PSZB17-210 - Seminar_4

Zoltan Kekecs

October 6, 2020

4. Ora - Adatexploracio

Az ora celja az adatexploracios modszerek elsajatitasa.

Package-ek betoltese

A kovetkező package-ekre lesz szuksegunk

```
if (!require("gridExtra")) install.packages("gridExtra")
library(gridExtra) # for grid.arrange
if (!require("psych")) install.packages("psych")
library(psych) # for describe
if (!require("tidyverse")) install.packages("tidyverse")
library(tidyverse) # for dplyr and ggplot2
```

Adatok betoltese

Beolvassuk a WHO altal 2020.09.28-an feltoltott COVID-19 adatokat a read_csv() funkcioval, es elmentjuk egy COVID_adat nevu objektumba. A **read_csv()** funkcio a tidyverse resze, es egybol tibble formatumban menti el az adatainkat.

```
COVID_adat <- read_csv("https://raw.githubusercontent.com/owid/covid-19-data/master/public/data/owid-covid-data.csv")
```

Adatok attekintese

Mindig erdemes azzal kezdeni, hogy **megismerkedunk az adat** szerkezetevel es tartalmaval.

A **tibble objektum** meghivasaval kapthatunk nemi informaciot az adattabla szerkezeterol. Lathatjuk hany sor es hany oszlop van az adattablaban, es lathatjuk milyen class-ba tartoznak (chr, dbl ...)

```
COVID_adat
```

```
## # A tibble: 48,379 x 41
##    iso_code continent location date       total_cases new_cases
new_cases_smoot~
##    <chr>    <chr>     <chr>    <date>           <dbl>     <dbl>
<dbl>
##  1 AFG      Asia      Afghani~ 2019-12-31           0         0
```

```
NA
##  2 AFG      Asia      Afghani~ 2020-01-01           0         0
NA
##  3 AFG      Asia      Afghani~ 2020-01-02           0         0
NA
##  4 AFG      Asia      Afghani~ 2020-01-03           0         0
NA
##  5 AFG      Asia      Afghani~ 2020-01-04           0         0
NA
##  6 AFG      Asia      Afghani~ 2020-01-05           0         0
NA
##  7 AFG      Asia      Afghani~ 2020-01-06           0         0
0
##  8 AFG      Asia      Afghani~ 2020-01-07           0         0
0
##  9 AFG      Asia      Afghani~ 2020-01-08           0         0
0
## 10 AFG      Asia      Afghani~ 2020-01-09           0         0
0
## # ... with 48,369 more rows, and 34 more variables: total_deaths <dbl>,
## #   new_deaths <dbl>, new_deaths_smoothed <dbl>, total_cases_per_million
<dbl>,
## #   new_cases_per_million <dbl>, new_cases_smoothed_per_million <dbl>,
## #   total_deaths_per_million <dbl>, new_deaths_per_million <dbl>,
## #   new_deaths_smoothed_per_million <dbl>, new_tests <lgl>, total_tests
<lgl>,
## #   total_tests_per_thousand <lgl>, new_tests_per_thousand <lgl>,
## #   new_tests_smoothed <lgl>, new_tests_smoothed_per_thousand <lgl>,
## #   tests_per_case <lgl>, positive_rate <lgl>, tests_units <lgl>,
## #   stringency_index <dbl>, population <dbl>, population_density <dbl>,
## #   median_age <dbl>, aged_65_older <dbl>, aged_70_older <dbl>,
## #   gdp_per_capita <dbl>, extreme_poverty <dbl>, cardiovasc_death_rate
<dbl>,
## #   diabetes_prevalence <dbl>, female_smokers <dbl>, male_smokers <dbl>,
## #   handwashing_facilities <dbl>, hospital_beds_per_thousand <dbl>,
## #   life_expectancy <dbl>, human_development_index <dbl>
```

Leiro statisztikak

Ha az egyes valtozok **leiro statisztikaira** (descriptive statistics) vagyunk kivancsiak, kerhetjuk ezt a mar tanult modon.

Peldaul lekerhetjuk a valtozo alapveto legalacsonyabb es legmagasabb erteket, atlagat, medianjat, a kvartiliseket, es hogy hany hianyzo adat van (ha van) a **summary()** funkcioval (miutan a select funkcioval kivalasztottuk, melyik valtozora vagyunk kivancsiak)

```
COVID_adat %>%
  select(total_cases) %>%
  summary()
```

```
##   total_cases
##  Min.   :       0
##  1st Qu.:      66
##  Median :    1118
##  Mean   :  110301
##  3rd Qu.:   12229
##  Max.   :35523518
##  NA's   :614
```

Vagy megkapthatjuk ugyanezt az osszes valtozora, ha ugyanezt az egesz adattablara futtatjuk le. Persze a karakter osztalyba tartozo valtozoknal mindezeknek a leiro statisztikaknak nincs ertelme, ott csak a class informaciot kaptjuk az output-ban.

```
COVID_adat %>%
  summary()
```

Gyakorlas

- Hany regisztralt eset volt osszesen Magyarorszagon a tegnapi napig (*total_cases*)?
- Mi volt a legmagasabb uj eset-szam Magyarorszagon (*new_cases*)?

Megtobb leiro statisztika

A **Psych** package segitsegevel a **describe()** funkcio megtobb hasznos informaciot adhat. Ez a funkcio elsosorban szam-valtozok leirasara szolgal, es karakter tipusu kategorikus valtozok eseten sok warning message-et ad, ezert erdemes a funciot csak a szam-valtozokra lefuttatni (ezt alabb a select() funkcioval erem el.)

```
COVID_adat %>%
  select(-date, -iso_code, -continent, -location, -contains("tests"), -
positive_rate) %>%
  describe()
```

```
##                                vars     n        mean           sd
median
## total_cases                       1 47765   110300.98   1140970.15
1118.00
## new_cases                         2 47548     1494.22     13508.82
11.00
## new_cases_smoothed                3 46766     1482.07     13323.14
15.71
## total_deaths                      4 47765     4304.99     39955.23
22.00
## new_deaths                        5 47548       43.85       368.45
0.00
## new_deaths_smoothed               6 46766       43.94       358.58
0.14
```

```
## total_cases_per_million             7 47484     2049.94      4286.88
305.83
## new_cases_per_million               8 47484       25.83        78.06
1.65
## new_cases_smoothed_per_million     9 46701       25.40        59.46
3.10
## total_deaths_per_million           10 47484       60.40       146.73
5.37
## new_deaths_per_million             11 47484        0.57         2.99
0.00
## new_deaths_smoothed_per_million    12 46701        0.57         1.89
0.02
## stringency_index                   13 40223       57.28        27.23
62.96
## population                         14 48098 87995367.45 611197259.64
8654618.00
## population_density                 15 45892      359.80      1651.54
88.12
## median_age                         16 43121       31.30         9.03
31.40
## aged_65_older                      17 42478        9.25         6.32
6.98
## aged_70_older                      18 42897        5.85         4.31
4.42
## gdp_per_capita                     19 42560    20856.09     20410.24
14048.88
## extreme_poverty                    20 28396       12.17        19.26
2.00
## cardiovasc_death_rate              21 43129      251.70       117.50
238.34
## diabetes_prevalence                22 44654        8.05         4.15
7.11
## female_smokers                     23 33755       10.80        10.48
6.40
## male_smokers                       24 33326       32.64        13.42
31.40
## handwashing_facilities             25 20226       52.40        31.61
55.18
## hospital_beds_per_thousand         26 38941        3.11         2.52
2.50
## life_expectancy                    27 47489       74.01         7.38
75.40
## human_development_index            28 41608        0.72         0.15
0.75
##                                    trimmed         mad      min
max
## total_cases                        8265.70     1653.10     0.00
3.552352e+07
## new_cases                           103.65       16.31 -8261.00
3.227800e+05
```

```
## new_cases_smoothed                   107.59       23.30  -552.00
2.978010e+05
## total_deaths                         191.06       32.62     0.00
1.042398e+06
## new_deaths                             1.87        0.00 -1918.00
1.049100e+04
## new_deaths_smoothed                    2.05        0.21  -232.14
7.456710e+03
## total_cases_per_million             1029.55      453.18     0.00
4.397409e+04
## new_cases_per_million                  9.91        2.45 -2212.55
4.944380e+03
## new_cases_smoothed_per_million        11.32        4.60  -269.98
8.829200e+02
## total_deaths_per_million              22.50        7.96     0.00
1.237550e+03
## new_deaths_per_million                 0.13        0.00   -67.90
2.153800e+02
## new_deaths_smoothed_per_million        0.18        0.03    -9.68
6.314000e+01
## stringency_index                      59.43       27.46     0.00
1.000000e+02
## population                       15635615.65 12405280.40   809.00
7.794799e+09
## population_density                   124.52       94.65     0.14
1.934750e+04
## median_age                            31.33       12.16    15.10
4.820000e+01
## aged_65_older                          8.67        5.93     1.14
2.705000e+01
## aged_70_older                          5.36        3.90     0.53
1.849000e+01
## gdp_per_capita                     17657.19    15797.71   661.24
1.169356e+05
## extreme_poverty                        7.72        2.67     0.10
7.760000e+01
## cardiovasc_death_rate                240.59      121.87    79.37
7.244200e+02
## diabetes_prevalence                    7.63        3.68     0.99
2.336000e+01
## female_smokers                         9.47        8.01     0.10
4.400000e+01
## male_smokers                          31.98       14.38     7.70
7.810000e+01
## handwashing_facilities                52.93       45.28     1.19
9.900000e+01
## hospital_beds_per_thousand             2.72        1.93     0.10
1.380000e+01
## life_expectancy                       74.70        7.09    53.28
8.675000e+01
```

```
## human_development_index                0.73        0.16     0.35
9.500000e-01
##                                       range  skew kurtosis         se
## total_cases                    3.552352e+07 20.75   498.06    5220.59
## new_cases                      3.310410e+05 16.55   311.18      61.95
## new_cases_smoothed             2.983530e+05 16.42   304.20      61.61
## total_deaths                   1.042398e+06 17.95   366.10     182.82
## new_deaths                     1.240900e+04 14.64   244.95       1.69
## new_deaths_smoothed            7.688860e+03 13.87   210.20       1.66
## total_cases_per_million        4.397409e+04  4.21    24.09      19.67
## new_cases_per_million          7.156920e+03 12.89   497.46       0.36
## new_cases_smoothed_per_million 1.152900e+03  5.13    39.84       0.28
## total_deaths_per_million       1.237550e+03  4.14    21.22       0.67
## new_deaths_per_million         2.832800e+02 30.50  1620.05       0.01
## new_deaths_smoothed_per_million 7.282000e+01  9.64   153.04       0.01
## stringency_index               1.000000e+02 -0.59    -0.65       0.14
## population                     7.794798e+09 11.83   144.95 2786877.56
## population_density             1.934736e+04  9.96   106.86       7.71
## median_age                     3.310000e+01 -0.02    -1.22       0.04
## aged_65_older                  2.591000e+01  0.65    -0.86       0.03
## aged_70_older                  1.797000e+01  0.79    -0.54       0.02
## gdp_per_capita                 1.162744e+05  1.65     3.47      98.93
## extreme_poverty                7.750000e+01  1.80     2.29       0.11
## cardiovasc_death_rate          6.450500e+02  0.91     0.86       0.57
## diabetes_prevalence            2.237000e+01  1.09     1.42       0.02
## female_smokers                 4.390000e+01  0.89    -0.30       0.06
## male_smokers                   7.040000e+01  0.55     0.33       0.07
## handwashing_facilities         9.781000e+01 -0.13    -1.45       0.22
## hospital_beds_per_thousand     1.370000e+01  1.77     3.95       0.01
## life_expectancy                3.347000e+01 -0.75    -0.12       0.03
## human_development_index        6.000000e-01 -0.49    -0.75       0.00
```

Gyakorlas

- Mi az egy millio fore eso uj esetek (*new_cases_per_million*) ferdesegi mutatoja (skew/skewness)?
- Hany valid (nem NA) adat szerepel az adatbazisban az egy fore eso gdp-rol (*gdp_per_capita*)?

Faktorok

Nehany karaktervaltozonak csak **korlatozott mennyisegu eleme** lehet, mint peldaul a continent (North America, Asia, Africa, Europe, South America, Oceania). Ezeket megjelolhetjuk faktor (factor) osztalyu valtozokent, es akkor az R tobb informaciot fog adni rola.

A **levels()** funkcio megmutatja mik a faktorunk szintjei, de lathato ez akkor is ha csak meghivjuk a valtozot magat.

