Bootcamp 4: Chloropleth maps

Lucien Baumgartner

10/26/2018

Okay, let's get down to business. In this bootcamp, we will produce a chloropleth map. The starting point is the following question:

Which districts have a lot of households with children?

I will first paraphrase this question to:

What is the median number of children (per household) per district over the last 5 years?

We will see that the answer to this question is simple: zero. Why? Singles kinda rule I guess.

Hence, I rephrase the whole thing to:

What is the median share of households with children per district over the last 5 years?

This should serve a reminder: even if your initial idea is stupid, think about it differently (think about *maximizing variance*), and try again.

The following script includes my whole workflow, including some exploratory stuff, just so that you see how I approached the whole thing.

```
library(dplyr) # data crunching
library(ggplot2) # plots
library(rgdal) # geodata handling
library(viridis) # color scale for lazy people
library(ggrepel) # text label repel
library(extrafont) # different fonts

# clean workspace
rm(list=ls())

# all the registered fonts
fonts()
```

```
##  [1] ".Keyboard"                     "System Font"
##  [3] "Alex Brush"                    "Amatic SC"
##  [5] "Andale Mono"                   "Apple Braille"
##  [7] "AppleMyungjo"                  "Arial Black"
##  [9] "Arial"                         "Arial Narrow"
## [11] "Arial Rounded MT Bold"         "Arial Unicode MS"
## [13] "Bad Script"                    "Bodoni Ornaments"
## [15] "Bodoni 72 Smallcaps"           ""
## [17] "Brush Script MT"               "Cabin Sketch"
## [19] "Codystar"                      "Comic Sans MS"
## [21] "Courier New"                   "Crimson Text"
## [23] "Crimson Text SemiBold"         "Cutive Mono"
## [25] "Dawning of a New Day"          "DIN Alternate"
## [27] "DIN Condensed"                 "Dosis"
## [29] "Economica"                     "Georgia"
## [31] "Goudy Bookletter 1911"         "Impact"
## [33] "Julius Sans One"               "Khmer Sangam MN"
## [35] "Lao Sangam MN"                 "Lekton"
## [37] "Libre Barcode 39 Extended Text" "Libre Barcode 39 Text"
## [39] "Luminari"                      "Megrim"
## [41] "Microsoft Sans Serif"          "Pompiere "
## [43] "PT Serif"                      "Raleway Dots "
## [45] "Sacramento"                    "Source Code Pro Black"
## [47] "Source Code Pro"               "Source Code Pro ExtraLight"
## [49] "Source Code Pro Light"         "Source Code Pro Medium"
## [51] "Source Code Pro Semibold"      "Space Mono"
## [53] "Suranna"                       "Tahoma"
## [55] "Tenor Sans"                    "Text Me One"
## [57] "Times New Roman"               "Trattatello"
## [59] "Trebuchet MS"                  "Verdana"
## [61] "Webdings"                      "Wingdings"
## [63] "Wingdings 2"                   "Wingdings 3"
```

```
# load custom functions available at: https://github.com/lucienbaumgartner/r-helpers
source('~/r-helpers/ggplot/ggplot-helper.R')

# set WD
setwd('~/ddj18/output/')

# load pop data
load('01-bevoelkerung-clean.RData')
str(df)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    9437083 obs. of  15 variables:
##  $ persnum        : int  911747 886098 347792 1073886 17361 966399 604793 797443 552
##  $ anzbestwir     : int  1 1 1 1 1 1 1 1 1 ...
##  $ stichtagdatjahr: int  1993 1993 1993 1993 1993 1993 1993 1993 1993 1993 ...
##  $ alterv05kurz   : chr  "40-44" "25-29" "20-24" "20-24" ...
##  $ sexcd          : int  1 1 1 2 1 1 2 2 1 1 ...
##  $ aufart2lang    : chr  "SchweizerIn" "andere" "andere" "SchweizerIn" ...
##  $ ziv2lang       : chr  "Ledig" "Ledig" "Ledig" "Ledig" ...
##  $ anzahlkinder   : int  0 0 0 0 0 0 3 0 0 1 ...
##  $ hhtyplang      : chr  NA NA NA NA ...
##  $ geblandhistlang: chr  "Asien" "Asien" "Asien" "Asien" ...
##  $ nationhistlang : chr  "Schweiz" "Asien" "Asien" "Schweiz" ...
##  $ kreislang      : chr  "Kreis 2" "Kreis 2" "Kreis 10" "Kreis 10" ...
##  $ quarlang       : chr  "Enge" "Enge" "Wipkingen" "Höngg" ...
##  $ gebnum         : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ ewid           : int  NA NA NA NA NA NA NA NA NA NA ...
```

