Gapminder; Assignment 3

MSB - 105

Ole Alexander Bakkevik & Sindre M. Espedal

Spørsmål 1

ddf–concepts.csv filen er en tekstfil uten verdier. Innholdet er beskrivelser av forskjellige variabler. Som for eksempel populasjonsforhold, dødsårsaker, sykdommer (HIV, tuberkulose), arbeidsledighet og yrkesaktive, aldersgrupper. Det er også beskrevet de respektive landenes mikro/makroøkonimiske forhold som BNP osv.

Spørsmål 2

ddf–entities–geo–country.csv henviser til en kort beskrivelse av alle verdens land og stater. Samtidig som informasjon om landet er anerkjent som et land eller ikke. Beskrivelsen innholder også geografisk lokasjon verdensdel/region, FN-tilhørighet og antatt levestandard.

Spørsmål 3

ddf–entities–geo–un_sdg_region.csv beskriver hvilke områder som er anerkjent som FN-regioner.

```
library(readr)
g_c <- read_csv(
  paste0(
    "ddf--gapminder--systema_globalis-master/",
    "ddf--entities--geo--country.csv"
    )
)
```

```
## Rows: 273 Columns: 22
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (17): country, g77_and_oecd_countries, income_3groups, income_groups, is...
## dbl  (3): iso3166_1_numeric, latitude, longitude
## lgl  (2): is--country, un_state
```

```
##
## 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.
```

```
print(g_c)
```

```
## # A tibble: 273 x 22
##    country   g77_and_oecd_countries income_3groups income_groups   `is--country`
##    <chr>     <chr>                  <chr>          <chr>           <lgl>
##  1 abkh      others                 <NA>           <NA>            TRUE
##  2 abw       others                 high_income    high_income     TRUE
```

```
##  3 afg       g77                   low_income     low_income      TRUE
##  4 ago       g77                   middle_income  lower_middle_i~ TRUE
##  5 aia       others                <NA>           <NA>            TRUE
##  6 akr_a_dhe others                <NA>           <NA>            TRUE
##  7 ala       others                <NA>           <NA>            TRUE
##  8 alb       others                middle_income  upper_middle_i~ TRUE
##  9 and       others                high_income    high_income     TRUE
## 10 ant       others                <NA>           <NA>            TRUE
## # ... with 263 more rows, and 17 more variables: iso3166_1_alpha2 <chr>,
## #   iso3166_1_alpha3 <chr>, iso3166_1_numeric <dbl>, iso3166_2 <chr>,
## #   landlocked <chr>, latitude <dbl>, longitude <dbl>,
## #   main_religion_2008 <chr>, name <chr>, un_sdg_ldc <chr>,
## #   un_sdg_region <chr>, un_state <lgl>, unhcr_region <chr>,
## #   unicef_region <chr>, unicode_region_subtag <chr>, world_4region <chr>,
## #   world_6region <chr>
```

Spørsmål 4

Gapminder datasettet fra gapminder pakken inneholder 6 variabler. disse er:

- *country* -> land, med 142 leveler.
- *continent* -> kontinenter, med 5 leveler.
- *year* -> år, varier fra 1952 til 2007 i trinn på 5 år.
- *lifeExp* -> forventet leveår, målt i år.
- *pop* -> befolkning
- *gdpPercap* -> BNP per innbygger, US$ inflasjon-justert

```
library(readr)
# Denne dobbel tilordningen trenger dere vel ikke
#oppg_4 <- ddf_entities_geo_country <- read_csv(
#"ddf--gapminder--systema_globalis-master/ddf--entities--geo--country.csv")
oppg_4 <- read_csv(
  paste0(
    "ddf--gapminder--systema_globalis-master/",
    "ddf--entities--geo--country.csv"
    )
  )
```

```
## Rows: 273 Columns: 22
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (17): country, g77_and_oecd_countries, income_3groups, income_groups, is...
## dbl  (3): iso3166_1_numeric, latitude, longitude
## lgl  (2): is--country, un_state
```

```
##
## 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.
```

```
print(oppg_4)
```

```
## # A tibble: 273 x 22
##    country   g77_and_oecd_countries income_3groups income_groups   `is--country`
##    <chr>     <chr>                  <chr>          <chr>           <lgl>
##  1 abkh      others                 <NA>           <NA>            TRUE
```

```
##  2 abw       others                 high_income    high_income     TRUE
##  3 afg       g77                    low_income     low_income      TRUE
##  4 ago       g77                    middle_income  lower_middle_i~ TRUE
##  5 aia       others                 <NA>           <NA>            TRUE
##  6 akr_a_dhe others                 <NA>           <NA>            TRUE
##  7 ala       others                 <NA>           <NA>            TRUE
##  8 alb       others                 middle_income  upper_middle_i~ TRUE
##  9 and       others                 high_income    high_income     TRUE
## 10 ant       others                 <NA>           <NA>            TRUE
## # ... with 263 more rows, and 17 more variables: iso3166_1_alpha2 <chr>,
## #   iso3166_1_alpha3 <chr>, iso3166_1_numeric <dbl>, iso3166_2 <chr>,
## #   landlocked <chr>, latitude <dbl>, longitude <dbl>,
## #   main_religion_2008 <chr>, name <chr>, un_sdg_ldc <chr>,
## #   un_sdg_region <chr>, un_state <lgl>, unhcr_region <chr>,
## #   unicef_region <chr>, unicode_region_subtag <chr>, world_4region <chr>,
## #   world_6region <chr>
```

Vi ser ut i fra datasettet *ddf:entities_geo_country.csv* at Australia og New Zealand tilhører *east_asia_pacific*, altså Asia.

