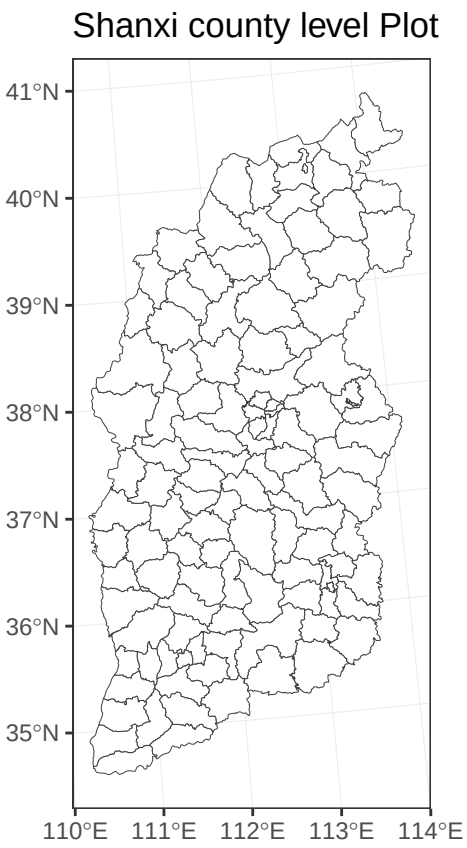
Code

Shanxi county

```
pacman::p_load(data.table, tidyverse, maptools, pinyin)
sx = sf::st_read("data/shp/sx_border.shp")
```

```
## Reading layer `sx_border' from data source `/Users/miaocai/Dropbox/@2018 SPRING SOC5670 Spatial Demog
## Simple feature collection with 118 features and 4 fields
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: 110.2145 ymin: 34.58557 xmax: 114.5565 ymax: 40.7418
## epsg (SRID):    4326
## proj4string:    +proj=longlat +datum=WGS84 +no_defs
```

```
ggplot() +
  geom_sf(data = sx, size = 0.05, color = "#404040", fill = "white") +
  ggtitle("Shanxi county level Plot") +
  coord_sf(crs = "+proj=aeqd +lat_0=37 +lon_0=104") + theme_bw()
```




```
# coord_sf(datum=NA)
```

Including Plots

You can also embed plots, for example:

The location of Shanxi Province in China

```
Sys.setlocale(category ="LC_ALL", locale ="zh_cn.utf-8")
```

```
## [1] "zh_cn.utf-8/zh_cn.utf-8/zh_cn.utf-8/C/zh_cn.utf-8/en_US.UTF-8"
```

```
library("scales")
```

```
##
## Attaching package: 'scales'
```

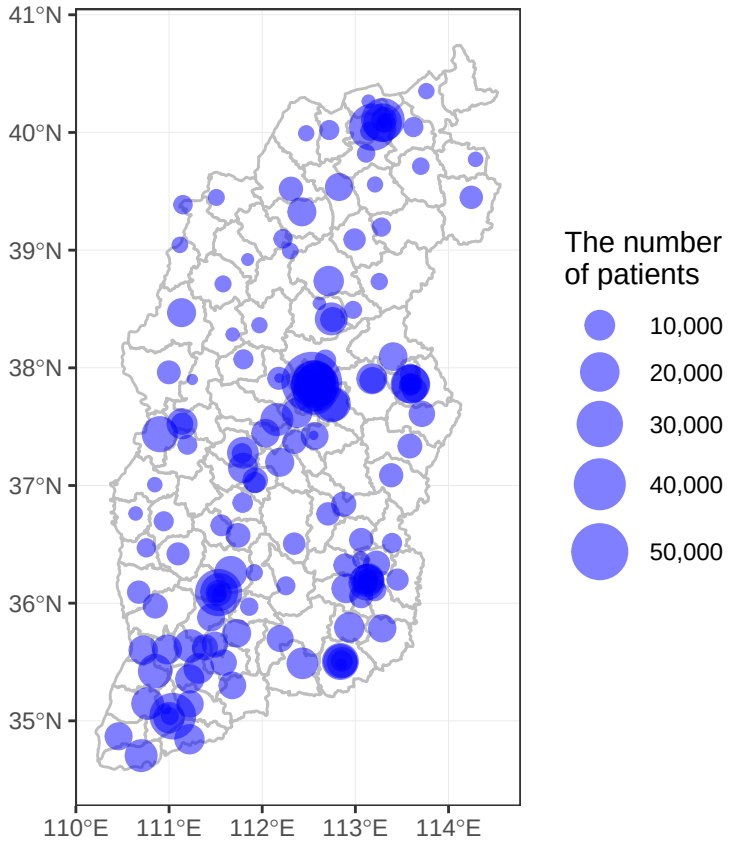
```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
load("data/hloca.Rdata")
names(hloca)[names(hloca) == "longtitude"] = "longitude"

