Data Visualization using ggplot2

Setia

Pengenalan Package ggplot2

ggpplot2 (http://ggplot2.org/) merupakan package R yang dikhususkan untuk visualisasi data, dan hingga saat ini merupakan salah satu package yang banyak digunakan dikarenakan beberapa keuntungan sebagau berikut:

1. sangat fleksible dan komplit
2. mengimplementasikan "Grammar Graphics"
3. Bekerja dalam layer/lapisan
4. Grafik yang dihasilkan sangat bagus
5. Dukungan komunitas

Fungsi ggplot mengimplementasikan Grammar Of Graphics dimana grafik/plot dibentuk dari beberapa bagian (building blocks) yang kemudian digabungkan untuk membuat grafik yang diinginkan. Building blocks dari grafik adalah: 1. data 2. aesthetic mapping 3. geometric object 4. statistical transformations 5. scales 6. coordinate system 7. position adjustments 8. faceting

Geometric Objects And Aesthetics

Setiap grafik yang dihasilkan oleh package ggplot2 memiliki 2 komponen utama yaitu:

1. data,
2. aesthetic. Merupakan sekumpulan opsi untuk menampilkan variabel yang terdapat di data dan juga opsi2 grafik lainnya.
3. Geometrik Objek (fungsi geom). Minimal terdapat satu layer/lapisan yang mendeskripsikan bagaimana data tersebut akan ditampilkan.

Aesthetic Mapping

Di dalam ggplot fungsi aesthetic (aes()) digunakan untuk memformat apa saja yang akan kita lihat dan di dalamnya terdapat beberapa opsi/argumen seperti:

a. position (i.e., on the x and y axes)
b. color ("outside" color)
c. fill ("inside" color)
d. shape (of points)
e. linetype
f. size

Geometic Objects (geom)

Pada objek Geometric merupakan setting untuk tanda yang akan kita tampilkan pada grafik yang terdiri dari banyak opsi antara lain:

a. points (geom_point, for scatter plots, dot plots, etc)
b. lines (geom_line, for time series, trend lines, etc)
c. boxplot (geom_boxplot, for, well, boxplots!)

Sebuah harus memiliki minimal sebuah layer/lapisan geom, dengan tidak ada batasan jumlah lapisan yang ditampilkan. Untuk menambah lapisan cukup dengan tanda "+". ‘

Beberapa "layers" untuk 1 variabel:

Untuk continuous variable: 1. geom_area() : area plot 2. geom_density() : density plot 3. geom_dotplot() : dot plot 4. geom_freqpoly() : frequency polygon 5. geom_histogram() : histogram plot 6. stat_ecdf() : empirical cumulative density function 7. stat_qq() : quantile - quantile plot

Untuk discrete variable:

geom_bar() : bar plot

Berikut ini layers untuk dua variable kontinu:

geom_point() : scatter plot geom_smooth() : menambah smoothed line geom_quantile() : menambah quantile lines geom_rug() : menambah a marginal rug geom_jitter() : menghindari overplotting geom_text() : menambah text

Untuk Variabel diskrit dan kontinu geom_boxplot() : box plot geom_violin() : violin plot geom_dotplot() : dot plot geom_jitter() : stripchart geom_line() : line plot geom_bar() : bar plot

