name = "germany", df = before22, title="Crimes per population by State (Dec 5th – 18th, 2016)")

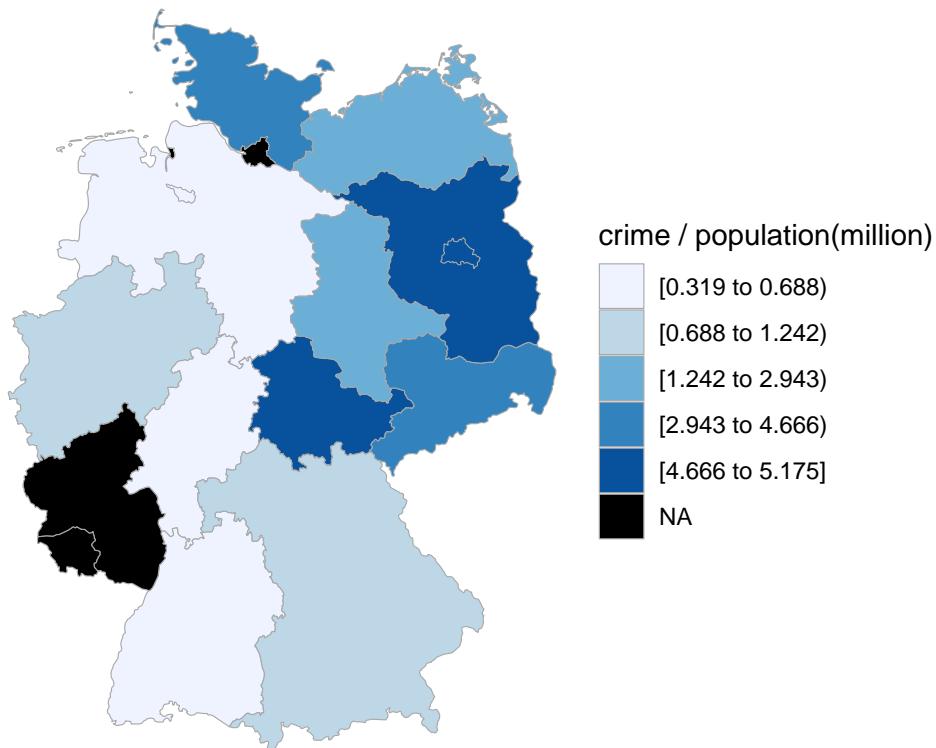
## Warning: Column `region` joining character vector and factor, coercing into
## character vector

## Warning in self$bind(): The following regions were missing and are being
## set to NA: bremen, freie und hansestadt hamburg, rheinland-pfalz, saarland

total24

```

Crimes per population by State (Dec 5th – 18th, 2016)



```

ccc = subset(d9, date >= "2016-12-19")
ddd = subset(ccc, date <= "2017-01-02")
after2 = aggregate(ddd$number, by=list(ddd$state), FUN=sum)
colnames(after2) = c("state", "number")

others <- read.csv('others.csv')
others <- others[-5,]
others <- others[-5,]
others <- others[-10,]
after2 <- cbind(after2, others)
after2 <- after2[-3]
after2$crime_pc <- 1000 * after2$number / after2$population
state_trans <- read.csv('state_trans2.csv')
state_trans <- state_trans[-5,]
state_trans <- state_trans[-5,]
state_trans <- state_trans[-10,]
after2$region <- state_trans$region

```

```

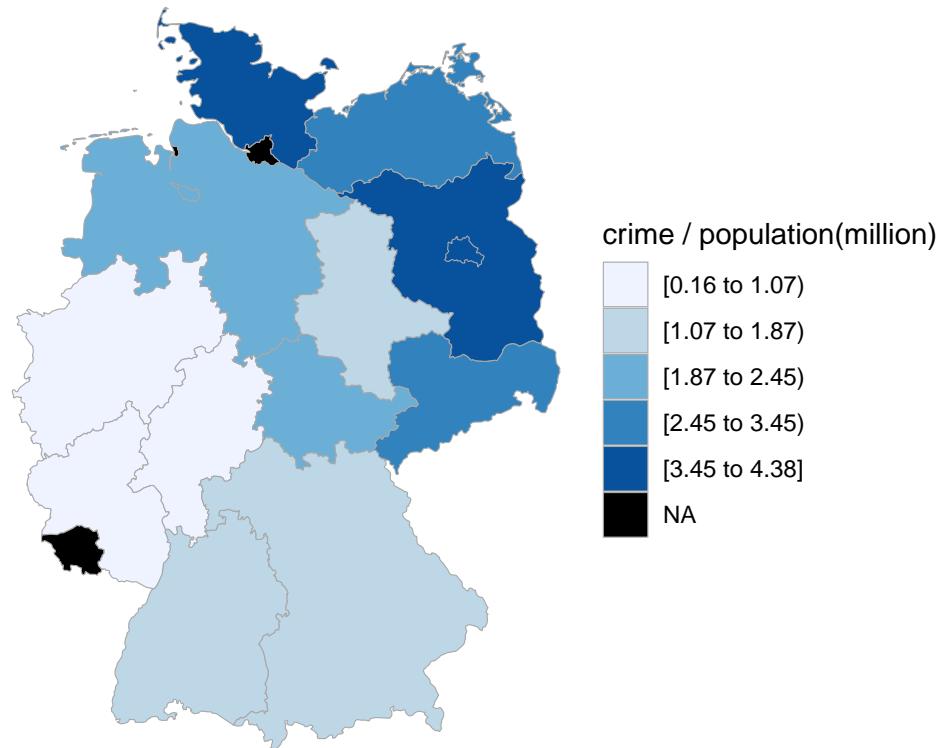
after22 <- data.frame(region=after2$region, value=after2$crime_pc)
total25 <- admin1_choropleth(country.name = "germany", df = after22, title="Crimes per population by Sta

## Warning: Column `region` joining character vector and factor, coercing into
## character vector

## Warning in self$bind(): The following regions were missing and are being
## set to NA: bremen, freie und hansestadt hamburg, saarland
total25

```

Crimes per population by State (Dec 19th, 2016 – Jan 2nd, 2017)



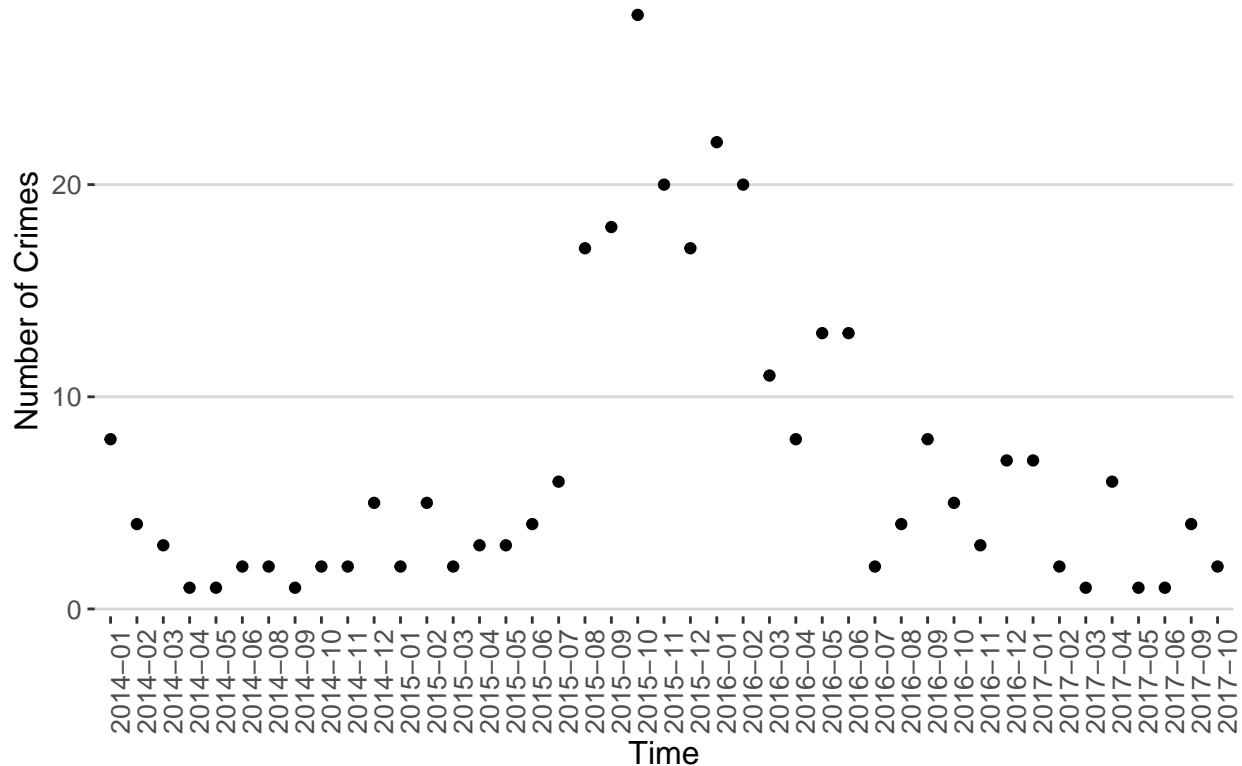
Arson graphs

```

# Arson
arson1 <- ggplot(subset(d2, category=='arson'),
                  aes(x = ym, y = number)) +
  geom_point() + theme(legend.position = "top") +
  labs(x="Time", y="Number of Crimes") + ggtitle("Arson") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
arson1

```

Arson

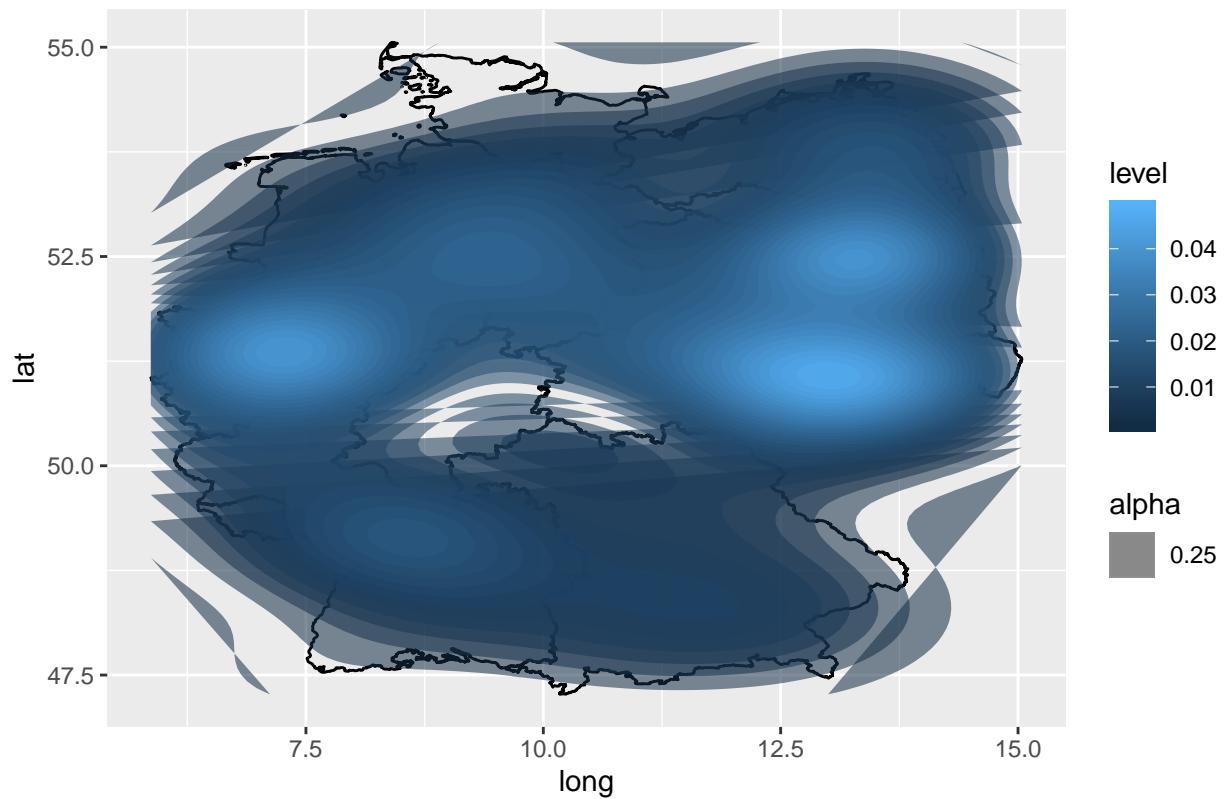


```
arson <- subset(b, category_en=='arson')

arson2 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(arson, y <= 2015),
    geom = "polygon") + ggttitle("Arson: 2014 & 2015")