# hloca1 = hloca %>%
#   select(org_id, longitude, latitude, N) %>%
#   sf::st_as_sf(coords = c("longitude", "latitude"),
#                crs = "+proj=aeqd +lat_0=37 +lon_0=104")

ggplot() +
  geom_sf(data = sx, fill = "white", color = "grey") +
  geom_point(data = hloca, aes(x=longitude, y=latitude, size=N),
             shape=21, colour = "blue", fill="blue", alpha = 0.5) +
  scale_size_continuous(labels = comma, range = c(1, 10),
                        name = "The number\nof patients")+
  theme_bw() + theme(axis.title.x=element_blank(),
                     axis.title.y=element_blank())
```




```
ggsave("figs/hospitals.pdf", height=7, width=6)
```

patient choropleth map

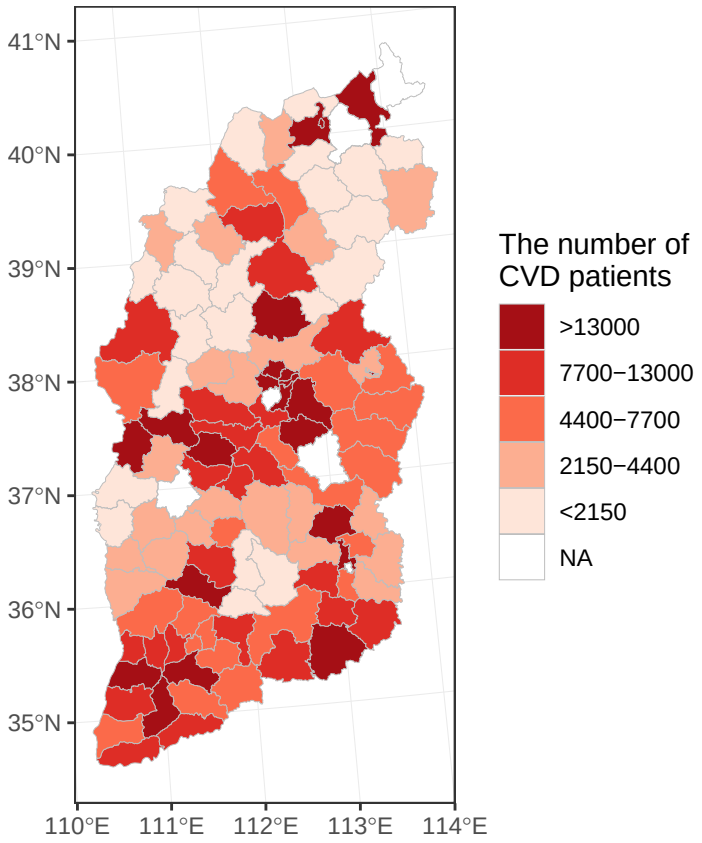
```
# Sys.setlocale(category ="LC_ALL", locale ="zh_cn.utf-8")
sx1 = sf::st_read("data/shp/sx_border.shp") %>%
  left_join(distpat, by = c("ADMINCODE" = "dist_old")) %>%
  mutate(pat_cat = NA,
         pat_cat = ifelse(Npat < 2150, "<2150", pat_cat),
         pat_cat = ifelse(Npat >= 2150 & Npat < 4400, "2150-4400", pat_cat),
         pat_cat = ifelse(Npat >= 4400 & Npat < 7700, "4400-7700", pat_cat),
         pat_cat = ifelse(Npat >= 7700 & Npat < 13000, "7700-13000", pat_cat),
         pat_cat = ifelse(Npat >= 13000, ">13000", pat_cat)) %>%