Spørsmål 5

```
g_c <- g_c %>%
  mutate(continent = case_when(
     world_4region == "asia" & un_sdg_region %in% c(
       "un_australia_and_new_zealand",
       "un_oceania_exc_australia_and_new_zealand"
       ) ~ "Oceania",
    world_4region ==  "asia" & !(un_sdg_region %in% c(
      "un_australia_and_new_zealand",
      "un_oceania_exc_australia_and_new_zealand"
      )
      ) ~ "Asia",
    world_4region == "africa" ~ "Africa",
    world_4region == "americas" ~ "Americas",
    world_4region ==  "europe" ~ "Europe")
  ) %>%
 filter(!is.na(iso3166_1_alpha3))
```

Spørsmål 6

a)

```
length(unique(g_c$country))
```

```
## [1] 247
```

Nå er det 247 land.

Spørsmål 6

b)

```
g_c %>%
  group_by(continent) %>%
  summarise(countries = length(unique(country)))
```

```
## # A tibble: 5 x 2
##   continent countries
##   <chr>         <int>
## 1 Africa           59
## 2 Americas         55
## 3 Asia             47
## 4 Europe           58
## 5 Oceania          28
```

Spørsmål 7

```
lifeExp <- read_csv("ddf--gapminder--systema_globalis-master/countries-etc-datapoints/ddf--datapoints--

col_types = cols(time = col_date(format = "%Y")))
lifeExp <- lifeExp %>%
rename(year = time)
names(lifeExp)
```

```
## [1] "geo"                  "year"                 "life_expectancy_years"
```

```
length(unique(lifeExp$geo))
```

```
## [1] 195
```

Spørsmål 8

```
length(unique(lifeExp$geo))
```

```
## [1] 195
```

Det er 195 land som informasjon om forventet levetid.

Spørsmål 9

```
g_c <- g_c %>%
select(country,
       name,
       iso3166_1_alpha3,
       un_sdg_region,
       world_4region,
       continent,
       world_6region,
       ) %>%
left_join(lifeExp, by = c("country" = "geo")) %>%
```

```
filter(!(is.na(year) & is.na(life_expectancy_years))) %>%
filter(year < "2020-01-01")
```

```
names(g_c)
```

```
## [1] "country"               "name"                  "iso3166_1_alpha3"
## [4] "un_sdg_region"         "world_4region"         "continent"
## [7] "world_6region"         "year"                  "life_expectancy_years"
```

Spørsmål 10

```
g_c_min <- g_c %>%
  group_by(country) %>%
  summarise(min_year = min(year))

table(g_c_min$min_year)
```

```
##
## 1800-01-01 1950-01-01
##        186          9
```

Fra 1800 er det 186 land som har forventet levealder, mens de resterende 9 andre landene har fra 1950.

Spørsmål 11

```
g_c_min <- g_c_min %>%
  left_join(g_c,
            by = "country") %>%
  filter(min_year == "1950-01-01")

tibble(country = unique(g_c_min$name))
```

```
## # A tibble: 9 x 1
##   country
##   <chr>
## 1 Andorra
## 2 Dominica
## 3 St. Kitts and Nevis
## 4 Monaco
## 5 Marshall Islands
## 6 Nauru
## 7 Palau
## 8 San Marino
## 9 Tuvalu
```

```
rm(g_c_min)
```

Dette er de 9 landene som har data om forventet levealder kun fra 1950.

Spørsmål 12

```
pop <- read_csv(
  # paste0() er bare paste() med sep=""
  paste0("ddf--gapminder--systema_globalis-master/",
         "countries-etc-datapoints/",
         "ddf--datapoints--population_total--by--geo--time.csv"
  ),
    col_types = cols(time = col_date(format = "%Y"))
  )
```

```
g_c <- g_c %>%
  left_join(pop, by = c("country" = "geo", "year" = "time"))
rm(pop)
```

Spørsmål 13

```
gdp_pc <- read_csv(
  paste0("ddf--gapminder--systema_globalis-master/",
         "countries-etc-datapoints/",
         "ddf--datapoints--gdppercapita_us_inflation_adjusted--by--geo--time.csv"
         ),
    col_types = cols(time = col_date(format = "%Y"))
  )
```

```
g_c <- g_c %>%
  left_join(gdp_pc, by = c("country" = "geo", "year" = "time"))
rm(gdp_pc)
```

```
g_c = g_c %>%
  rename("lifeExp" = "life_expectancy_years",
         "pop" = "population_total",
         "gdpPercap" = "gdppercapita_us_inflation_adjusted")
```

```
names(g_c)
```

```
##  [1] "country"          "name"             "iso3166_1_alpha3" "un_sdg_region"
##  [5] "world_4region"    "continent"        "world_6region"    "year"
##  [9] "lifeExp"          "pop"              "gdpPercap"
```

Her har vi gitt nytt navn til 3 variabler, slik at de har samme navn som i *gapminder*-datasettet.

Spørsmål 14

```
t1 <- paste(c(seq(1800, 2015, by = 5), 2019), "01-01", sep = "-") %>%
  parse_date(format = "%Y-%m-%d")

g_c_5 <- g_c %>%
  filter(year %in% t1) %>%
  select(country, name, continent, year, lifeExp, pop, gdpPercap)

dim(g_c_5)
```

```
## [1] 8505    7
```

```
g_c_gdpprc <- g_c_5 %>%
  group_by(gdpPercap) %>%
  summarise(min_year = min(year))

table(g_c_gdpprc$min_year)
```

```
##
## 1800-01-01 1960-01-01 1965-01-01 1970-01-01 1975-01-01 1980-01-01 1985-01-01
##          1         86         93        108        112        133        142
## 1990-01-01 1995-01-01 2000-01-01 2005-01-01 2010-01-01 2015-01-01 2019-01-01
##        161        178        186        189        191        188        186
```

Spørsmål 15

```
g_c <- g_c %>%
  filter(!is.na(gdpPercap)) %>%
  group_by(name) %>%
  summarise(nr = n()) %>%
  arrange((name))

print(g_c)
```

```
## # A tibble: 191 x 2
##    name                   nr
##    <chr>               <int>
##  1 Afghanistan            18
##  2 Albania                40
##  3 Algeria                60
##  4 Andorra                50
##  5 Angola                 40
##  6 Antigua and Barbuda    43
##  7 Argentina              60
##  8 Armenia                30
##  9 Australia              60
## 10 Austria                60
## # ... with 181 more rows
```

