探究的データ分析：難民

鈴木寛

2024年2月20日

# 難民人口

世界の出身国（origin）ごとの難民数と、庇護国（asylum）ごとの難民数について、特に、世界の7つの地域における難民と、紛争下にある、シリア、ウクライナ、アフガニスタン、南スーダン、ミャンマー、今後民主共和国、トルコ、パレスチナに焦点を合わせ調べる。

## データ

### データ情報

1. 難民受け入れ人口：Refugee population by country or territory of asylum：SM.POP.REFG [[Link](https://data.worldbank.org/indicator/SM.POP.REFG)]
2. 難民となる人口：Refugee population by country or territory of origin：SM.POP.REFG.OR [[Link](https://data.worldbank.org/indicator/SM.POP.REFG.OR)]

* 難民の庇護国とは、庇護請求が提出され、認められた国のことである。出身国とは、一般的に請求者の国籍または市民権のある国を指す。
* **Description:** Refugees are people who are recognized as refugees under the 1951 Convention Relating to the Status of Refugees or its 1967 Protocol, the 1969 Organization of African Unity Convention Governing the Specific Aspects of Refugee Problems in Africa, people recognized as refugees in accordance with the UNHCR statute, people granted refugee-like humanitarian status, and people provided temporary protection. Asylum seekers–people who have applied for asylum or refugee status and who have not yet received a decision or who are registered as asylum seekers–are excluded. Palestinian refugees are people (and their descendants) whose residence was Palestine between June 1946 and May 1948 and who lost their homes and means of livelihood as a result of the 1948 Arab-Israeli conflict. Country of asylum is the country where an asylum claim was filed and granted. Country of origin generally refers to the nationality or country of citizenship of a claimant.
* **概要:** 難民とは、「難民の地位に関する1951年条約」またはその「1967年議定書」、「アフリカにおける難民問題の特定の側面を管理する1969年アフリカ統一機構条約」に基づき難民と認められた人、UNHCRの法令に基づき難民と認められた人、難民に準ずる人道的地位を与えられた人、一時的な保護を与えられた人を指す。庇護申請者–庇護または難民認定を申請し、まだ決定を受けていない人、または庇護申請者として登録されている人–は除外される。パレスチナ難民とは、1946年6月から1948年5月までの間に居住地がパレスチナであり、1948年のアラブ・イスラエル紛争の結果、住居と生計手段を失った人々（およびその子孫）である。庇護国とは、庇護請求が提出され、認められた国のことである。出身国とは、一般的に請求者の国籍または市民権のある国を指す。

### データの取得

#### 準備

*これまでとも同じように二つのパッケージを読み込み（load）ます。*

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.4.4 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(WDI)

library(showtext)

## Loading required package: sysfonts

## Loading required package: showtextdb

# showtext\_auto()  
# showtext\_opts(dpi = 300)  
knitr::opts\_chunk$set(fig.showtext=TRUE)  
#showtext\_auto(FALSE) # if you do not use showtext\_auto()

*変数名を指定します。ここでは、refugee\_asylum と、refugee\_origin としました。なるべくわかりやすい名前にします。*

*この変数名は、今後使いますから、重要です。一応、例として、わたしが使った、変数名を書いてあります。他の変数名を使っても構いません。extra = TRUE とすると、地域名（region）や、収入レベル（income）などが追加されます。*

df\_refugee <- WDI(indicator = c(refugee\_asylum = "SM.POP.REFG",   
 refugee\_origin = "SM.POP.REFG.OR"),  
 extra = TRUE)

*data ディレクトリに書き込んでおきます。*

write\_csv(df\_refugee, "data/refugee.csv")

*data ディレクトリから読み出します。Run All や、Run All Chunks Above をする時は、上の二つをスキップする設定にしてあります。*

df\_refugee <- read\_csv("data/refugee.csv")

## Rows: 16758 Columns: 14  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (7): country, iso2c, iso3c, region, capital, income, lending  
## dbl (5): year, refugee\_asylum, refugee\_origin, longitude, latitude  
## lgl (1): status  
## date (1): lastupdated  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