## Regions defined for each Polygons
arson2
```

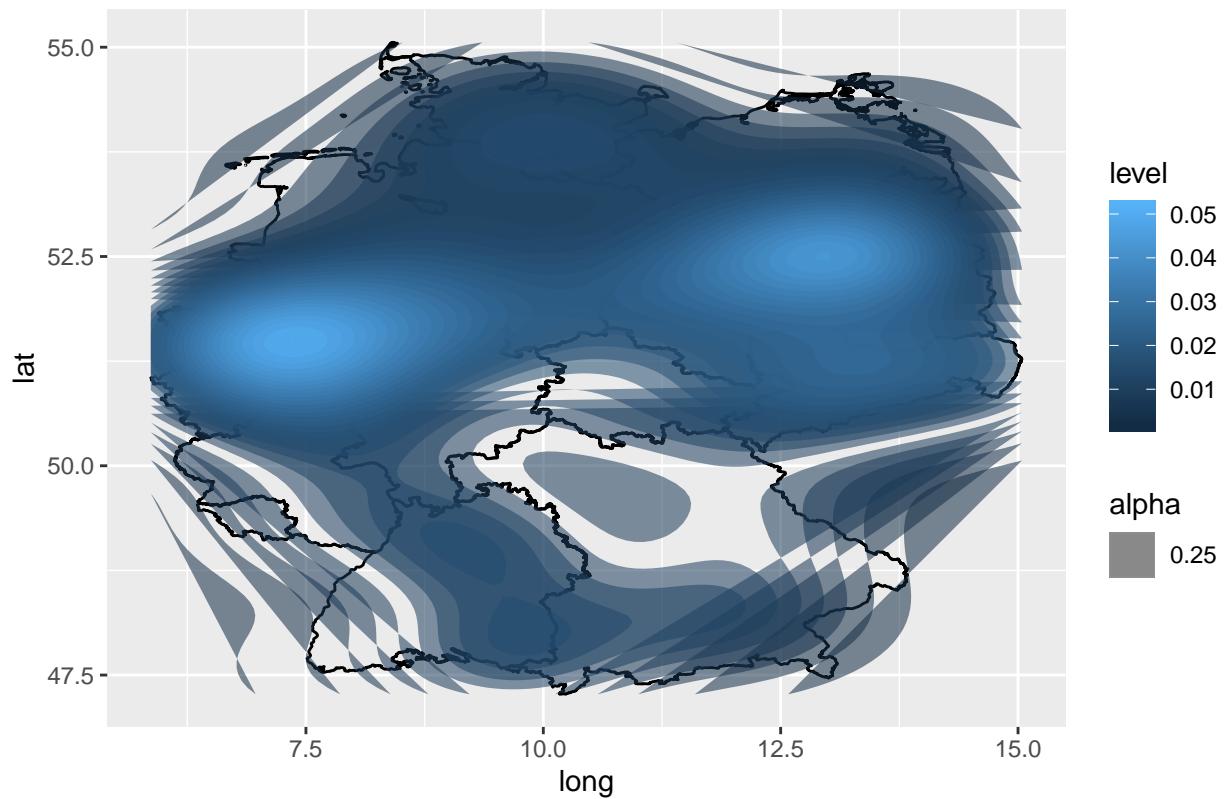
Arson: 2014 & 2015



```
arson3 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(arson, y >= 2016),
    geom = "polygon") + ggttitle("Arson: 2016 & 2017")
```

```
## Regions defined for each Polygons
arson3
```

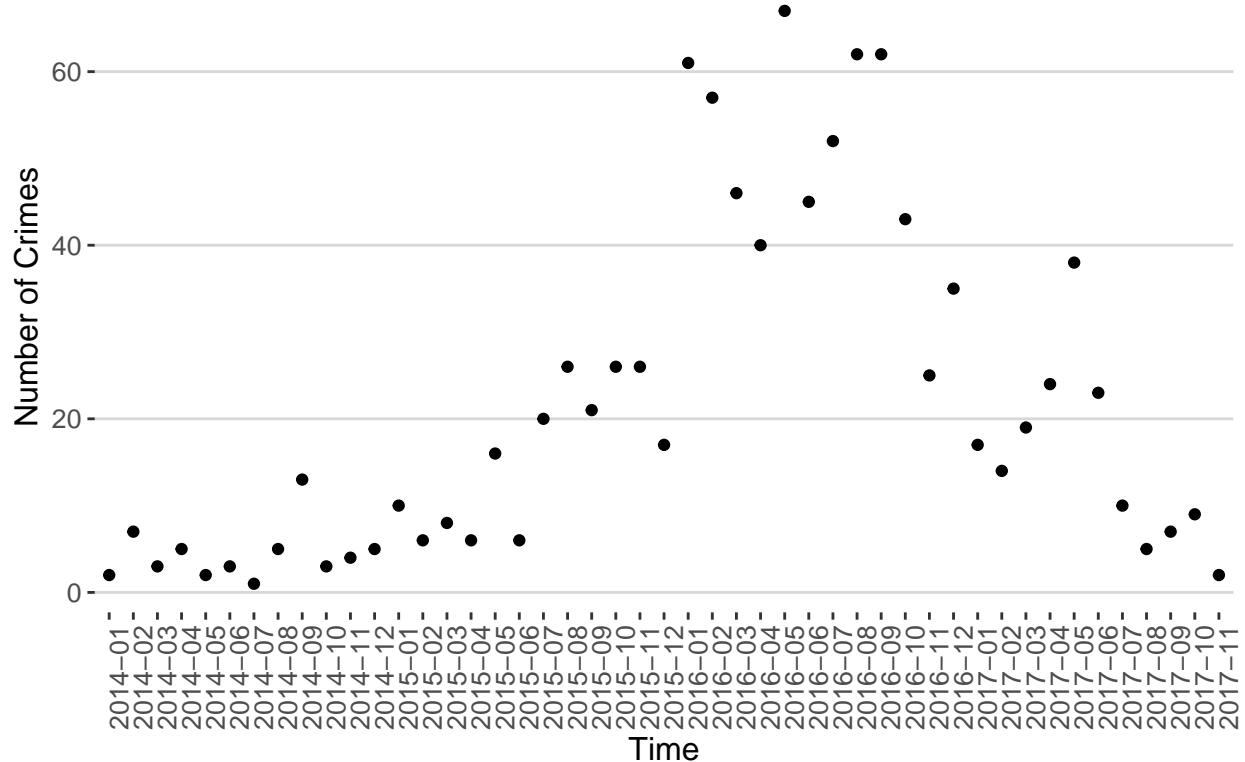
Arson: 2016 & 2017



Assault graphs

```
# Assault
assault1 <- ggplot(subset(d2, category=='assault'),
  aes(x = ym, y = number)) +
  geom_point() + theme(legend.position = "top") +
  labs(x="Time", y="Number of Crimes") + ggtitle("Assault") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
assault1
```

Assault

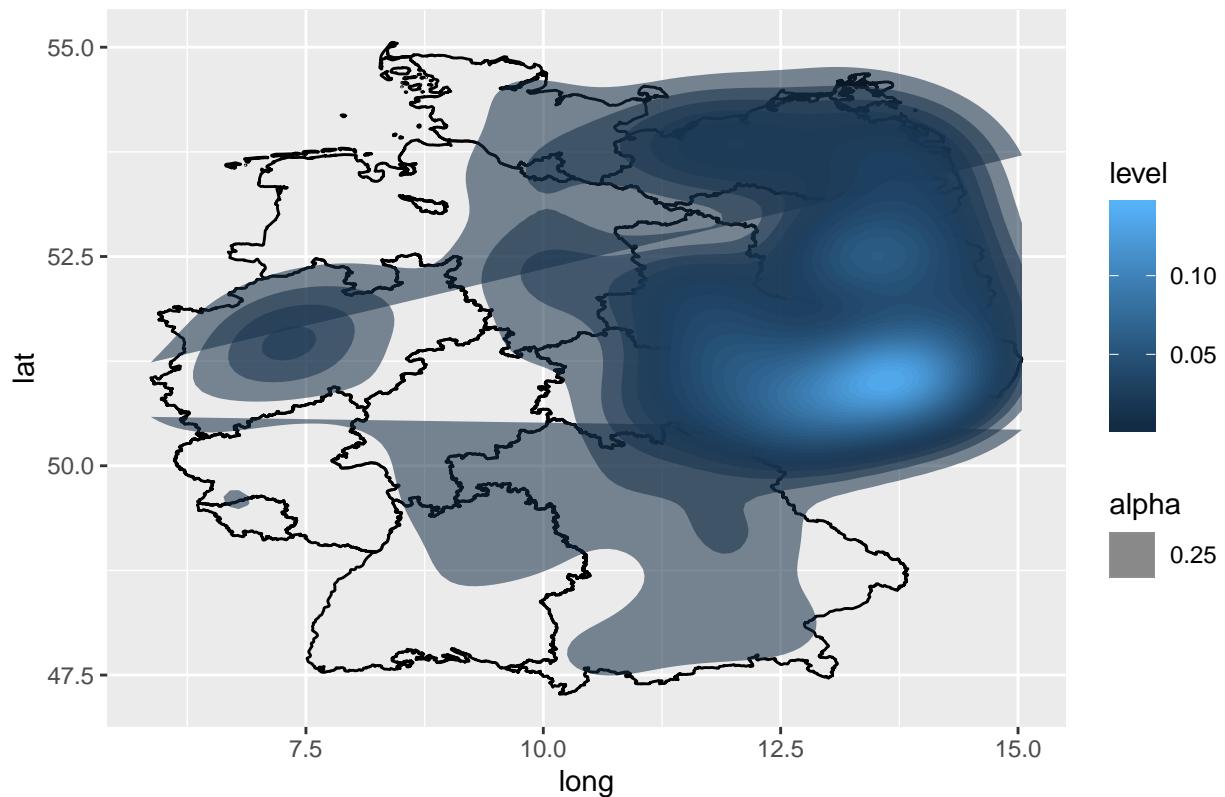


```
assault <- subset(b, category_en=='assault')

assault2 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(assault, y <= 2015),
    geom = "polygon") + ggttitle("Assault: 2014 & 2015")

## Regions defined for each Polygons
assault2
```

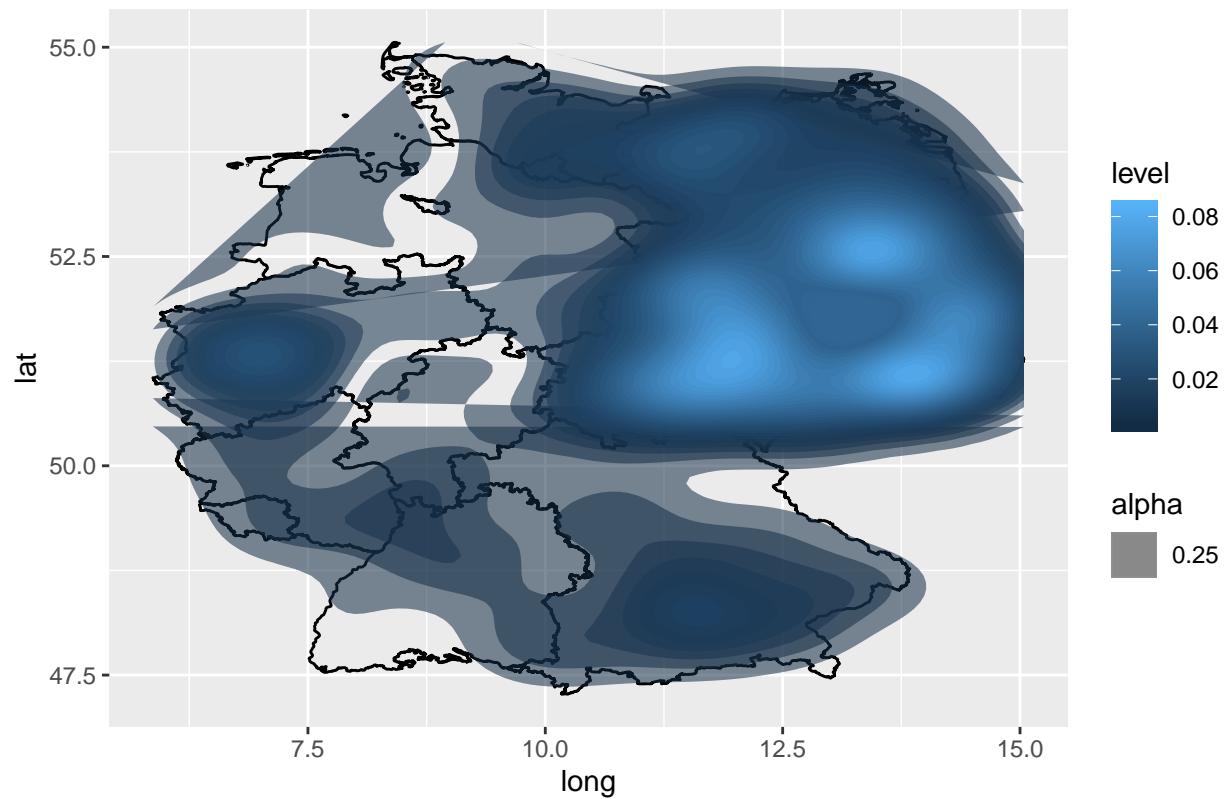
Assault: 2014 & 2015



```
assault3 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(assault, y >= 2016),
    geom = "polygon") + ggttitle("Assault: 2016 & 2017")

## Regions defined for each Polygons
assault3
```

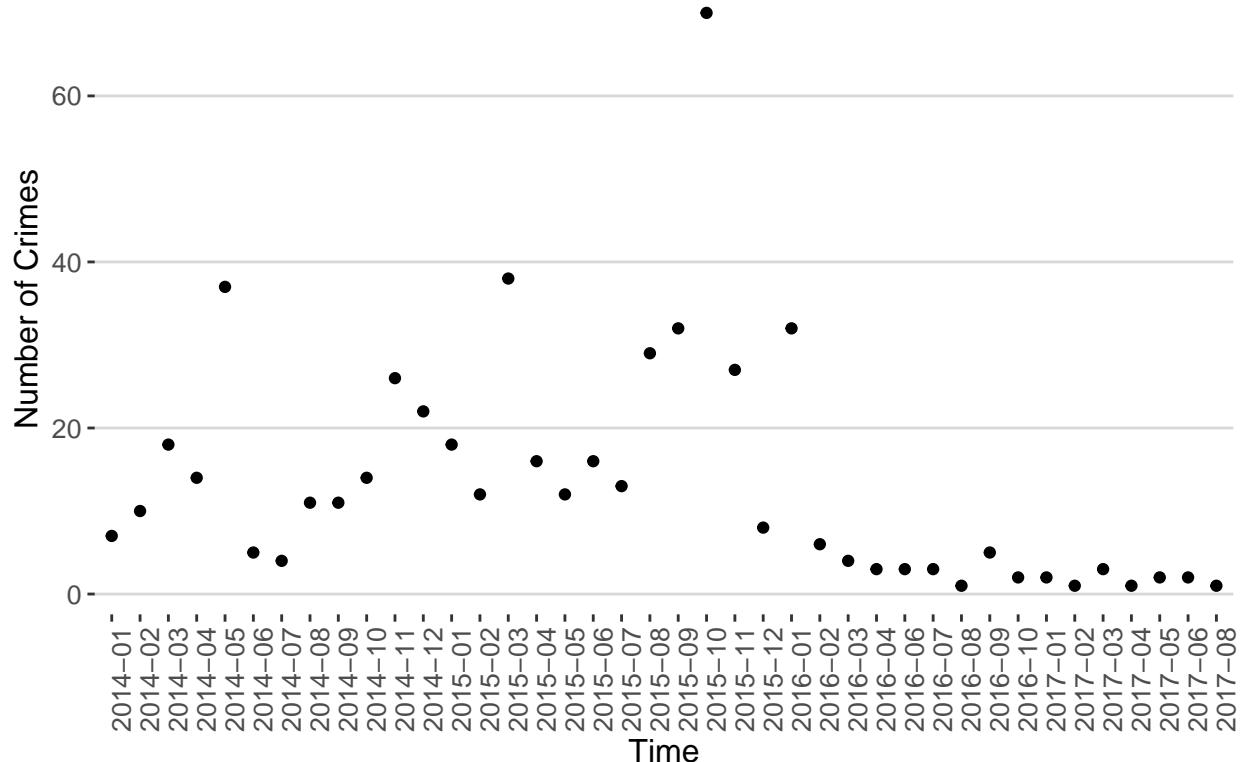
Assault: 2016 & 2017



Demonstration graphs