  mutate(pat_cat = factor(pat_cat, levels =
                            c(">13000", "7700-13000", "4400-7700", "2150-4400", "<2150")))
```

```
## Reading layer `sx_border' from data source `/Users/miaocai/Dropbox/@2018 SPRING SOC5670 Spatial Demog
## Simple feature collection with 118 features and 4 fields
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: 110.2145 ymin: 34.58557 xmax: 114.5565 ymax: 40.7418
## epsg (SRID):    4326
## proj4string:    +proj=longlat +datum=WGS84 +no_defs
```

```
## Warning: Column `ADMINCODE`/`dist_old` joining factor and character vector,
## coercing into character vector
```

```
ggplot() +
  geom_sf(data = sx1, size = 0.1,
```

```
          color = "grey", aes(fill = pat_cat)) + # #404040
  scale_fill_brewer(palette="Reds", direction = -1)+
  coord_sf(crs = "+proj=aeqd +lat_0=37 +lon_0=104") +
  labs(fill = "The number of \nCVD patients") +
  theme_bw() # + geom_sf_text(data = sx1, aes(label = PYNAME), colour = "blue")
```




```
ggsave("figs/choropleth_patients.pdf", height=7, width=6)
```

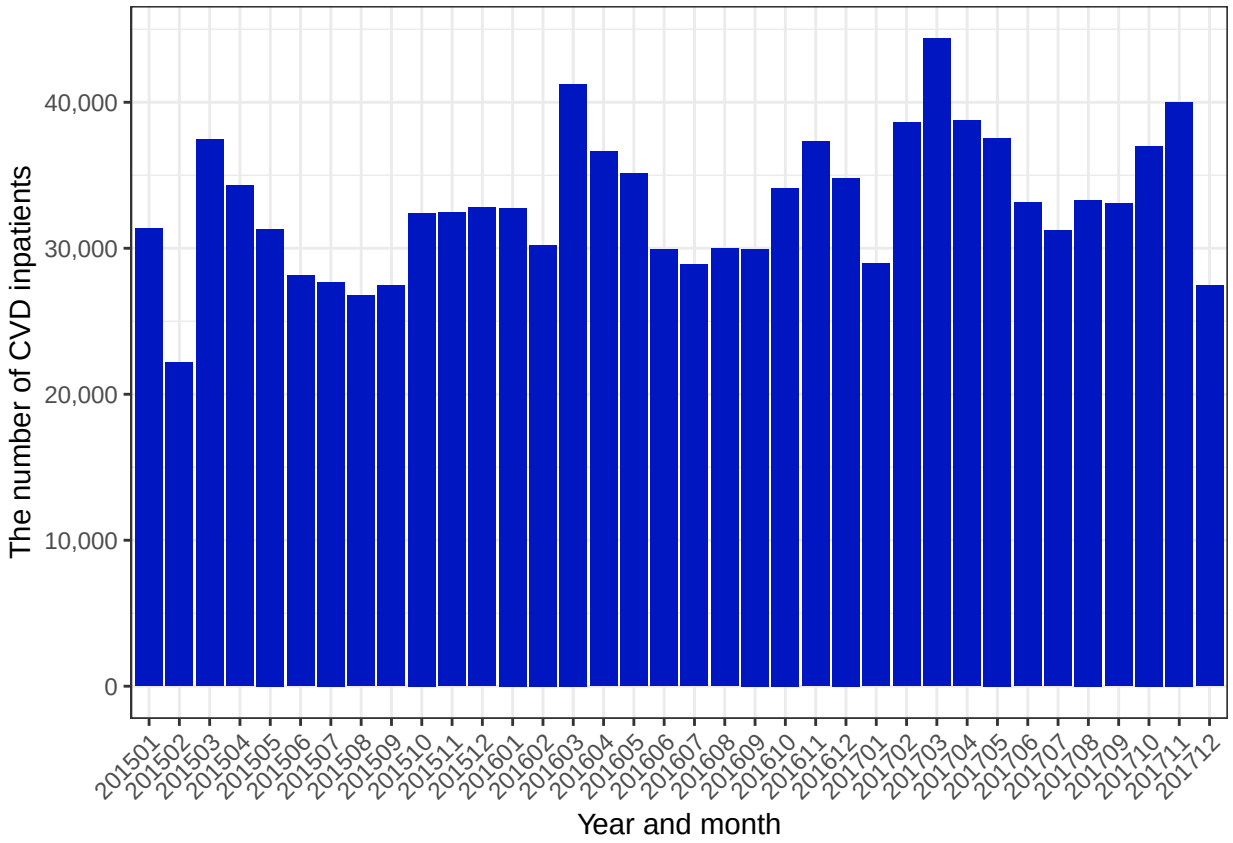
Deal with weather data

```
CVDsx = fread("data/CVDsx.csv")
hosp_dist = fread("data/hosp_district.csv") %>%
  set_names(c("hospid", "district", "city", "county"))