A **table()** funkcio pedig tablazatot keszit arrol, hogy az egyes csoportokban hany megfigyeles talalhato

Amikor kilistazzuk a faktor valtozot, akkor is kiirja az R a lista aljara, hogy milyen faktorszintek vannak. (Alabb csinalunk egy COVID_adat_tegnap valtozot, amivel csak a tegnapi adatokat nezzuk, hogy kisebb legyen az adattabla amivel dolgozunk.)

```
COVID_adat <- COVID_adat %>%
              mutate(continent = factor(continent),
                     location = factor(location))

levels(COVID_adat$continent)

## [1] "Africa"        "Asia"          "Europe"        "North America"
## [5] "Oceania"       "South America"

table(COVID_adat$continent)

##
##        Africa          Asia        Europe North America       Oceania
##         11327         11546         12677          7577          1807
## South America
##          2883

COVID_adat_tegnap = COVID_adat %>%
  filter(date == "2020-09-28")

COVID_adat_tegnap$continent

##   [1] Asia          Europe        Africa        Europe        Africa
##   [6] North America North America South America Asia          North
America
##  [11] Oceania       Europe        Asia          North America Asia
##  [16] Asia          North America Europe        Europe        North
America
##  [21] Africa        North America Asia          South America North
America
##  [26] Europe        Africa        South America North America Asia
##  [31] Europe        Africa        Africa        Asia          Africa
##  [36] North America Africa        North America Africa        Africa
##  [41] South America Asia          South America Africa        Africa
##  [46] North America Africa        Europe        North America North
America
##  [51] Europe        Europe        Africa        Europe        Africa
##  [56] North America North America South America Africa        North
America
##  [61] Africa        Africa        Europe        Africa        Europe
##  [66] South America Oceania       Europe        Europe        Oceania
```

```
##  [71] Africa        Africa        Asia          Europe        Africa
##  [76] Europe        Europe        North America North America Oceania
##  [81] North America Europe        Africa        Africa        South
America
##  [86] North America North America Europe        Europe        Asia
##  [91] Asia          Asia          Asia          Europe        Europe
##  [96] Asia          Europe        North America Asia          Europe
## [101] Asia          Asia          Africa        Europe        Asia
## [106] Asia          Asia          Europe        Asia          Africa
## [111] Africa        Africa        Europe        Europe        Europe
## [116] Europe        Africa        Africa        Asia          Asia
## [121] Africa        Europe        Africa        Africa        North
America
## [126] Europe        Europe        Asia          Europe        North
America
## [131] Africa        Africa        Asia          Africa        Asia
## [136] Europe        Oceania       Oceania       North America Africa
## [141] Africa        Oceania       Europe        Asia          Asia
## [146] Asia          North America Oceania       South America South
America
## [151] Asia          Europe        Europe        North America Asia
## [156] Europe        Europe        Africa        North America North
America
## [161] North America Europe        Africa        Asia          Africa
## [166] Europe        Africa        Africa        Asia          North
America
## [171] Europe        Europe        Africa        Africa        Asia
## [176] Africa        Europe        Asia          Africa        South
America
## [181] Africa        Europe        Europe        Asia          Asia
## [186] Asia          Africa        Asia          Asia          Africa
## [191] North America Africa        Asia          North America Africa
## [196] Europe        Asia          Europe        North America North
America
## [201] South America Asia          Europe        South America Asia
## [206] Africa        Asia          Africa        Africa        <NA>
## [211] <NA>
## Levels: Africa Asia Europe North America Oceania South America
```

Igy mar a fenti **summary()** funkcio is kiadja az **egyes faktorszintekrol** hogy hanyan tartoznak oda.

```
COVID_adat_tegnap %>%
  select(continent) %>%
  summary()

##          continent
##  Africa       :55
##  Asia         :46
##  Europe       :51
```

```
##  North America:36
##  Oceania      : 8
##  South America:13
##  NA's         : 2
```

Van, hogy szeretnenk **kizarni** bizonyos **faktorszinteket** az elemzesbol. Pl. ha valamelyik faktor szintbol nagyon keves megfigyeles van, mondjuk Oceaniat, mondjuk mert ugy gondoljuk hogy az tulsagosan “elszigetelt” a vilag tobbi reszetol, oket lehet hogy szeretnenk kizarni a kesobbi elemzesekbol hogy egyszerusitsuk az eredmenyeink ertelmezeset. Ezt a mar korabban tanult **filter()** funkcio segitsegevel konnyeden megtehetjuk, azonban arra figyelnunk kell, hogy az R megjegyzi a faktorszinteket, es azt azt kovetoen is a **valtozohoz rendelve tartja**, miutan mar az adott faktorszintbol nincs egy megifgyeles sem az adattablaban.

```
COVID_adat_tegnap %>%
  filter(continent != "Oceania") %>%
  select(total_cases, continent) %>%
  summary()
```

```
##   total_cases              continent
##  Min.   :      3   Africa       :55
##  1st Qu.:   1743   Asia         :46
##  Median :   9682   Europe       :51
##  Mean   : 165022   North America:36
##  3rd Qu.:  72691   Oceania      : 0
##  Max.   :7115046   South America:13
```

Igy ezeket a szinteket ejthetjuk a **droplevels()** funkcioval.

```
COVID_adat_tegnap_noOceania = COVID_adat_tegnap %>%
  filter(continent != "Oceania") %>%
  mutate(continent = droplevels(continent))



COVID_adat_tegnap_noOceania %>%
  select(continent) %>%
  summary()
```

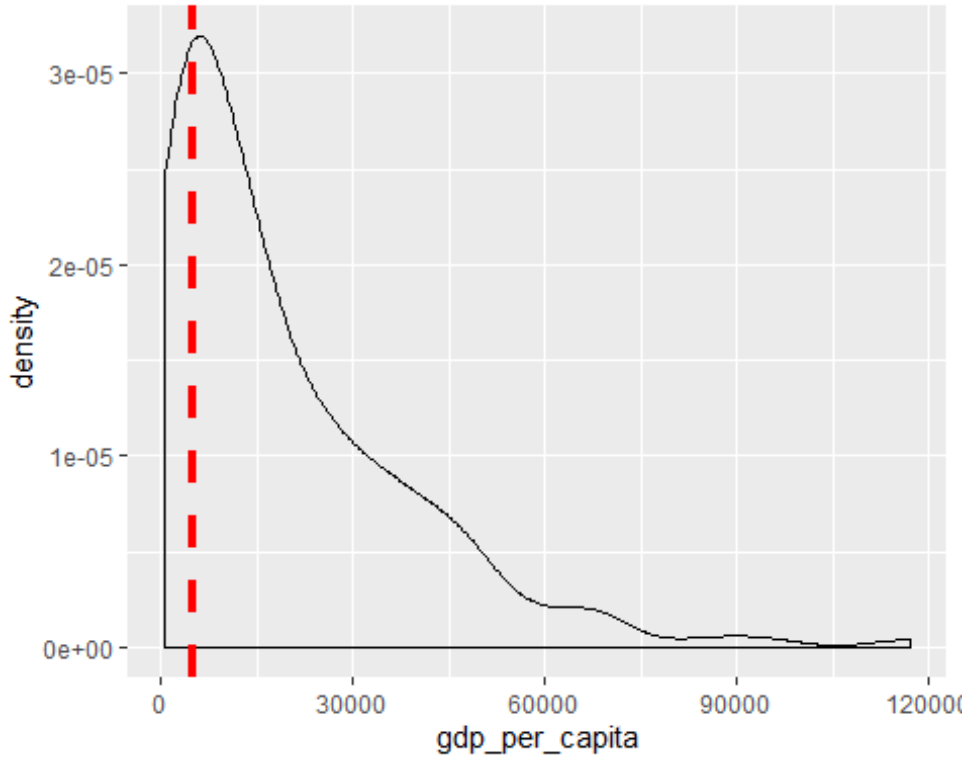
```
##          continent
##  Africa       :55
##  Asia         :46
##  Europe       :51
##  North America:36
##  South America:13
```

Elofordul, hogy egy **numerikus valtozot akarunk atalakitani faktorra**, pl. elkepzelheto hogy ossze akarjuk hasonlitani azokat az orszagokat ahol 5000 alatti a gdp_per_capita azokkal akinel e feletti, hogy hogyan kulonboznek a COVID adatok.

```r
COVID_adat_tegnap %>%
  select(gdp_per_capita, continent) %>%
  drop_na() %>%
  group_by(continent) %>%
  summarize(mean_gdp = mean(gdp_per_capita))
```

```
## # A tibble: 6 x 2
##   continent     mean_gdp
##   <fct>            <dbl>
## 1 Africa           5444.
## 2 Asia            22185.
## 3 Europe          33361.
## 4 North America   21655.
## 5 Oceania         23315.
## 6 South America   13841.
```

```r
COVID_adat_tegnap %>%
  select(gdp_per_capita) %>%
  drop_na() %>%
  ggplot() +
  aes(x = gdp_per_capita) +
  geom_density() +
  geom_vline(xintercept = 5000, linetype="dashed",
                color = "red", size=1.5)
```



Folytonos valtozok atkodolasa kategorikus valtozova

Ilyenkor hasznalhatjuk a **mutate()** es **case_when()** funkciok kombinaciojat hogy csinaljunk egy uj valtozot. Ebbe a kodba beleepitettem a **factor()** funkciot is, hogy azonnal meghatarozzuk, hogy ez az uj valtozo egy faktor, es nem egy egyszeru karaktervektor. A factor() funkcio nelkul is lefut a kod, de akkor meg kellene egy kulon sor ahol megadjuk hogy ez egy faktorvaltozo.

```
COVID_adat = COVID_adat %>%
  mutate(gdp_per_capita_kat = factor(
                                     case_when(gdp_per_capita < 5000 ~
"small",
                                               gdp_per_capita >= 5000 &
gdp_per_capita < 10000 ~ "medium",
                                               gdp_per_capita > 10000 ~
"large")))
levels(COVID_adat$gdp_per_capita_kat)

## [1] "large"  "medium" "small"

# ugyanez a COVID_adat_tegnap -al

COVID_adat_tegnap = COVID_adat_tegnap %>%
  mutate(gdp_per_capita_kat = factor(
                                     case_when(gdp_per_capita < 5000 ~
"small",
                                               gdp_per_capita >= 5000 &
gdp_per_capita < 10000 ~ "medium",
                                               gdp_per_capita > 10000 ~
"large")))
```

Kategorikus valtozo ujrakodolasa

Hasonlo eset ha kategorikus valtozokat szeretnenk atkodolni. Mondjuk ha szeretnenk a deli felteket az eszaki feltekevel osszehasonlitani. Ezt a **recode()** funkcioval lehet megoldani.

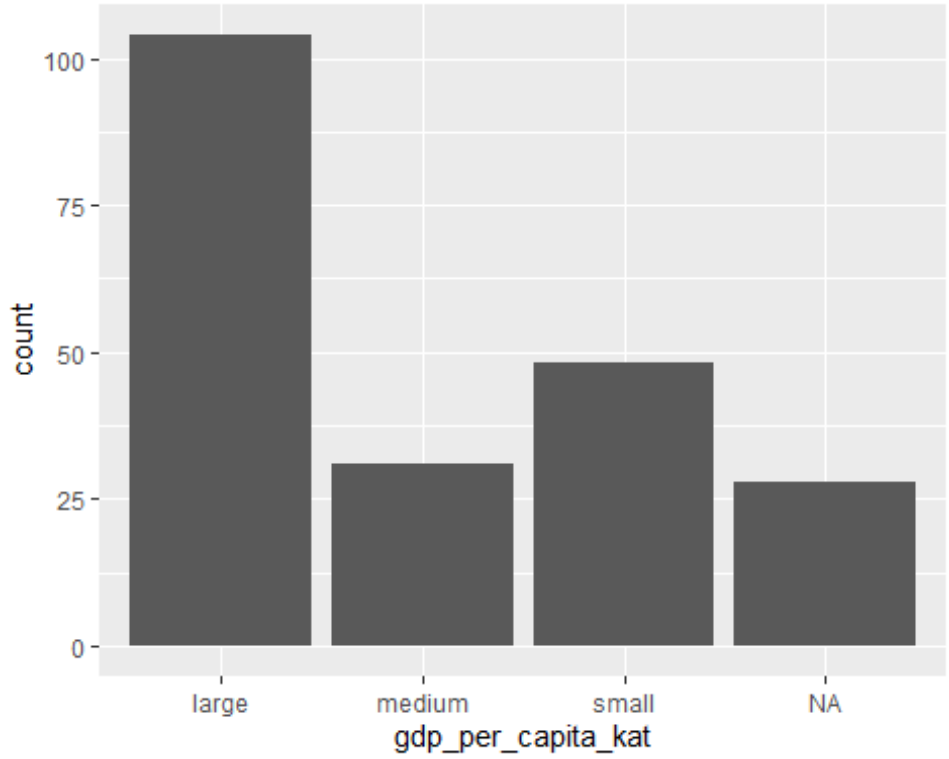
```
COVID_adat = COVID_adat %>%
  mutate(continent_south_north = factor(recode(continent,
                                               "Oceania" = "South",
                                               "South America" = "South",
                                               "Africa" = "South",
                                               "Asia" = "North",
                                               "Europe" = "North",
                                               "North America" = "North")))

levels(COVID_adat$continent_south_north)

## [1] "South" "North"

COVID_adat_tegnap = COVID_adat_tegnap %>%
  mutate(continent_south_north = factor(recode(continent,
```

```
                                        "Oceania" = "South",
                                        "South America" = "South",
                                        "Africa" = "South",
                                        "Asia" = "North",
                                        "Europe" = "North",
                                        "North America" = "North")))
```

