SSPs Human Capitals - National

James Millington

Libraries

```
library(tidyr)
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
library(readxl)
library(ggplot2)
library(ggrepel)
```

Load Data

```
load_ssp <- function(filepath,varlab){
  #for each SSP data sheet
  for(n in 1:5){
    ssp <- paste0("ssp", n)
    sheet <- read_excel(filepath, sheet = ssp, na='<Null>')
```

```
    #pivot longer, add ssp id column
    lsheet <- sheet %>%
      pivot_longer(
        cols = !OID:unit,
        names_to='year',
        values_to='value'
      ) %>%
      mutate(ssp=n)

    #calculate ranks of values
    lranks <- lsheet %>%
      group_by(year) %>%
      mutate(unit='rank',
             value=min_rank(desc(value)))

    #combine into single dataframe
    ldat <- rbind(lsheet,lranks)
    if(n > 1){
      alldat <- rbind(alldat,ldat)
    } else { alldat <- ldat}
  }
  alldat <- mutate(alldat,variable=varlab)
  return(alldat)
}
```

```
edu <- load_ssp("data/edu.xlsx", "edu")
ma <- load_ssp("data/ma.xlsx", "ma")

countries <-edu %>%
  select(OID, GID_0) %>%
  distinct()
```

```
#sum of squared differences function
ssd <- function(x) {
  med = median(x, na.rm=TRUE)
  return(sum(sqrt((x-med)^2)))
}

out = countries
for(nm in list(edu,ma)){
```

```
  ssdnm = paste0(nm$variable[1],"_ssd")
  mednm = paste0(nm$variable[1],"_med")
  sumnm = paste0(nm$variable[1],"_sum")
  out <- nm %>%
  #edu %>%
    filter(unit=='rank') %>%
    group_by(GID_0, ssp) %>%
    #calc median of the ssp mean year rank for each ssp
    summarise(medrank = median(value, na.rm=TRUE)) %>%
    group_by(GID_0) %>%
    #calc sum of sq differences from ssp median
    summarise(!!ssdnm := ssd(medrank),
              !!mednm := median(medrank),
              !!sumnm := sum(medrank)) %>%
    left_join(out, ., by='GID_0')
}
```

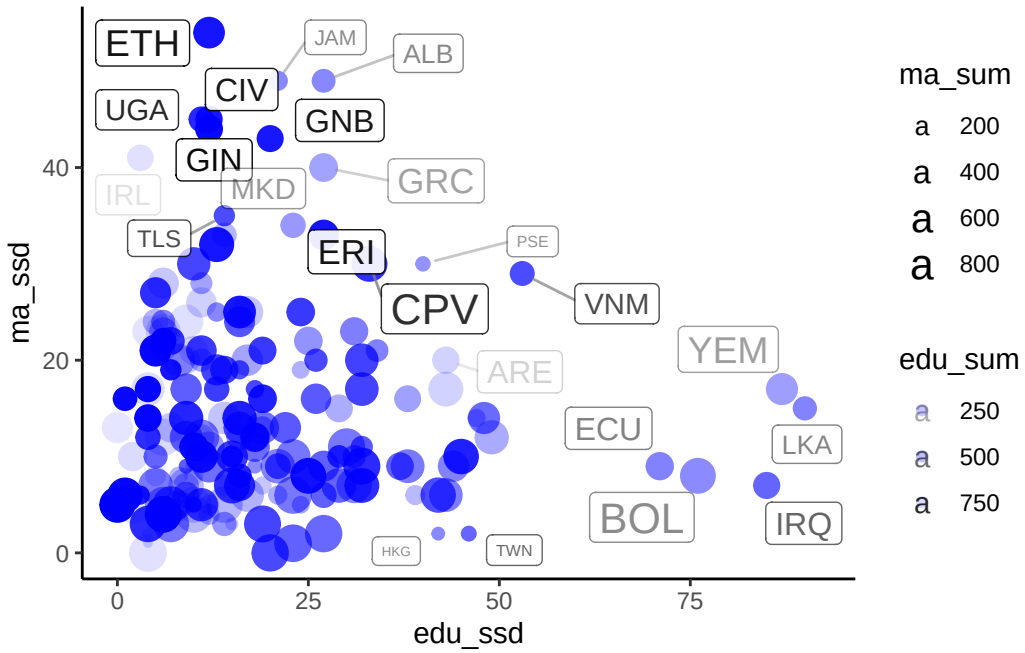
```
`summarise()` has grouped output by 'GID_0'. You can override using the
`.groups` argument.
`summarise()` has grouped output by 'GID_0'. You can override using the
`.groups` argument.
```

```
out %>%
  ggplot(aes(x=edu_ssd, y=ma_ssd, alpha=edu_sum, size=ma_sum)) +
  geom_point(color = "blue", fill="black") +
  geom_label_repel(aes(label = GID_0),
                   box.padding   = 0.35,
                   point.padding = 0.5,
                   segment.color = 'grey50') +
  theme_classic()
```

```
Warning: Removed 30 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 30 rows containing missing values (`geom_label_repel()`).
```

```
Warning: ggrepel: 151 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

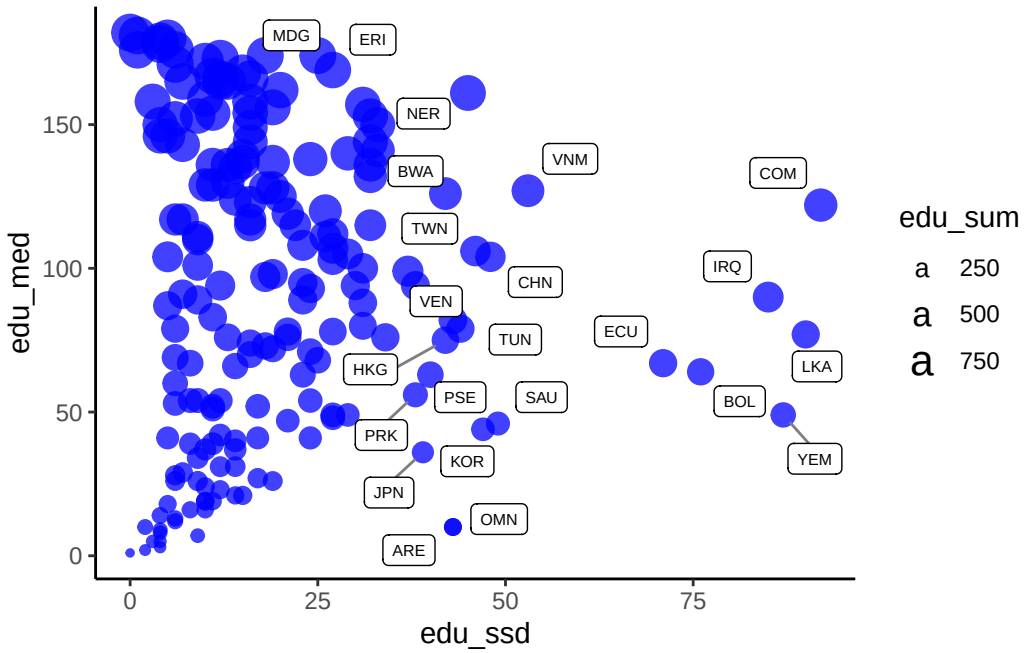