### データの確認

*データ名で、中身を表示できます。head(df\_refugee) とすると、6行表示されます。*

df\_refugee

## # A tibble: 16,758 × 14  
## country iso2c iso3c year status lastupdated refugee\_asylum refugee\_origin  
## <chr> <chr> <chr> <dbl> <lgl> <date> <dbl> <dbl>  
## 1 Afghanist… AF AFG 2014 NA 2023-12-19 300421 2596259  
## 2 Afghanist… AF AFG 1971 NA 2023-12-19 NA NA  
## 3 Afghanist… AF AFG 2006 NA 2023-12-19 34 2107510  
## 4 Afghanist… AF AFG 2013 NA 2023-12-19 16861 2556483  
## 5 Afghanist… AF AFG 1995 NA 2023-12-19 19605 2679132  
## 6 Afghanist… AF AFG 2005 NA 2023-12-19 31 2166139  
## 7 Afghanist… AF AFG 1997 NA 2023-12-19 5 2676675  
## 8 Afghanist… AF AFG 1998 NA 2023-12-19 NA 2667118  
## 9 Afghanist… AF AFG 1999 NA 2023-12-19 NA 2601690  
## 10 Afghanist… AF AFG 1996 NA 2023-12-19 18779 2674234  
## # ℹ 16,748 more rows  
## # ℹ 6 more variables: region <chr>, capital <chr>, longitude <dbl>,  
## # latitude <dbl>, income <chr>, lending <chr>

*変数の情報を得ることができます。*

str(df\_refugee)

## spc\_tbl\_ [16,758 × 14] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ country : chr [1:16758] "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...  
## $ iso2c : chr [1:16758] "AF" "AF" "AF" "AF" ...  
## $ iso3c : chr [1:16758] "AFG" "AFG" "AFG" "AFG" ...  
## $ year : num [1:16758] 2014 1971 2006 2013 1995 ...  
## $ status : logi [1:16758] NA NA NA NA NA NA ...  
## $ lastupdated : Date[1:16758], format: "2023-12-19" "2023-12-19" ...  
## $ refugee\_asylum: num [1:16758] 300421 NA 34 16861 19605 ...  
## $ refugee\_origin: num [1:16758] 2596259 NA 2107510 2556483 2679132 ...  
## $ region : chr [1:16758] "South Asia" "South Asia" "South Asia" "South Asia" ...  
## $ capital : chr [1:16758] "Kabul" "Kabul" "Kabul" "Kabul" ...  
## $ longitude : num [1:16758] 69.2 69.2 69.2 69.2 69.2 ...  
## $ latitude : num [1:16758] 34.5 34.5 34.5 34.5 34.5 ...  
## $ income : chr [1:16758] "Low income" "Low income" "Low income" "Low income" ...  
## $ lending : chr [1:16758] "IDA" "IDA" "IDA" "IDA" ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. country = col\_character(),  
## .. iso2c = col\_character(),  
## .. iso3c = col\_character(),  
## .. year = col\_double(),  
## .. status = col\_logical(),  
## .. lastupdated = col\_date(format = ""),  
## .. refugee\_asylum = col\_double(),  
## .. refugee\_origin = col\_double(),  
## .. region = col\_character(),  
## .. capital = col\_character(),  
## .. longitude = col\_double(),  
## .. latitude = col\_double(),  
## .. income = col\_character(),  
## .. lending = col\_character()  
## .. )  
## - attr(\*, "problems")=<externalptr>

## 変形

### 変数の選択（selecting）

df\_ref <- df\_refugee |>   
 select(country, iso2c, year, refugee\_asylum, refugee\_origin, region, income)  
df\_ref

## # A tibble: 16,758 × 7  
## country iso2c year refugee\_asylum refugee\_origin region income   
## <chr> <chr> <dbl> <dbl> <dbl> <chr> <chr>   
## 1 Afghanistan AF 2014 300421 2596259 South Asia Low income  
## 2 Afghanistan AF 1971 NA NA South Asia Low income  
## 3 Afghanistan AF 2006 34 2107510 South Asia Low income  
## 4 Afghanistan AF 2013 16861 2556483 South Asia Low income  
## 5 Afghanistan AF 1995 19605 2679132 South Asia Low income  
## 6 Afghanistan AF 2005 31 2166139 South Asia Low income  
## 7 Afghanistan AF 1997 5 2676675 South Asia Low income  
## 8 Afghanistan AF 1998 NA 2667118 South Asia Low income  
## 9 Afghanistan AF 1999 NA 2601690 South Asia Low income  
## 10 Afghanistan AF 1996 18779 2674234 South Asia Low income  
## # ℹ 16,748 more rows

