Praktikum 3

Visualisasi Data Kategorik dan Transformasi Data

Variabel Kategorik

Sebuah variabel disebut kategorikal ketika skala pengukurannya berupa himpunan kategori. Contoh variabel kategorikal adalah status pernikahan (dengan kategori seperti lajang, menikah, bercerai, janda/duda), moda transportasi utama ke tempat kerja (mobil, sepeda, bus, kereta bawah tanah, berjalan kaki), tujuan utama berbelanja pakaian (pusat kota, internet, mal, lainnya), dan jenis musik favorit (klasik, country, folk, jazz, rap/hip-hop, rock). Variabel kategorikal yang hanya memiliki dua kategori, seperti status pekerjaan (ya, tidak), disebut biner. Untuk variabel kategorikal, kategori yang berbeda menunjukkan perbedaan kualitas, bukan besarnya nilai numerik. Variabel kategorikal sering juga disebut kualitatif.

Variabel kategorikal memiliki dua jenis skala pengukuran. Untuk beberapa variabel kategorikal, seperti yang baru saja disebutkan, kategorinya tidak memiliki urutan. Skala ini tidak memiliki ujung "tinggi" atau "rendah". Kategori tersebut dikatakan membentuk skala nominal. Sebaliknya, beberapa skala kategorikal memiliki urutan alami dari nilai-nilainya. Kategori tersebut membentuk skala ordinal. Contohnya adalah tingkat kebahagiaan yang dirasakan (tidak terlalu bahagia, cukup bahagia, sangat bahagia), tingkat rasa sakit sakit kepala (tidak ada, ringan, sedang, parah), dan pandangan politik (sangat liberal, agak liberal, moderat, agak konservatif, sangat konservatif).

Diagram batang

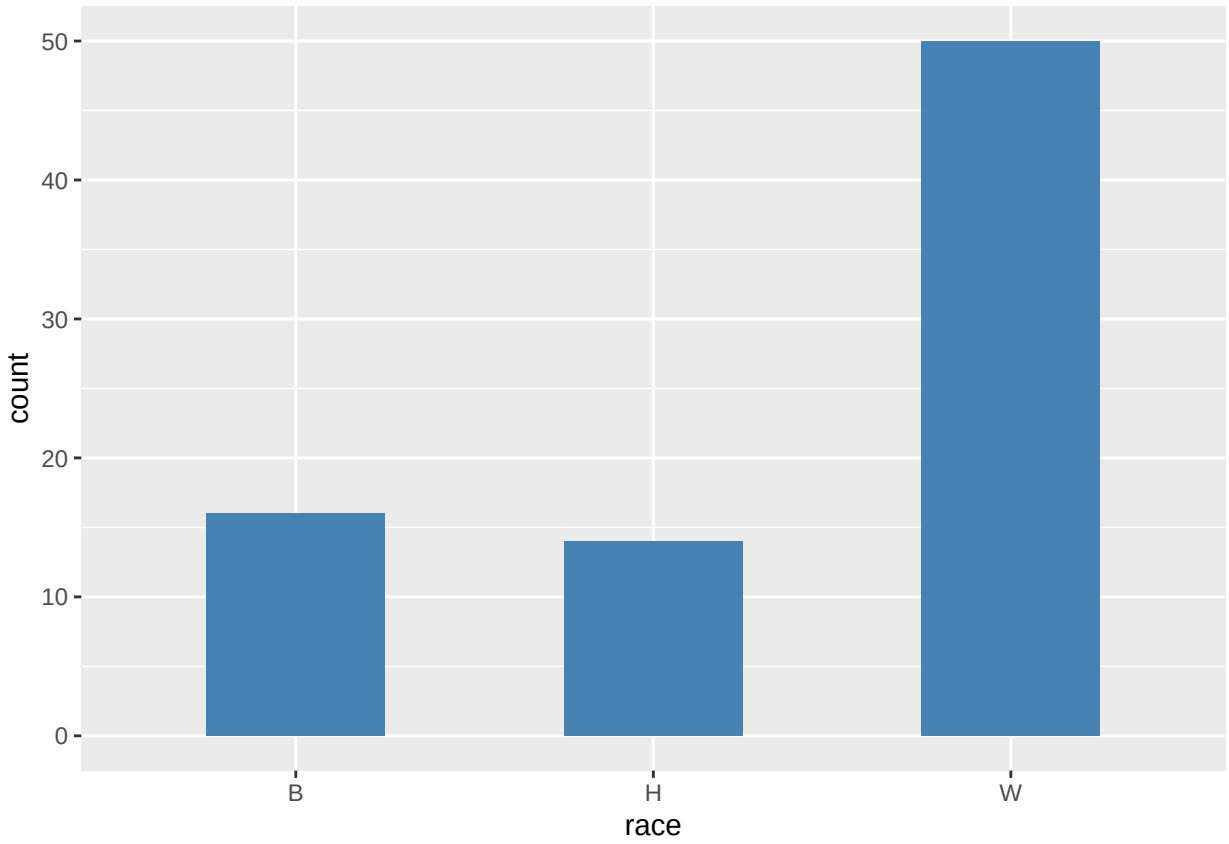
Contoh 1

Sebuah variabel kategorikal diringkas dengan **tabel frekuensi** dan dapat divisualisasikan melalui **diagram batang**, dengan tinggi batang mewakili frekuensi atau proporsi. Kita ilustrasikan untuk status ras-etnis dalam berkas data *Income*, menggunakan fungsi **ggplot** dari paket **ggplot2**. Data Income memuat pendapatan tahunan (dalam ribuan dolar) dari 80 subjek yang diklasifikasikan ke dalam tiga kategori status ras-etnis (Black, Hispanic, White).

```
library(ggplot2)
Inc <- read.table("http://stat4ds.rwth-aachen.de/data/Income.dat", header=TRUE)
head(Inc, 3) # shows first 3 lines of data file
```