Den lengste tidsserien med data om BNP per innbygger er 60. Filtrerer disse ut og viser alle landene:

```
g_c_60 <- g_c %>%
  filter(nr == 60)
print(g_c_60)
```

```
## # A tibble: 85 x 2
##    name          nr
##    <chr>      <int>
##  1 Algeria       60
##  2 Argentina     60
##  3 Australia     60
##  4 Austria       60
##  5 Bahamas       60
##  6 Bangladesh    60
##  7 Belgium       60
##  8 Belize        60
```

```
##  9 Benin         60
## 10 Bolivia       60
## # ... with 75 more rows
```

Vi har 85 observasjoner med 60 år med data om BNP per innbygger.

Spørsmål 16

Lager ny datasett som inkluderer land med data fra 1960 til 2019 og uten NA-verdier.

```
my_gapminder_60 <- g_c_5 %>%
  filter(!is.na(gdpPercap)) %>%
  group_by(country) %>%
  summarise(min_year = min(year))
```

```
dim(my_gapminder_60)
```

```
## [1] 191   2
```

Vi har 191 land i dette datasettet.

```
c_m_y_60 <- my_gapminder_60$country[my_gapminder_60$min_year == "1960-01-01"]
g_c_1960 <- g_c_5 %>%
  filter(country %in% c_m_y_60)
```

```
dim(g_c_1960)
```

```
## [1] 3870    7
```

```
length(unique(g_c_1960$country))
```

```
## [1] 86
```

Det er 86 land med data melloim 1960-2019.

Her er landene fordelt utover kontinentene:

```
g_c_1960 %>%
  distinct(country, continent) %>%
  group_by(continent) %>%
  count() %>%
  kable()
```

continent	n
Africa	29
Americas	25
Asia	14
Europe	15
Oceania	3

```
(num_NA <- g_c_1960[is.na(g_c_1960$gdpPercap) == TRUE, ])
```

```
## # A tibble: 2,754 x 7
##    country name      continent year       lifeExp    pop gdpPercap
##    <chr>   <chr>     <chr>     <date>       <dbl>  <dbl>     <dbl>
##  1 arg     Argentina Americas  1800-01-01    33.2 534000        NA
##  2 arg     Argentina Americas  1805-01-01    33.2 465622        NA
```

```
##  3 arg     Argentina Americas  1810-01-01    33.2 419661        NA
##  4 arg     Argentina Americas  1815-01-01    33.2 465972        NA
##  5 arg     Argentina Americas  1820-01-01    33.2 530996        NA
##  6 arg     Argentina Americas  1825-01-01    33.2 582027        NA
##  7 arg     Argentina Americas  1830-01-01    33.2 634974        NA
##  8 arg     Argentina Americas  1835-01-01    33.2 698047        NA
##  9 arg     Argentina Americas  1840-01-01    33.2 776366        NA
## 10 arg     Argentina Americas  1845-01-01    33.2 920317        NA
## # ... with 2,744 more rows
```

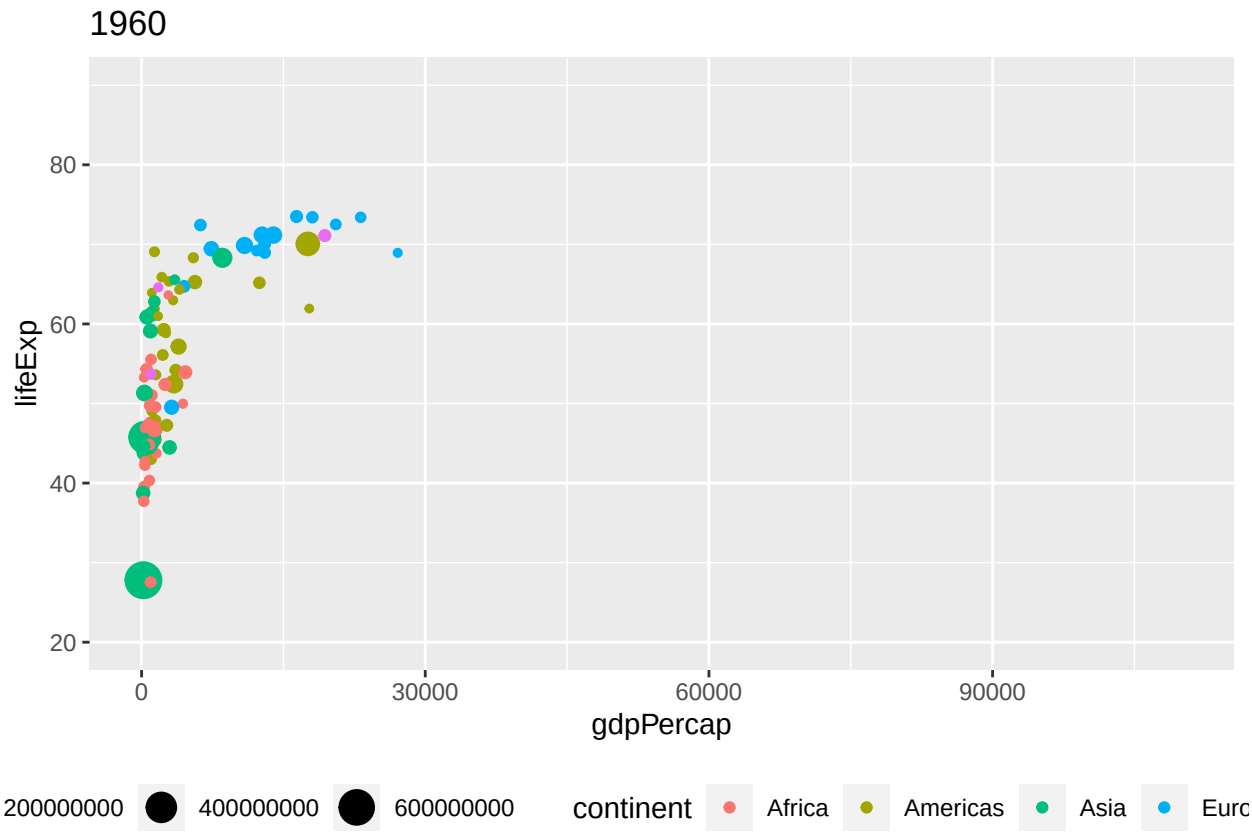
Her har vi sjekket NA-verdiene. Men vi gjør den mer oversiktelig med hjelp av paste-funksjonen.

```
paste("Number of NAs in g_c_1960 is", dim(num_NA)[1], sep = " ")
```

```
## [1] "Number of NAs in g_c_1960 is 2754"
```