res_date = c(201501:201512, 201601:201612, 201701:201712) %>% as.character()
zdist = CVDsx[, .N, c("hospid", "yearmonth")][
  hosp_dist, on="hospid"][
    ,.(N = sum(N)),
    by = c("district", "city", "county", "yearmonth")][
      ,yearmonth := factor(yearmonth, levels = res_date)]
zcity = CVDsx[, .N, c("hospid", "yearmonth")][
  hosp_dist, on="hospid"][
    ,.(N = sum(N)), by = c("city", "yearmonth")][
      ,yearmonth := factor(yearmonth, levels = res_date)]
```

```
pday = CVDsx[,yearmonth := factor(
  yearmonth, levels = res_date)][,.N, "yearmonth"]


pday %>%
  ggplot(aes(yearmonth, N)) +
  geom_bar(stat = "identity", fill = "#0016c1") + theme_bw()+
  scale_y_continuous(name="The number of CVD inpatients", labels = scales::comma)+
  xlab("Year and month")+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
ggsave("figs/Npat_by_month.pdf", width = 10, height = 6.18)
```

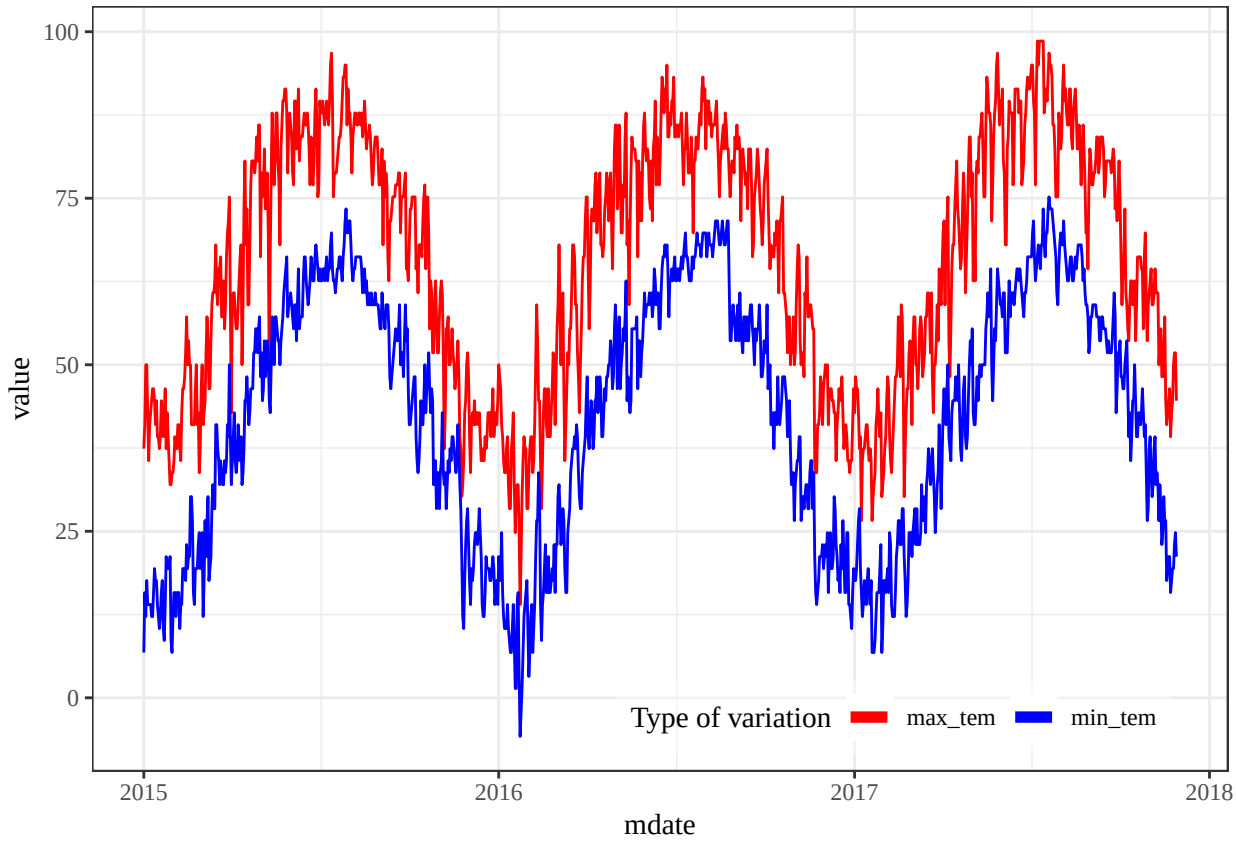
```
require('pinyin')
require(data.table)
mypy = pydic(method = 'toneless', dic = 'pinyin2')
load("data/weather.Rdata")
res_date = c(201501:201512, 201601:201612, 201701:201712) %>%
  as.character()

w = weather %>%
  select(1,2, mdate = ymd, max_tem = bWendu,
         min_tem = yWendu, aqi, aqiLevel) %>%
  mutate(myear = as.integer(substr(mdate, 1, 4))) %>%