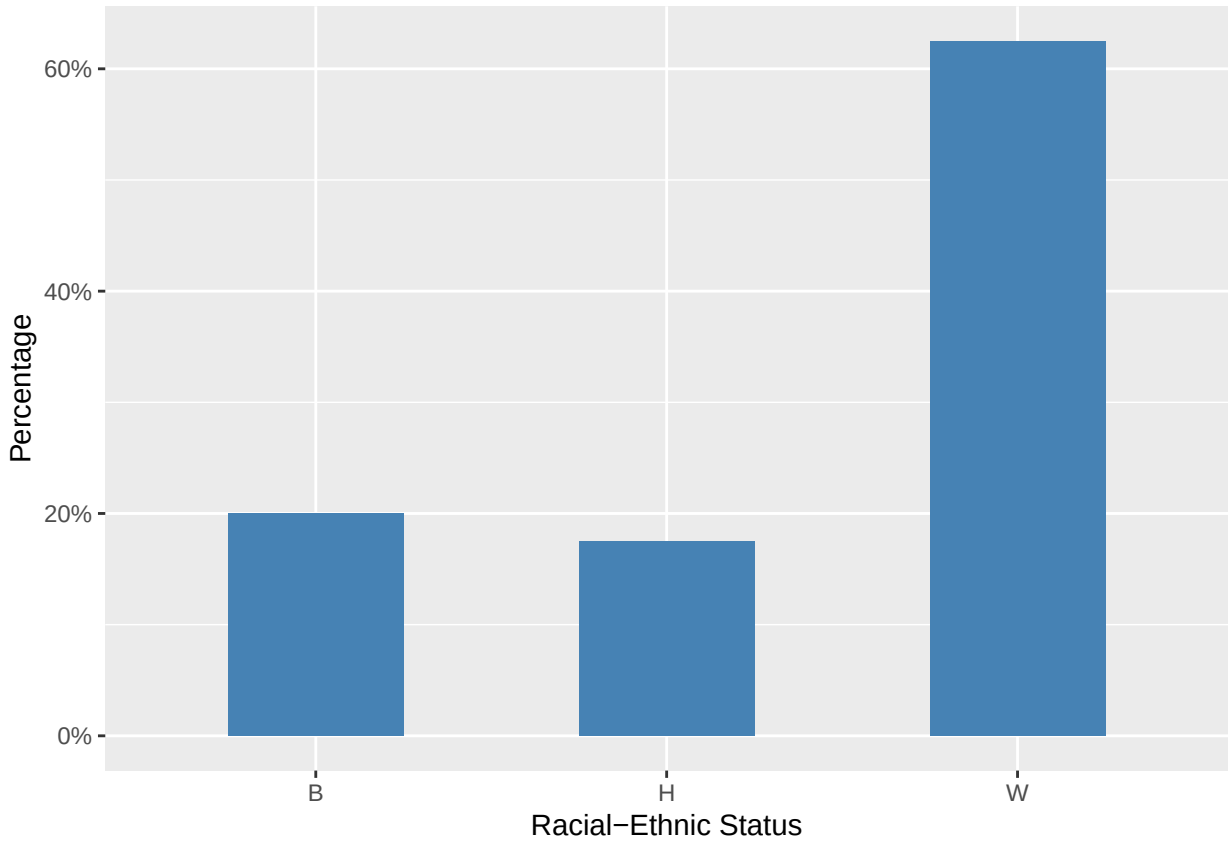
```
##   income education race
## 1     16        10    B
## 2     18         7    B
## 3     26         9    B
```

```
# Diagram batang untuk jumlah (tidak ditampilkan):
ggplot(data=Inc, aes(x=race)) + geom_bar(width=0.5, fill="steelblue")
```




```
# Diagram batang untuk persentase:
library(scales)
ggplot(data=Inc, aes(race)) +
geom_bar(aes(y=..prop.., group = 1), width=0.5, fill="steelblue") +
scale_y_continuous(labels=percent_format()) +
ylab("Percentage") + xlab("Racial-Ethnic Status")
```