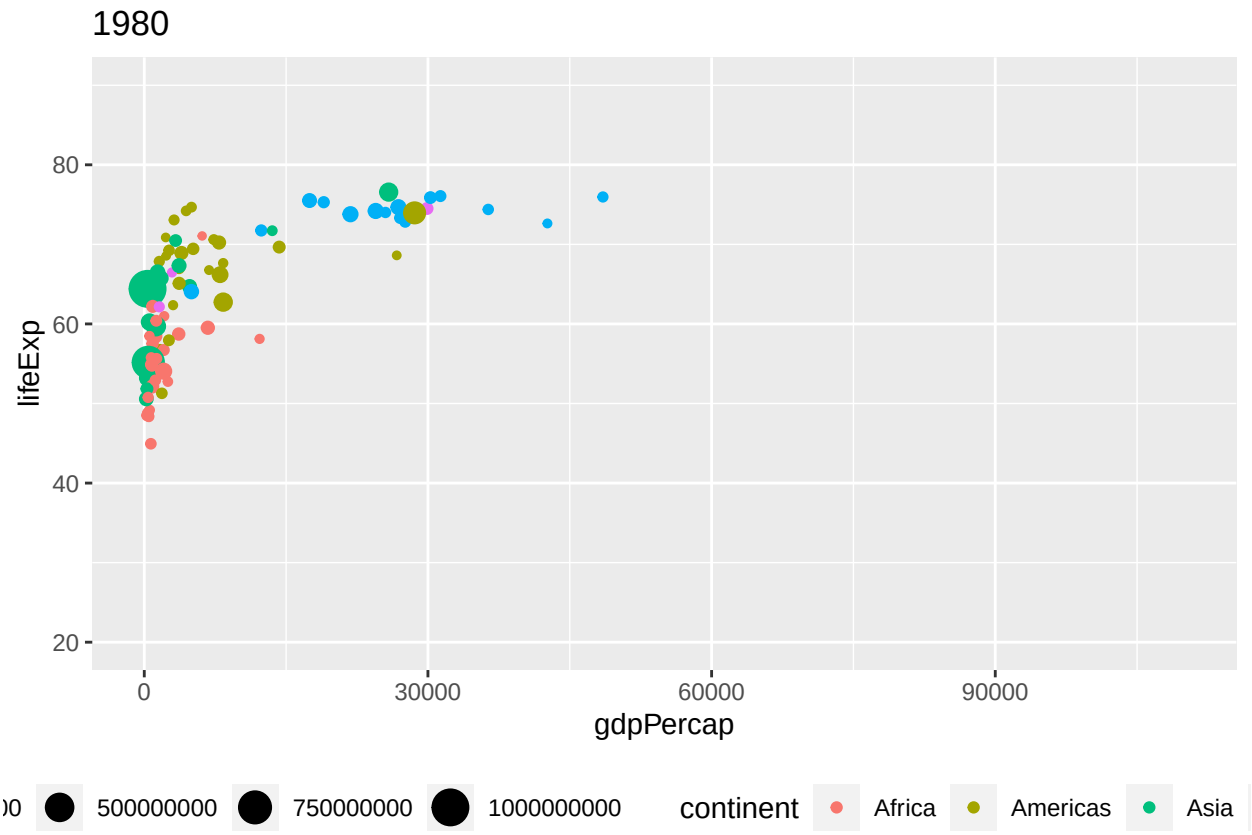
Spørsmål 17

```
g_c_1960 %>%
filter(year == "1960-01-01") %>%
  ggplot(mapping = aes(x = gdpPercap,
                       y = lifeExp,
                       size = pop,
                       colour = continent)) +
  geom_point() +
  # coord_cartesian() virker helt fint, men funksjonene xlim og ylim er kanskje lettere
  # å huske. Endrer xlim til 0, 110000 slik at vi kan ha samme aksjer på alle figurene
  # Like akser gjør det enklere å sammenligne over tid
  ylim(20,90) +
  xlim(0,110000) +
  # coord_cartesian(ylim = c(20, 90),
  #                 xlim = c(0,30000)) +
  ggtitle("1960") +
  theme(legend.position = "bottom")
```