*データには country のところに、国だけでなく、地域も入っているので、地域のリストを、iso2c で REGION に入れておきます。*

REGION <- c("1A", "1W", "4E", "6F", "6N", "6X", "7E", "8S", "A4", "A5",   
"A9", "B1", "B2", "B3", "B4", "B6", "B7", "B8", "C4", "C5", "C6",   
"C7", "C8", "C9", "D2", "D3", "D4", "D5", "D6", "D7", "EU", "F1",   
"F6", "M1", "M2", "N6", "OE", "R6", "S1", "S2", "S3", "S4", "T2",   
"T3", "T4", "T5", "T6", "T7", "V1", "V2", "V3", "V4", "XC", "XD",   
"XE", "XF", "XG", "XH", "XI", "XJ", "XL", "XM", "XN", "XO", "XP",   
"XQ", "XT", "XU", "XY", "Z4", "Z7", "ZB", "ZF", "ZG", "ZH", "ZI",   
"ZJ", "ZQ", "ZT")

*地域名にはどのようなものがあるか見ておきます。下の、地域と、国名を選択するときに参考にします。*

df\_ref |> filter(iso2c %in% REGION) |> distinct(country, iso2c)

## # A tibble: 49 × 2  
## country iso2c  
## <chr> <chr>  
## 1 Africa Eastern and Southern ZH   
## 2 Africa Western and Central ZI   
## 3 Arab World 1A   
## 4 Caribbean small states S3   
## 5 Central Europe and the Baltics B8   
## 6 Early-demographic dividend V2   
## 7 East Asia & Pacific Z4   
## 8 East Asia & Pacific (excluding high income) 4E   
## 9 East Asia & Pacific (IDA & IBRD countries) T4   
## 10 Euro area XC   
## # ℹ 39 more rows

*国名も、地域（region）と、所得レベル（income）と共に、表示しておきます。*

df\_ref |> filter(!(iso2c %in% REGION)) |> distinct(country, iso2c, region, income)

## # A tibble: 217 × 4  
## country iso2c region income   
## <chr> <chr> <chr> <chr>   
## 1 Afghanistan AF South Asia Low income   
## 2 Albania AL Europe & Central Asia Upper middle income  
## 3 Algeria DZ Middle East & North Africa Lower middle income  
## 4 American Samoa AS East Asia & Pacific Upper middle income  
## 5 Andorra AD Europe & Central Asia High income   
## 6 Angola AO Sub-Saharan Africa Lower middle income  
## 7 Antigua and Barbuda AG Latin America & Caribbean High income   
## 8 Argentina AR Latin America & Caribbean Upper middle income  
## 9 Armenia AM Europe & Central Asia Upper middle income  
## 10 Aruba AW Latin America & Caribbean High income   
## # ℹ 207 more rows

### 分析する国のリスト

世界の難民を、7つの地域別に調るとともに、個人的に、関心がある、８カ国を選んで調べてみることにする。

世界銀行は、世界を７地域に分けています。

regions <- c("South Asia", "Europe & Central Asia",   
"Middle East & North Africa", "East Asia & Pacific", "Sub-Saharan Africa",   
"Latin America & Caribbean", "North America")

*自分で分析したい国のリストを作ってください。ここでは、紛争に関係がありそうな国を CONFLICTS にします。*

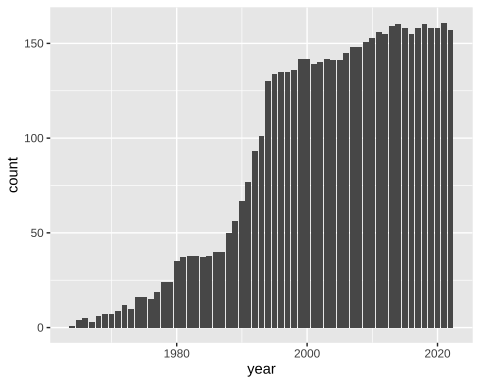
CONFLICTS <- c("Syrian Arab Republic", "Ukraine", "Afghanistan", "South Sudan", "Myanmar", "Congo, Dem. Rep.", "Turkiye", "West Bank and Gaza")