```
# Demonstration
demo1 <- ggplot(subset(d2, category=='demonstration'),
                 aes(x = ym, y = number)) +
  geom_point() + theme(legend.position = "top") +
  labs(x="Time", y="Number of Crimes") + ggtitle("Demonstration") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
demo1
```

Demonstration

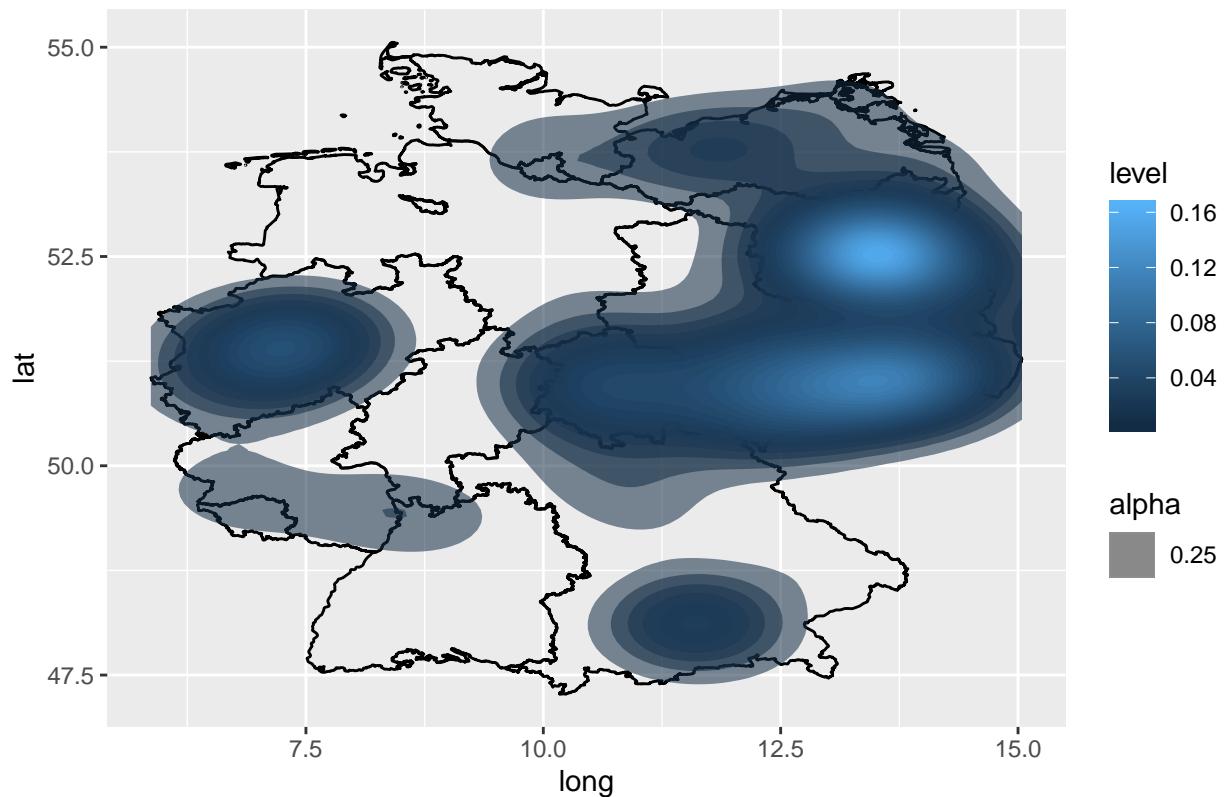


```
demo <- subset(b, category_en=='demonstration')

demo2 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(demo, y <= 2015),
    geom = "polygon") + ggtitle("Demonstration: 2014 & 2015")

## Regions defined for each Polygons
demo2
```

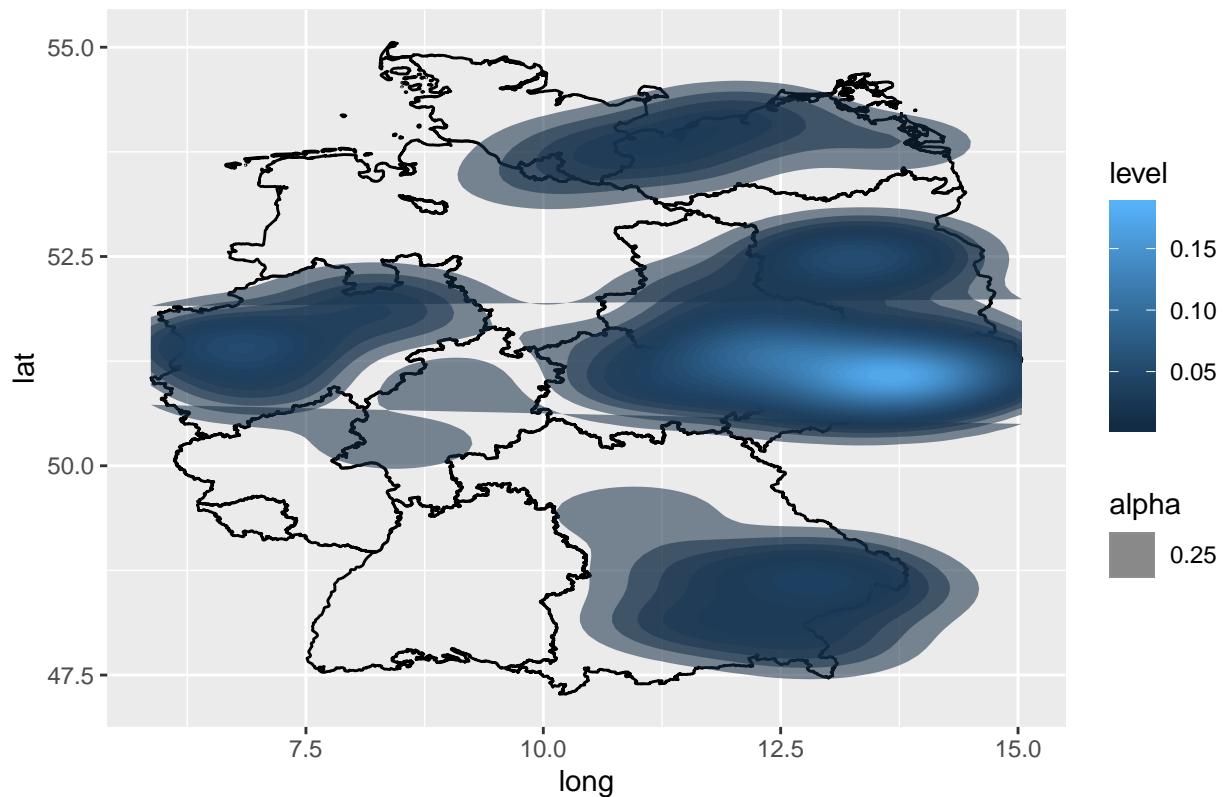
Demonstration: 2014 & 2015



```
demo3 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(demo, y >= 2016),
    geom = "polygon") + ggttitle("Demonstration: 2016 & 2017")

## Regions defined for each Polygons
demo3
```

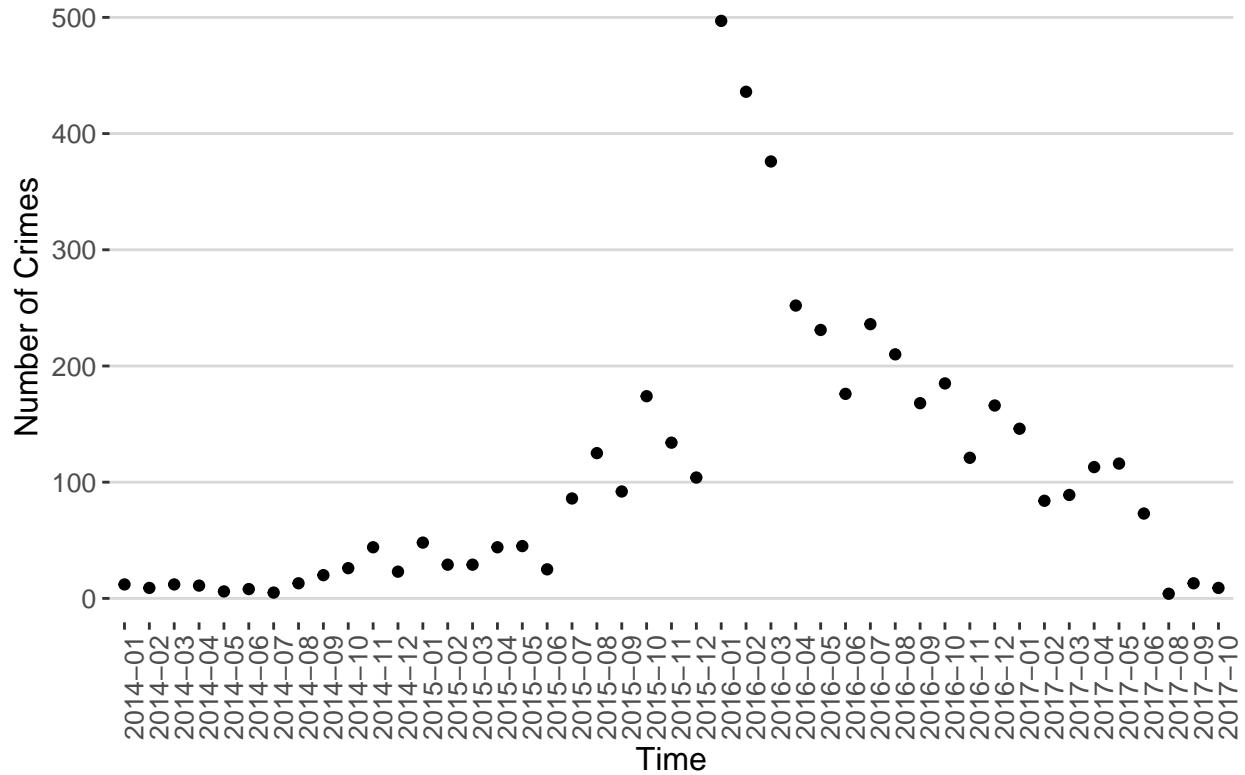
Demonstration: 2016 & 2017



Miscellaneous Attack graphs

```
# Miscellaneous Attack
att1 <- ggplot(subset(d2, category=='miscellaneous attack'),
               aes(x = ym, y = number)) +
  geom_point() + theme(legend.position = "top") +
  labs(x="Time", y="Number of Crimes") +
  ggtitle("Miscellaneous Attack") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
att1
```

Miscellaneous Attack

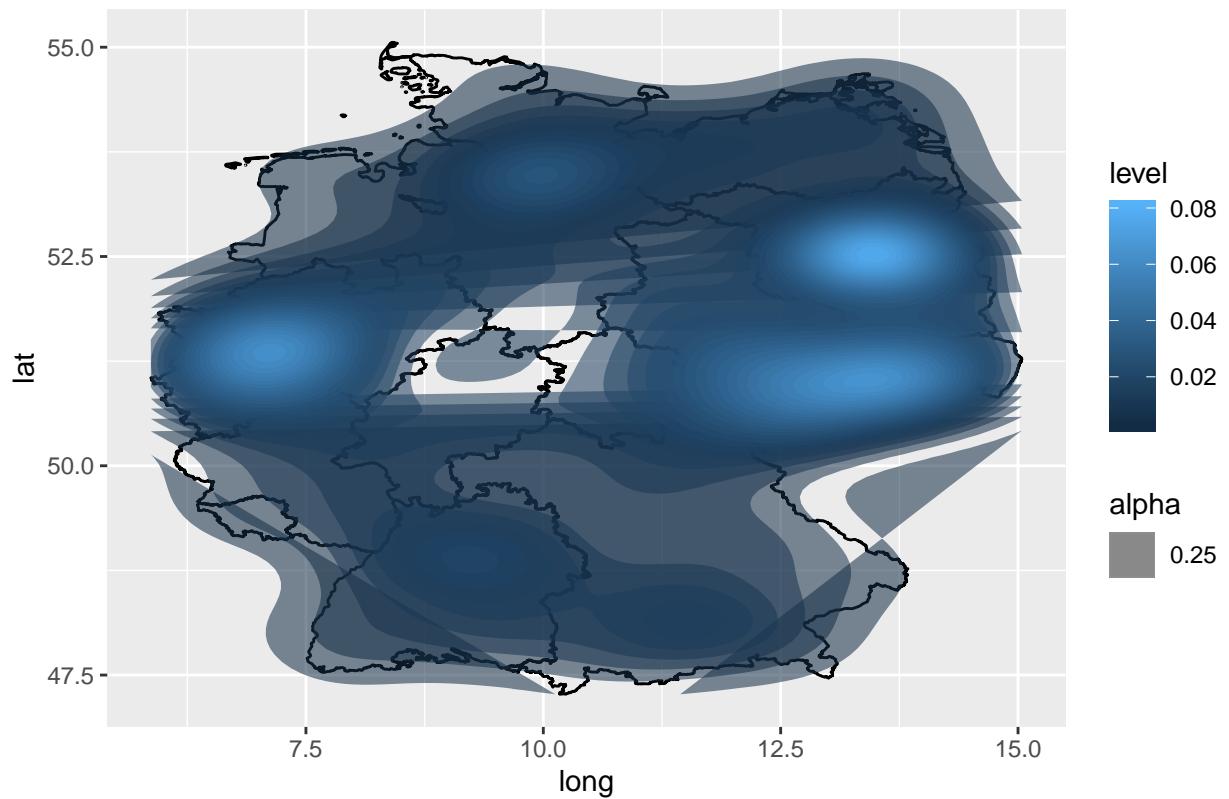