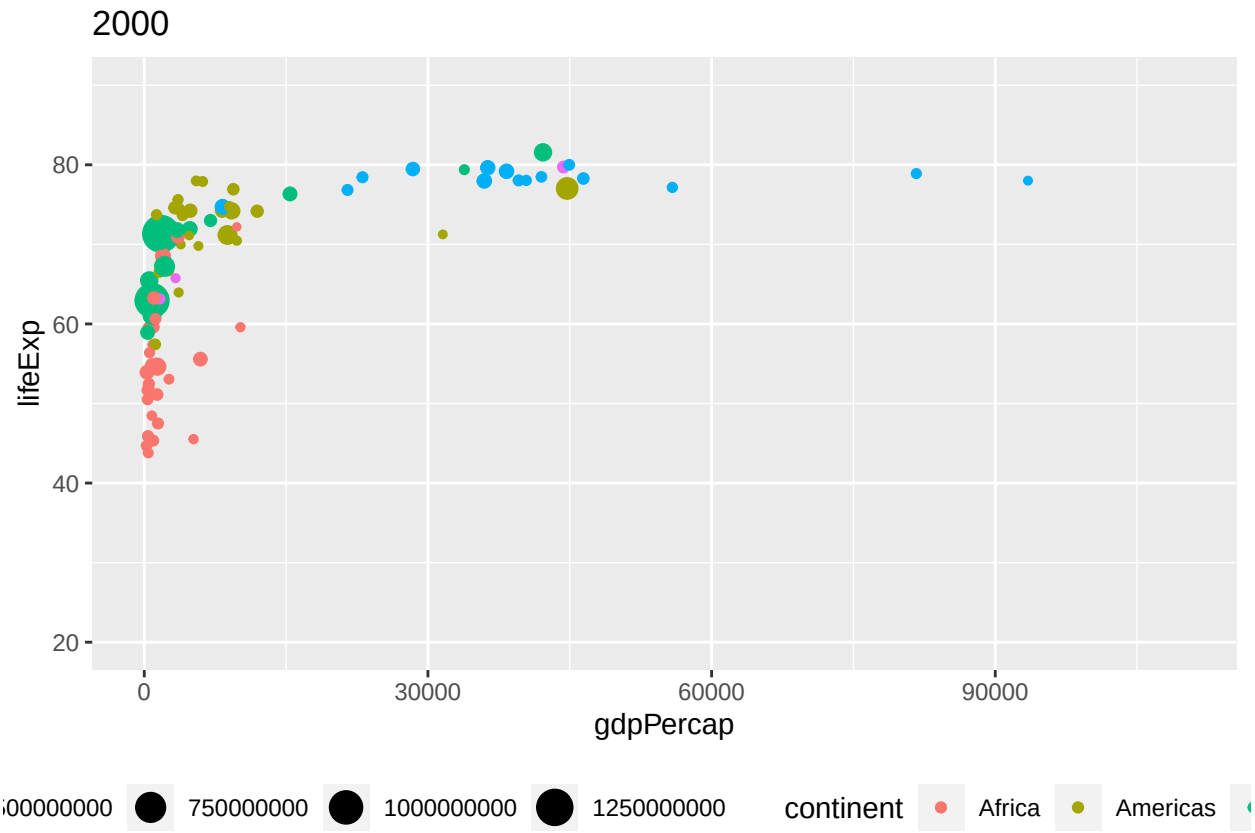
i 1960 er det mindre land som er registrert og vi ser at Europa dominerer både med **lifeExp** og **gdpPercap**.

```
g_c_1960 %>%
filter(year == "1980-01-01") %>%
  ggplot(mapping = aes(x = gdpPercap,
                       y = lifeExp,
                       size = pop,
                       colour = continent)) +
  geom_point() +
  coord_cartesian(ylim = c(20, 90),
                  xlim = c(0,110000)) +
  ggtitle("1980") +
  theme(legend.position = "bottom")
```



i 1980 er det mange flere registrerte land. Europa har nå fått selskap av Amerika. Vi ser landene i Asia og Afrika henger bak med både forventet levealder og BNP per innbygger.

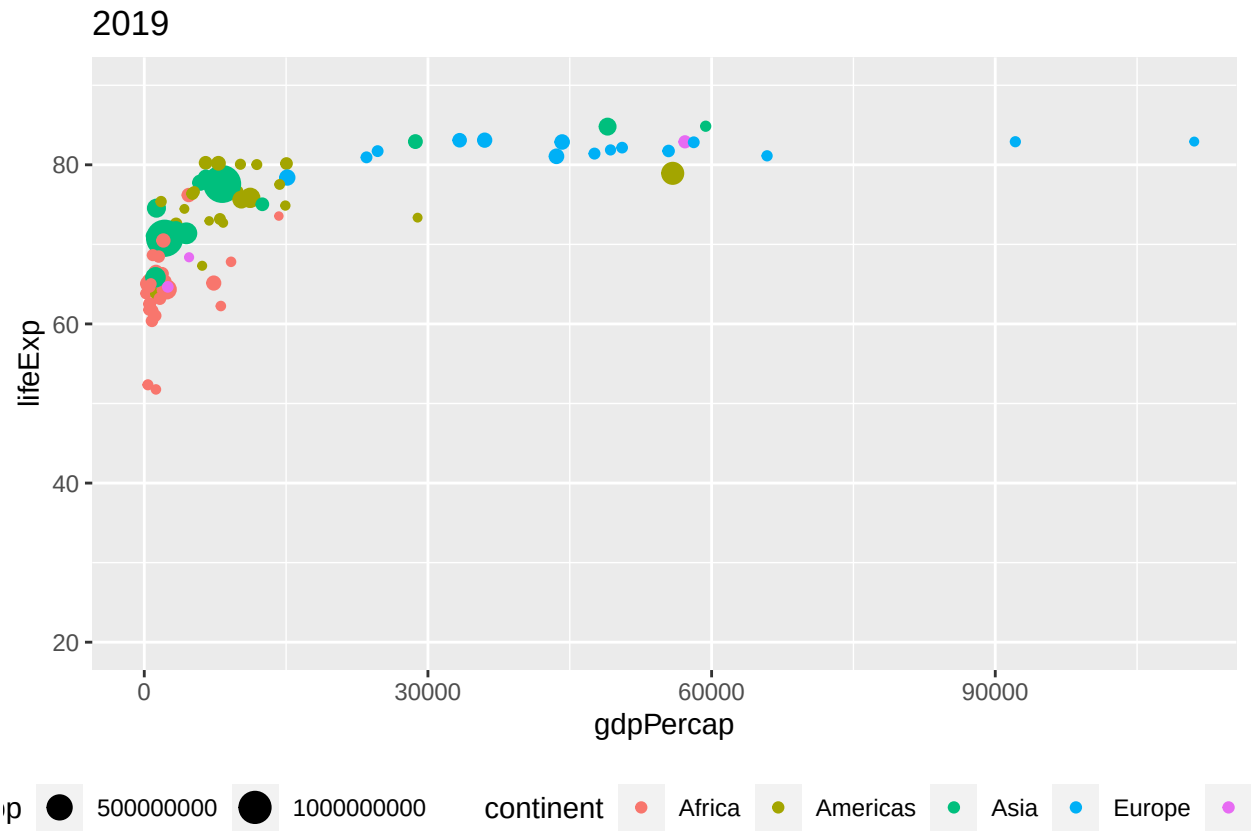
```
g_c_1960 %>%
filter(year == "2000-01-01") %>%
  ggplot(mapping = aes(x = gdpPercap,
                       y = lifeExp,
                       size = pop,
                       colour = continent)) +
  geom_point() +
  coord_cartesian(ylim = c(20, 90),
                  xlim = c(0,110000)) +
  ggtitle("2000") +
  theme(legend.position = "bottom")
```