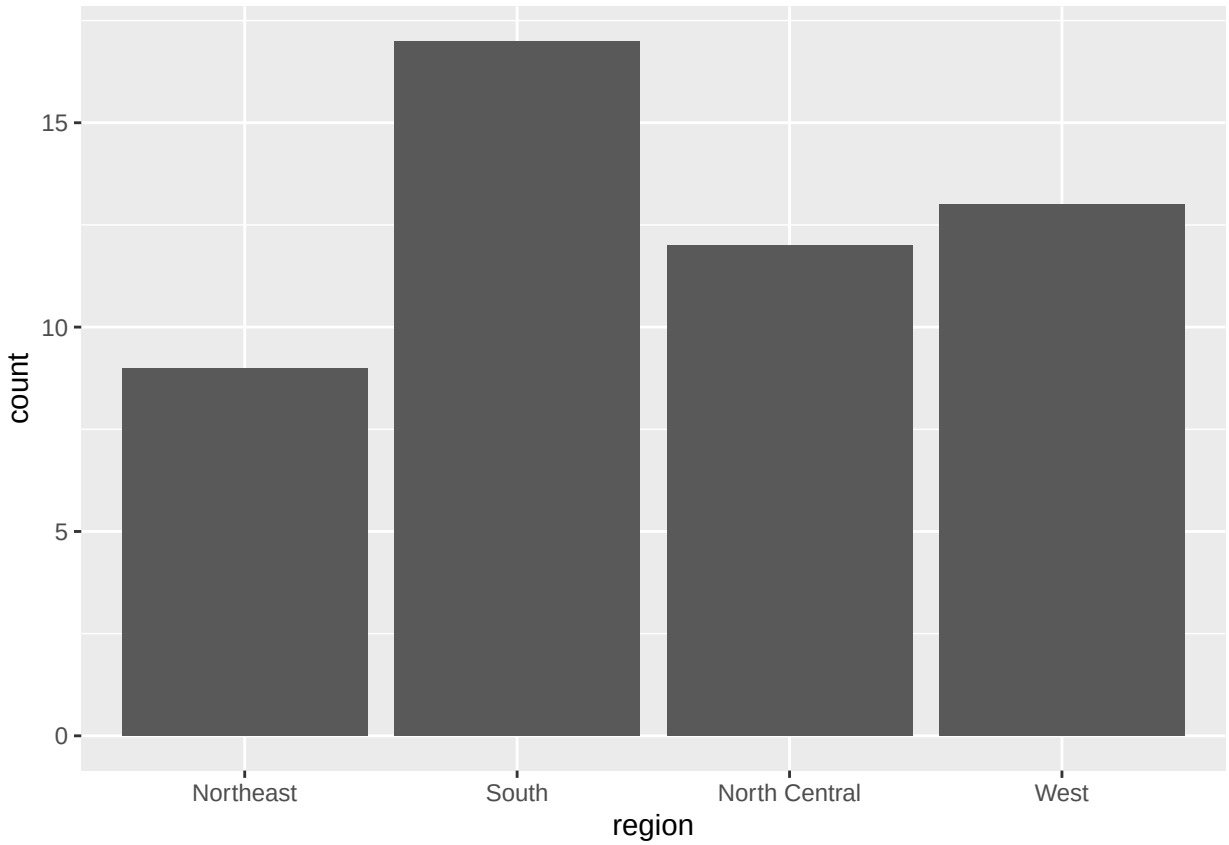
```
## Warning: The dot-dot notation (`..prop..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(prop)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```




```
# Pada perintah ggplot, simbol "+" harus berada di akhir setiap baris,
# bukan di awal baris berikutnya.
```

Contoh 2 Untuk membuat barplot, kita dapat menggunakan geometri geom_bar. Secara bawaan (default), fungsi ini menghitung jumlah setiap kategori dan menggambar batangnya. Berikut contoh plot untuk wilayah di Amerika Serikat:

```
library(dslabs)
murders |> ggplot(aes(region)) + geom_bar()
```



Namun, sering kali kita sudah memiliki tabel dengan angka-angka yang ingin kita tampilkan sebagai barplot. Berikut contoh tabel seperti itu:

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

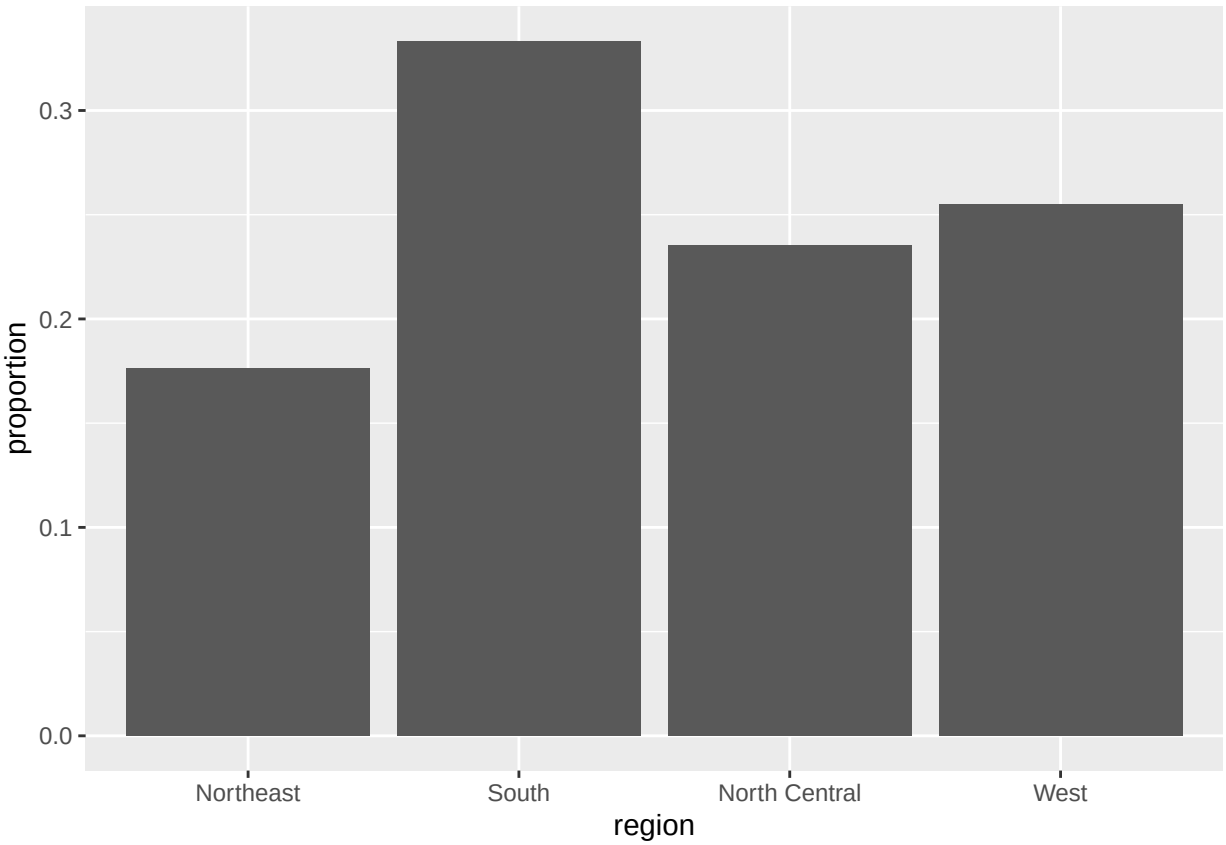
```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
tab <- murders |>
count(region) |>
mutate(proportion = n/sum(n))
```

Dalam kasus ini, kita menggunakan geom_col sebagai ganti dari geom_bar:

```
tab |> ggplot(aes(region, proportion)) + geom_col()
```



Tabel Distribusi frekuensi

Studi Kasus : General Social Survey (GSS)

Beberapa basis data tersedia secara bebas di Internet. Salah satu basis data penting di Amerika Serikat berisi hasil General Social Survey (GSS) sejak tahun 1972, yang dilaksanakan setiap dua tahun oleh National Opinion Research Center di Universitas Chicago. Survei ini mengumpulkan informasi melalui wawancara langsung terhadap sampel sekitar n = 2000 subjek dari populasi orang dewasa di AS untuk memberikan gambaran tentang opini dan perilaku.

Para peneliti menggunakannya untuk menyelidiki bagaimana orang dewasa Amerika menjawab beragam pertanyaan, seperti: “Apakah Anda percaya pada kehidupan setelah kematian?” dan “Apakah Anda bersedia membayar harga lebih tinggi demi melindungi lingkungan?”

Survei sosial serupa juga dilakukan di negara lain, seperti General Social Survey yang dilaksanakan oleh Statistics Canada, British Social Attitudes Survey, serta survei Eurobarometer dan European Social Survey untuk negara-negara di Uni Eropa.

- Kunjungi situs web https://sda.berkeley.edu/archive.htm di Survey Documentation and Analysis milik University of California, Berkeley.
- Klik pada General Social Survey (GSS) Cumulative Datafile yang tersedia paling baru.
- Anda kemudian akan melihat daftar pemilihan variabel di sisi kiri yang berisi karakteristik yang diukur selama bertahun-tahun, dan sebuah menu di sisi kanan untuk memilih karakteristik tertentu yang diminati.
- Ketik nama karakteristik yang diminati pada kotak Row, lalu klik Run the table. Situs GSS kemudian akan menghasilkan sebuah tabel yang menampilkan kemungkinan nilai untuk karakteristik tersebut beserta jumlah orang dan persentase yang memberikan setiap jawaban.

Sebagai contoh, dalam salah satu survei GSS diajukan pertanyaan: "Kira-kira berapa banyak teman dekat yang Anda miliki?" Nama variabel GSS untuk karakteristik ini adalah NUMFREND. Tabel yang disediakan GSS menunjukkan bahwa jawaban 1, 2, 3, 4, 5, dan 6 teman dekat masing-masing memiliki persentase 6,1; 16,2; 15,7; 14,2; 11,3; dan 8,8, sedangkan sisa 27,7% tersebar pada kemungkinan jawaban lainnya.

```
GSS <- read.table("http://stat4ds.rwth-aachen.de/data/GSS2018.dat", header=TRUE)
```

Paket dplyr dapat menggabungkan fungsi count dan group_by untuk membangun distribusi frekuensi dari sebuah variabel kategorikal di dalam nilai variabel lain. Sebagai contoh, untuk variabel yang merepresentasikan respons terhadap pernyataan "Agar sebuah masyarakat adil, perbedaan dalam standar hidup orang seharusnya kecil" pada berkas data GSS2018, kode berikut membuat distribusi frekuensi dari SMALLGAP untuk laki-laki dan perempuan:

```
library(dplyr)
GSS$SEX <- factor(GSS$SEX, levels=c(1:2), labels = c("male", "female"))
GSS$SMALLGAP <- factor(GSS$SMALLGAP, levels=c(1:5), labels = c("strongly agree",
                                "agree","neutral","disagree","strongly disagree"))
GSS %>% group_by(SEX) %>% count(SMALLGAP) # result not shown
```