```
# load shapefile
shp.raw <- readOGR('~/ddj18/input/st-zh/shapefile/Quartier_Shapefile/', layer = 'Statis
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "/Users/lucienbaumgartner/Dropbox/ddj18-data/input/st-zh/Shapefile/Quartier_
## with 34 features
## It has 4 fields
```

```
# inspect shp meta info
str(shp.raw@data)
```

```
## 'data.frame':    34 obs. of  4 variables:
##  $ QNR  : num  92 101 111 21 33 72 74 119 91 23 ...
##  $ QNAME: Factor w/ 34 levels "Affoltern","Albisrieden",..: 4 15 1 34 9 16 33 26 2 1
##  $ KNR  : num  9 10 11 2 3 7 7 11 9 2 ...
##  $ KNAME: Factor w/ 12 levels "Kreis 1","Kreis 10",..: 12 2 3 5 6 10 10 3 12 5 ...
```

```
# we have both kreis and districtict vars!
# this means we will have to join the data by one of the two

## let's compute the median number of children over the last 5 years per district
# number of obs per district
table(df$quarlang)
```

```
##
##             Affoltern          Albisrieden         Alt-Wiedikon
##                525258               444571               387803
##           Altstetten                 City                Enge
##                735195                21880               212536
##          Escher Wyss             Fluntern          Friesenberg
##                 73539               186758               259957
##        Gewerbeschule                 Hard           Hirslanden
##                242783               322196               173751
##           Hirzenbach          Hochschulen               Höngg
##                284778                17912               523133
##            Hottingen          Langstrasse             Leimbach
##                260170               265635               127334
##            Lindenhof            Mühlebach          Oberstrass
##                 24545               141645               252486
##             Oerlikon             Rathaus             Saatlen
##                488149                78130               173552
## Schwamendingen-Mitte             Seebach             Seefeld
##                265649               514926               125552
##             Sihlfeld          Unterstrass             Weinegg
##                522292               518702               122484
##                Werd             Wipkingen             Witikon
##                101721               393761               248979
##          Wollishofen
##                399321
```

```
# number of obs per district in the last five years
df %>%
  filter(stichtagdatjahr%in%2012:2017) %>%
  group_by(quarlang) %>%
  summarise(n=n()) %>%
  print(n=100)
```

```
## # A tibble: 34 x 2
##    quarlang                 n
##    <chr>                <int>
##  1 Affoltern           153944
##  2 Albisrieden         119956
##  3 Alt-Wiedikon        101734
##  4 Altstetten          190369
##  5 City                  4818
##  6 Enge                 54789
##  7 Escher Wyss          29327
##  8 Fluntern             47712
##  9 Friesenberg          64937
## 10 Gewerbeschule        57773
## 11 Hard                 78487
## 12 Hirslanden           43985
## 13 Hirzenbach           71256
## 14 Hochschulen           3930
## 15 Höngg               134484
## 16 Hottingen            65403
## 17 Langstrasse          65689
## 18 Leimbach             35507
## 19 Lindenhof             5714
## 20 Mühlebach            36181
## 21 Oberstrass           63597
## 22 Oerlikon            131826
## 23 Rathaus              19283
## 24 Saatlen              47140
## 25 Schwamendingen-Mitte 66847
## 26 Seebach             147755
## 27 Seefeld              30094
## 28 Sihlfeld            127133
## 29 Unterstrass         132914
## 30 Weinegg              30205
## 31 Werd                 26439
## 32 Wipkingen            95907
## 33 Witikon              62825
## 34 Wollishofen          98806
```

```
# actual stuff that we are interested in:
df %>%
  filter(stichtagdatjahr%in%2012:2017) %>%
  summarise(n.kids=median(anzahlkinder)) %>%
  print(n=100)
```

```
## # A tibble: 1 x 1
##   n.kids
##    <dbl>
## 1      0
```

```