```
att <- subset(b, category_en=='miscellaneous attack')

att2 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(att, y <= 2015),
    geom = "polygon") +
  ggtitle("Miscellaneous Attack: 2014 & 2015")

## Regions defined for each Polygons
att2
```

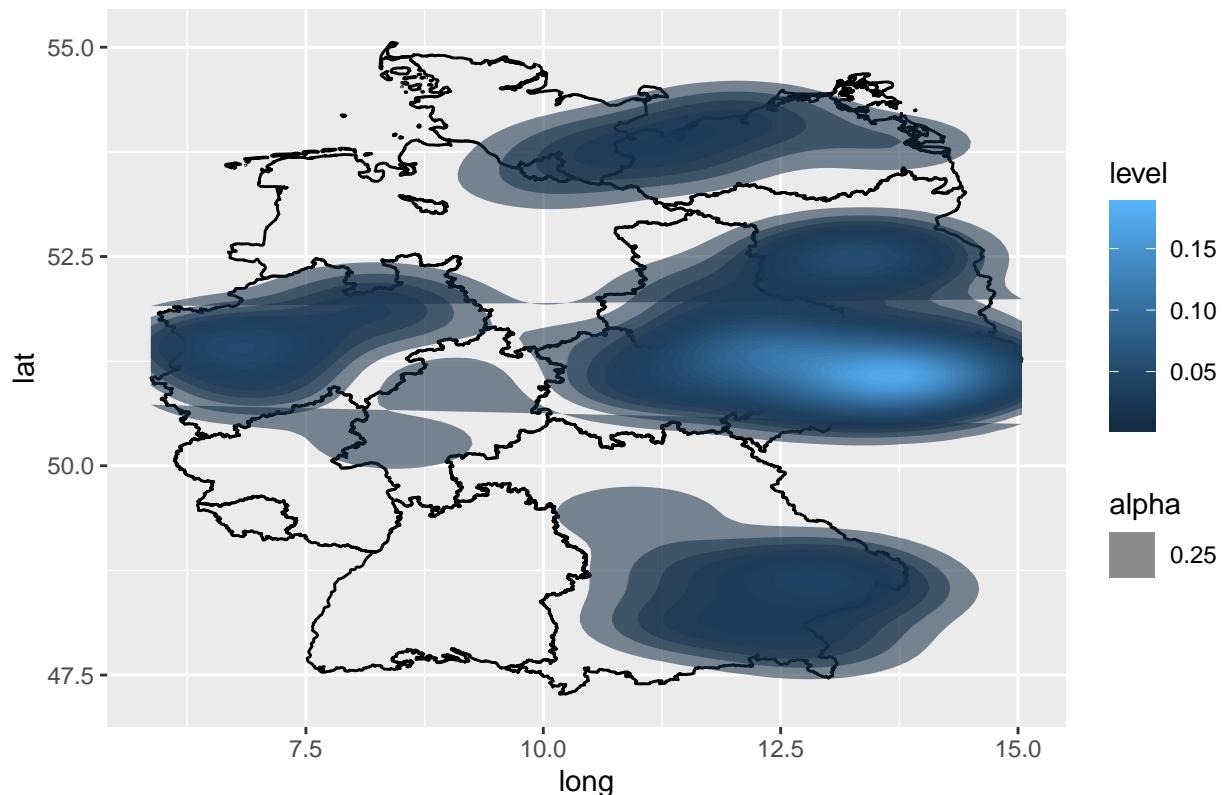
Miscellaneous Attack: 2014 & 2015



```
att3 <- ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(demo, y >= 2016),
    geom = "polygon") +
  ggtitle("Miscellaneous Attack: 2016 & 2017")

## Regions defined for each Polygons
att3
```

Miscellaneous Attack: 2016 & 2017

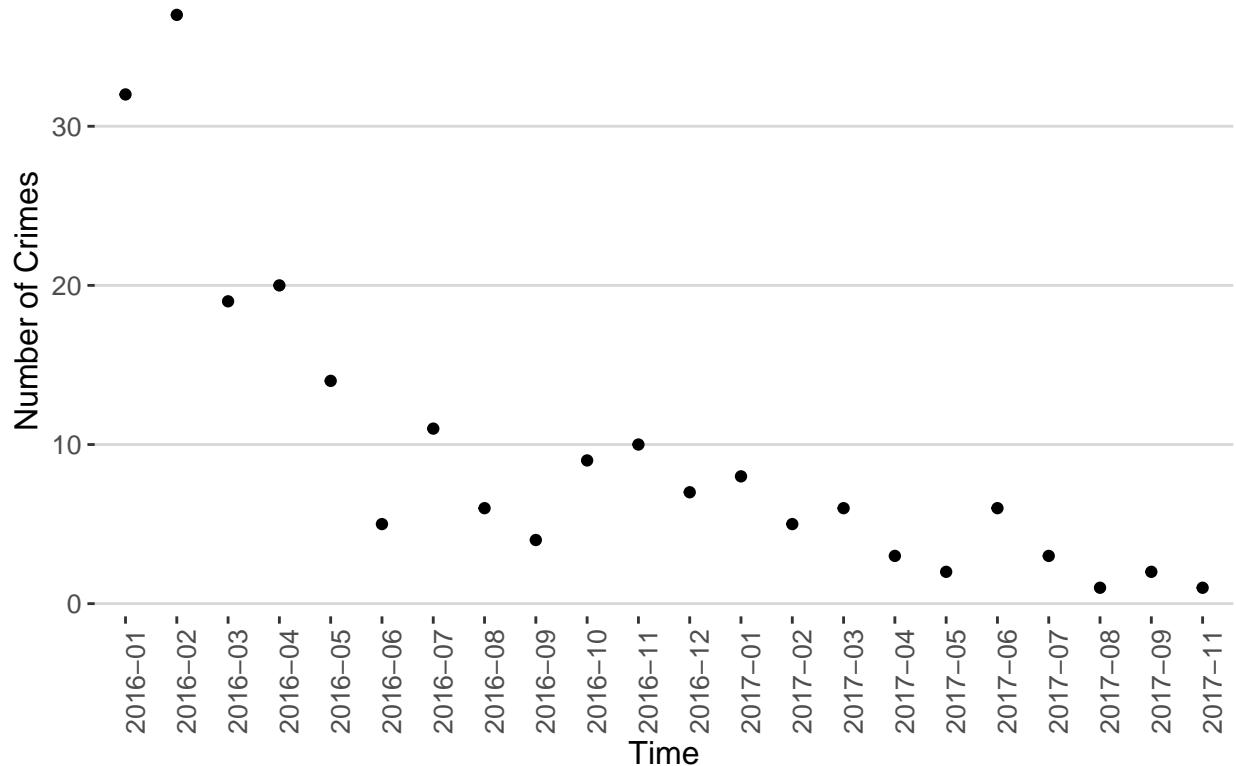


Comment: For Miscellaneous Attack, the crime area may shrink after Jan-2016.

Suspicion graphs

```
# Suspicion
ggplot(subset(d2, category=='suspicion'),
       aes(x = ym, y = number)) +
  geom_point() + theme(legend.position = "top") +
  labs(x="Time", y="Number of Crimes") + ggtitle("Suspicion") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
```

Suspicion

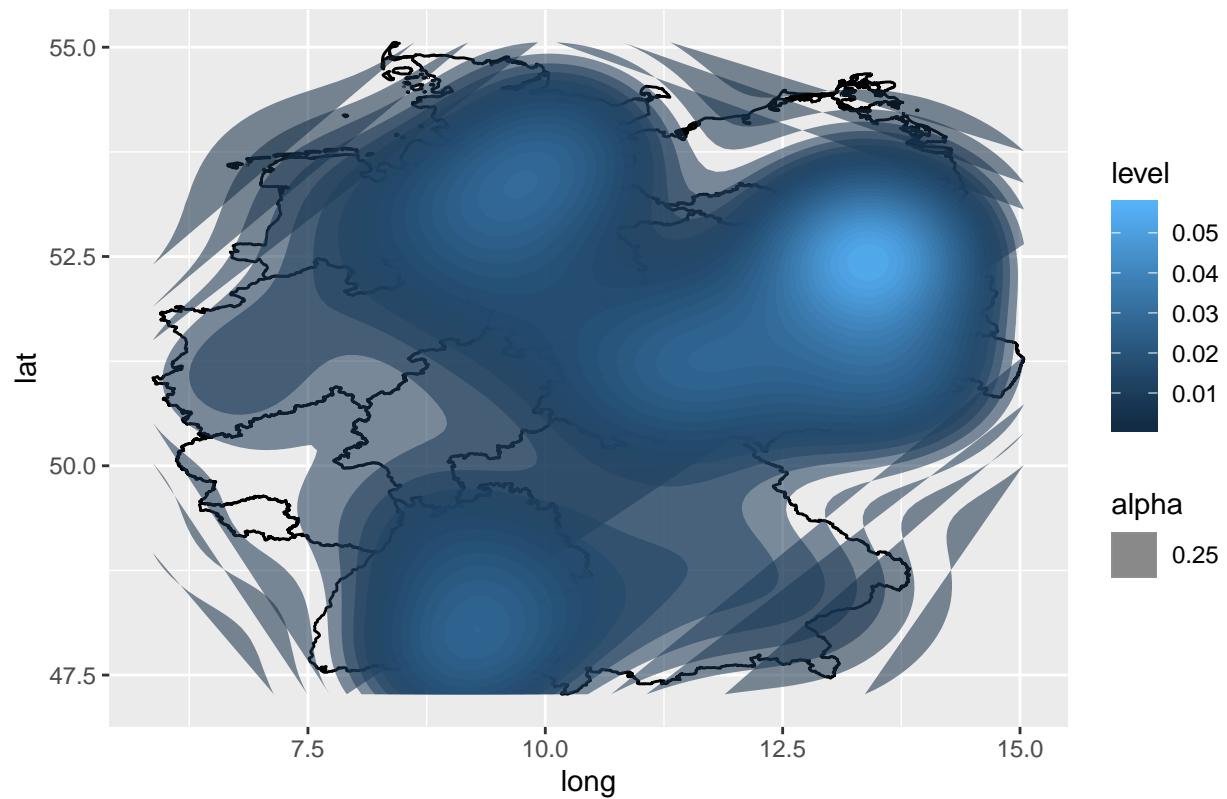


```
sus <- subset(b, category_en=='suspicion')

ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(sus, y >= 2016),
    geom = "polygon") + ggttitle("Suspicion: 2016 & 2017")

## Regions defined for each Polygons
```

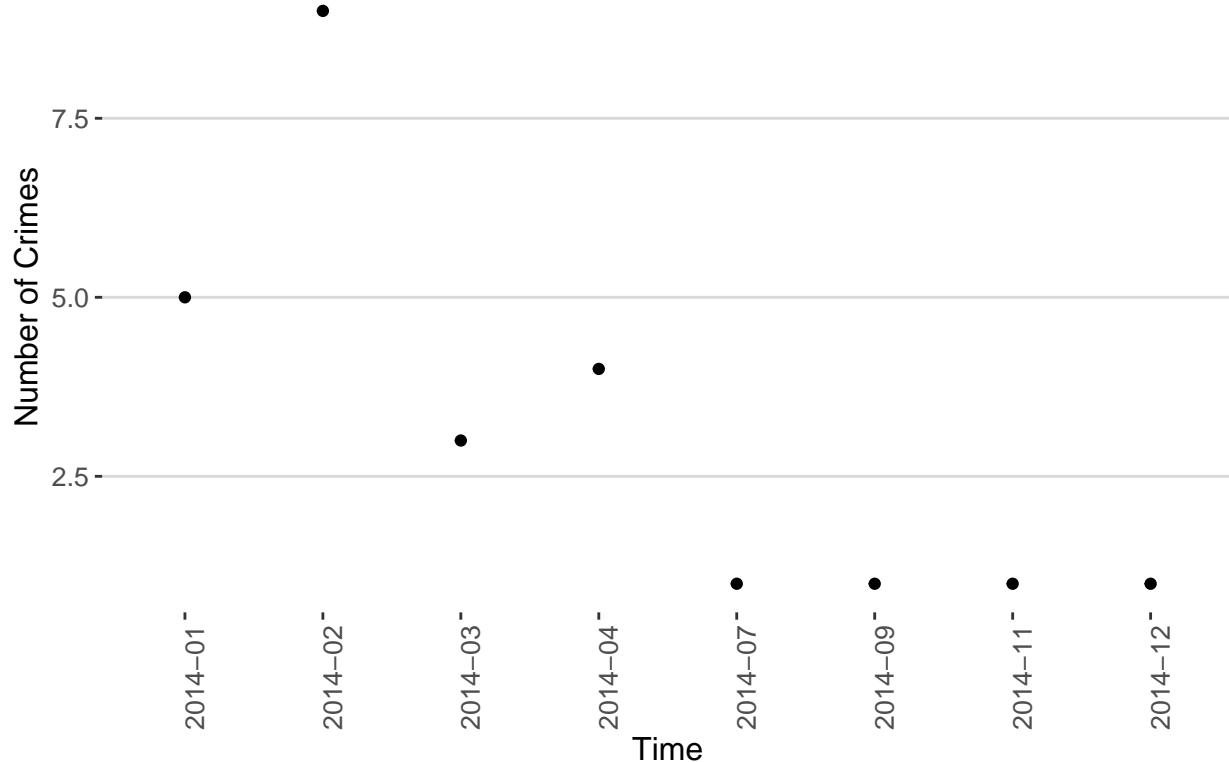
Suspicion: 2016 & 2017



Other graphs