  filter(myear >= 2015) %>%
  mutate(mdate = lubridate::ymd(mdate),
```

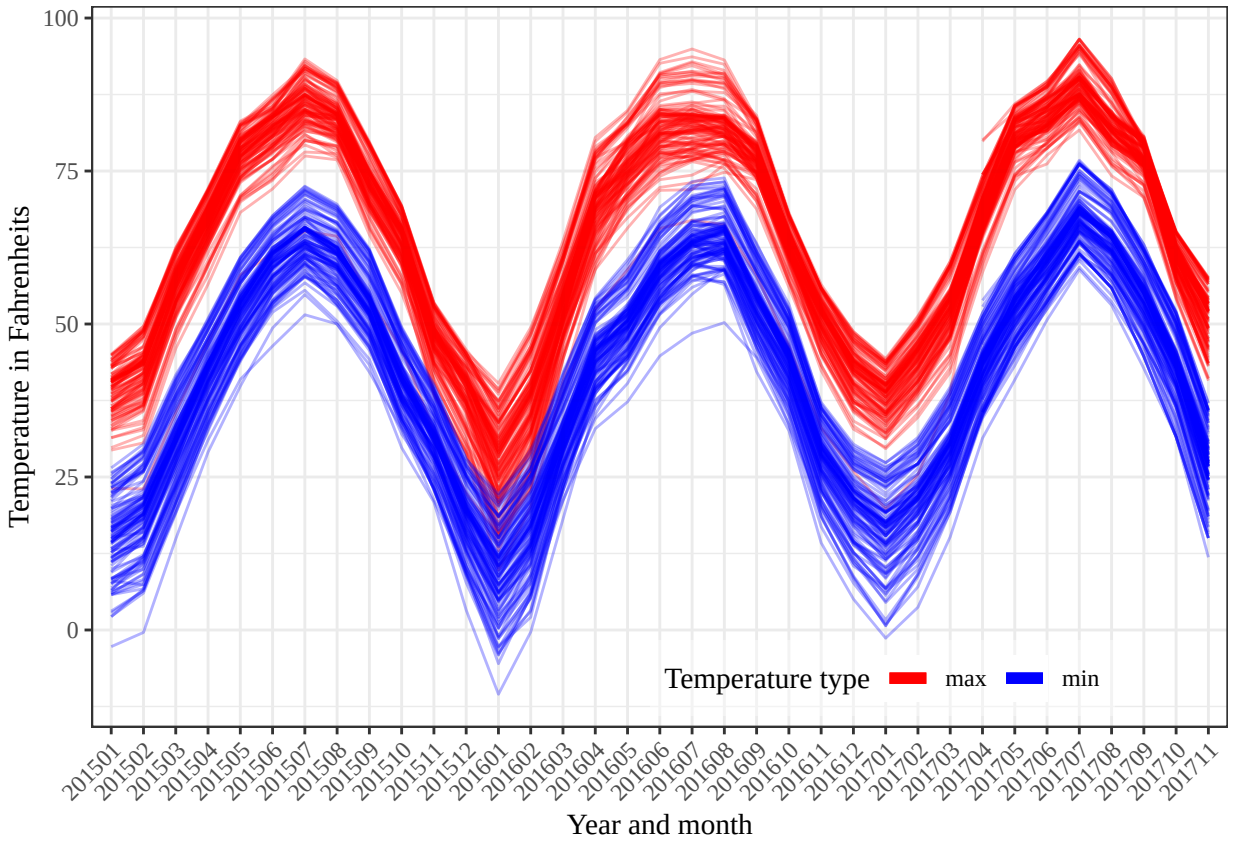
```
         max_tem = gsub("([0-9]+).*$", "\\1",max_tem) %>% as.numeric,
         min_tem = gsub("([0-9]+).*$", "\\1",min_tem) %>% as.numeric,
         yearmonth = factor(paste0(substr(mdate, 1, 4),
                                   substr(mdate, 6, 7))),
         city = py(city,  dic = mypy, sep = ''),
         county = py(county,  dic = mypy, sep = ''),
         tem_diff = max_tem - min_tem,
         tem_mean = (max_tem + min_tem)/2) %>%
  mutate(city = gsub('[[:digit:]]+', '', city),
         county = gsub('[[:digit:]]+', '', county)) %>%
  arrange(city, county, mdate)

c2f = function(x) return((x*9/5) + 32)

w %>%
  filter(county == "taiyuan") %>%
  select(mdate, max_tem, min_tem) %>%
  mutate(max_tem = c2f(max_tem),
         min_tem = c2f(min_tem)) %>%
  gather(key = "tem_type", value = "value", -mdate) %>%
  ggplot(aes(mdate, value, color = tem_type)) +
  geom_line() + scale_color_manual(values=c("red", "blue"))+
  theme_bw()+
  theme(legend.justification = c(1, 1), legend.position = c(0.95, 0.12),
        legend.background = element_rect(fill=alpha('white', 0.4)),