# ... mmh okay.. that's kinda stupid to plot...
# let's compute the share of household with more than 1 kid per district over the last
kids <- df %>% mutate(has.kids=ifelse(anzahlkinder>0,1,0)) %>%
  filter(stichtagdatjahr%in%2012:2017) %>%
  group_by(quarlang, stichtagdatjahr, has.kids) %>%
  summarise(n=n()) %>%
  mutate(p=n/sum(n)) %>%
  ungroup %>%
  group_by(quarlang, has.kids) %>%
  summarise(median.p=median(p)) %>%
  print(n=100)
```

```
## # A tibble: 68 x 3
## # Groups:   quarlang [?]
##    quarlang             has.kids median.p
##    <chr>                   <dbl>    <dbl>
##  1 Affoltern                   0   0.804
##  2 Affoltern                   1   0.196
##  3 Albisrieden                 0   0.832
##  4 Albisrieden                 1   0.168
##  5 Alt-Wiedikon                0   0.851
##  6 Alt-Wiedikon                1   0.149
##  7 Altstetten                  0   0.839
##  8 Altstetten                  1   0.161
##  9 City                        0   0.872
## 10 City                        1   0.128
## 11 Enge                        0   0.841
## 12 Enge                        1   0.159
## 13 Escher Wyss                 0   0.872
## 14 Escher Wyss                 1   0.128
## 15 Fluntern                    0   0.836
## 16 Fluntern                    1   0.164
## 17 Friesenberg                 0   0.770
## 18 Friesenberg                 1   0.230
## 19 Gewerbeschule               0   0.877
## 20 Gewerbeschule               1   0.123
## 21 Hard                        0   0.860
## 22 Hard                        1   0.140
## 23 Hirslanden                  0   0.840
## 24 Hirslanden                  1   0.160
## 25 Hirzenbach                  0   0.803
## 26 Hirzenbach                  1   0.197
## 27 Hochschulen                 0   0.931
## 28 Hochschulen                 1   0.0691
## 29 Höngg                       0   0.830
## 30 Höngg                       1   0.170
## 31 Hottingen                   0   0.839
## 32 Hottingen                   1   0.161
## 33 Langstrasse                 0   0.921
## 34 Langstrasse                 1   0.0791
## 35 Leimbach                    0   0.781
## 36 Leimbach                    1   0.219
## 37 Lindenhof                   0   0.922
## 38 Lindenhof                   1   0.0781
## 39 Mühlebach                   0   0.869
## 40 Mühlebach                   1   0.131
## 41 Oberstrass                  0   0.849
## 42 Oberstrass                  1   0.151
## 43 Oerlikon                    0   0.856
## 44 Oerlikon                    1   0.144
## 45 Rathaus                     0   0.922
## 46 Rathaus                     1   0.0784
## 47 Saatlen                     0   0.778
## 48 Saatlen                     1   0.222
## 49 Schwamendingen-Mitte        0   0.843
## 50 Schwamendingen-Mitte        1   0.157
## 51 Seebach                     0   0.828
## 52 Seebach                     1   0.172
## 53 Seefeld                     0   0.880
```

```
## 54 Seefeld                    1   0.120
## 55 Sihlfeld                   0   0.862
## 56 Sihlfeld                   1   0.138
## 57 Unterstrass                0   0.838
## 58 Unterstrass                1   0.162
## 59 Weinegg                    0   0.859
## 60 Weinegg                    1   0.141
## 61 Werd                       0   0.881
## 62 Werd                       1   0.119
## 63 Wipkingen                  0   0.865
## 64 Wipkingen                  1   0.135
## 65 Witikon                    0   0.832
## 66 Witikon                    1   0.168
## 67 Wollishofen                0   0.840
## 68 Wollishofen                1   0.160
```

```
# already looks more promising

# now we have to check whether the names of the districts in shp actually match those i
table(unique(shp.raw$QNAME)%in%unique(kids$quarlang))
```

```
##
## TRUE
##   34
```

```
# yep! this makes everything a lot easier!

## now let's join the data to the shp
# since we work with ggplot, and we don't do super fancy stuff with geodata, we just co
class(shp.raw)
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
shp <- fortify(shp.raw, region='QNAME') # use QNAME as region (otherwise it will be dro
class(shp)
```

```
## [1] "data.frame"
```