```
## # A tibble: 12 x 3
## # Groups:   SEX [2]
##    SEX    SMALLGAP              n
##    <fct>  <fct>             <int>
##  1 male   strongly agree       47
##  2 male   agree               140
##  3 male   neutral             149
##  4 male   disagree            183
##  5 male   strongly disagree    38
##  6 male   <NA>                495
##  7 female strongly agree       58
##  8 female agree               156
##  9 female neutral             192
## 10 female disagree            159
## 11 female strongly disagree    26
## 12 female <NA>                705
```

Data Kategorikal Bivariat: Tabel Kontingensi

Untuk dua variabel kategorikal, tabel kontingensi adalah tabel berbentuk persegi panjang yang menyilangkan (cross-classify) variabel-variabel tersebut. Sel-selnya menunjukkan kombinasi kategori dan jumlahnya. Sebagai contoh, Tabel 1 adalah tabel kontingensi untuk identifikasi partai politik (ID = Demokrat, Independen, atau Republik) dan ras, menggunakan data dari 2018 General Subject Survey.

	Political Party ID		
Race	Democrat	Independent	Republican
Black	281	66	30
Other	124	77	52
White	633	272	704

Dengan memperlakukan identifikasi partai politik sebagai variabel respons, kita dapat membuat ringkasan dengan menghitung persentase pada setiap kategori ID, berdasarkan ras. Misalnya, 44% orang kulit putih dan 8% orang kulit hitam mengidentifikasi diri sebagai Republikan.

Kode berikut menggunakan R dengan data Party ID untuk membangun tabel kontingensi dan menghitung proporsi ID secara terpisah untuk setiap kategori ras:

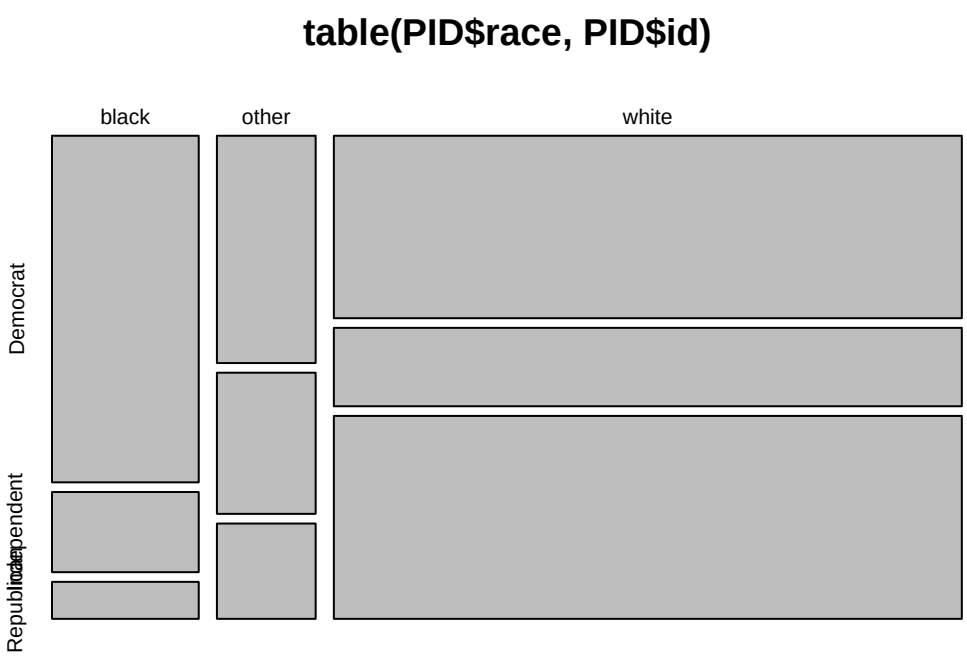
```
PID <- read.table("http://stat4ds.rwth-aachen.de/data/PartyID.dat", header=TRUE)
table(PID$race, PID$id)   # membentuk tabel kontingensi
```

```
##
##         Democrat Independent Republican
##   black      281          65         30
##   other      124          77         52
##   white      633         272        704
```

```
options(digits=2)
# Untuk margin=1, proporsi dijumlahkan hingga 1.0 dalam setiap baris
prop.table(table(PID$race, PID$id), margin=1)
```

```
##
##         Democrat Independent Republican
##   black     0.75        0.17       0.08
##   other     0.49        0.30       0.21
##   white     0.39        0.17       0.44
```