## 視覚化

### 1. 各年毎のデータの数の棒グラフ

*まずは、NA の値を削除します。そのあとで、国の情報の数を数えたいので、!(iso2c %in% REGION) で、上でおいた、地域の iso2c を選択し、! は否定でしたから、地域ではないものを選択し、その数を棒グラフにしています。*

df\_ref |> drop\_na(refugee\_asylum, refugee\_origin) |> filter(!(iso2c %in% REGION)) |>  
 ggplot(aes(year)) + geom\_bar()



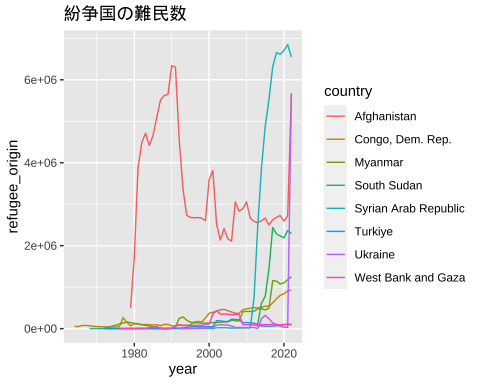
**気づいたこと・疑問**

* 1995年ごろからは、十分データがあるが、その前は少ない。2022年までありそう。

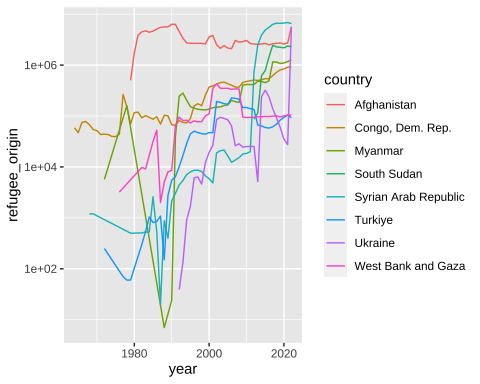
### 2. 経年変化を表す折れ線グラフ

CONFLICTS は、country ですから、下のようにします。6e+06 = 1,000,000 などの、記法になっていますが、0 に近いところが多くなってしまうので、log10 スケールにしてみてみます。

df\_ref |> filter(country %in% CONFLICTS) |> drop\_na(refugee\_origin) |>  
 ggplot(aes(year, refugee\_origin, col = country)) + geom\_line() +  
 labs(title = "紛争国の難民数")



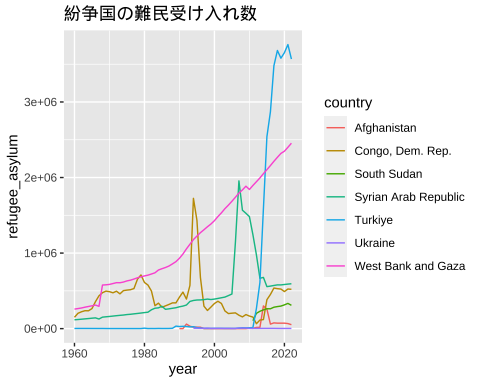
df\_ref |> filter(country %in% CONFLICTS) |> drop\_na(refugee\_origin) |>  
 ggplot(aes(year, refugee\_origin, col = country)) + geom\_line() + scale\_y\_log10()



labs(title = "紛争国の難民数（log10 スケール）")

## $title  
## [1] "紛争国の難民数（log10 スケール）"  
##   
## attr(,"class")  
## [1] "labels"

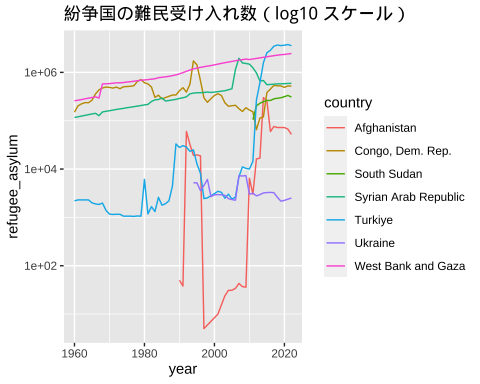
df\_ref |> filter(country %in% CONFLICTS) |> drop\_na(refugee\_asylum) |>  
 ggplot(aes(year, refugee\_asylum, col = country)) + geom\_line() +  
 labs(title = "紛争国の難民受け入れ数")