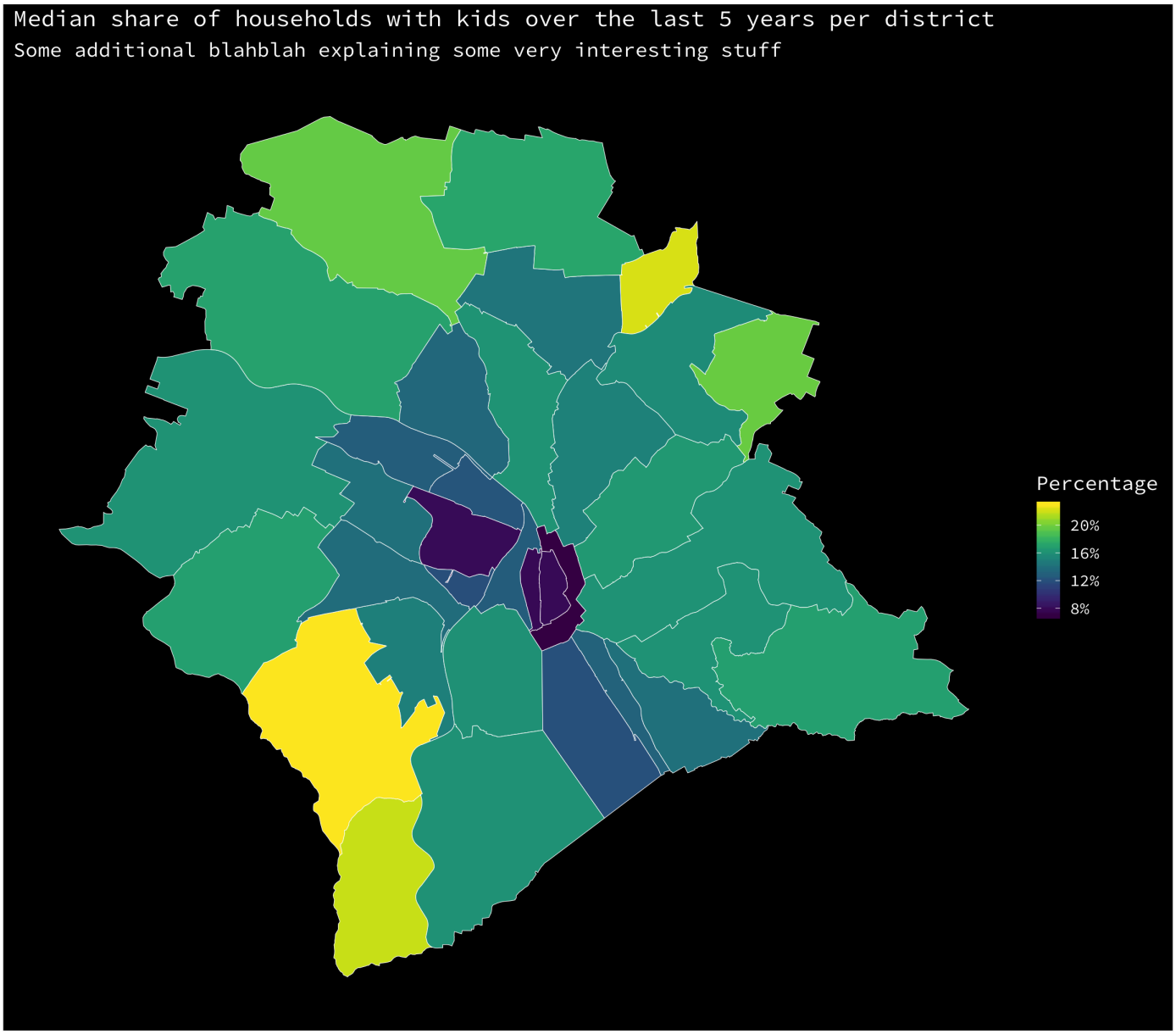
```
# dims before join
dim(shp)
```

```
## [1] 17725     7
```

```
# join; NOTE: use a left join!!!
# shp$region is now shp$id; to join we will have to rename kids$quarlang to kids$id
shp <- left_join(shp, rename(kids, id=quarlang))
# dims after join
dim(shp)
```

```
## [1] 35450     9
```

```
# the number of rows doubled and we got three additional variables...
# why? because kids$quarlang contains every district twice (because we grouped by quarl
# the addtional variables are all other var in kids (those are the reason why we actual
# since we will only plot the share of the pop who has children, we can drop all the ot
kp <- shp %>%
  filter(has.kids==1) %>%
  ggplot(.) + # feed the subset to the ggplot
  geom_polygon(aes(x=long, y=lat, group=group, fill=median.p),
               color='white',
               size=0.2) + # plot the polygon and fill it with the percentages
  coord_equal() + # make x and y equistant
  scale_fill_viridis(option = 'D', name='Percentage',  label=scales::percent) +
  labs(title='Median share of households with kids over the last 5 years per district',
    subtitle='Some additional blahblah explaining some very interesting stuff'
    ) +
  # get rid of the background, axis, grids, etc, and add title and subtitle spacing
  theme_empty(
    title.spacing = 10,
    subtitle.spacing = 10
  ) +
  # add another background, change text color
  theme(
    plot.background = element_rect(fill='black'),
    legend.background = element_rect(fill='black'),
    text = element_text(colour='white', family = 'Source Code Pro'),
    plot.title = element_text(size=17),
    plot.subtitle = element_text(size=15),
    legend.title = element_text(size=15),
    legend.text = element_text(size=12)
  )
kp
```




```
# now we would like to show the reader in which people with kids add up to 20% of the d
# for that we need to compute the anker of the annotation: in this case the district po
centroids <- getSpPPolygonsLabptSlots(shp.raw)
```