```
# Other
ggplot(subset(d2, category=='other'),
       aes(x = ym, y = number)) +
  geom_point() + theme(legend.position = "top") +
  labs(x="Time", y="Number of Crimes") + ggtitle("Other") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
```

Other

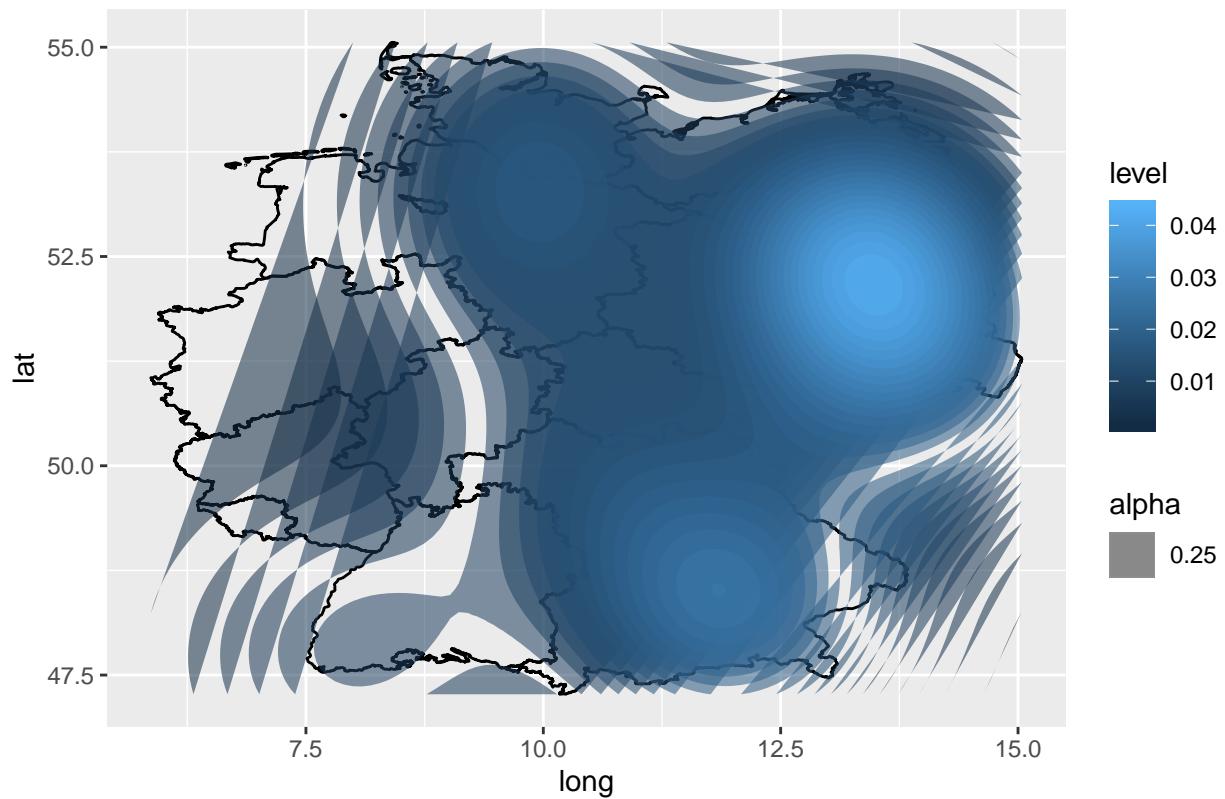


```
other <- subset(b, category_en=='other')

ggplot() +
  geom_path(data = germany_shape_files,
            aes(x = long, y = lat, group = group)) +
  stat_density2d(
    aes(x=longitude, y=latitude, fill = ..level.., alpha = 0.25),
    size = 0.01, bins = 30, data = subset(other, y <= 2015),
    geom = "polygon") + ggttitle("Other: 2014 & 2015")

## Regions defined for each Polygons
```

Other: 2014 & 2015

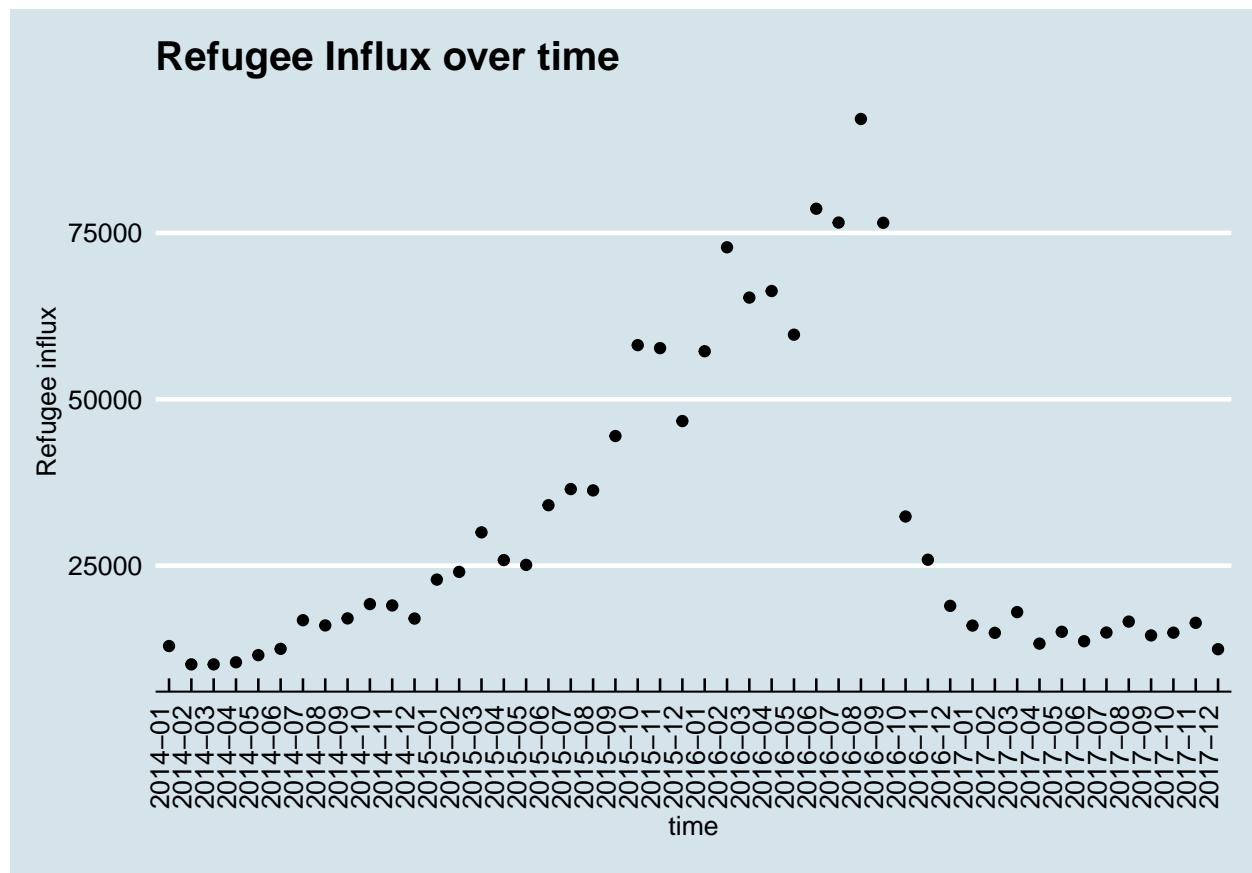


Refugee graphs

```
refugee <- read.csv('refugee.csv')
r <- subset(
  refugee, year == 2014 | year == 2015 | year == 2016 | year == 2017)
ref <- aggregate(r$number, by=list(r$ym), FUN=sum)
colnames(ref) = c("ym", "number")

ref1 <- ggplot(data = ref, aes(x = ym, y = number)) +
  geom_point() + geom_smooth(color="blue", lwd=1, se=FALSE) +
  xlab("time") + ylab("Refugee influx") +
  theme_economist() + theme(legend.position="none") +
  ggtitle("Refugee Influx over time") +
  theme(axis.text.x = element_text(angle=90, hjust=1))
ref1

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



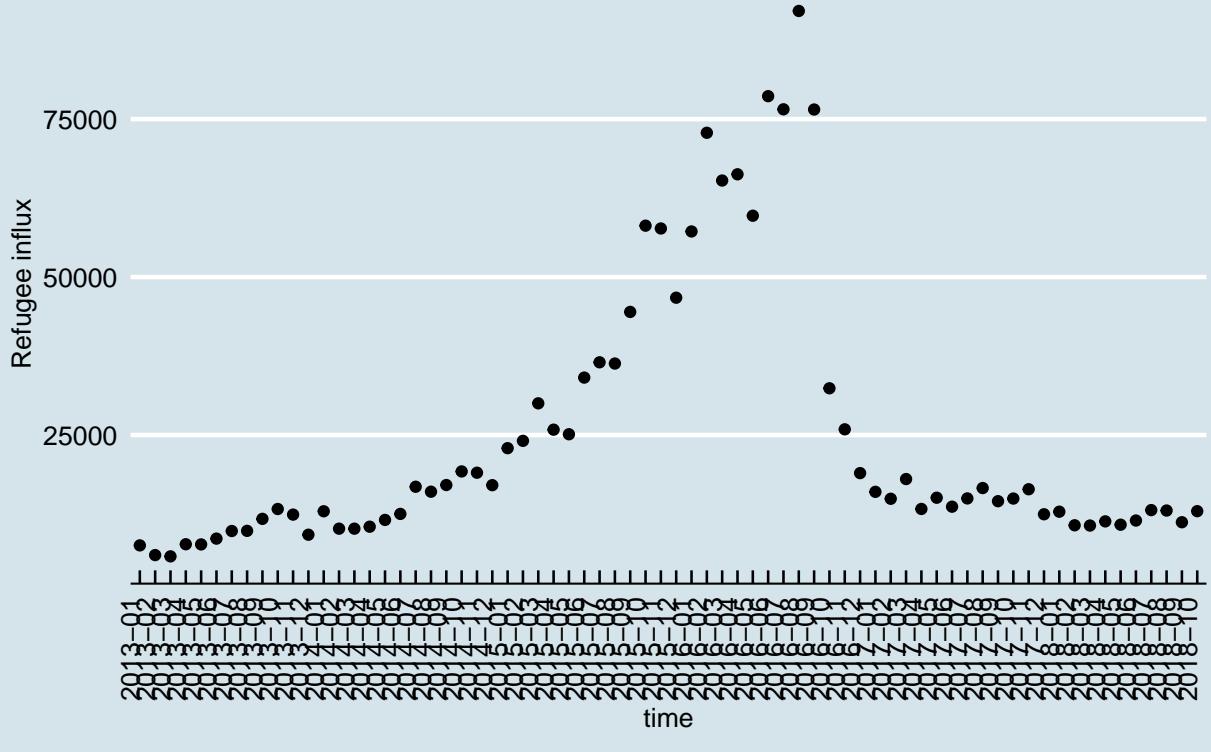
Comment: Refugee number in Germany looks strongly correlated with crime data. Although I could not find state level data for refugee data, we may analyze something related to refugee data.

```
refugee <- read.csv('refugee.csv')
ref <- aggregate(refugee$number, by=list(refugee$ym), FUN=sum)
colnames(ref) = c("ym", "number")

ref2 <- ggplot(data = ref, aes(x = ym, y = number)) +
  geom_point() + geom_smooth(color="blue", lwd=1, se=FALSE) +
  xlab("time") + ylab("Refugee influx") +
  theme_economist() + theme(legend.position="none") +
  ggtitle("Refugee Influx over time") +
  theme(axis.text.x = element_text(angle=90, hjust=1))
ref2

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

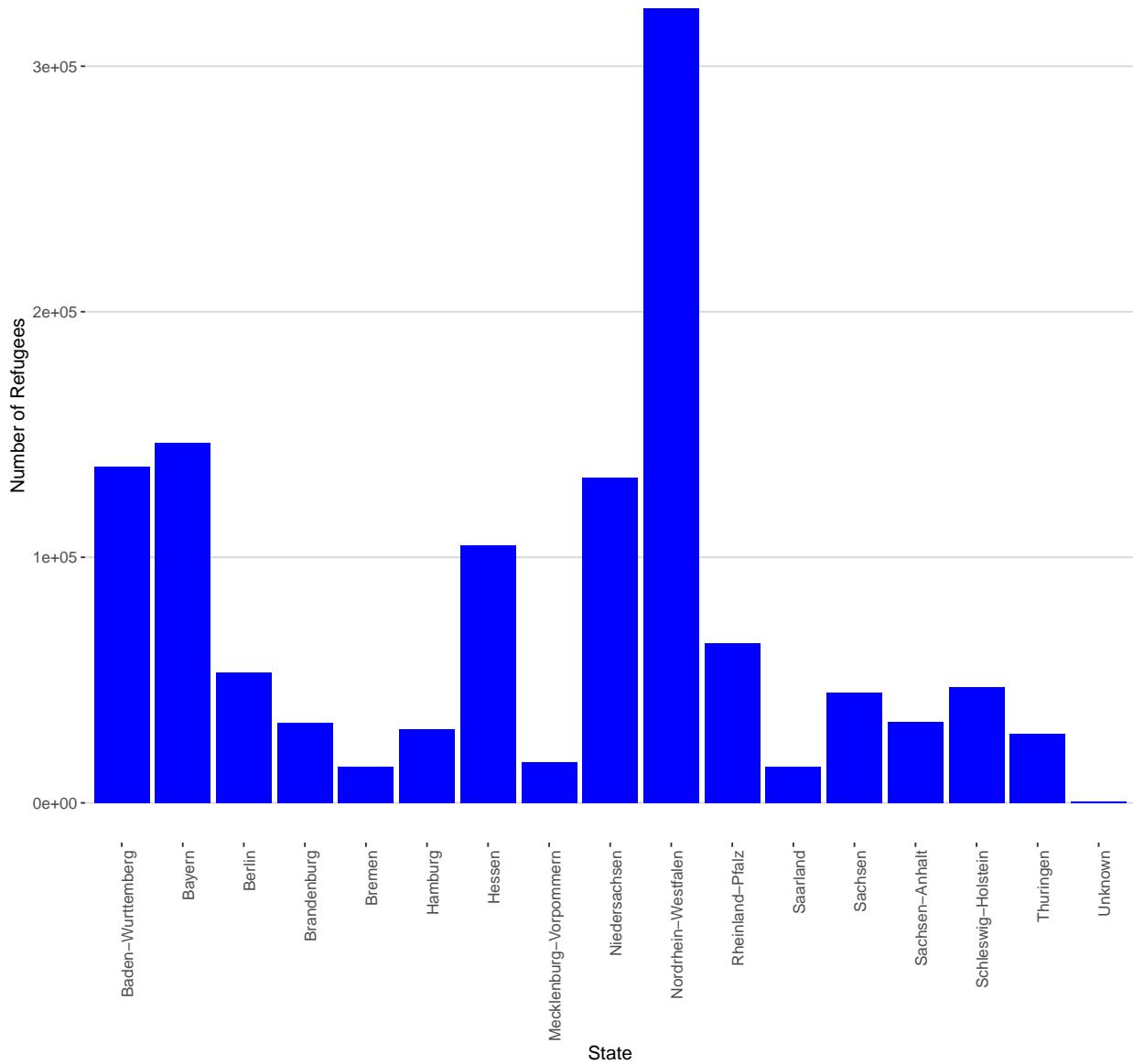
Refugee Influx over time