Faktorszintek sorrendje, ordinalis valtozok

Amikor van ertelme a **sorrendisegnek** a faktorszintek kozott, **ordinalis valtozokrol** beszelunk (vagyis az egyik faktorszint alacsonyabb, vagy kisebb "erteku" mint a masik). Arra figyelnunk kell, hogy amikor faktorokat hozunk letre, az R automatikusan a faktorszintek neveinek **ABC sorrendje** alapjan rakja oket sorba, es az abrakon is igy szemlelteti majd oket.

```
COVID_adat_tegnap %>%
  ggplot() +
  aes(x = gdp_per_capita_kat) +
  geom_bar()
```



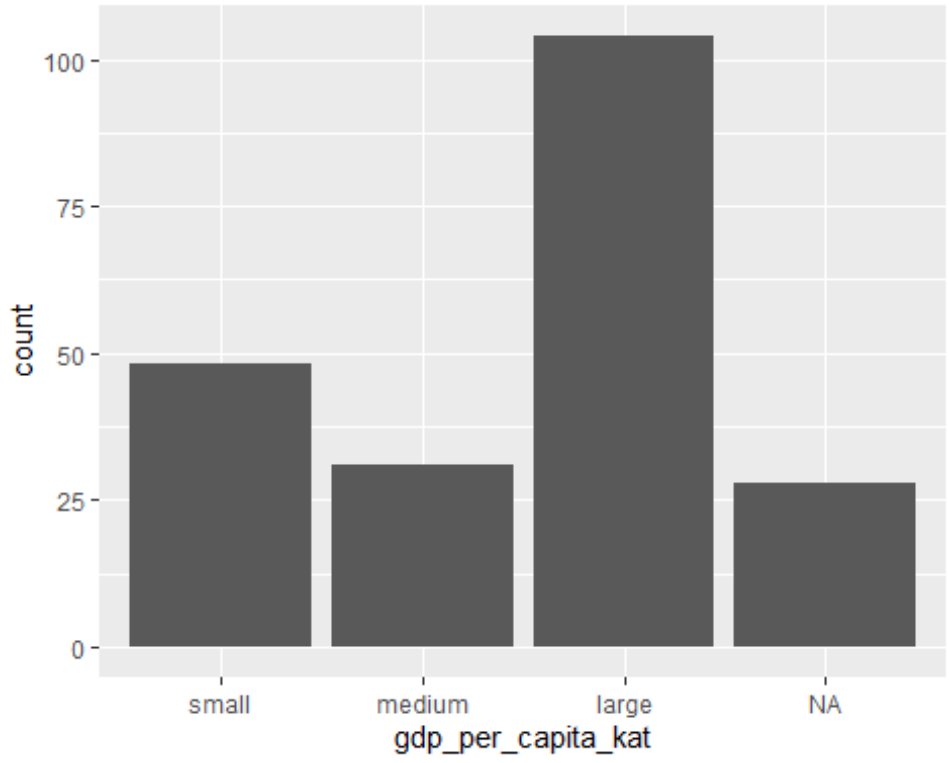
Ilyenkor erdemes meghatarozni a faktorszintek sorrendjet (**order**). Ezt legegyszerubben a factor() funkcion belul tehetjuk meg, az **ordered = T** beallitasaval, es a **levels =** resznel a szintek sorrendjenek meghatarozasaval.

```
COVID_adat_tegnap = COVID_adat_tegnap %>%
mutate(gdp_per_capita_kat = factor(gdp_per_capita_kat, ordered = T, levels =
c(
```

```
                                                "small",
                                                 "medium",
                                                 "large")))
```

Igy mar az R minden funkcioja tudni fogja, hogy egy ordinalis valtozorol van szo, ahol fontos a sorrend, es tudni fogja a sorrendet is.

```
COVID_adat_tegnap %>%
  ggplot() +
  aes(x = gdp_per_capita_kat) +
  geom_bar()
```



Gyakorlas

- szurd az adatokat ugy hogy csak a 2020-09-28-ai adatokkal dolgozzunk csak.
- csinalj egy uj kategorikus valtozot (nevezzuk ezt *new_cases_per_million_kat*-nak) a mutate() funkcio hasznalataval amiben azok az orszagok ahol a *new_cases_per_million* valtozo 20 alatt van “small”, ahol 20 vagy a felett van “large” kategoriaba keruljenek.
- figyelj oda hogy faktorkent jelold meg ezt az uj valtozot (Ezt lehet az elozo lepesben a mutate() funkcion belul, vagy egy kulon lepesben, de mindenkeppen a factor() vagy az as.factor() funkciokat erdemes hozza hasznalni)
- mentsd el ezt a valtozot az eredeti adatobjektumban ugy hogy kesobb is lehessen vele dolgozni

- keszits egy tablazatot arrol, hogy hanyan esnek a *new_cases_per_million_kat* egyes kategoriaiba.
- Add meg a faktorszintek helyes sorrendjet: small, large (Ird felul a *new_cases_per_million_kat* korabbi valtozatat ezzel a valtozattal ahol a szintek mar helyes sorrendben vannak, vagy ezt a sorrendezest is bele vonhatod az eredeti funkcioba, amivel a valtozot generaltad)
- Ellenorizd, hogy valoban helyes sorrendben szerepelnek-e a faktor szintjei.

Exploracio vizualizacion keresztul

Egyes valtozok vizualizacioja

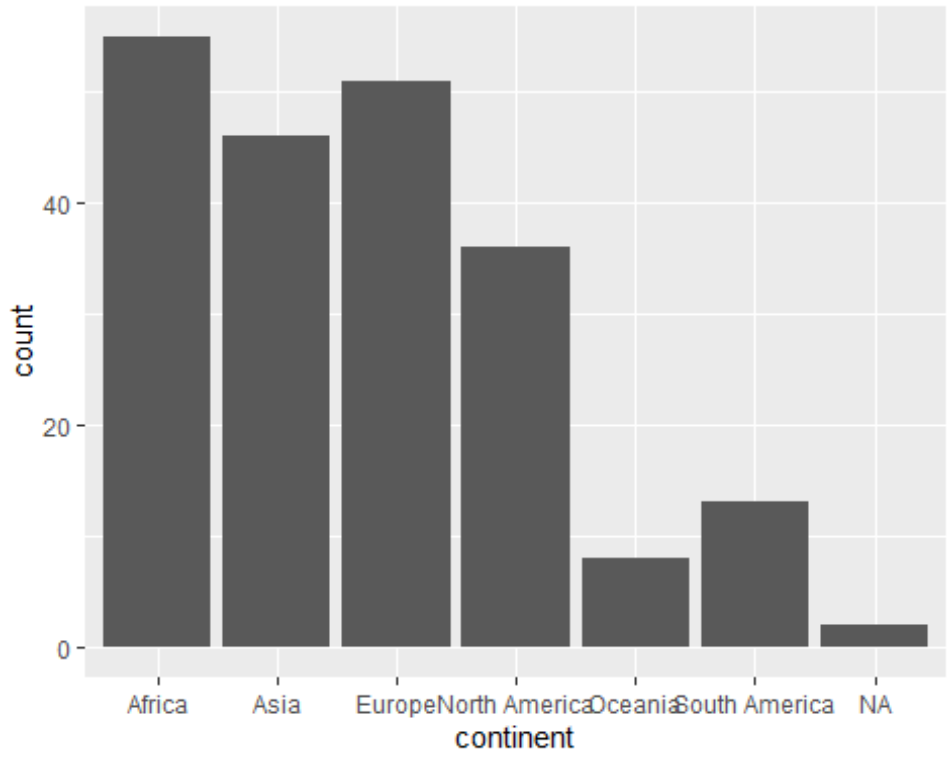
Az egyes valtozok **abrak** (plot) segitsegevel is megvizsgalhatok. A **kategorikus** valtozokat gyakran oszlopdiagrammal (**geom_bar**) abrazoljuk,

Mig a **numerikus** valtozokat inkabb **dotplot** , **histogram**, vagy **density plot** segitsegevel szoktuk abrazolni.

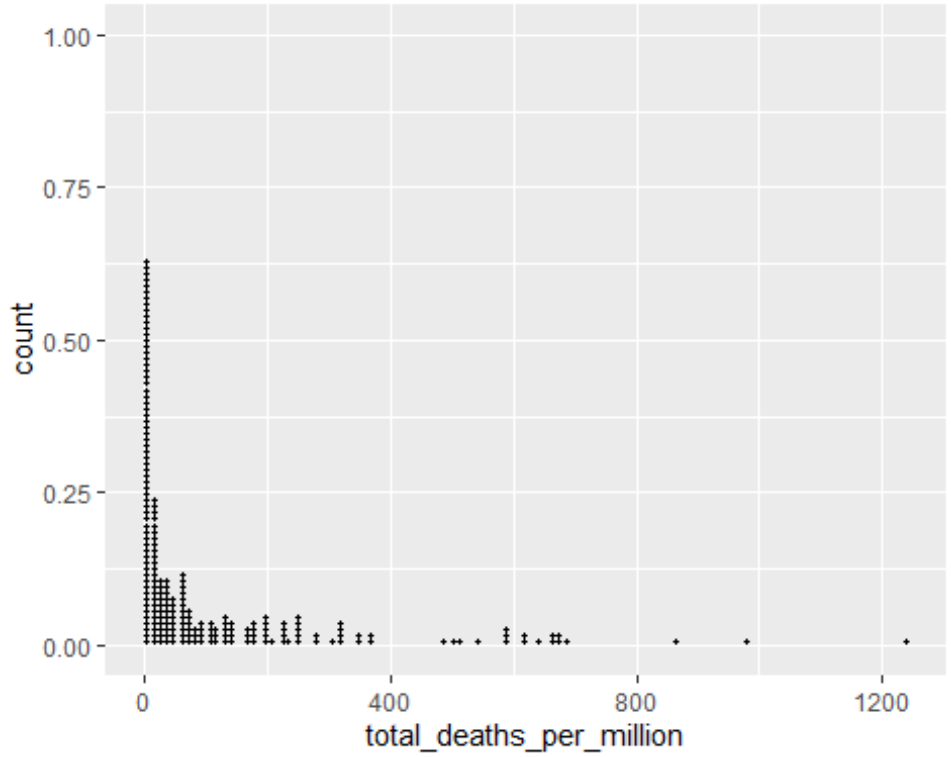
Az egyes valtozok vizualizacioja es a leiro statisztikak atvizsgalasa elengedhetetlen hogy azonositsuk az esetleges adatbeviteli **hibakat es egyeb nemvart furcsasagokat** az adataink kozott.

MINDING ellenorizd az adataidat ezekkel a modszerekkel mielott komolyabb adatelemzesbe kezdesz, hogy meggyozodj rola, hogy az adatok tisztak es megfelenek az elvarasaidnak.