Fra 1980 til 2000 ser vi forventet levealder har økt betraktelig, spesielt for land med lav BNP per innbygger. Vi ser også at BNP per innbygger har økt for de fleste land.

```
g_c_1960 %>%
filter(year == "2019-01-01") %>%
  ggplot(mapping = aes(x = gdpPercap,
                       y = lifeExp,
                       size = pop,
                       colour = continent)) +
  geom_point() +
  coord_cartesian(ylim = c(20, 90),
                  xlim = c(0,110000)) +
  ggtitle("2019") +
  theme(legend.position = "bottom")
```

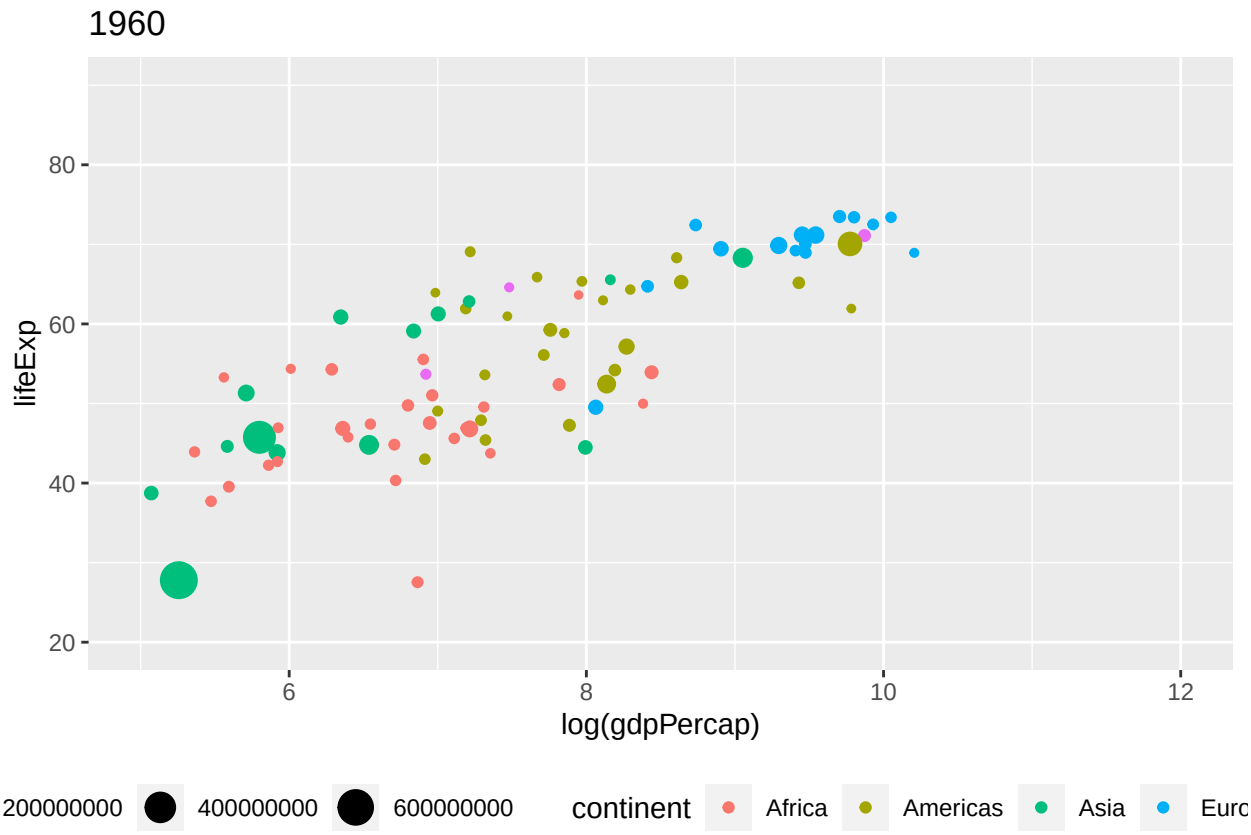
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