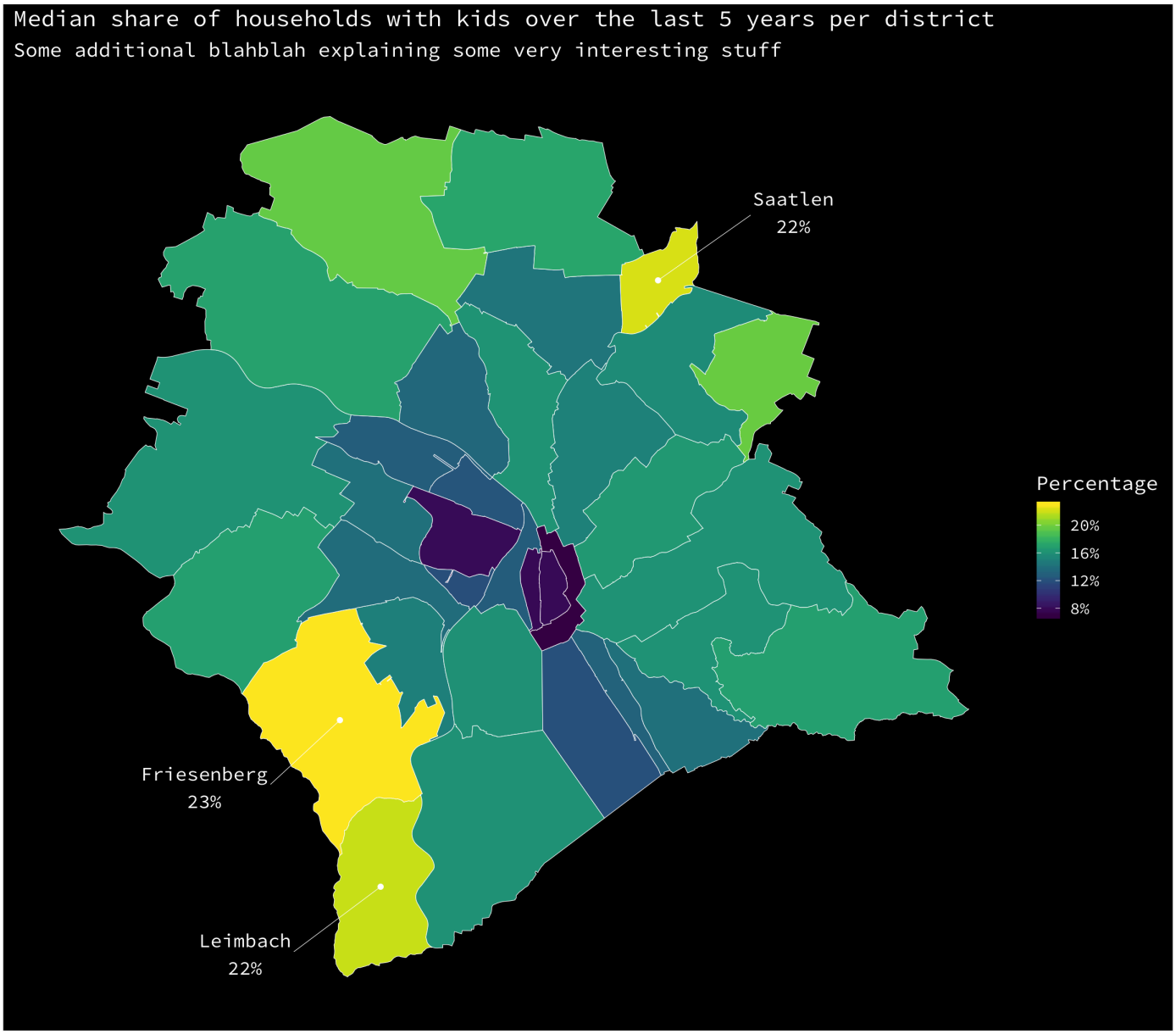
```
## Warning: use coordinates method
```

```
centroids <- centroids[order(shp.raw@data$QNAME),]

# join points to the shp via QNAME
kids <- left_join(kids,
                  # add district names to centroids
                  cbind(as_tibble(centroids),
                        as.character(shp.raw@data$QNAME)[order(shp.raw@data$QNAME)]) %>
                    setNames(., c('long.c', 'lat.c', 'quarlang')),
                  by='quarlang')
```