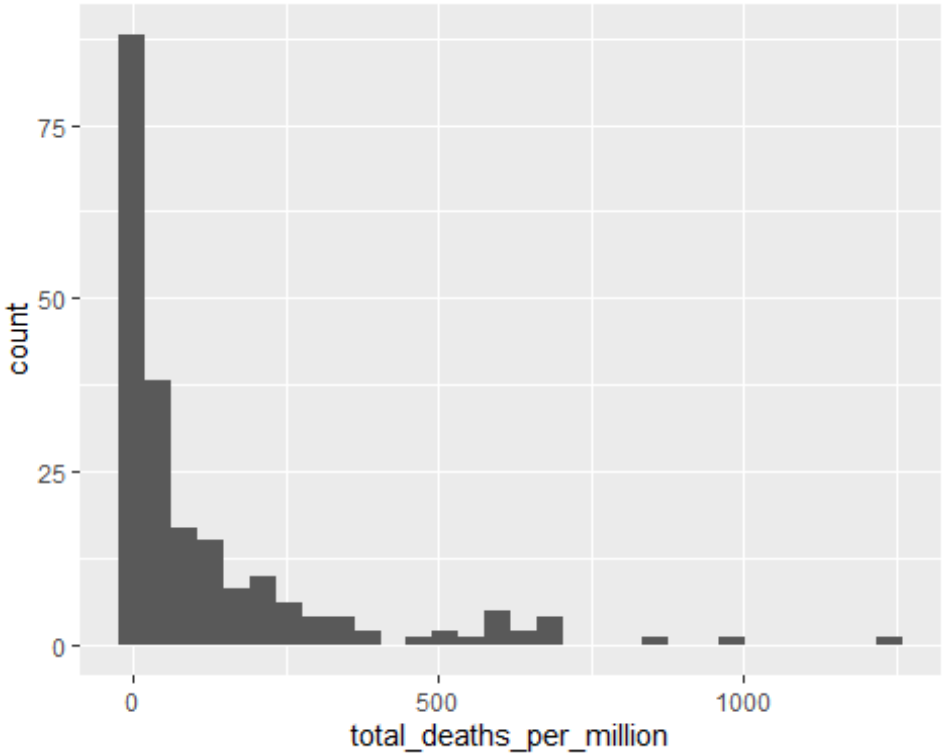
```
COVID_adat_tegnap %>%
ggplot() +
  aes(x = continent) +
  geom_bar()
```



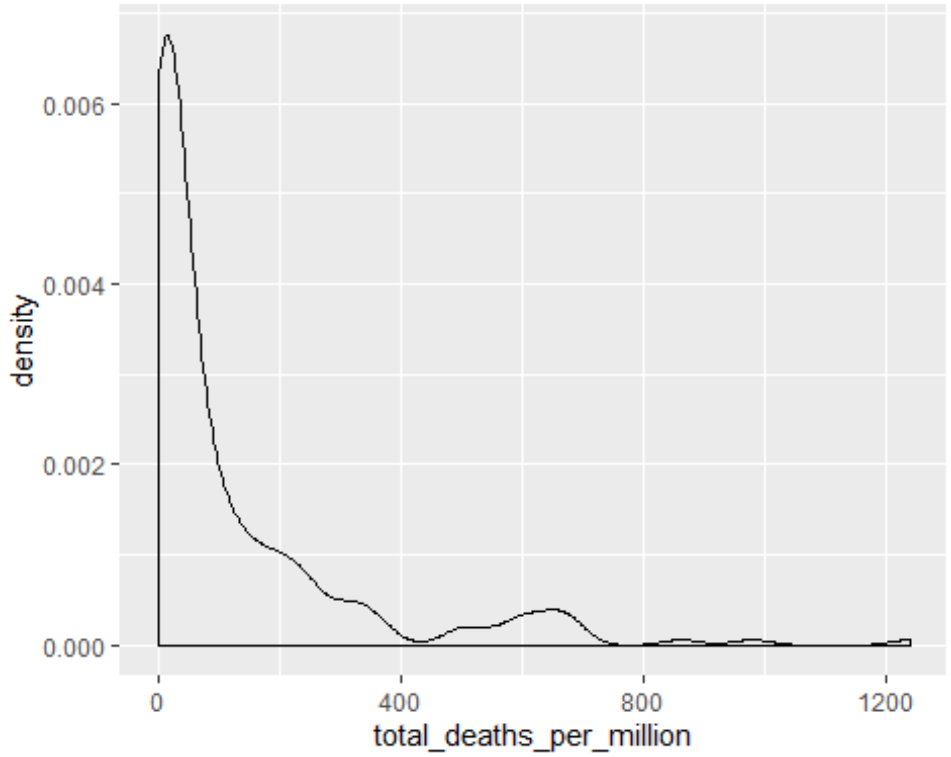

```
COVID_adat_tegnap %>%
ggplot() +
  aes(x = total_deaths_per_million) +
  geom_dotplot(binwidth = 10)

## Warning: Removed 1 rows containing non-finite values (stat_bindot).
```




```
COVID_adat_tegnap %>%
ggplot() +
  aes(x = total_deaths_per_million) +
  geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 1 rows containing non-finite values (stat_bin).
```




```
COVID_adat_tegnap %>%
ggplot() +
  aes(x = total_deaths_per_million) +
  geom_density()

## Warning: Removed 1 rows containing non-finite values (stat_density).
```



Gyakorlas

Szurd az adatokat ugy hogy csak a 2020-09-07-en jeletett adatokkal dolgozzunk

Hasznald a fent tanult modszereket, hogy **azonositsd az COVID_adat adattablaban levo hibakat** vagy nem vart furcsasagokat.

- A vizualizacion tul a View(), describe(), es summary() funciokat erdemes hasznalni az adatok elso attekintesere
- A numerikus (vagy eppen folytonos) valtozoknal vizsgald meg a minimum es maximum erteket es a hianyzo adatok mennyiseget, valamint az eloszlast.
- A kategorikus valtozoknal vizsgald meg az osszes faktorszintet es az egyes szintekhez tartozo megfigyelesek mennyiseget.

A hibakat a kovetkezokeppen javithatjuk.

A **mutate()** es a **replace()** funkciok hasznalataval **cserelhetunk ki** ertekeket mas ertekekre. Azt, hogy ilyenkor hianyzo adatra (NA), vagy egy masik, valoszinu ertekre kell megvaltoztatni az erteket, a szituaciotol fogg. Altalaban a biztosabb megoldas ha hianyzo adatnak jeloljuk a kerdeses erteket (NA), de ez sok adatveszteshez vezethet. Ha eleg valoszinu hogy mi a helyes valasz, beirhatjuk, DE **minden javitast fel kell tuntetni** a

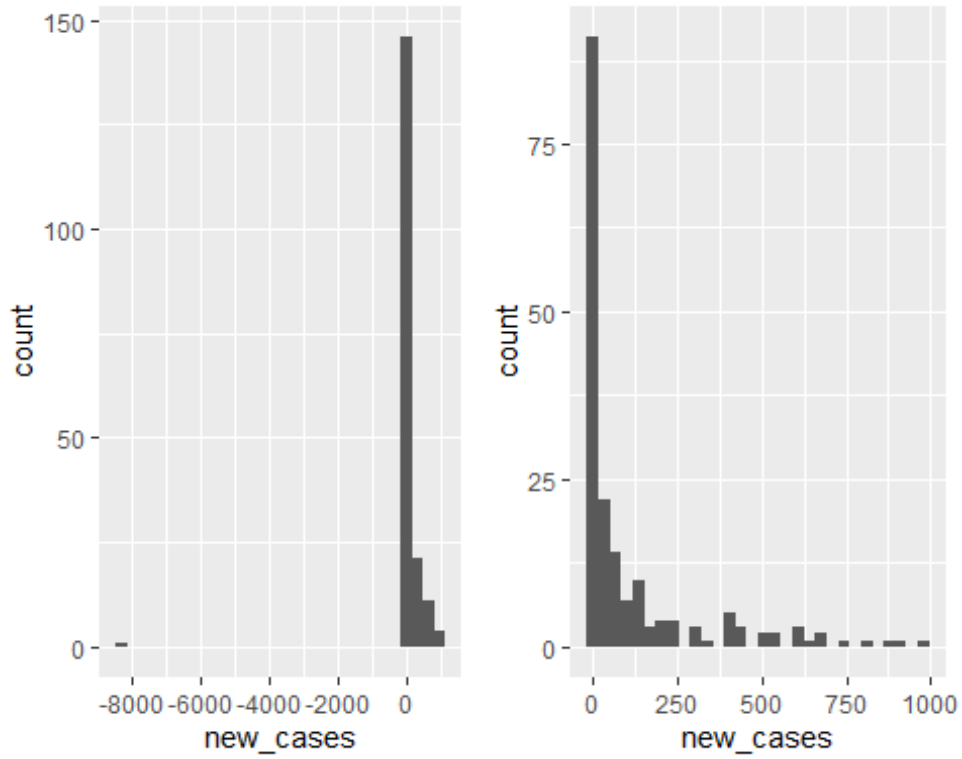
kutatasi jelentesben (es a ZH soran is), hogy az olvaso szamara tiszta legyen, hogy itt egy adathelyettesites vagy kizaras tortent!

Mindig erdemes a javitott adatokat **uj adattablaba** elmenteni. A mi esetunkben az COVID_adat_corrected nevet adtuk a javitott objektumnak. Igy a nyers adataink megmaradnak, ami hasznos lehet kesobbi muveleteknel.

```
COVID_adat_corrected <- COVID_adat %>%
  mutate(new_cases = replace(new_cases,  new_cases=="-8261", NA))
```

Erdemes **megbizonyosodni rola**, hogy az adatcsere sikeres volt. Alabb az adatok vizualizaciojaval gyozodunk meg errol, de az adatok megjelenitesevel, vagy a leiro statisztikak lekerdezesevel is megteheto ez, ha az informativ.

```
# hasznalhatnak meg az alabbiakat is arra,
# hogy megbizonyosodjunk abban, hogy sikeres volt a csere
# View(COVID_adat_corrected)
# describe(COVID_adat_corrected)
# summary(COVID_adat_corrected$szocmedia_3)
# COVID_adat_corrected$szocmedia_3

old_plot <-
  COVID_adat %>%
  filter(date == "2020-09-07", new_cases < 1000) %>%
  ggplot()+
    aes(x = new_cases)+
    geom_histogram()

new_plot <-
  COVID_adat_corrected %>%
  filter(date == "2020-09-07", new_cases < 1000) %>%
  ggplot()+
    aes(x = new_cases)+
    geom_histogram()


grid.arrange(old_plot, new_plot, ncol=2)
```



Tobb valtozo kapcsolatanak felterkepezese

Tobb valtozo kapcsolatat is felterkepezhetjuk tablazatok es abrak segitsegevel.

Ket kategorikus (csoportosito) valtozo kapcsolatanak felterkepezese

Feltaro elemzes

Most vizsgaljuk meg azt, hogy 2020-09-28-an mi az osszefuggese a gdp kategorianak (*gdp_per_capita_kat*) a kontinenssel (*continent*) ahol az orszag elhelyezkedik.

A legegyszerubb modja ket csoportosito valtozo kapcsolatanak megvizsgalasara a ket valtozo **kereszt-tablazatanak (crosstab)** elkezsitese a **table()** funkcioval.

```
table(COVID_adat_tegnap$gdp_per_capita_kat, COVID_adat_tegnap$continent)
```

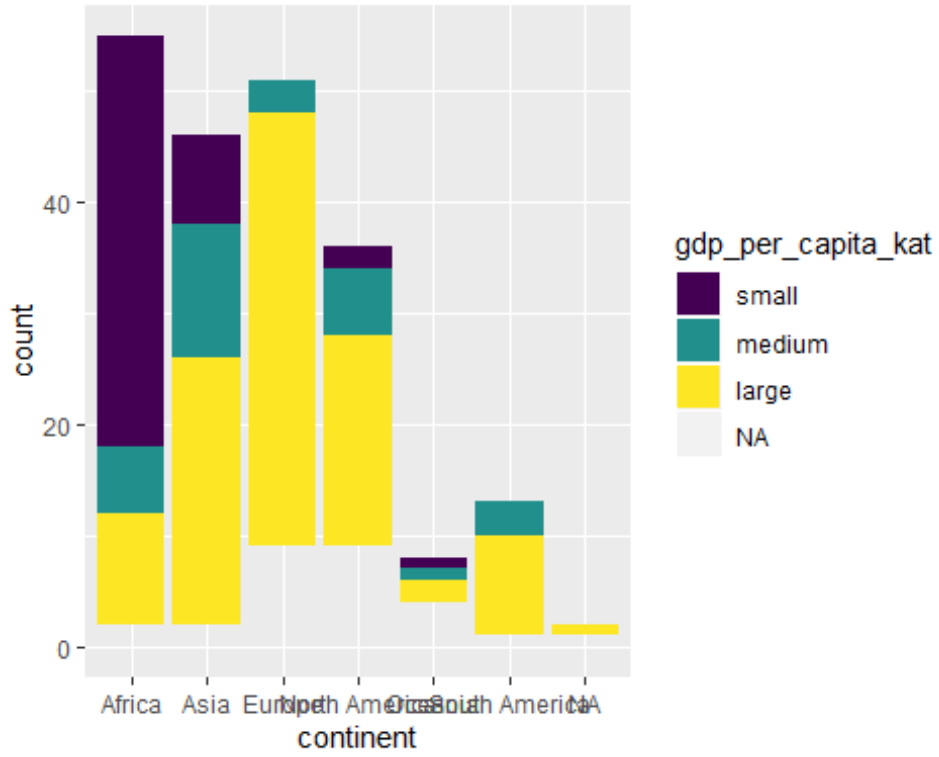
```
##
##          Africa Asia Europe North America Oceania South America
##   small      37    8      0             2       1             0
##   medium      6   12      3             6       1             3
##   large      10   24     39            19       2             9
```

Sokszor ennel sokkal **szemleletesebb az abrak** (plot) hasznalata.