**気づいたこと・疑問**

* ミャンマーは登場していない。
* 下の方に固まっているので、scale\_y\_log10() を使ってみる
  + 1e+06 は、1,000,000 人ですから、それ以下の国が多いということになります。
  + scale\_y\_log10() にすると、1e+02 = 100, 1e+04 = 10,000, 1e+06 = 1,000,000 y 座標軸に現れます。

df\_ref |> filter(country %in% CONFLICTS) |> drop\_na(refugee\_asylum) |>  
 ggplot(aes(year, refugee\_asylum, col = country)) + geom\_line() + scale\_y\_log10() +  
 labs(title = "紛争国の難民受け入れ数（log10 スケール）")



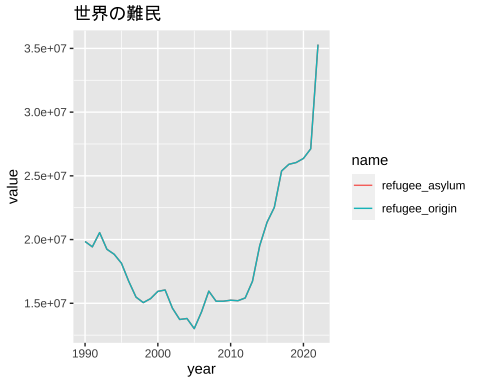
### 3. 縦長の表（Long Table）

*cols = c(refugee\_asylum, refugee\_origin) と、一つの指標にまとめるものをリストすると、name に指標名が入り、value に、その値が入る。書式は、pivot\_longer(cols = ..., names\_to = ..., values\_to = ...) で、初期設定では、names\_to = "name", values\_to = "value" となっているので、指定しないと、下のようになる。Help 参照。*

df\_ref\_long <- df\_ref |> pivot\_longer(cols = c(refugee\_asylum, refugee\_origin))  
df\_ref\_long

## # A tibble: 33,516 × 7  
## country iso2c year region income name value  
## <chr> <chr> <dbl> <chr> <chr> <chr> <dbl>  
## 1 Afghanistan AF 2014 South Asia Low income refugee\_asylum 300421  
## 2 Afghanistan AF 2014 South Asia Low income refugee\_origin 2596259  
## 3 Afghanistan AF 1971 South Asia Low income refugee\_asylum NA  
## 4 Afghanistan AF 1971 South Asia Low income refugee\_origin NA  
## 5 Afghanistan AF 2006 South Asia Low income refugee\_asylum 34  
## 6 Afghanistan AF 2006 South Asia Low income refugee\_origin 2107510  
## 7 Afghanistan AF 2013 South Asia Low income refugee\_asylum 16861  
## 8 Afghanistan AF 2013 South Asia Low income refugee\_origin 2556483  
## 9 Afghanistan AF 1995 South Asia Low income refugee\_asylum 19605  
## 10 Afghanistan AF 1995 South Asia Low income refugee\_origin 2679132  
## # ℹ 33,506 more rows

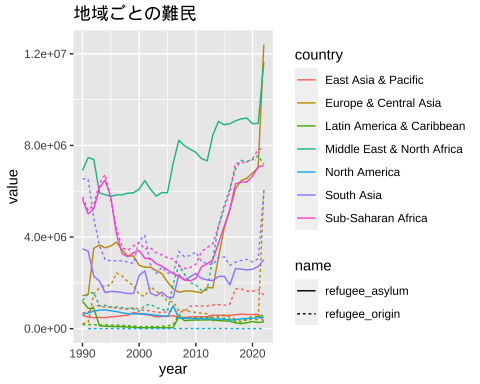
df\_ref\_long |> filter(country == "World") |> drop\_na(value) |>  
 ggplot(aes(year, value, col = name)) + geom\_line() +  
 labs(title = "世界の難民")