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

```
# add everything to the plot
kp <- kp +
  # centroids
  geom_point(data=filter(kids, has.kids==1&median.p>.2),
             aes(x=long.c, y=lat.c),
             color='white') +
  # text label for Leimbach and Friesenberg
  geom_text_repel(data=filter(kids, has.kids==1&quarlang%in%c('Friesenberg', 'Leimbach'
                  aes(x=long.c, y=lat.c, label=paste0(quarlang,
                                                      '\n',
                                                      format(round(median.p*100,1), dig
                                                      '%')
                      ),
                  # some additional repel and viz parameters
                  nudge_y = -1000,
                  nudge_x = -2000,
                  size=5,
                  segment.size = 0.2,
                  color='white',
                  family='Source Code Pro') +
  # text label for Saatlen
  geom_text_repel(data=filter(kids, has.kids==1&quarlang%in%c('Saatlen')),
                  aes(x=long.c, y=lat.c, label=paste0(quarlang,
                                                      '\n',
                                                      format(round(median.p*100,1), dig
                                                      '%')
                      ),
                  # some additonal repel and viz parameters
                  nudge_y = 1000,
                  nudge_x = 2000,
                  size=5,
                  segment.size = 0.2,
                  color='white',
                  family='Source Code Pro')
kp
```


Median share of households with kids over the last 5 years per district
Some additional blahblah explaining some very interesting stuff
Saatlen
22%
Friesenberg
23%
Leimbach
22%
Percentage
20%
16%
12%
8%