Erre az egyik lehetoseg a **stacked bar chart** (egymasra tornyozott oszlopdiagram, a **geom_bar()** geomot hasznaljuk) hasznalata. Itt az egyik valtozo kategoriai adjak meg hany oszlop lesz (ez a valtozo lesz az x tengelyen reprezentalva, igy ezt az “x =” reszen adhatjuk

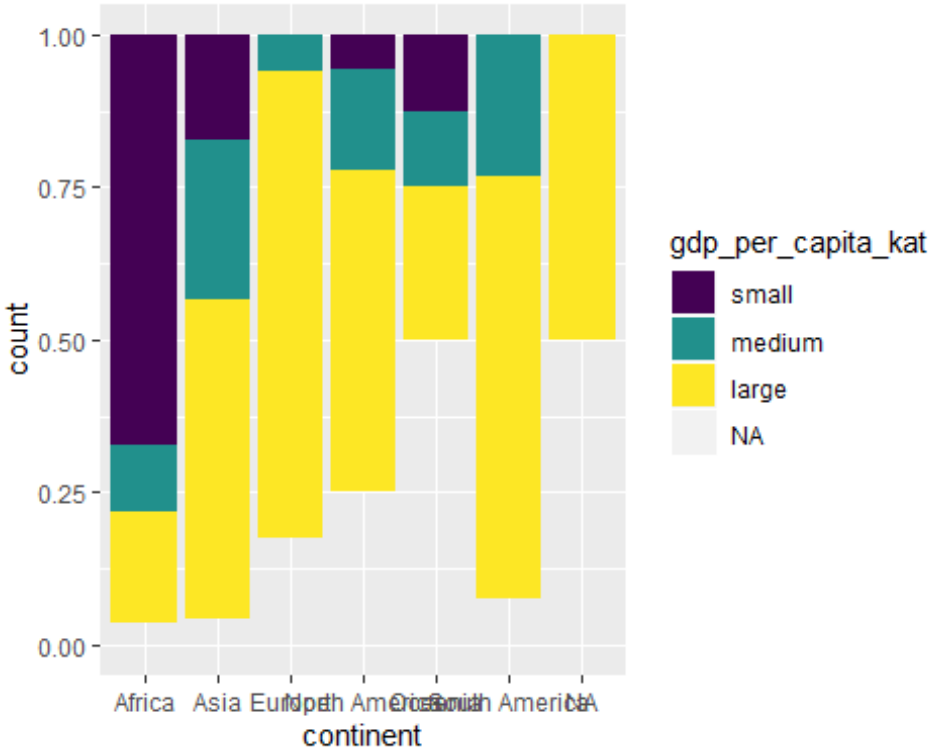
meg), a masik valtozo az oszlopokat szinekkel szegmentalja, ezt pedig a "**fill =**" reszen adhatjuk meg.

```
COVID_adat_tegnap %>%
ggplot() +
  aes(x = continent, fill = gdp_per_capita_kat) +
  geom_bar()
```



Ha az egyes faktorszinteken nagyon **kulonbozo mennyisegu megfigyeles** van, ez a megjelenites neha felrevezeto kovetkeztetesekhez vezethet, igy neha hasznosabb ha az oszlopok nem szamossagot (count), hanem **reszaranyt (proportion)** jelolnek. Ha ezt szeretnenk, ahelyett hogy uresen hagynank a geom_bar() funkciot, a kovetkezot adjuk meg: **geom_bar(position = "fill")**.

```
COVID_adat_tegnap %>%
ggplot() +
  aes(x = continent, fill = gdp_per_capita_kat) +
  geom_bar(position = "fill")
```



Gyakorlas

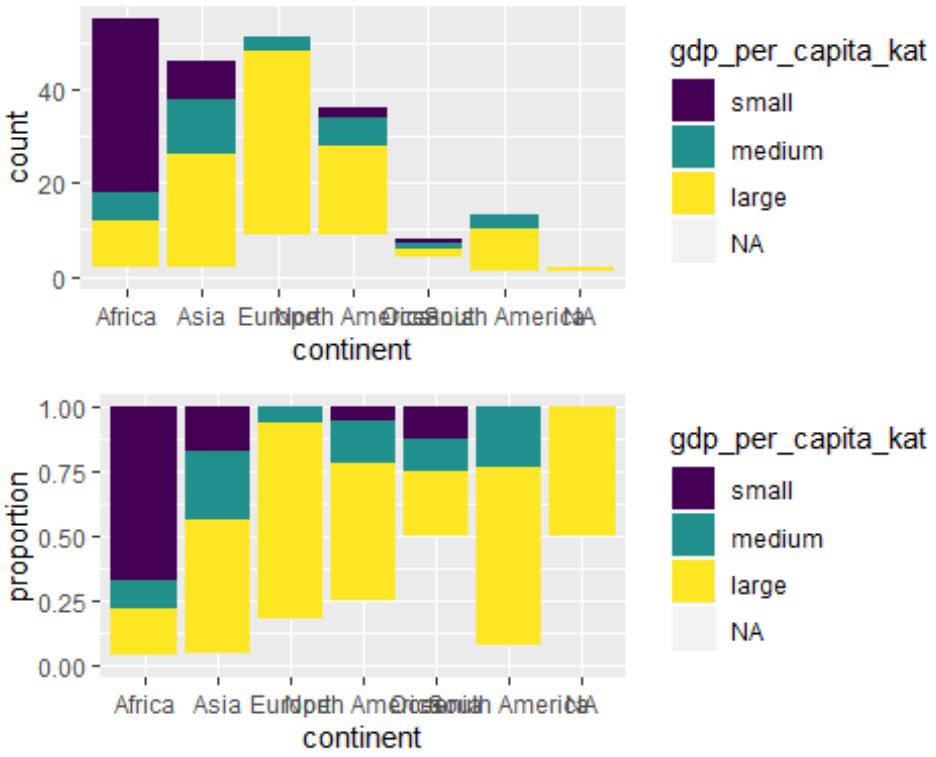
Hasznald a fent tanult modszereket, hogy megvizsgald a COVID_adat_tegnap adatbazisban a **new_cases_per_million_kat** es a **continent** valtozok kozotti osszefuggest. - hasznalj **geom_bar()** geomot a megjeleniteshez - probald meg mind a **szamossagot**, mind a **reszaranyt** kifejezo abrat megvizsgalni geom_bar(position = “fill”) - milyen **kovetkeztetest** tudsz levonni az abrakrol?

```
# a fenti gyakorlashoz a new_cases_per_million_kat valtozot igy lehet legeneralni:

COVID_adat = COVID_adat %>%
  mutate(new_cases_per_million_kat = factor(
                                      case_when(new_cases_per_million < 20 ~ "small",
                                                new_cases_per_million >= 20 ~ "large"), ordered = T, levels = c("small", "large")))

levels(COVID_adat$new_cases_per_million_kat)

## [1] "small" "large"
```

```r
# ugyanez a COVID_adat_tegnap -al

COVID_adat_tegnap = COVID_adat_tegnap %>%
  mutate(new_cases_per_million_kat = factor(
                                      case_when(new_cases_per_million < 20 ~
"small",
                                                new_cases_per_million >= 20 ~
"large"), ordered = T, levels = c("small", "large")))
```

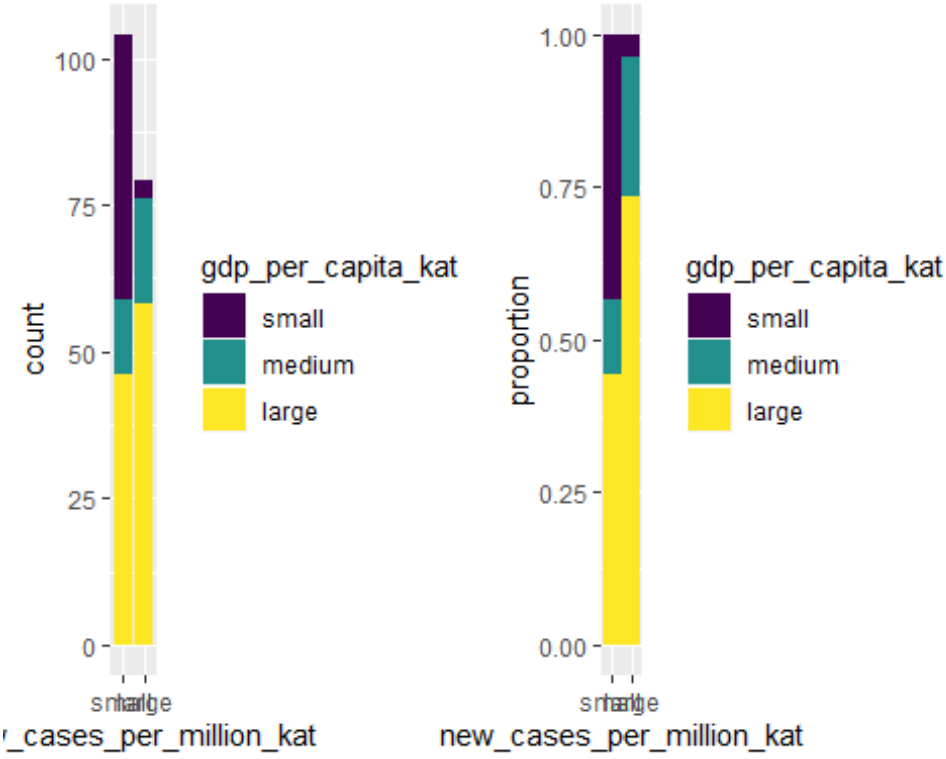
geom_bar() megjelenitesnel fontos hogy ha az egyes megfigyelesek **keves megfigyelesbol allnak**, az abra megteveszto lehet, mert az abra nem jelzi a megfigyelesek szamat es igy azt, hogy milyen biztosak lehetunk az eredmenyben. Ilyen esetekben az egyik kategoriat ki lehet venni az abrarol, vagy a **szamossagot es a reszaranyt abrazolo abrakat egymas mellet** lehet bemutatni, hogy igy kiegeszitsek egymast. Ehhez hasznalhatjuk a **grid.arrange()** funkciot.

```r
szamossag_plot <-
COVID_adat_tegnap %>%
ggplot() +
  aes(x = continent, fill = gdp_per_capita_kat) +
  geom_bar()

reszarany_plot <-
COVID_adat_tegnap %>%
ggplot() +
  aes(x = continent, fill = gdp_per_capita_kat) +
  geom_bar(position = "fill") +
  ylab("proportion")

grid.arrange(szamossag_plot, reszarany_plot, nrow=2)
```



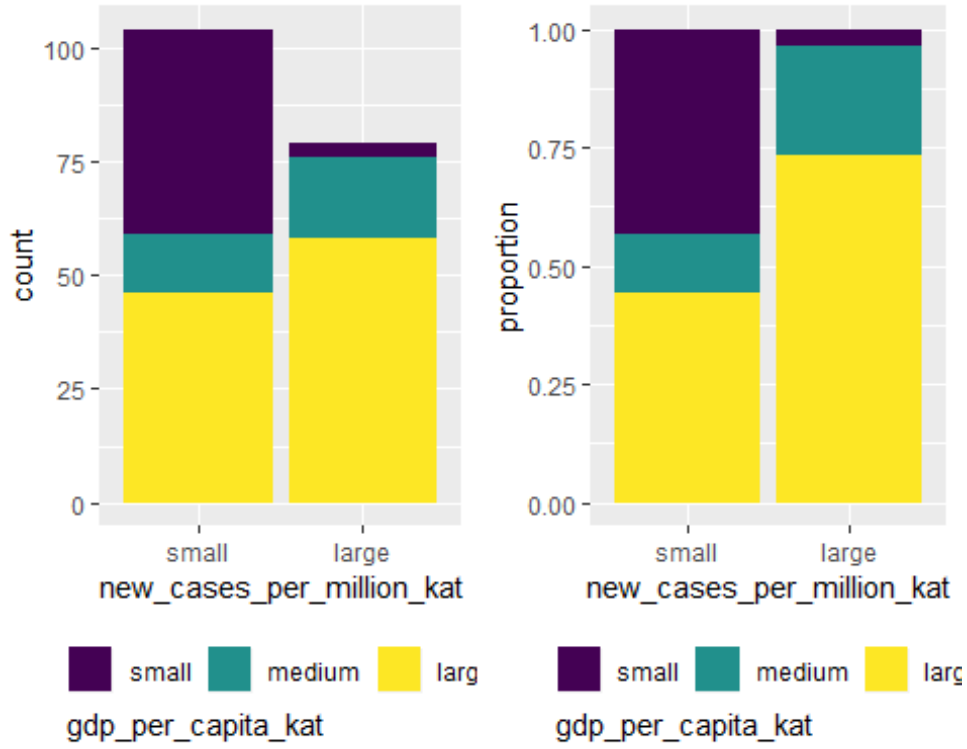
A theme(legend.position) es a guides() funciok hasznalataval kontrollalhatjuk hogy hol es hogyan jelenjen meg a **jelmagyarazat** az abran. Az abra **interpretalhatosaga** attol fuggoen is **valtozhat**, hogy melyik valtozot tesszuk az x-tengelyre es melyiket szinkent abrazolva. Az alabbi abrakon az egymillio fore vetitett uj esetek szamanak kapcsolatat nezzuk meg a gdp-vel. Mindket valtozo eseten a csoportositott valtozot (_kat) hasznaljuk.

```
barchart_plot_3 <-
COVID_adat_tegnap %>%
  select(new_cases_per_million_kat, gdp_per_capita_kat) %>%
  drop_na() %>%
ggplot() +
  aes(x = new_cases_per_million_kat, fill = gdp_per_capita_kat) +
  geom_bar()


barchart_plot_4 <-
COVID_adat_tegnap %>%
  select(new_cases_per_million_kat, gdp_per_capita_kat) %>%
  drop_na() %>%
ggplot() +
  aes(x = new_cases_per_million_kat, fill = gdp_per_capita_kat) +
  geom_bar(position = "fill") +
  ylab("proportion")

grid.arrange(barchart_plot_3, barchart_plot_4, ncol=2)
```