**気づいたこと・疑問**

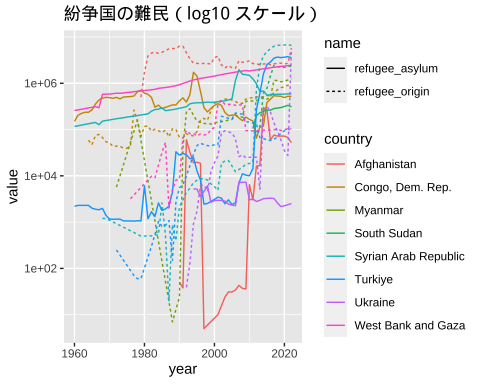
* 世界では、難民受け入れと、難民数が一致している。2010年ごろから、増加している。

df\_ref\_long |> filter(country %in% regions) |> drop\_na(value) |>  
 ggplot(aes(year, value, col = country, linetype = name)) + geom\_line() +  
 labs(title = "地域ごとの難民")



#### 紛争国の難民受け入れ数と難民のこれらの出身国別経年変化のグラフ

df\_ref\_long |> filter(country %in% CONFLICTS) |> drop\_na(value) |>  
 ggplot(aes(year, value, col = country, linetype = name)) + geom\_line() + scale\_y\_log10() +  
 labs(title = "紛争国の難民（log10 スケール）")



**気づいたこと・疑問**

* もうすこし丁寧にみないとわからない。国の数が多い
* log10 スケールの方がみやすい。

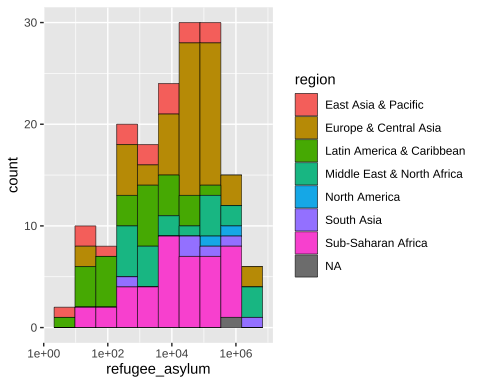
### 4. ヒストグラム

それぞれのデータが十分ある最近の年の値のヒストグラム（度数分布）

*分け方の数 bins や、幅 binwidth を適切に変更*

#### 受け入れ国別難民数

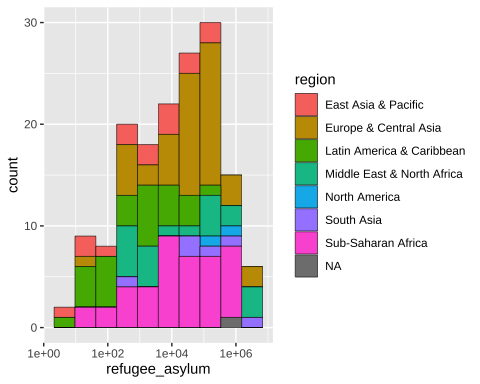
df\_ref |> filter(!(iso2c %in% REGION)) |> filter(year == 2022) |> drop\_na(refugee\_asylum) |>  
 ggplot(aes(refugee\_asylum, fill = region)) + geom\_histogram(bins = 10, col = "black", linewidth = 0.2) + scale\_x\_log10()



#### 出身国別難民数

df\_ref |> filter(!(iso2c %in% REGION)) |> filter(year == 2022, refugee\_origin >0) |>  
 drop\_na(refugee\_origin) |>  
 ggplot(aes(refugee\_asylum, fill = region)) + geom\_histogram(bins = 10, col = "black", linewidth = 0.2) + scale\_x\_log10()