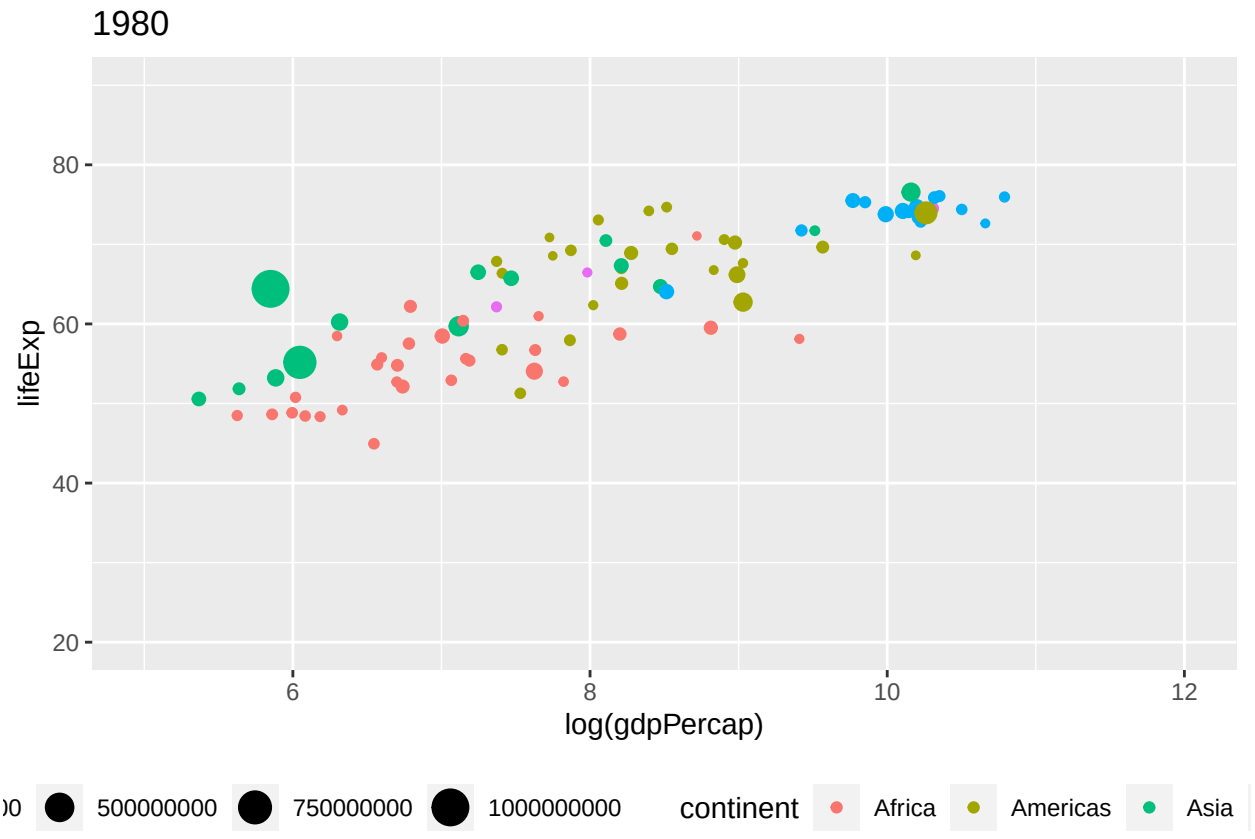
Alle land har økende BNP per innbygger. Forventet levealder har ikke økt noe særlig fra 2000.

Spørsmål 18

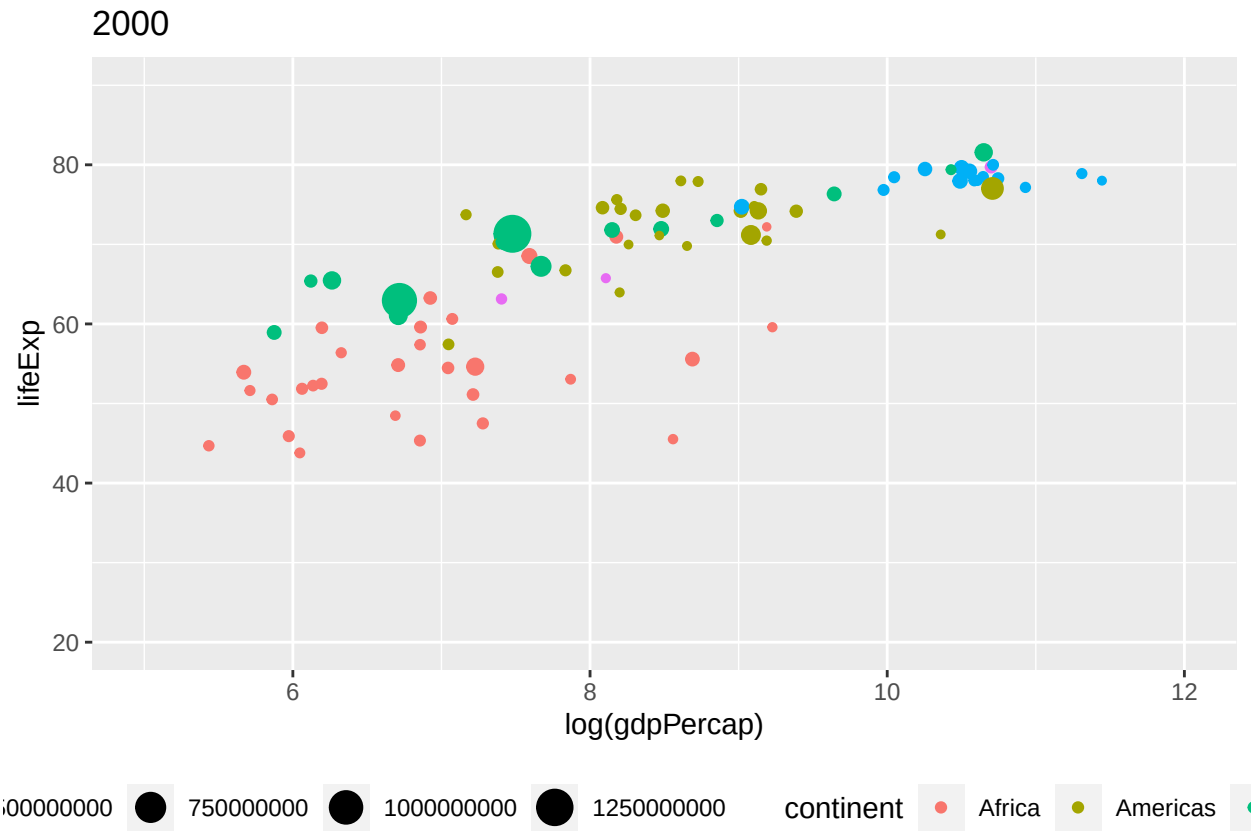
```
g_c_1960 %>%
filter(year == "1960-01-01") %>%
  ggplot(mapping = aes(x = log(gdpPercap),
                       y = lifeExp,
                       size = pop,
                       colour = continent)) +
  geom_point() +
  coord_cartesian(ylim = c(20, 90),
                  xlim = c(5,12)) +
  ggtitle("1960") +
  theme(legend.position = "bottom")
```




```
g_c_1960 %>%
filter(year == "1980-01-01") %>%
  ggplot(mapping = aes(x = log(gdpPercap),
                       y = lifeExp,
                       size = pop,
                       colour = continent)) +
  geom_point() +
  coord_cartesian(ylim = c(20, 90),
                  xlim = c(5,12)) +
  ggtitle("1980") +
  theme(legend.position = "bottom")
```