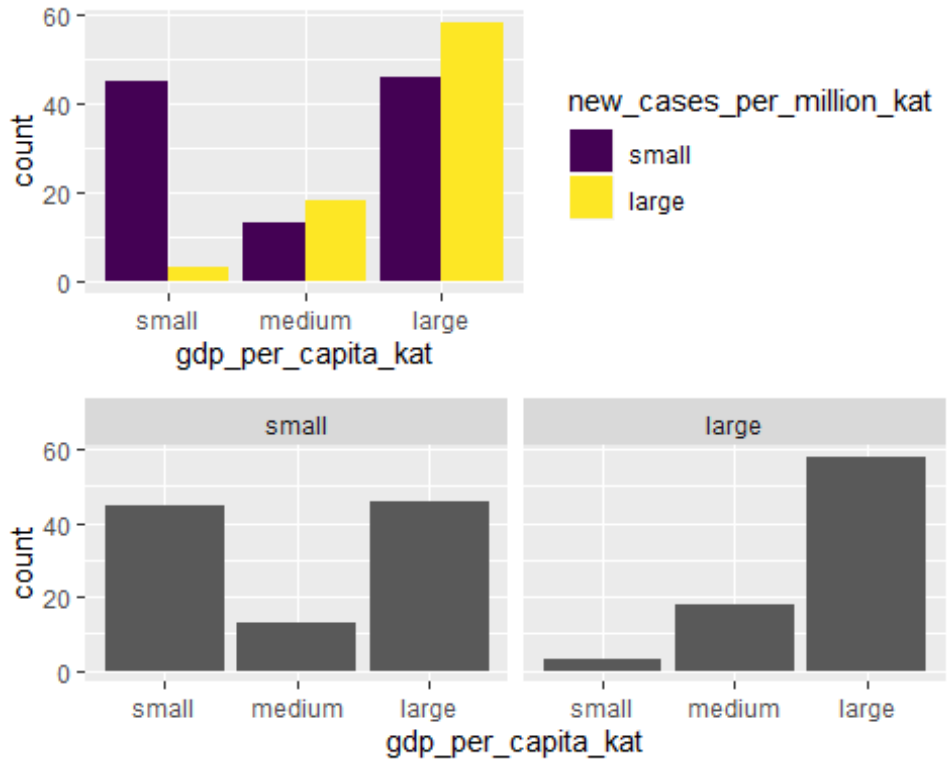

```r
# a theme(legend.position) es a guides() funciok
# hasznalataval kontrollalhatjuk hogy hol es hogyan
# jelenjen meg a jelmagyarazat az abran

barchart_plot_3 <-
COVID_adat_tegnap %>%
  select(new_cases_per_million_kat, gdp_per_capita_kat) %>%
  drop_na() %>%
ggplot() +
    aes(x = new_cases_per_million_kat, fill = gdp_per_capita_kat) +
    geom_bar() +
    theme(legend.position="bottom") +
    guides(fill = guide_legend(title.position = "bottom"))


barchart_plot_4 <-
COVID_adat_tegnap %>%
  select(new_cases_per_million_kat, gdp_per_capita_kat) %>%
  drop_na() %>%
ggplot() +
  aes(x = new_cases_per_million_kat, fill = gdp_per_capita_kat) +
  geom_bar(position = "fill") +
  theme(legend.position="bottom") +
  guides(fill = guide_legend(title.position = "bottom")) +
  ylab("proportion")
```

```
grid.arrange(barchart_plot_3, barchart_plot_4, ncol=2)
```



Ujabb modja a barchart segitsegevel valo megjelenitesnek ha az oszlopok nem egymasra tornyozva, hanem **egymas mellett** jelennek meg, vagy ha a masodik valtozo szerint **kulon paneleken (facet)** jelennek meg.

```
barchart_plot_5 <-
COVID_adat_tegnap %>%
  select(new_cases_per_million_kat, gdp_per_capita_kat) %>%
  drop_na() %>%
ggplot() +
  aes(x = gdp_per_capita_kat, fill = new_cases_per_million_kat) +
  geom_bar(position = "dodge")

barchart_plot_6 <-
COVID_adat_tegnap %>%
  select(new_cases_per_million_kat, gdp_per_capita_kat) %>%
  drop_na() %>%
ggplot() +
  aes(x = gdp_per_capita_kat) +
  geom_bar() +
  facet_wrap(~ new_cases_per_million_kat)

grid.arrange(barchart_plot_5, barchart_plot_6, nrow=2)
```



Egy kategorikus es egy numerikus valtozo kapcsolata

Vizsgaljuk meg hogy hogyan alakul az egy fore juto GDP kontinensenkent. A GDP ebben az esetben egy folytonos valtozó (gdp_per_capita), es ennek az osszefuggeset szeretnenk megvizsgalni egy kategorikus valtozoval (continent).

Az exploraciot kezdhetjuk leiro statisztikak lekerdezesevel csoportonkent. Peldaul ha arra vagyunk kivancsiak, milyen a GDP atlaga es szorasa kontinensenkent, ezt megvizsgalhatjuk a **group_by()** es a **summarize()** segitsegevel.

```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
  drop_na() %>%
  group_by(continent) %>%
    summarize(mean = mean(gdp_per_capita),
              sd = sd(gdp_per_capita))
```

```
## # A tibble: 6 x 3
##   continent       mean     sd
##   <fct>          <dbl>  <dbl>
## 1 Africa         5444.  6183.
## 2 Asia          22185. 25406.
## 3 Europe        33361. 18030.
## 4 North America 21655. 15404.
## 5 Oceania       23315. 20097.
## 6 South America 13841.  5110.
```

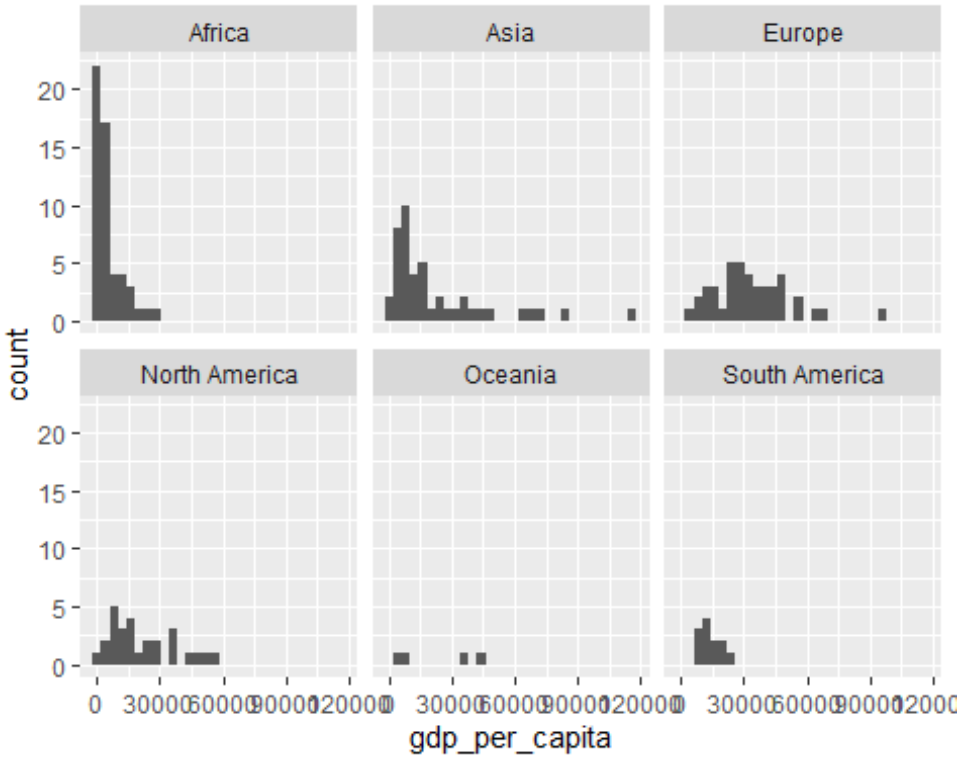
A ket valtozo kapcsolatat megvizsgalhatjuk **abrakkal** is. Pl. hasznalhatjuk a

- **facet_wrap()** fuggvenyt egy **geom_histogram()** vagy **geom_dotplot()** -al kobinalva
- a **geom_boxplot()** -ot
- esetleg hasznalhatunk egy egymasra illesztett **geom_density()** plot-ot.
- talan ebben az esetben a legtisztabb kepet a **geom_violin()** mutatja, ami a geom_boxplot() es a geom_density() keverekenek tekintheto. Ezt kiegeszithetunk egy **geom_point()** -al, hogy pontosan latsszon, hany megfigyelesen alapulnak az abra adatai.

Mindig erdemes **tobb megkozelitest** is hasznalni az adat-exploracio kozben, hogy minel reszletesebb kepet kaphassunk, es csokkentsuk a valoszinuseget hogy egyik vagy masik megkozelites hianyossagai felrevezetnek minket.

```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
  drop_na() %>%
  ggplot() +
    aes(x = gdp_per_capita) +
    geom_histogram() +
    facet_wrap(~ continent)
```

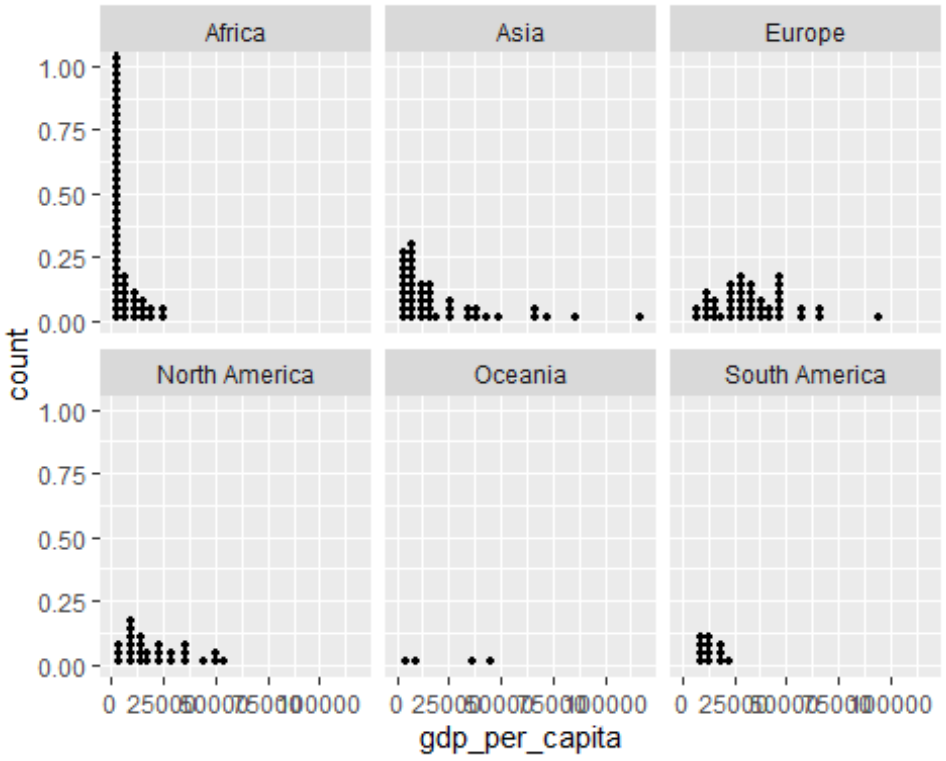
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