## Warning: Removed 30 rows containing non-finite values (`stat\_bin()`).

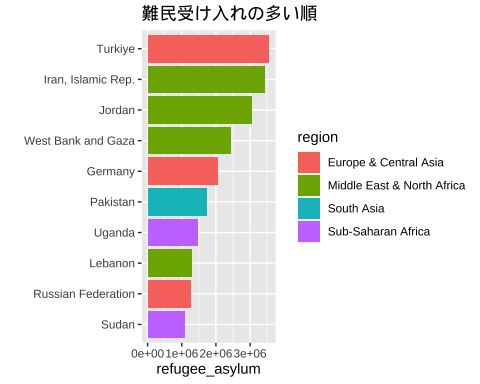


### 5. 棒グラフ

データが十分ある最近の年の値の10カ国の値の棒グラフ

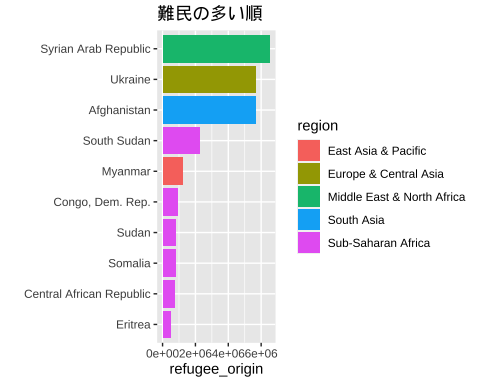
#### 難民受け入れ国

df\_ref |> filter(!(iso2c %in% REGION)) |> filter(year == 2022) |> drop\_na(refugee\_asylum) |>  
 arrange(desc(refugee\_asylum)) |> slice\_head(n = 10) |>  
 ggplot(aes(fct\_rev(fct\_inorder(country)), refugee\_asylum, fill = region)) + geom\_col() + coord\_flip() + labs(title = "難民受け入れの多い順", x = "")



#### 出身国の難民数

df\_ref |> filter(!(iso2c %in% REGION)) |> filter(year == 2022) |> drop\_na(refugee\_origin) |>  
 arrange(desc(refugee\_origin)) |> slice\_head(n = 10) |>  
 ggplot(aes(fct\_rev(fct\_inorder(country)), refugee\_origin, fill = region)) + geom\_col() + coord\_flip() + labs(title = "難民の多い順", x = "")

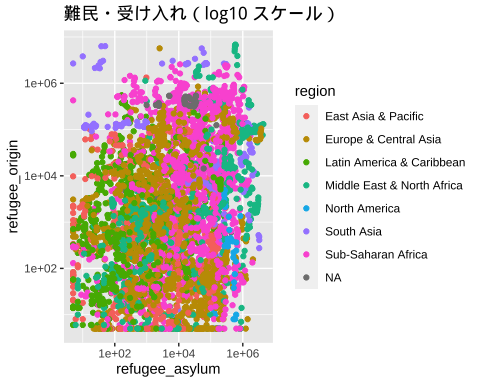


### 6. 散布図

#### a.（地域を除き）国のみの散布図

*LOG10 スケールを使っています。*

df\_ref |> filter(!(iso2c %in% REGION)) |> drop\_na(refugee\_asylum, refugee\_origin) |>  
 ggplot(aes(refugee\_asylum, refugee\_origin)) + geom\_point(aes(col = region)) + scale\_x\_log10() + scale\_y\_log10() +  
 labs(title = "難民・受け入れ（log10 スケール）")

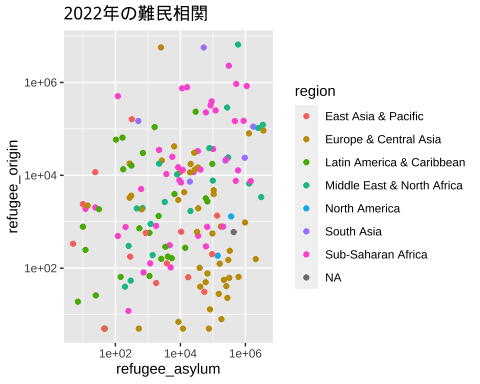


**気づいたこと・疑問**

* 相関はみられないが、地域性はみて取れるように思う。

#### 2022年の（地域を除き）国のみの散布図

df\_ref |> filter(!(iso2c %in% REGION)) |> filter(year == 2022) |>   
 drop\_na(refugee\_asylum, refugee\_origin) |>  
 ggplot(aes(refugee\_asylum, refugee\_origin)) + geom\_point(aes(col = region)) +　scale\_x\_log10() + scale\_y\_log10() +  
 labs(title = "2022年の難民相関")