        legend.direction="horizontal", text=element_text(family="Times"))+
  guides(color=guide_legend(title="Type of variation",
                            override.aes = list(alpha = 1, size = 2)))
```




```
w %>%
  group_by(county, yearmonth) %>%
  summarise(min_tem = mean(min_tem),
            max_tem = mean(max_tem)) %>%
  ungroup() %>%
  mutate(max_tem = c2f(max_tem),
         min_tem = c2f(min_tem)) %>%
  ggplot() +
  geom_line(aes(yearmonth, max_tem, group = county,
            color = "max"), alpha = 0.3) +
  geom_line(aes(yearmonth, min_tem, group = county,
                color = "min"), alpha = 0.3) +
  scale_colour_manual(name="Line Color",
    values = c(max = "red", min = "blue")) +
  theme_bw() + ylab("Temperature in Fahrenheits") +
  xlab("Year and month") +
  theme(legend.justification = c(1, 1), legend.position = c(0.9, 0.12),
        legend.background = element_rect(fill=alpha('white', 0.4)),
        legend.direction="horizontal", text=element_text(family="Times"),
        axis.text.x = element_text(angle = 45, hjust = 1))+
  guides(color=guide_legend(title="Temperature type",
                            override.aes = list(alpha = 1, size = 2)))
```

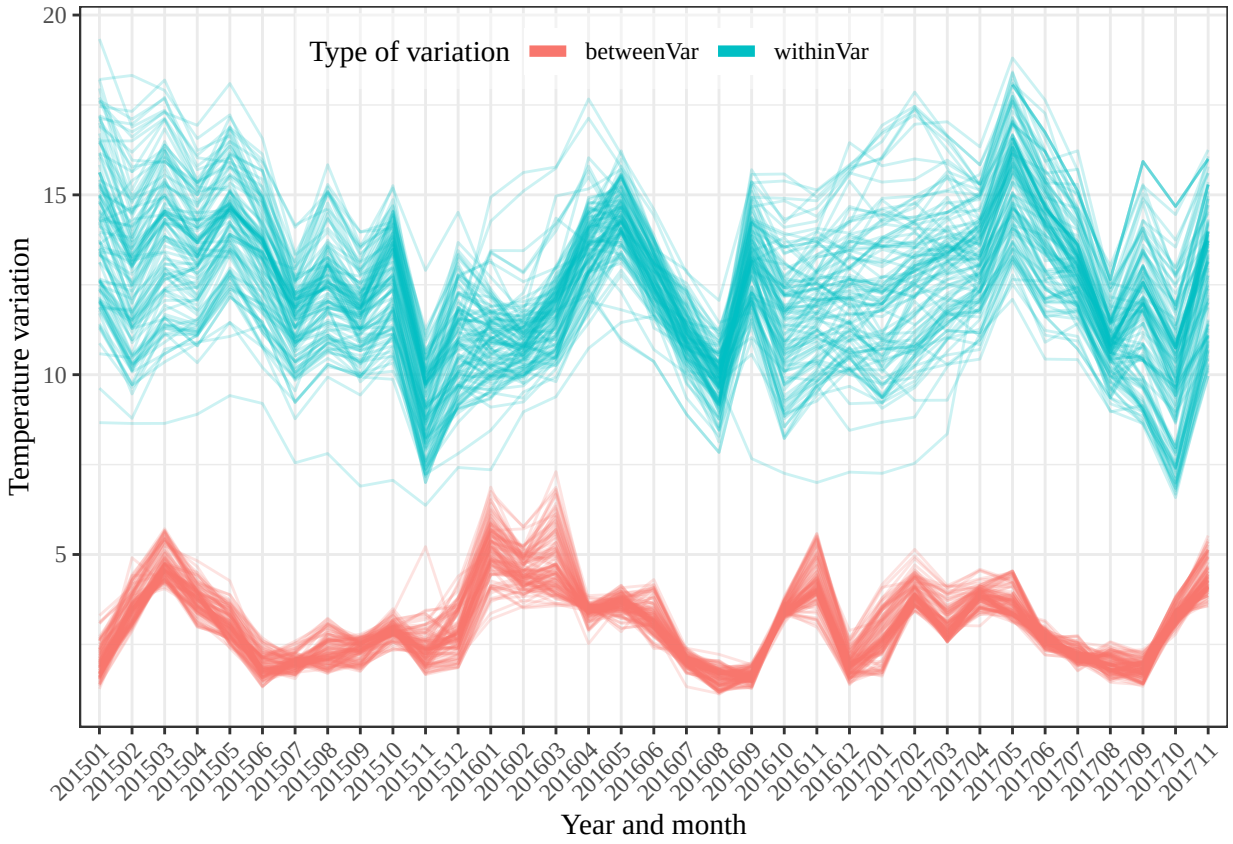



```
ggsave("figs/temperature_in_groups.pdf", width = 10, height = 6.18)