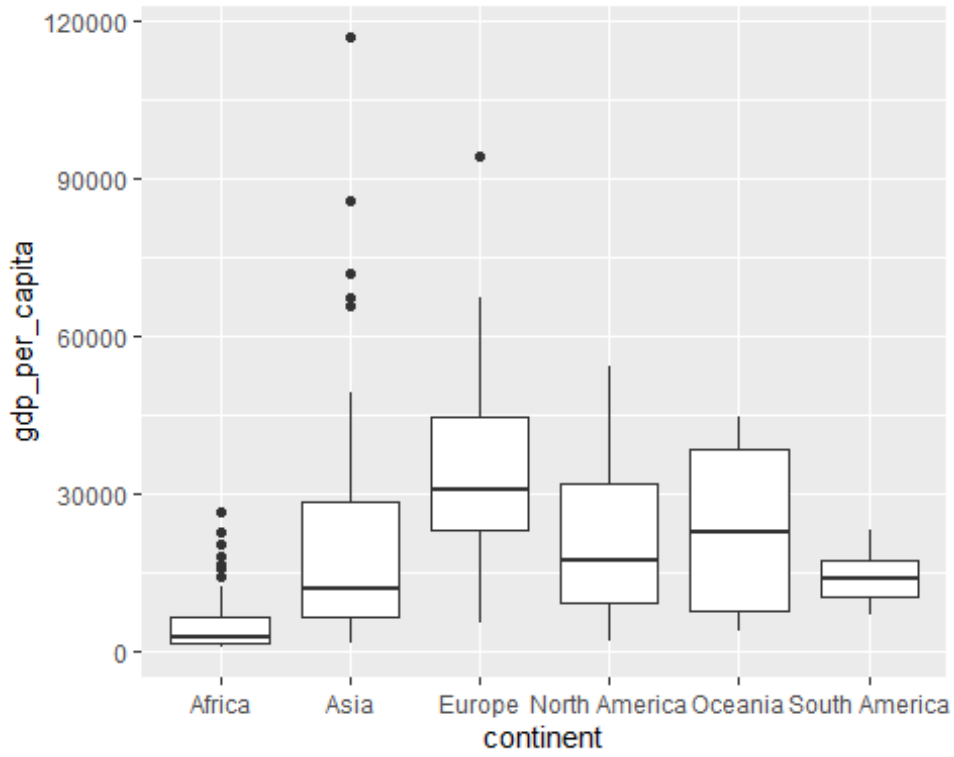

```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
```

```
  drop_na() %>%
  ggplot() +
    aes(x = gdp_per_capita) +
    geom_dotplot() +
    facet_wrap(~ continent)
```

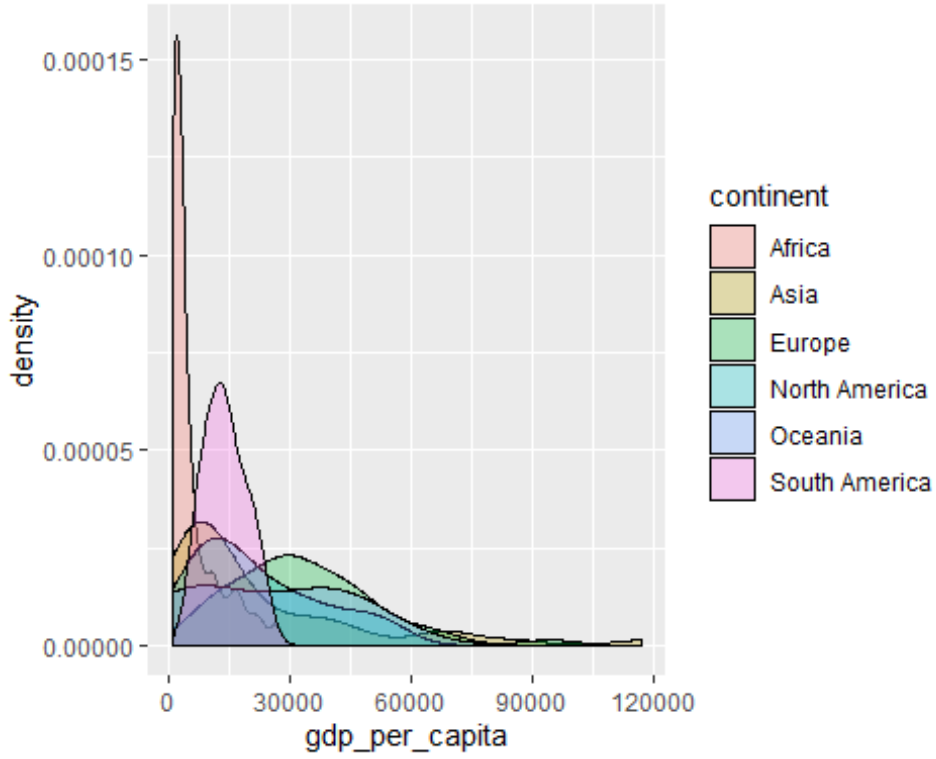
```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```



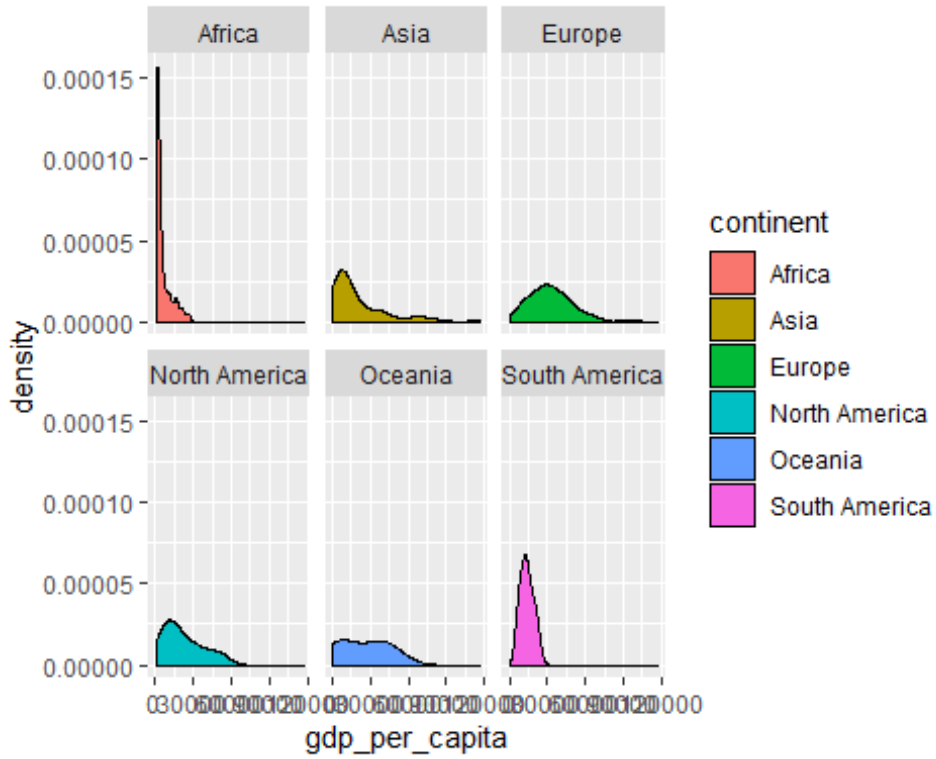

```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
  drop_na() %>%
  ggplot() +
    aes(x = continent, y = gdp_per_capita) +
    geom_boxplot()
```



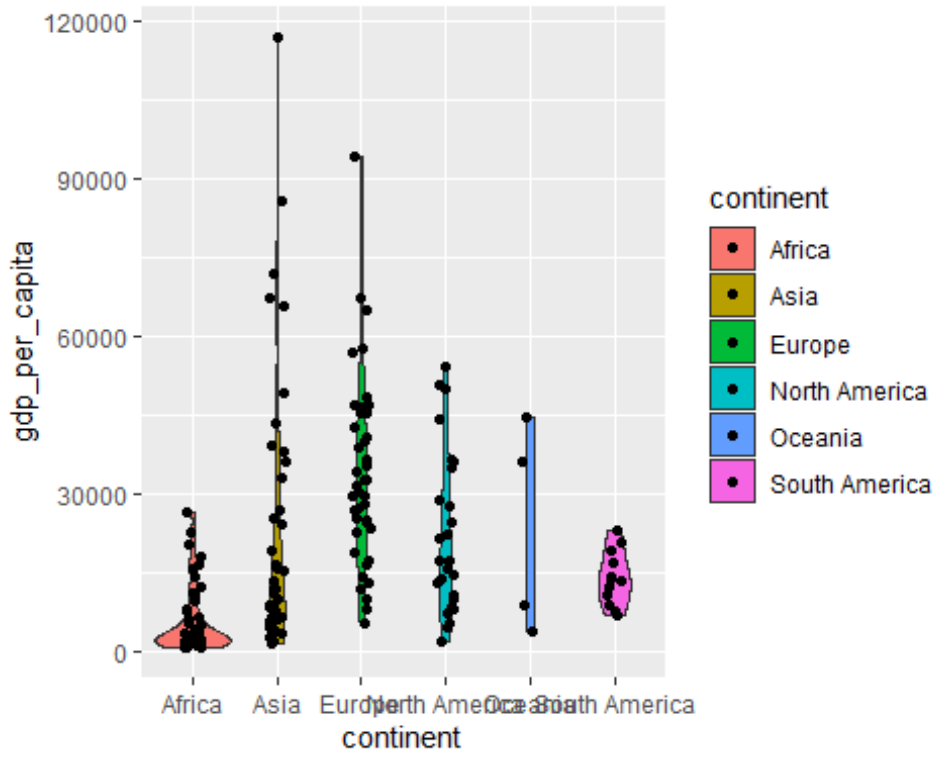

```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
  drop_na() %>%
  ggplot() +
    aes(x = gdp_per_capita, fill = continent) +
    geom_density(alpha = 0.3)
```




```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
  drop_na() %>%
  ggplot() +
    aes(x = gdp_per_capita, fill = continent) +
    geom_density()+
  facet_wrap(~continent)
```



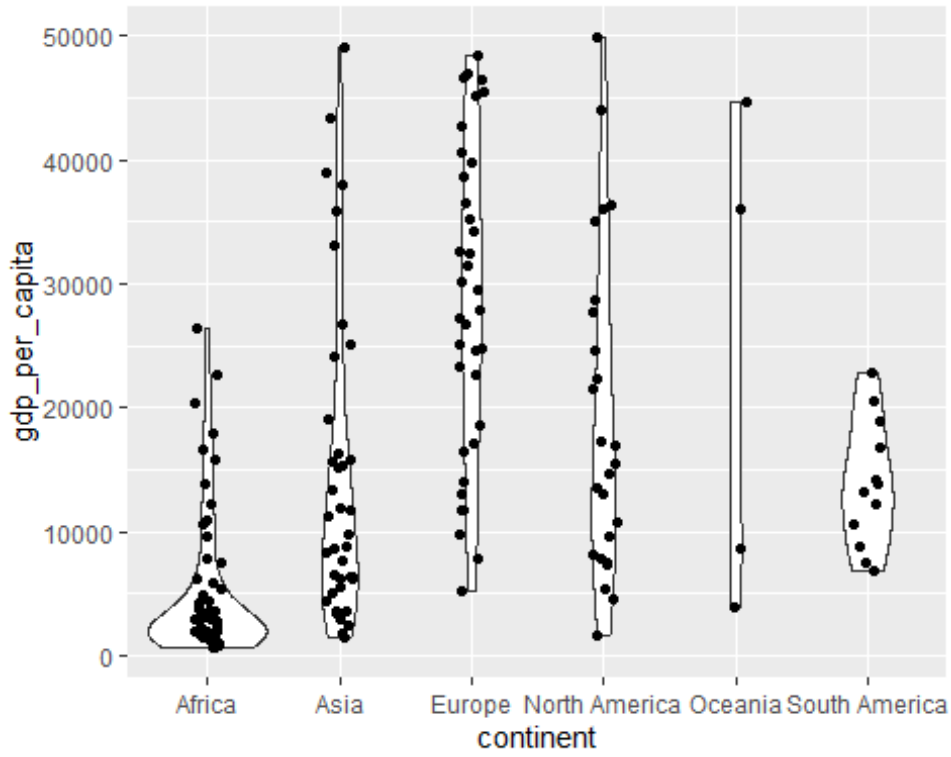

```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
  drop_na() %>%
  ggplot() +
    aes(x = continent, y = gdp_per_capita, fill = continent) +
    geom_violin() +
    geom_jitter(width = 0.1)
```



A fenti abran latszik, hogy Azsiaban a legtobb orszagban viszonylag alacsony a gdp, viszont van nehany **kiurgo ertek**, az atlagot felhuzza ebben a csoportban.

Ha szeretnenk **kizarni az elemzesunkbol** az extrem ertekekt, a **filter()** funkcio beekelesevel a pipe-ba megepithetjuk a fenti abrankat es tablazatokat ugy, hogy csak a 50000-nel alancsonyabb GDP-ju orszagok keruljenek az abrara.