```
http://popstats.unhcr.org/en/asylum_seekers_monthly
refstate <- read.csv('refugee_state.csv')
colnames(refstate) = c("State", "Y2016", "Y2017", "Y2018", "Y2019")
refstate$total <- refstate$Y2016 + refstate$Y2017 +
  refstate$Y2018 + refstate$Y2019

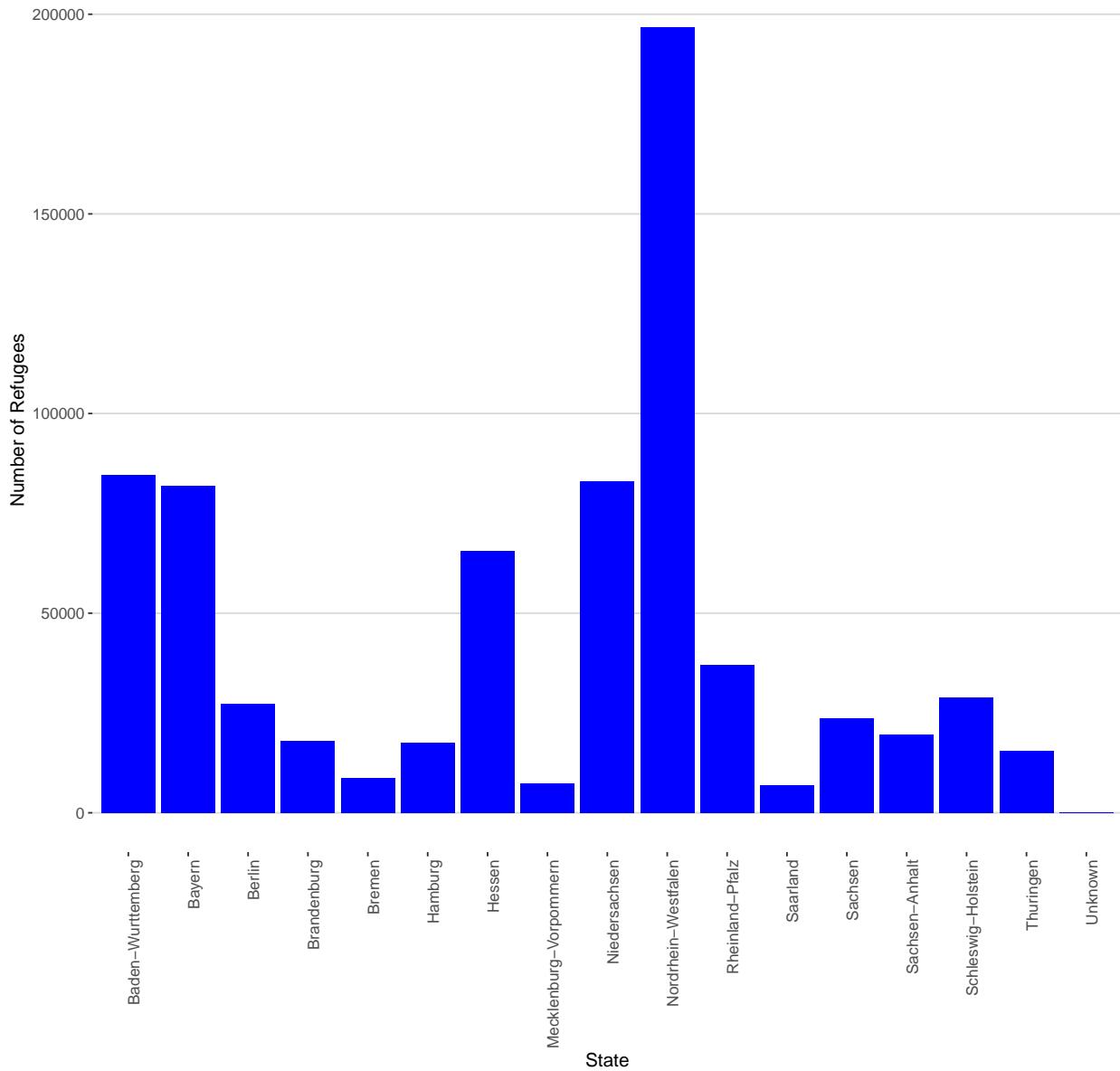
ref3 <- ggplot(refstate) +
  geom_bar(aes(x=State, y=total), stat="identity", fill='blue') +
  theme(legend.position = "top") +
  ggtitle("Refugees by state (2016-2019)") +
  labs(x="State", y="Number of Refugees") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
ref3
```

Refugees by state (2016–2019)



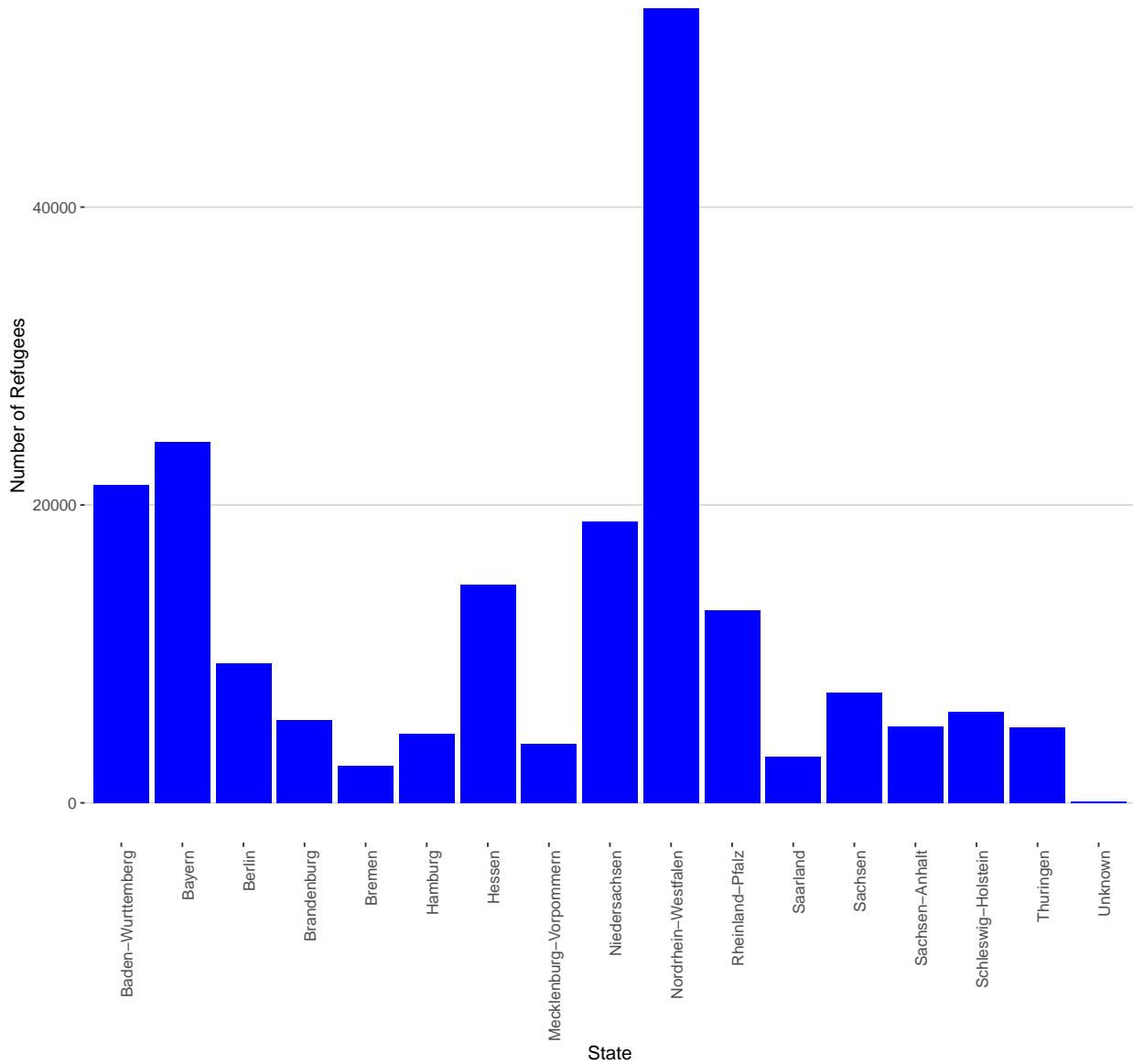
```
ref4 <- ggplot(refstate) +
  geom_bar(aes(x=State, y=Y2016), stat="identity", fill='blue') +
  theme(legend.position = "top") +
  ggtitle("Refugees by state (2016)") +
  labs(x="State", y="Number of Refugees") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
ref4
```

Refugees by state (2016)



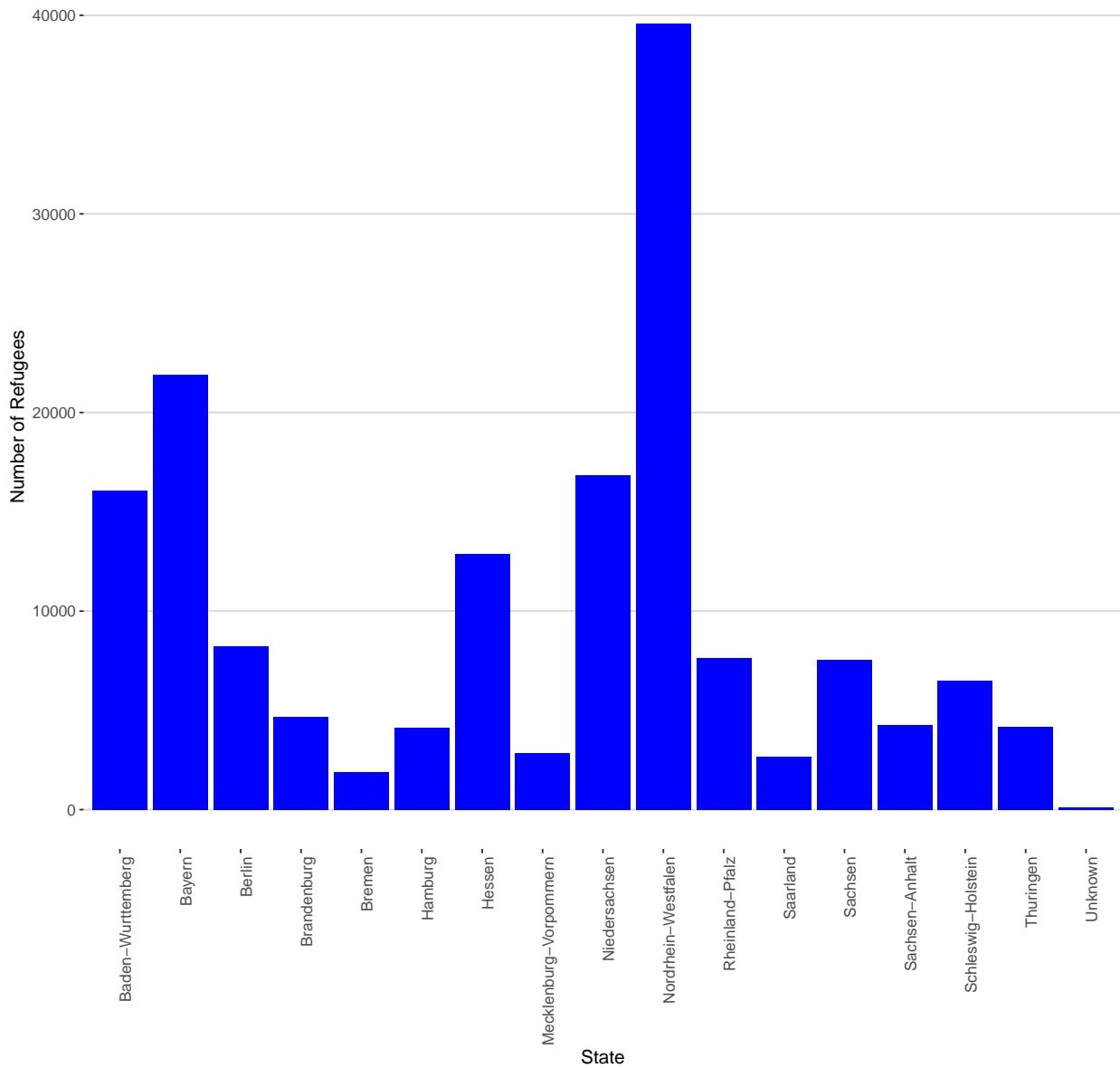
```
ref5 <- ggplot(refstate) +
  geom_bar(aes(x=State, y=Y2017), stat="identity", fill='blue') +
  theme(legend.position = "top") +
  ggtitle("Refugees by state (2017)") +
  labs(x="State", y="Number of Refugees") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
ref5
```

Refugees by state (2017)



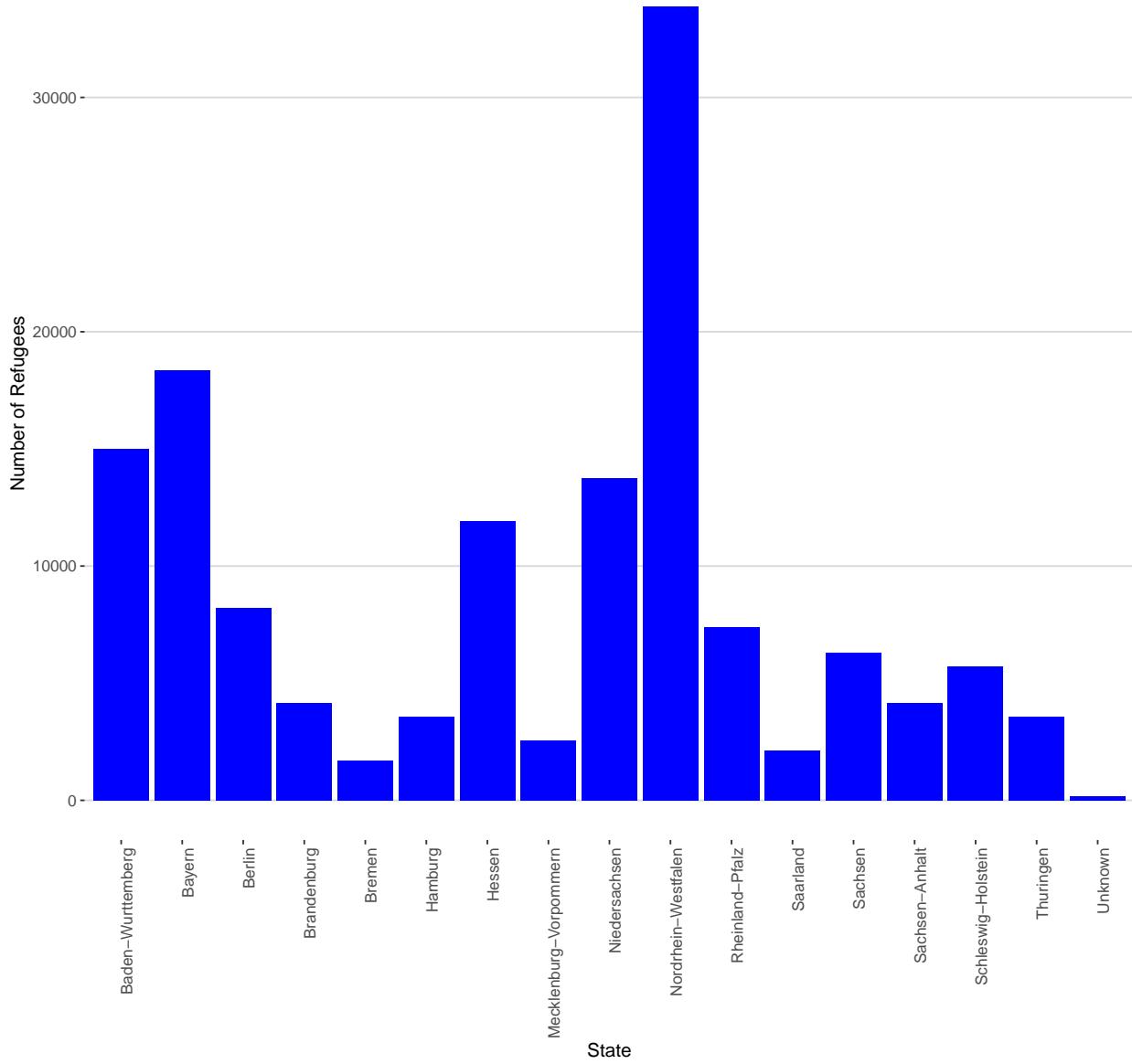
```
ref6 <- ggplot(refstate) +
  geom_bar(aes(x=State, y=Y2018), stat="identity", fill='blue') +
  theme(legend.position = "top") +
  ggtitle("Refugees by state (2018)") +
  labs(x="State", y="Number of Refugees") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
ref6
```

Refugees by state (2018)