**気づいたこと・疑問**

* 地域ごとの特性があるていどあるように見える。

#### 相関係数

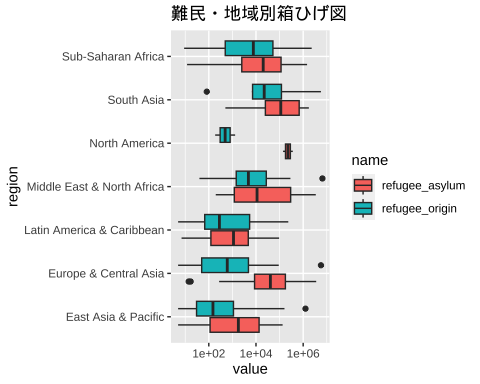
df\_ref |> filter(!(iso2c %in% REGION)) |> filter(year == 2022) |>   
 drop\_na(refugee\_asylum, refugee\_origin) |> select(refugee\_asylum, refugee\_origin) |> cor()

## refugee\_asylum refugee\_origin  
## refugee\_asylum 1.00000000 0.02499224  
## refugee\_origin 0.02499224 1.00000000

相関係数が、0.025 で、ほとんど相関はないと言って良い。

### 7. 箱ヒゲ図（Boxplot）

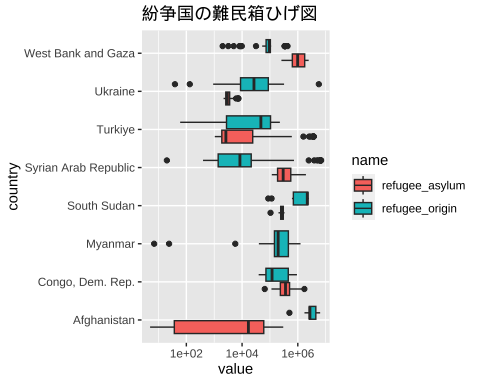
df\_ref\_long |> filter(!(iso2c %in% REGION)) |> filter(year == 2022) |>   
 drop\_na(value, region) |>  
 ggplot(aes(region, value, fill = name)) + geom\_boxplot() + coord\_flip() +scale\_y\_log10() +  
 labs(title = "難民・地域別箱ひげ図")



**気づいたこと・疑問**

* 地域差がある。これだけでは、あまりわからない。

df\_ref\_long |> filter(country %in% CONFLICTS) |> drop\_na(value) |>   
 ggplot(aes(country, value, fill = name)) + geom\_boxplot() + coord\_flip() + scale\_y\_log10() +  
 labs(title = "紛争国の難民箱ひげ図")

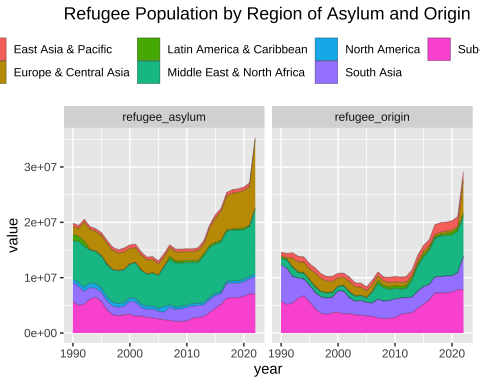


**気づいたこと・疑問**

* それぞれの国のデータは、データのある年の中での、値である。
* アフガニスタンは、継続して、難民が多いことがわかる。

## 参考

df\_ref\_long |> filter(country %in% regions) |> drop\_na(value) |>  
 ggplot(aes(year, value, fill = country)) +   
 geom\_area(col = "black", linewidth = 0.1) +  
 facet\_wrap(~name) +  
 labs(title = "Refugee Population by Region of Asylum and Origin", fill = "") +  
 theme(legend.position = "top")



実数で、積み上げグラフを作成すると、 それぞれの地域の和をとったもので割って、百聞率を出すと、下のようになる。

df\_ref |> filter(country %in% regions) |>   
 select(country, year, refugee\_asylum, refugee\_origin) |> drop\_na(refugee\_origin) |> group\_by(year) |>   
 mutate(ASYLUM = refugee\_asylum/sum(refugee\_asylum), ORIGIN = refugee\_origin/sum(refugee\_origin)) |> ungroup() |>  
 pivot\_longer(cols = c(ASYLUM, ORIGIN)) |>  
 ggplot(aes(year, value, fill = country)) +   
 geom\_area(col = "black", linewidth = 0.1) +   
 facet\_wrap(~name) + scale\_y\_continuous(labels = scales::percent\_format(accuracy = 1)) +  
 labs(title = "Refugee Ratio by Region of Asylum and Origin", fill = "", y = "") +  
 theme(legend.position = "top")