```
out %>%
  ggplot(aes(x=edu_ssd, y=edu_med, size=edu_sum)) +
  geom_point(color = "blue", fill="black", alpha=0.75) +
  geom_label_repel(aes(label = GID_0, size = 50),
                   box.padding   = 0.35,
                   point.padding = 0.5,
                   segment.color = 'grey50') +
  theme_classic()
```

```
Warning: Removed 22 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 22 rows containing missing values (`geom_label_repel()`).
```

```
Warning: ggrepel: 159 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```

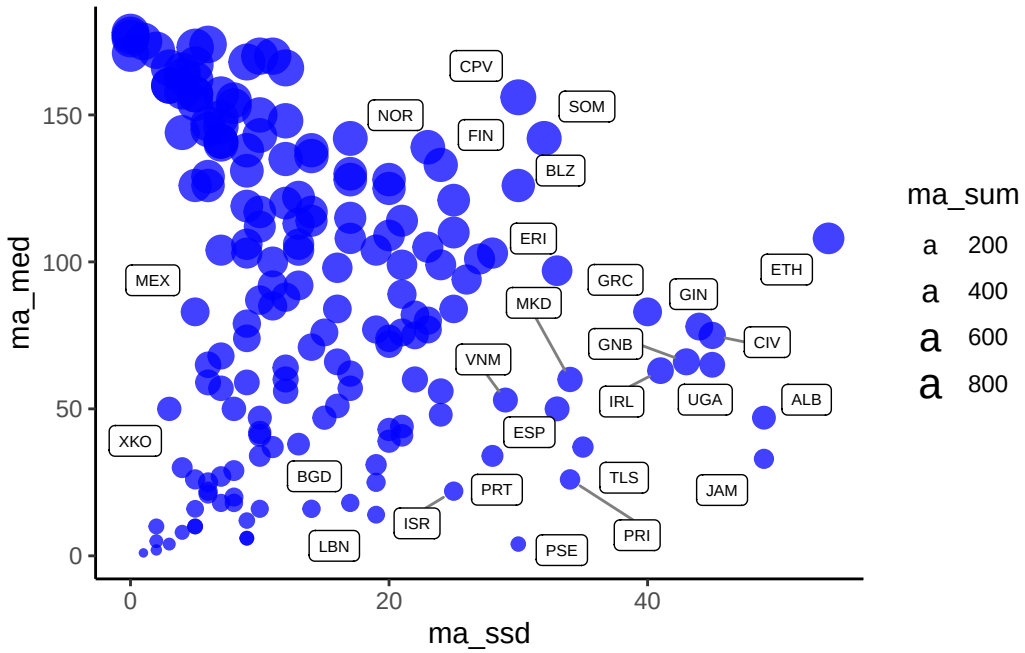



```
out %>%
  ggplot(aes(x=ma_ssd, y=ma_med, size=ma_sum)) +
  geom_point(color = "blue", fill="black", alpha=0.75) +
  geom_label_repel(aes(label = GID_0, size = 50),
                   box.padding   = 0.35,
                   point.padding = 0.5,
                   segment.color = 'grey50') +
  theme_classic()
```

```
Warning: Removed 26 rows containing missing values (`geom_point()`).
```

```
Warning: Removed 26 rows containing missing values (`geom_label_repel()`).
```

```
Warning: ggrepel: 151 unlabeled data points (too many overlaps). Consider
increasing max.overlaps
```


CPV
SOM
NOR
FIN
BLZ
ERI
ETH
MEX
GRC
GIN
MKD
GNB
CIV
VNM
IRL
UGA
ALB
ESP
XKO
BGD
TLS
PRT
JAM
ISR
PRI
LBN
PSE
150
100
50
0
0
20
40
ma_med
ma_ssd
ma_sum
a 200
a 400
a 600
a 800