```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
  drop_na() %>%
  filter(gdp_per_capita < 50000) %>%
    ggplot() +
      aes(x = continent, y = gdp_per_capita) +
      geom_violin() +
      geom_jitter(width = 0.1)
```




```
COVID_adat_tegnap %>%
  select(continent, gdp_per_capita) %>%
  drop_na() %>%
  filter(gdp_per_capita < 50000) %>%
    group_by(continent) %>%
      summarize(mean = mean(gdp_per_capita),
                sd = sd(gdp_per_capita))
```

```
## # A tibble: 6 x 3
##   continent       mean     sd
##   <fct>          <dbl>  <dbl>
## 1 Africa         5444.  6183.
## 2 Asia          14591. 12710.
## 3 Europe        28661. 12390.
## 4 North America 19192. 13095.
## 5 Oceania       23315. 20097.
## 6 South America 13841.  5110.
```

Ha szeretnenk latni hogy a kisebb vagy nagyobb uj esetszammal jellemezheto orszagok (new_cases_per_million_kat) hogyan kulonboznek a GDP tekinteteben kontinensenkent akkor mar **harom valtozo** kapcsolatat kell abrazolnunk. Ehhez a facet_grid() funkciot lehet hasznalni, vagy kulonbozo esztetikai elemeket (aes()) lehet a kulonbozo valtozokhoz rendelni. (Az alabbi peldanal csak Europara es Eszak Amerikara korlatoztuk az adatbazist, hogy az abrak szemleletesebbek legyenek.)

Gyakorlas

Hasznald a fent tanult modszereket, hogy megvizsgald a **total_cases_per_million** es a **gdp_per_capita_kat** valtozok kozotti osszefuggest.

- hasznald a fenti geomokat, es keszits legalabb ket kulonbozo abrat mas-mas geomokkal

---

Ket numerikus valtozo kapcsolata

Ket numerikus valtozo kozotti kapcsolat jellemzesere altalaban a korrelacios egyutthatot szoktuk hasznalni (cor()). A **cor()** funkciot akar tobb mint ket valtozo paronkenti korrelaciojanak meghatarozasara is lehet hasznalni.

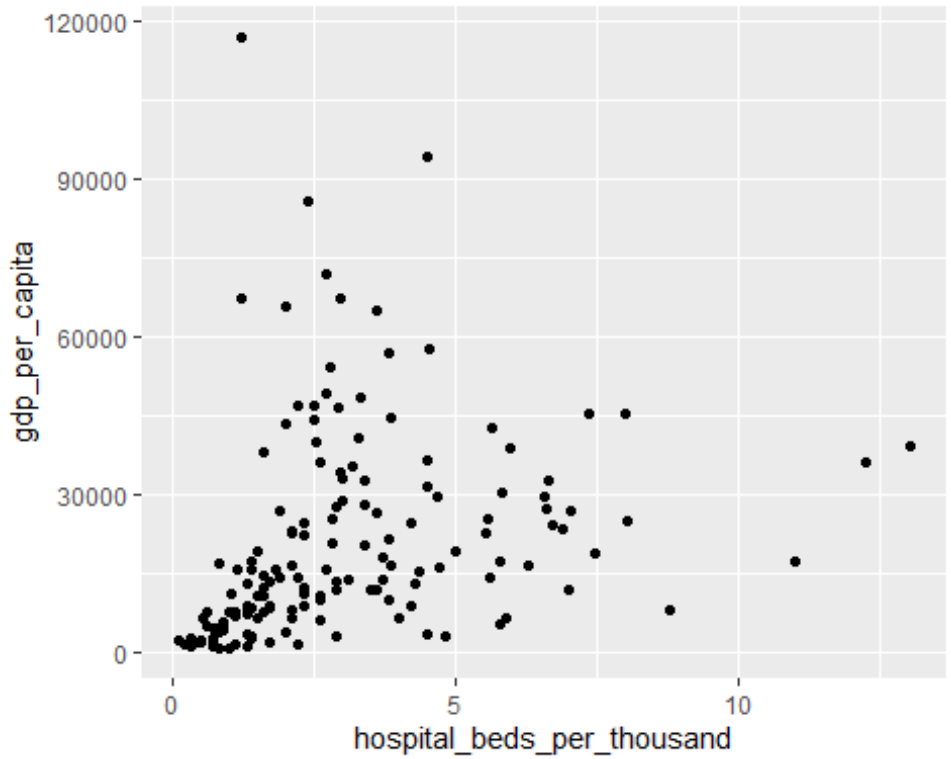
A **drop_na()** funkcioval kiejthetjuk azokat a megfigyeleseket, ahol a valtozok barmelyikeben hianyzo adat (NA) van. Ha ezt nem tesszuk meg, a cor() fuggveny NA eredmenyt adhna ha valamelyik valtozoban NA-val talalkozik.

```
COVID_adat_tegnap %>%
  select(new_cases_per_million, gdp_per_capita) %>%
  drop_na() %>%
      cor()

##                       new_cases_per_million gdp_per_capita
## new_cases_per_million             1.0000000      0.2495567
## gdp_per_capita                    0.2495567      1.0000000

COVID_adat_tegnap %>%
  select(new_cases_per_million, gdp_per_capita, hospital_beds_per_thousand)
%>%
  drop_na() %>%
      cor()

##                             new_cases_per_million gdp_per_capita
## new_cases_per_million                  1.00000000      0.2246956
## gdp_per_capita                         0.22469556      1.0000000
## hospital_beds_per_thousand             0.04896543      0.2970931
##                             hospital_beds_per_thousand
## new_cases_per_million                       0.04896543
## gdp_per_capita                              0.29709314
## hospital_beds_per_thousand                  1.00000000
```

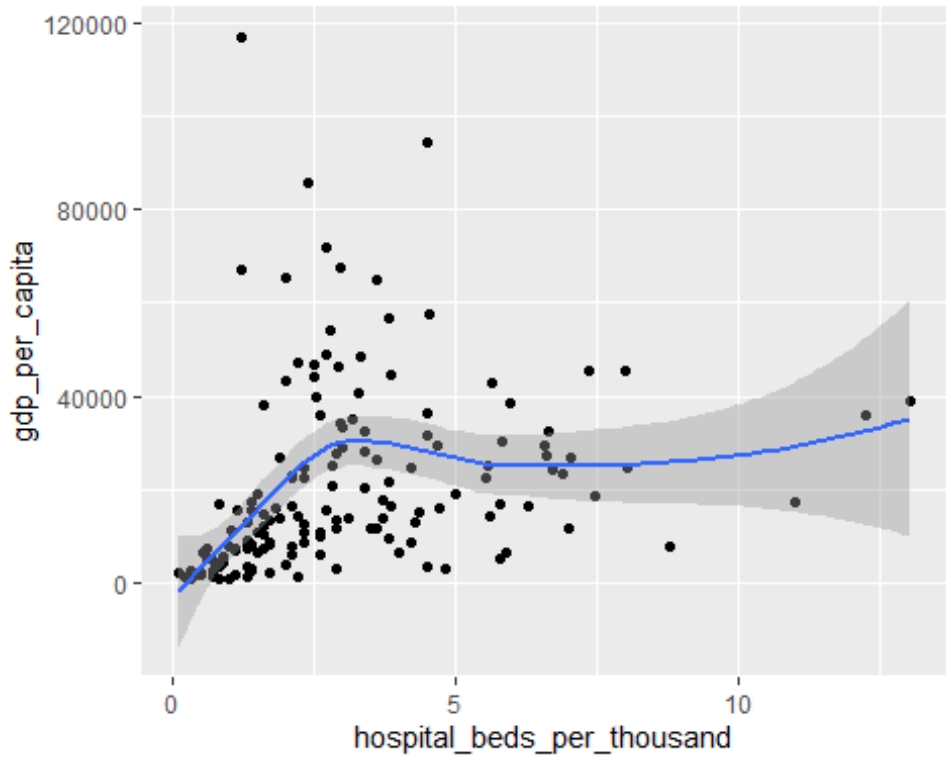
A numerikus valtozok kozotti kapcsolatot altalaban pont diagrammal szoktuk abrazolni (**geom_point()**)

A **geom_smooth()** layer hozzaadasaval kaphatunk a pontok kozott meghuzodo trendrol egy kepet. A kek vonal az ugyevezett trendvonal, a szurke sav a konfidencia intervallum. Ezekrol kesobb meg reszletesebben beszelunk majd

```r
COVID_adat_tegnap %>%
  select(hospital_beds_per_thousand, gdp_per_capita) %>%
  drop_na() %>%
  ggplot() +
    aes(x = hospital_beds_per_thousand, y = gdp_per_capita) +
    geom_point()
```




```r
COVID_adat_tegnap %>%
  select(hospital_beds_per_thousand, gdp_per_capita) %>%
  drop_na() %>%
  ggplot() +
    aes(x = hospital_beds_per_thousand, y = gdp_per_capita) +
    geom_point() +
    geom_smooth()
```



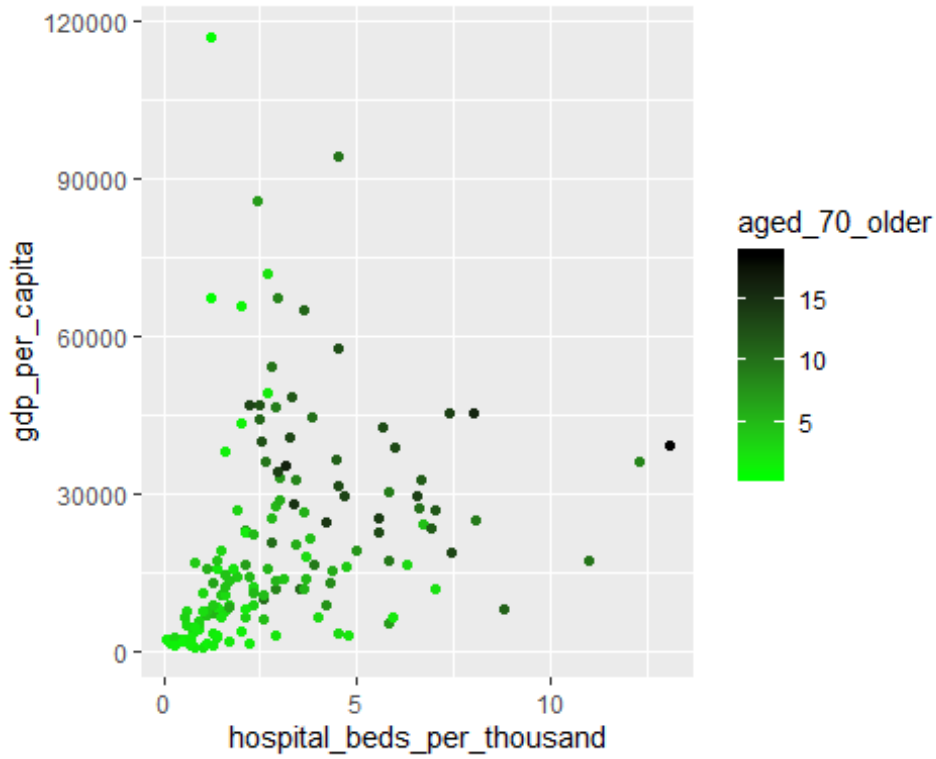
Gyakorlas

Milyen eros a kapcsolat a aged_70_older es a gdp_per_capita kozott?

- hatarozd meg a korrelacios egyutthatot a valtozok kozott
- abrazold a valtozok kapcsolatat

Tobb folytonos valtozo kapcsolata megjelenitheto peldaul ugy, hogy az egyik valtozot egy szinskalahoz rendeljuk az alabbi modon.

```
COVID_adat_tegnap %>%
  select(hospital_beds_per_thousand, gdp_per_capita, aged_70_older) %>%
  drop_na() %>%
  ggplot() +
    aes(x = hospital_beds_per_thousand, y = gdp_per_capita, col =
aged_70_older) +
    geom_point()+
  scale_colour_gradientn(colours=c("green","black"))
```


120000
90000
60000
30000
0
gdp_per_capita
0
5
10
hospital_beds_per_thousand
aged_70_older
15
10
5