```
ref7 <- ggplot(refstate) +
  geom_bar(aes(x=State, y=Y2019), stat="identity", fill='blue') +
  theme(legend.position = "top") +
  ggtitle("Refugees by state (2019)") +
  labs(x="State", y="Number of Refugees") +
  theme_hc() + theme(axis.text.x = element_text(angle=90, hjust=1))
ref7
```

Refugees by state (2019)



Crimes by month/asylum applications

```

q <- read.csv("applicants.csv")
q$crime <- 0
for (i in 1:22){
  q$crime[i] = d3$number[3*(i-1)+1]+d3$number[3*(i-1)+2]+d3$number[3*(i-1)+3]
}

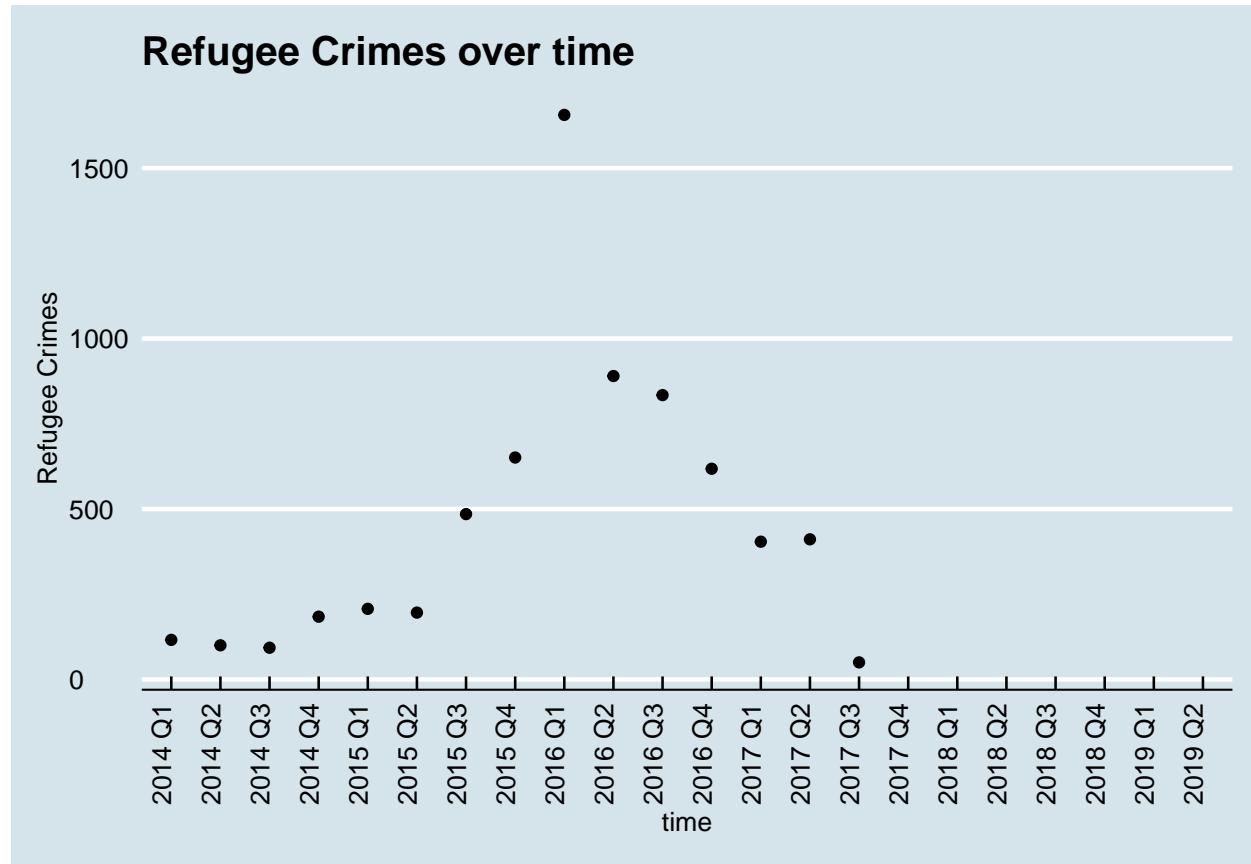
ref8 <- ggplot(data = q, aes(x = time, y = crime)) +
  geom_point() + geom_smooth(color="blue", lwd=1, se=FALSE) +
  xlab("time") + ylab("Refugee Crimes") +
  theme_economist() + theme(legend.position="none") +
  ggtitle("Refugee Crimes over time") +
  
```

```

theme(axis.text.x = element_text(angle=90, hjust=1))
ref8

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## Warning: Removed 7 rows containing non-finite values (stat_smooth).
## Warning: Removed 7 rows containing missing values (geom_point).

```

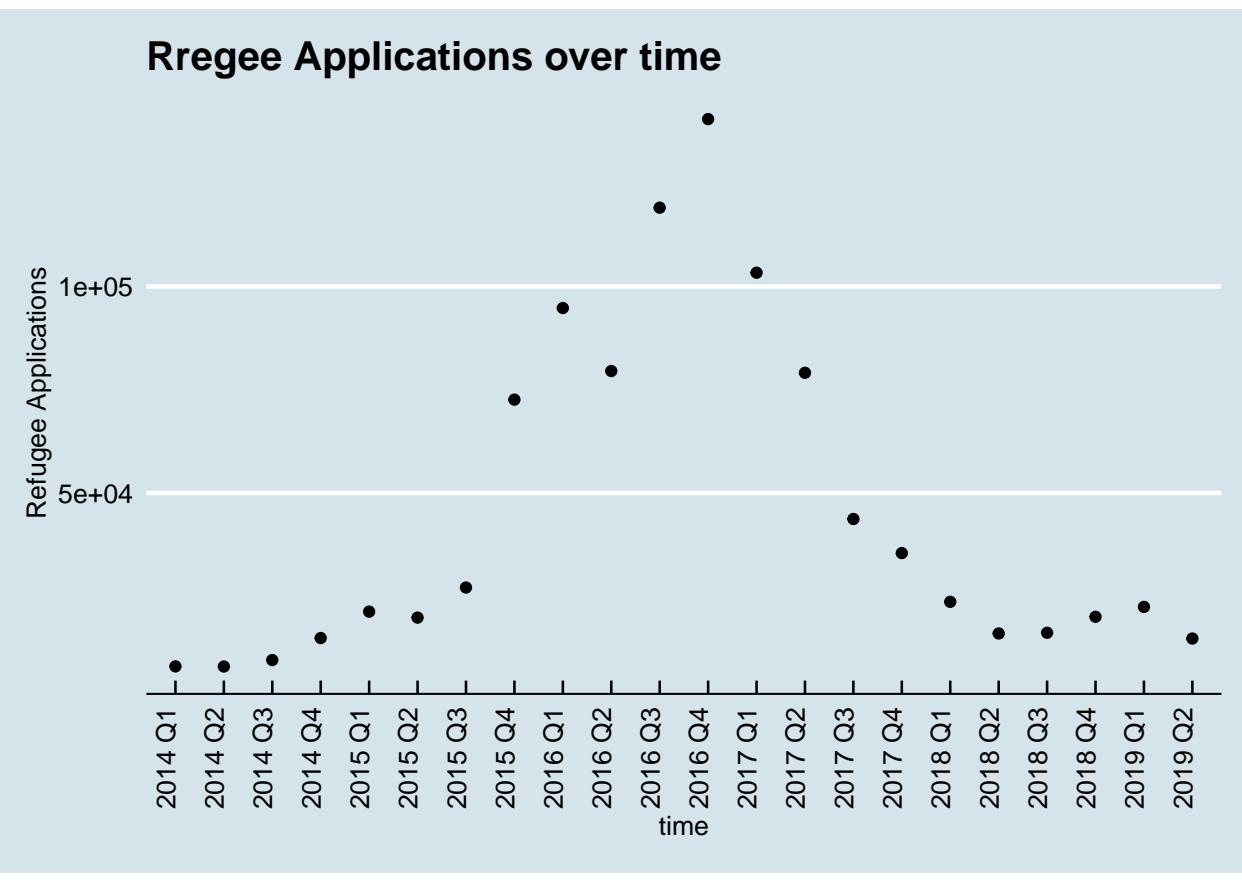


```

ref9 <- ggplot(data = q, aes(x = time, y = applicants)) +
  geom_point() + geom_smooth(color="blue", lwd=1, se=FALSE) +
  xlab("time") + ylab("Refugee Applications") +
  theme_economist() + theme(legend.position="none") +
  ggtitle("Refugee Applications over time") +
  theme(axis.text.x = element_text(angle=90, hjust=1))
ref9

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

```



Prediction

```

library(forecast)

## Warning: package 'forecast' was built under R version 3.6.2
## Registered S3 method overwritten by 'quantmod':
##   method           from
##   as.zoo.data.frame zoo

library(urca)

## Warning: package 'urca' was built under R version 3.6.2
crime_ts <- ts(d3$number, start=c(2014,1), frequency=12)

# Unit Root Test
summary(ur.kpss(crime_ts))

##
## #####
## # KPSS Unit Root Test #
## #####
## 
## Test is of type: mu with 3 lags.
##

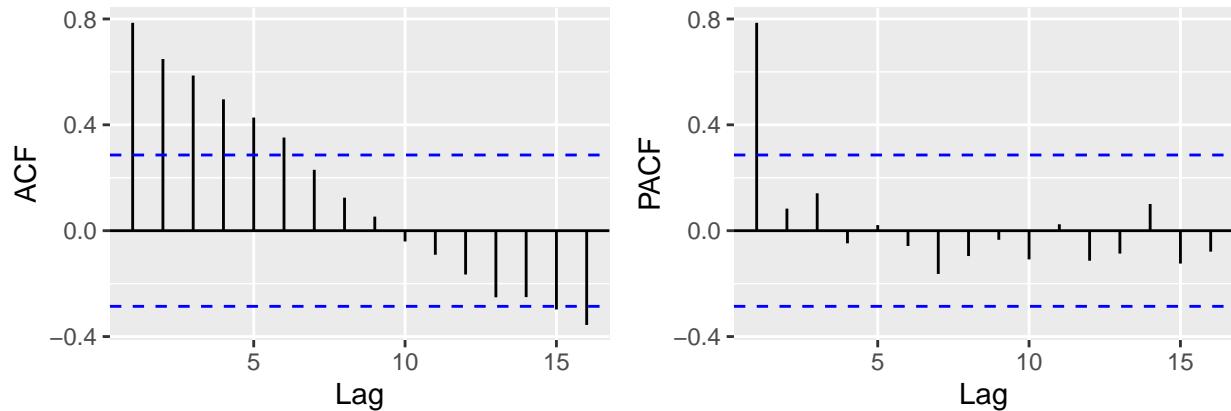
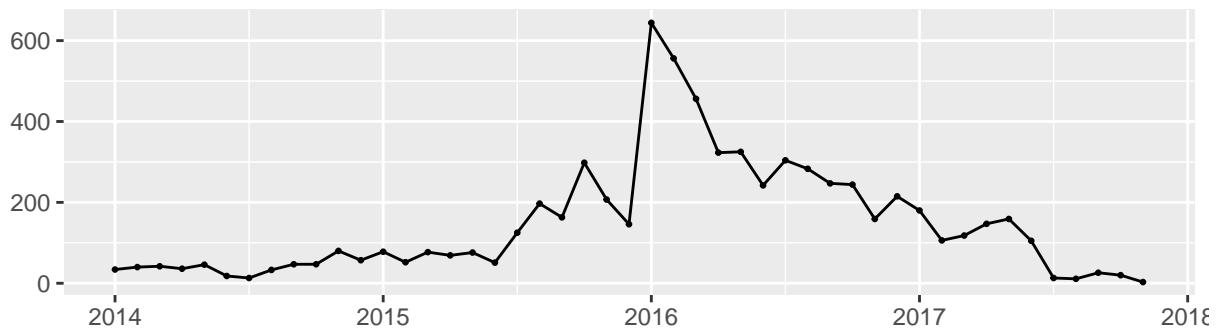
```