```
mosaicplot(table(PID$race, PID$id))   # representasi grafis dari ukuran sel
```



Sebagai latihan, periksa apa yang digambarkan oleh fungsi mosaicplot. Tabel kontingensi dapat diperluas menjadi tabel multi-dimensi untuk menangani beberapa variabel sekaligus.

Tranformasi Data

Studi Kasus : wawasan baru tentang kemiskinan

Ketika khalayak umum ditanya apakah negara miskin menjadi semakin miskin dan negara kaya menjadi semakin kaya, mayoritas menjawab "ya". Pertama-tama kita akan mempelajari bagaimana transformasi terkadang dapat membantu memberikan ringkasan dan grafik yang lebih informatif. Tabel data gapminder menyertakan sebuah kolom dengan Produk Domestik Bruto (PDB) suatu negara. PDB mengukur nilai pasar barang dan jasa yang diproduksi oleh sebuah negara dalam satu tahun. PDB per orang sering

digunakan sebagai ringkasan kasar mengenai tingkat kekayaan suatu negara. Di sini kita membagi besaran tersebut dengan 365 untuk memperoleh ukuran yang lebih mudah dipahami, yaitu dolar per hari. Dengan menggunakan dolar AS saat ini sebagai satuan, seseorang yang bertahan hidup dengan pendapatan kurang dari 2 dolar per hari didefinisikan sebagai hidup dalam kemiskinan absolut. Kita menambahkan variabel ini ke dalam tabel data.

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.4     v tibble    3.2.1
## v purrr     1.0.4     v tidyr     1.3.1
## v readr     2.1.5
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x readr::col_factor() masks scales::col_factor()
## x purrr::discard()    masks scales::discard()
## x dplyr::filter()     masks stats::filter()
## x dplyr::lag()        masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

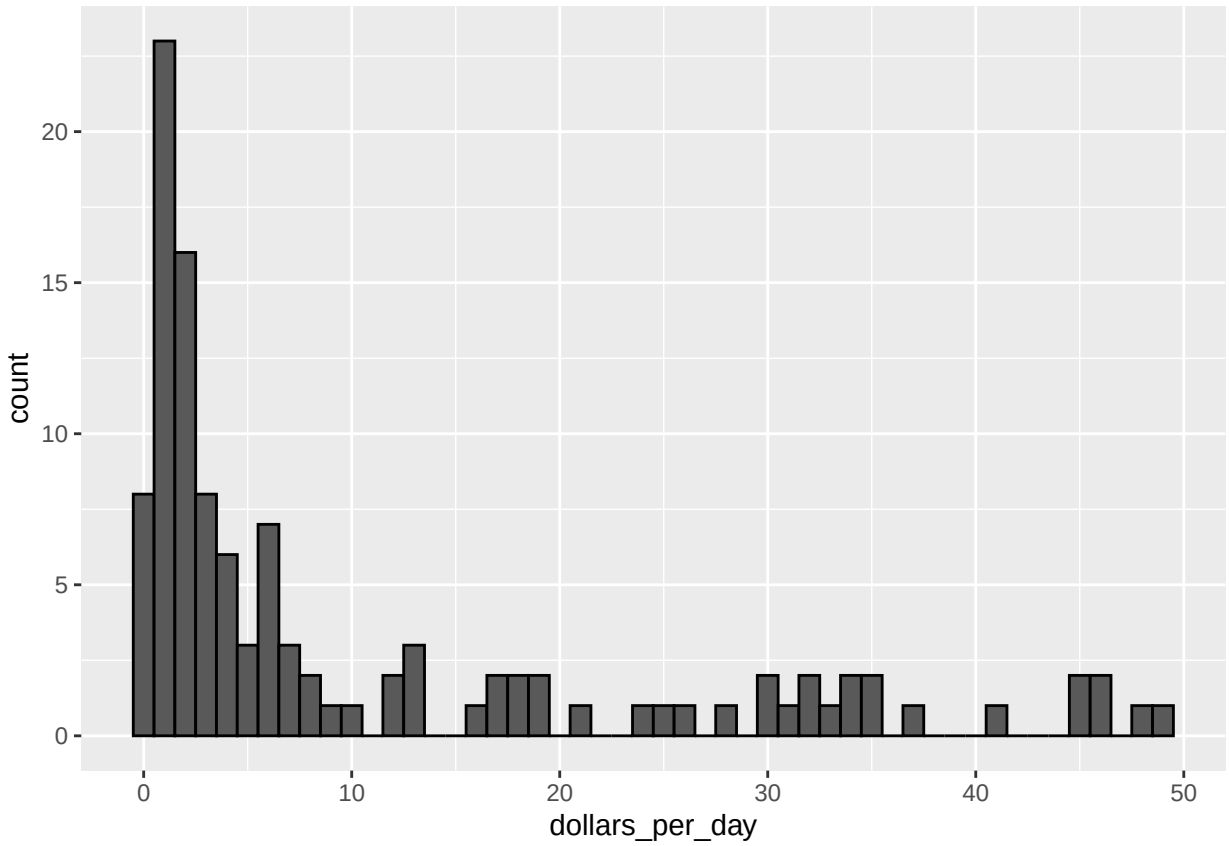
```
gapminder <- gapminder |>
mutate(dollars_per_day = gdp/population/365)
```

Nilai-nilai PDB tersebut telah disesuaikan terhadap inflasi dan merepresentasikan dolar AS saat ini, sehingga nilai-nilai ini dimaksudkan agar dapat dibandingkan sepanjang tahun. Tentu saja, nilai ini adalah rata-rata negara dan di dalam tiap negara terdapat banyak variasi. Semua grafik dan wawasan yang dijelaskan di bawah ini berkaitan dengan rata-rata negara, bukan individu.

Log transformation

Berikut adalah histogram pendapatan per hari pada tahun 1970:

```
past_year <- 1970
gapminder |>
filter(year == past_year & !is.na(gdp)) |>
ggplot(aes(dollars_per_day)) +
geom_histogram(binwidth = 1, color = "black")
```



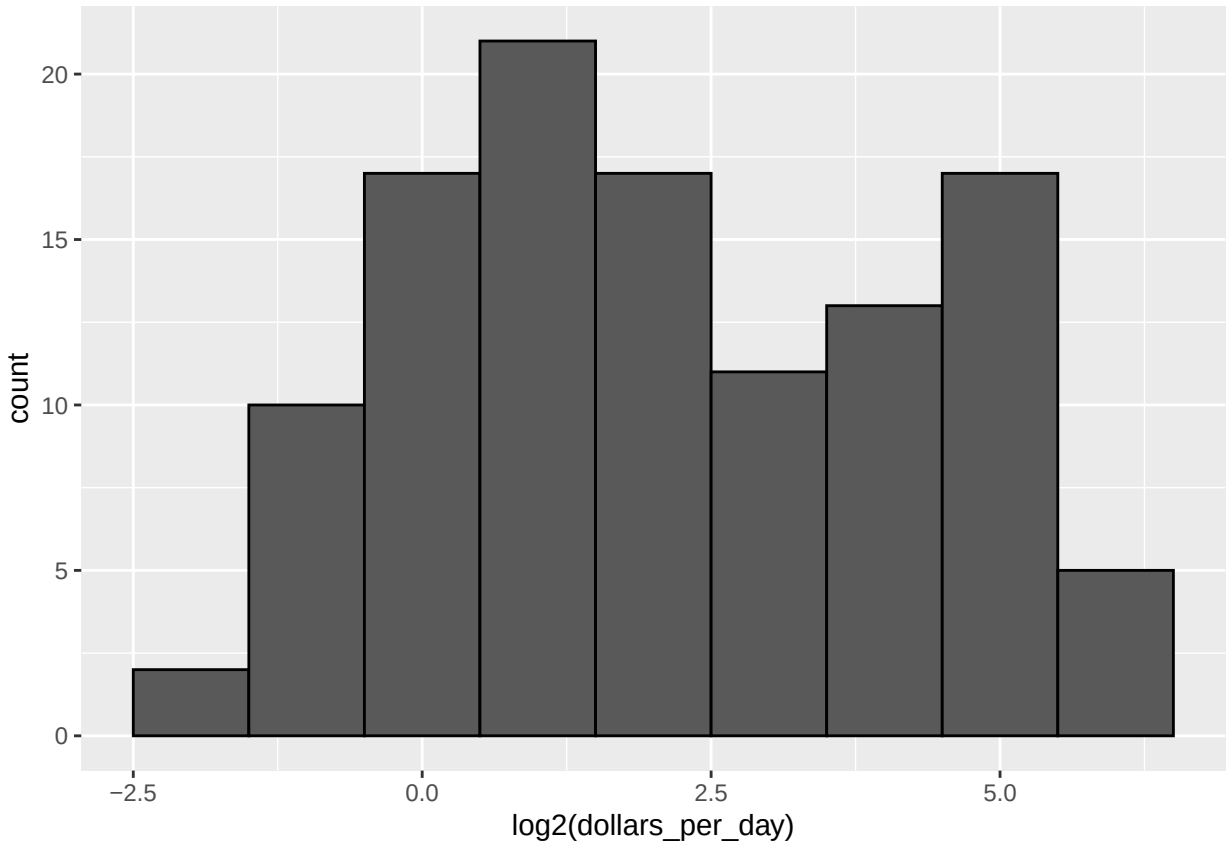
Kita menggunakan argumen color = “black” untuk menggambar batas dan secara jelas membedakan bins.

Dalam plot ini, kita melihat bahwa bagi sebagian besar negara, rata-rata pendapatan berada di bawah $10 per hari. Namun, sebagian besar sumbu-x digunakan untuk 35 negara dengan rata-rata di atas $10. Jadi, plot ini tidak terlalu informatif mengenai negara-negara dengan nilai di bawah $10 per hari.

Akan lebih informatif jika kita dapat dengan cepat melihat berapa banyak negara yang memiliki rata-rata pendapatan harian sekitar $1 (sangat miskin), $2 (miskin sekali), $4 (miskin), $8 (menengah), $16 (cukup sejahtera), $32 (kaya), dan $64 (sangat kaya) per hari. Perubahan ini bersifat multiplikatif, dan transformasi log mengubah perubahan multiplikatif menjadi aditif: ketika menggunakan basis 2, pelipatan ganda dari suatu nilai menjadi kenaikan sebesar 1.

Berikut distribusinya jika kita menerapkan transformasi log basis 2:

```
gapminder |>
  filter(year == past_year & !is.na(gdp)) |>
  ggplot(aes(log2(dollars_per_day))) +
  geom_histogram(binwidth = 1, color = "black")
```



Dalam contoh sebelumnya, kita menggunakan **basis 2** pada transformasi log. Pilihan umum lainnya adalah **basis *e*** (logaritma natural) dan **basis 10**.

Secara umum, **kami tidak merekomendasikan penggunaan logaritma natural** untuk eksplorasi dan visualisasi data. Alasannya:

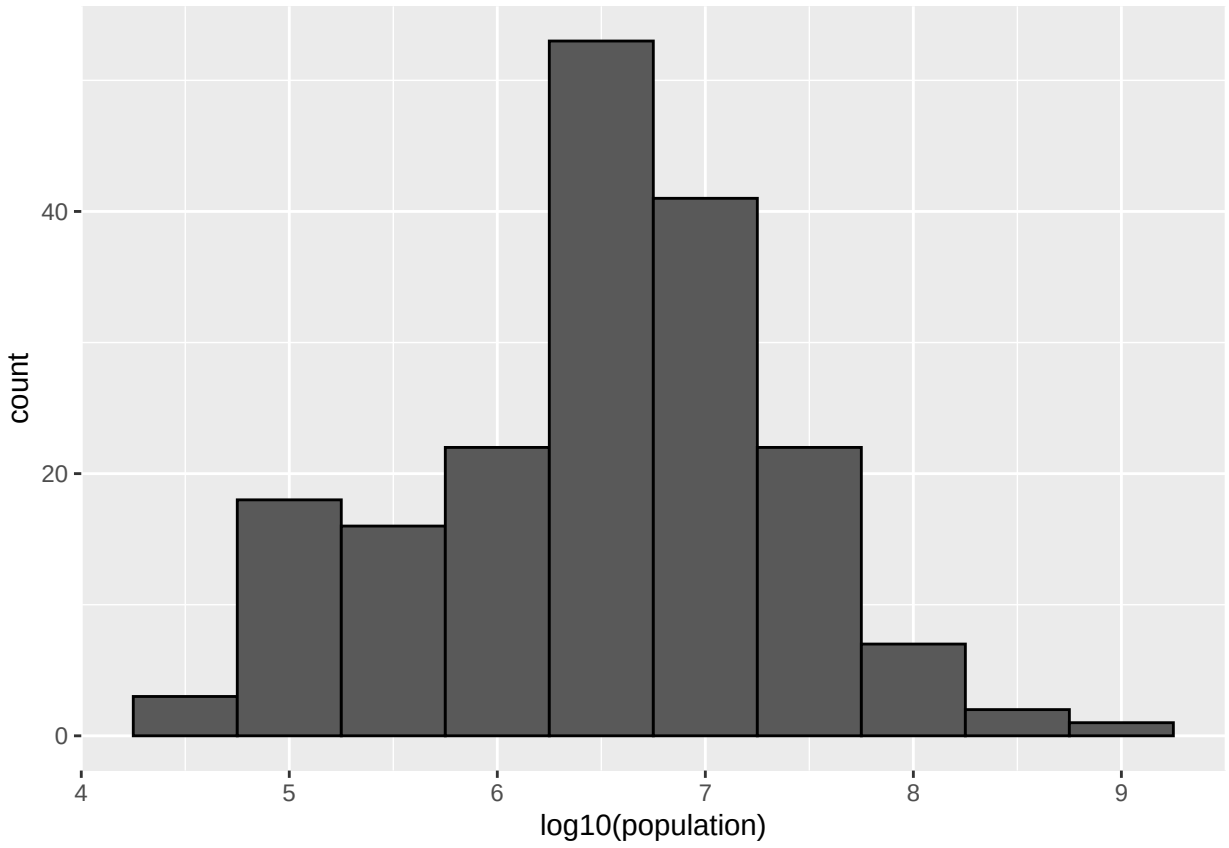
- $2^2, 2^3, 2^4, ...$ atau $10^2, 10^3, ...$ mudah dihitung secara mental.
- Hal yang sama **tidak berlaku** untuk $e^2, e^3, ...$
- Skala logaritma natural **tidak intuitif** dan sulit ditafsirkan.

```
filter(gapminder, year == past_year) |>
summarize(min = min(population), max = max(population))
```

```
##     min     max
## 1 46075 8.1e+08
```

Berikut adalah histogram dari nilai-nilai yang telah ditransformasi:

```
gapminder |>
filter(year == past_year) |>
ggplot(aes(log10(population))) +
geom_histogram(binwidth = 0.5, color = "black")
```



Transformasi Nilai atau Skala?

Ada dua cara kita dapat menggunakan **transformasi log** dalam plot:

1. **Mentransformasi nilai dengan log sebelum mem-plot**

2. **Menggunakan skala log pada sumbu**

Plot yang dihasilkan akan terlihat sama, kecuali pada angka yang ditampilkan di sumbu.
Kedua pendekatan ini sama-sama berguna dan memiliki kelebihan masing-masing.

Jika Data Ditrasformasi Log Kita bisa lebih mudah menafsirkan nilai-nilai di antara skala.
Misalnya:
—-1—-x—-2———3—-

Untuk data yang ditransformasi log, kita tahu bahwa nilai x = 1.5.

Jika Skala Sumbu Ditrasformasi Log Tampilannya menjadi:
—-10—x—100——1000—

Untuk mengetahui nilai x, kita harus menghitung 10^1.5, yang **tidak mudah dilakukan di kepala**.

Kelebihan Skala Log pada Sumbu Keuntungan dari menampilkan **skala log pada sumbu** adalah nilai **asli** tetap muncul di plot, sehingga lebih mudah diinterpretasikan.

Contoh:

- Tampilan dengan skala log: **“32 dolar per hari”**

- Tampilan dengan data log: **"5 log basis 2 dolar per hari"**

Implementasi di R Seperti yang telah kita pelajari sebelumnya, jika ingin melakukan penskalaan sumbu dengan log, kita dapat menggunakan fungsi:

```
gapminder |>
filter(year == past_year & !is.na(gdp)) |>
ggplot(aes(dollars_per_day)) +
geom_histogram(binwidth = 1, color = "black") +
scale_x_continuous(trans = "log2")
```