```

```
lagterm = 7
aw = w %>%
  group_by(city, county) %>%
  mutate(lagtem_mean = lag(tem_mean, lagterm),
         lagtem_diff = lag(tem_diff, lagterm)) %>%
  ungroup %>%
  group_by(city, county, yearmonth) %>%
  summarise(betweenVar = sd(lagtem_mean, na.rm = TRUE),
            withinVar = mean(lagtem_diff, na.rm = TRUE)) %>%
  ungroup()

aw %>%
  gather("var_type", "value", -city, -county, -yearmonth) %>%
  mutate(group0 = paste0(city, county, var_type)) %>%
  ggplot(aes(yearmonth, value, group = group0, color = var_type)) +
  geom_line(alpha = 0.2) + theme_bw() +
  xlab("Year and month")+ ylab("Temperature variation") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))+
  theme(legend.justification = c(1, 1), legend.position = c(0.7, 0.99),
        legend.background = element_rect(fill=alpha('white', 0.4)),
        legend.direction="horizontal", text=element_text(family="Times"))+
  guides(color=guide_legend(title="Type of variation",
                            override.aes = list(alpha = 1, size = 2)))
```

```
## Warning: Removed 24 rows containing missing values (geom_path).
```




```
ggsave("figs/two_tem_variation.pdf", width = 10, height = 6.18)
```

```
## Warning: Removed 24 rows containing missing values (geom_path).
```