```

## Value of test-statistic is: 0.3261
##
## Critical value for a significance level of:
##          10pct  5pct 2.5pct 1pct
## critical values 0.347 0.463 0.574 0.739
# Autocorrelation
ggtstdisplay(crime_ts, main="Autocorrelation")

```

Autocorrelation

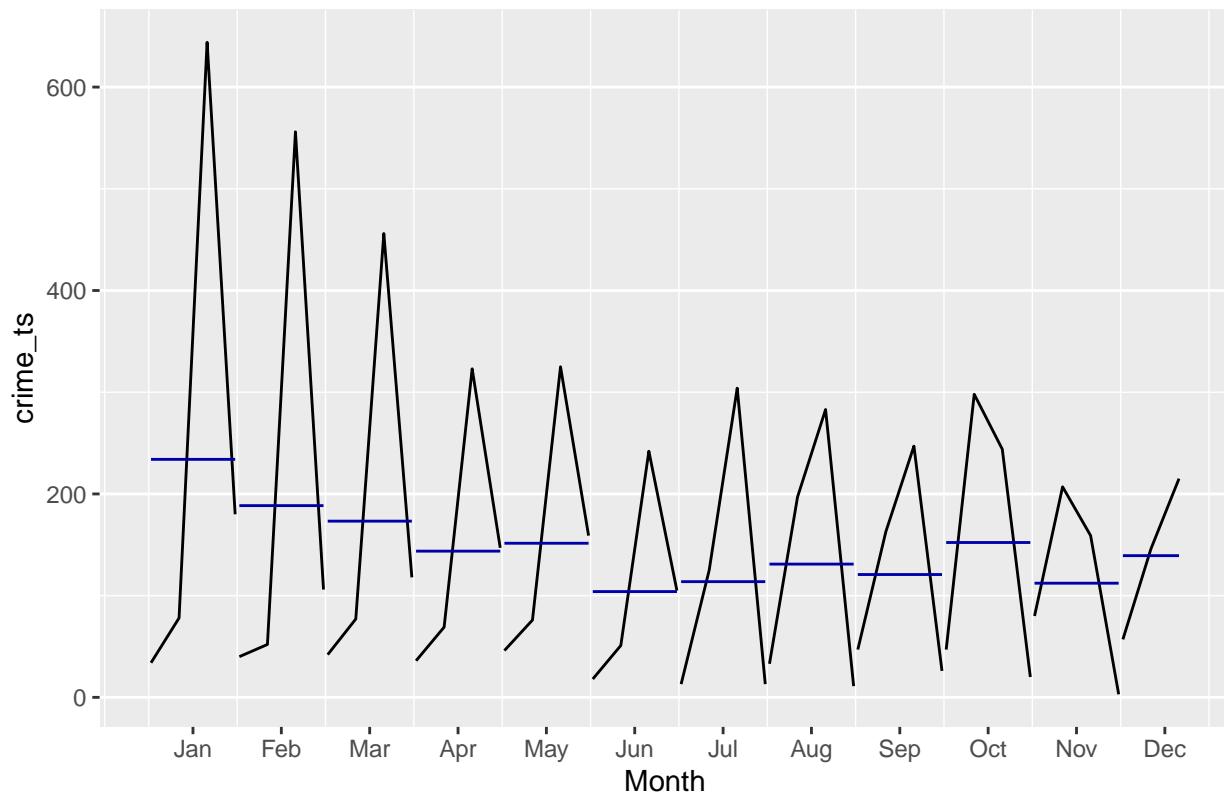


```

# Seasonality
ggsubseriesplot(crime_ts, main="Seasonality")

```

Seasonality



```
# Model Selection using only crime data
model <- auto.arima(
  y=crime_ts, ic="aic", max.order=7,
  stepwise=F, approximation=F, parallel=T, num.cores=4
)

# Best Model using only crime data
model

## Series: crime_ts
## ARIMA(0,1,1)
##
## Coefficients:
##         ma1
##       -0.2973
## s.e.   0.1643
##
## sigma^2 estimated as 7810:  log likelihood=-270.97
## AIC=545.93  AICc=546.21  BIC=549.59

# Prediction using only crime data
prediction <- forecast(
  model, h=11, level=c(95,70)
)
prediction
```

```
##          Point Forecast      Lo 70      Hi 70      Lo 95      Hi 95
```

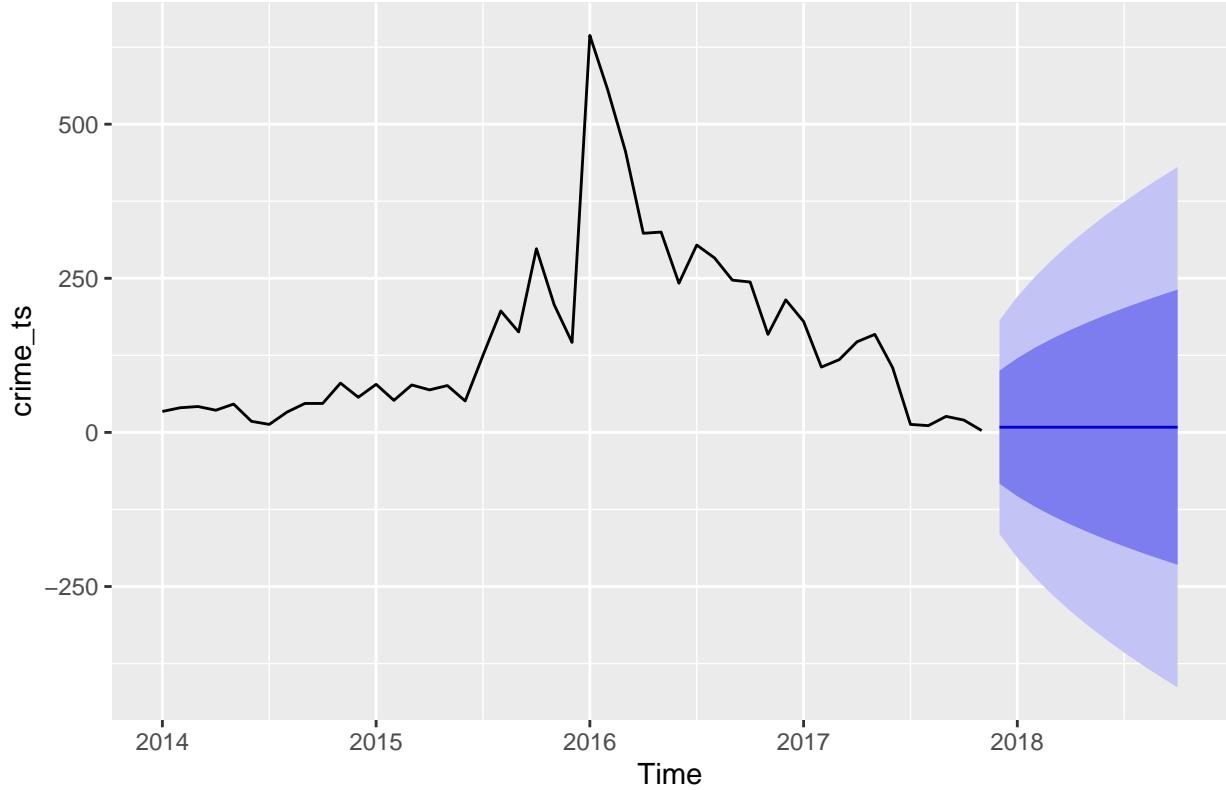
```

## Dec 2017      8.453808 -83.14285 100.0505 -164.7615 181.6691
## Jan 2018     8.453808 -103.49375 120.4014 -203.2464 220.1540
## Feb 2018     8.453808 -120.67622 137.5838 -235.7396 252.6472
## Mar 2018     8.453808 -135.82673 152.7343 -264.3902 281.2978
## Apr 2018     8.453808 -149.53094 166.4386 -290.3058 307.2134
## May 2018     8.453808 -162.13778 179.0454 -314.1462 331.0538
## Jun 2018     8.453808 -173.87501 190.7826 -336.3420 353.2496
## Jul 2018     8.453808 -184.90107 201.8087 -357.1930 374.1006
## Aug 2018     8.453808 -195.33141 212.2390 -376.9175 393.8251
## Sep 2018     8.453808 -205.25329 222.1609 -395.6804 412.5880
## Oct 2018     8.453808 -214.73453 231.6421 -413.6101 430.5177

autoplot(prediction, predict.colour=1,
         main="Prediction using only crime data")

```

Prediction using only crime data



```

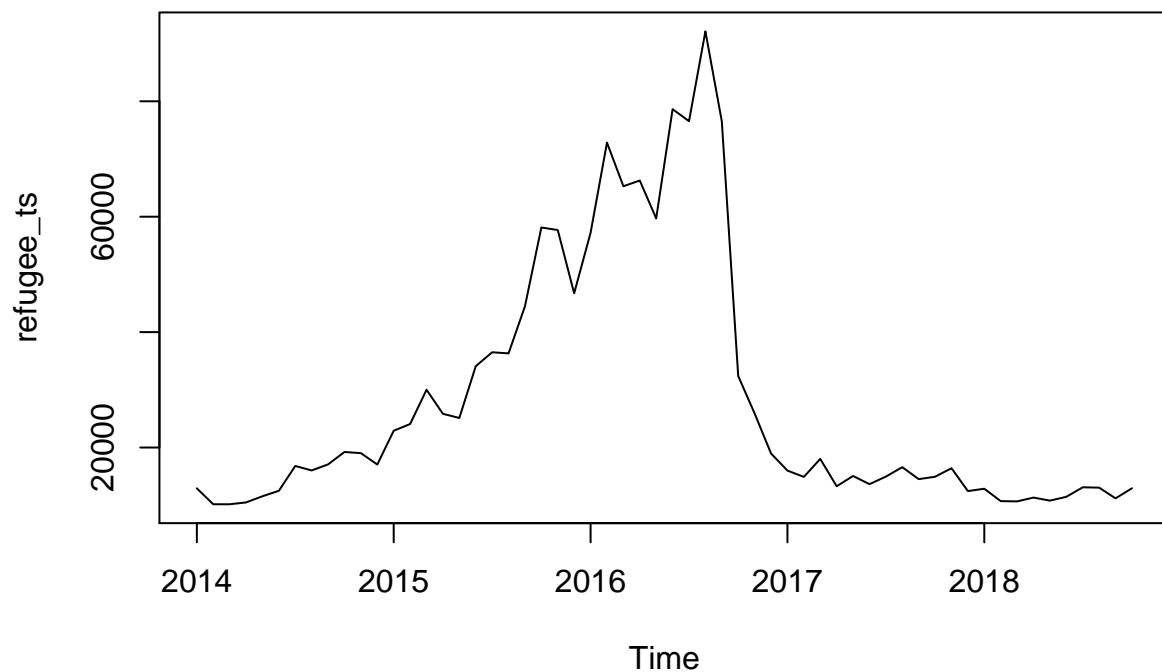
# Loading refugee data
r2 <- subset(refugee, year == 2014 | year == 2015 |
              year == 2016 | year == 2017 | year == 2018)
reff <- aggregate(r2$number, by=list(r2$ym), FUN=sum)
colnames(reff) = c("ym", "number")

refugee_ts <- ts(reff$number, start=c(2014,1), frequency=12)
refugee_ts_train <- window(refugee_ts, end=c(2017,11))
refugee_ts_test <- window(refugee_ts, start=c(2017,12))

# Refugee data
plot(refugee_ts, main="Refugee influx data")

```

Refugee influx data



```
# Model Selection with refugee data
model <- auto.arima(
  y=crime_ts, xreg=refugee_ts_train, ic="aic", max.order=7,
  stepwise=F, approximation=F, parallel=T, num.cores=4
)

# Best Model with refugee data
model

## Series: crime_ts
## Regression with ARIMA(1,0,0) errors
##
## Coefficients:
##             ar1      xreg
##             0.4889   0.0043
## s.e.   0.1354   0.0006
##
## sigma^2 estimated as 6677:  log likelihood=-272.75
## AIC=551.51  AICc=552.07  BIC=557.06

# Prediction with refugee data
prediction <- forecast(
  model, xreg=refugee_ts_test, h=11, level=c(95,70)
)
prediction
```

```
##          Point Forecast      Lo 70      Hi 70      Lo 95      Hi 95
```

```

## Dec 2017      20.72458 -63.96229 105.4115 -139.4239 180.8730
## Jan 2018     39.56307 -54.70129 133.8274 -138.6971 217.8232
## Feb 2018     38.57185 -57.84056 134.9843 -143.7504 220.8941
## Mar 2018     42.46626 -54.45243 139.3849 -140.8134 225.7459
## Apr 2018     47.32990 -49.70939 144.3692 -136.1778 230.8376
## May 2018     46.01044 -51.05765 143.0785 -137.5517 229.5726
## Jun 2018     49.36876 -47.70621 146.4437 -134.2064 232.9439
## Jul 2018     56.77306 -40.30355 153.8497 -126.8052 240.3513
## Aug 2018     56.59580 -40.48120 153.6728 -126.9832 240.1748
## Sep 2018     48.59801 -48.47909 145.6751 -134.9812 232.1772
## Oct 2018     56.23977 -40.83735 153.3169 -127.3395 239.8190

autoplot(prediction, predict.colour=1,
         main="Prediction with refugee data")

```

Prediction with refugee data