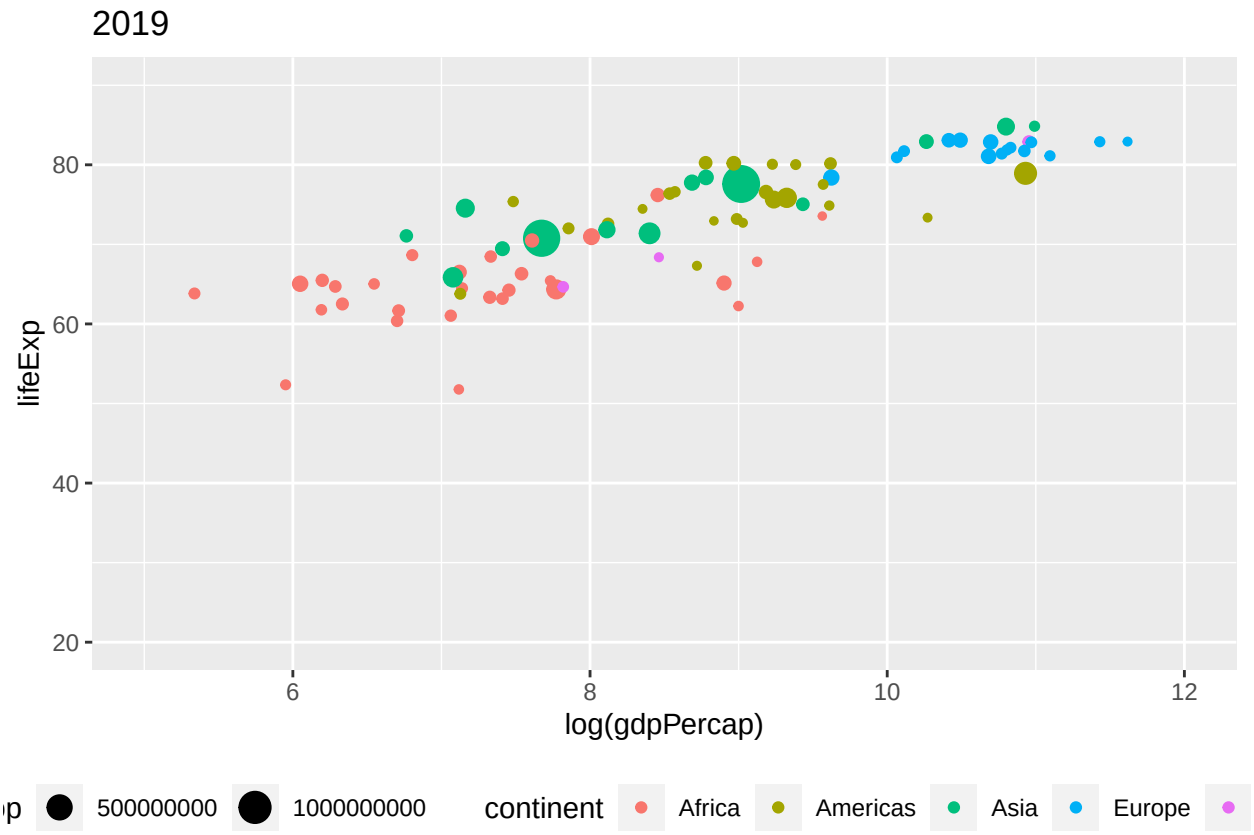



```
g_c_1960 %>%
filter(year == "2000-01-01") %>%
  ggplot(mapping = aes(x = log(gdpPercap),
                       y = lifeExp,
                       size = pop,
                       colour = continent)) +
  geom_point() +
  coord_cartesian(ylim = c(20, 90),
                  xlim = c(5,12)) +
  ggtitle("2000") +
  theme(legend.position = "bottom")
```




```
g_c_1960 %>%
filter(year == "2019-01-01") %>%
  ggplot(mapping = aes(x = log(gdpPercap),
                       y = lifeExp,
                       size = pop,
                       colour = continent)) +
  geom_point() +
  coord_cartesian(ylim = c(20, 90),
                  xlim = c(5,12)) +
  ggtitle("2019") +
  theme(legend.position = "bottom")
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



Spørsmål 19

Fra 1960 til 1980 er det hentet inn masse mer data på landene i verden. Fra 1980 til 2000 øker forventet levealder spesielt for land med lav BNP per innbygger. Ved hjelp av **log(gdpPercap)** får vi en mye mindre spredning i BNP per innbygger enn kun ved **gdpPercap**. Fortsatt dominerer land i Asia og Afrika med lavest BNP per innbygger, men med hjelp av log-funksjonen så er avvikket fra de europeeiske landene lengst til høyre mindre.

Oppsummert kan vi si at alle landene beveger seg oppover og til høyre i diagrammet, som vil si at BNP per innbygger øker og leveralder øker for samtlige land.

Spørsmål 20

```
write.table(g_c_5, file="my_gapminder.csv", sep = ",")

#write.table(g_c_1960, file="my_gapminder_red.csv", sep = ",")
# Vi bør bruke tidyverse funksjonen. Ofte kjappere og kanskje mer robust enn den klassiske
write_csv(g_c_1960, file="my_gapminder_red.csv")
```