Shortcut: qplot

```
#install.packages("ggplot2"")

library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.4
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.4
```

```
##
## 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
```

```
setwd("C:/Users/stis/Documents/Training R Data Science Pusdiklat")
country <- read.csv("CountryData.csv")
dim(country)
```

```
## [1] 256  77
```

```
country <- country %>% mutate(lpop= log10(pop),
                              lGDP=log(GDP),
                              )
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

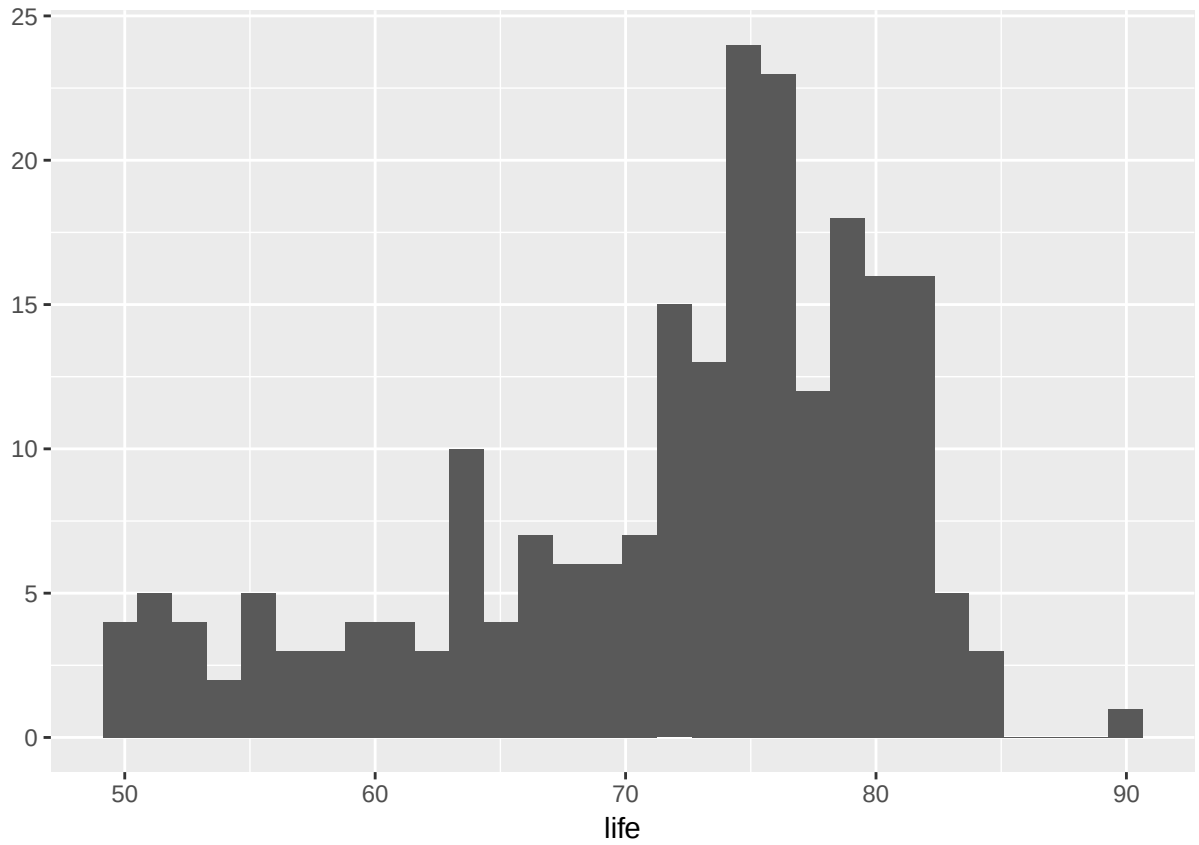
```
#country$lpop <- log10(country$pop)


country$grpGDPcap <- cut(country$GDPcapita,
```

```
                  breaks=c(-Inf, 0.3, 0.6, Inf),
                  labels=c("low","middle","high"))

country <- country %>%
  mutate(grpGDPcap=cut(GDPcapita,
  breaks=c(quantile(GDPcapita,probs=c(0,0.3, 0.6,1 ),na.rm = T)),
  labels=c("low","middle","high")),
  grpArea=cut(area,
                breaks=c(quantile(area,probs=c(0,0.25, 0.75,1 ),na.rm = T)),
                labels=c("Small","Medium","Large"))

  )

table(country$grpGDPcap)
```

```
##
##    low middle   high
##     68     68     91
```

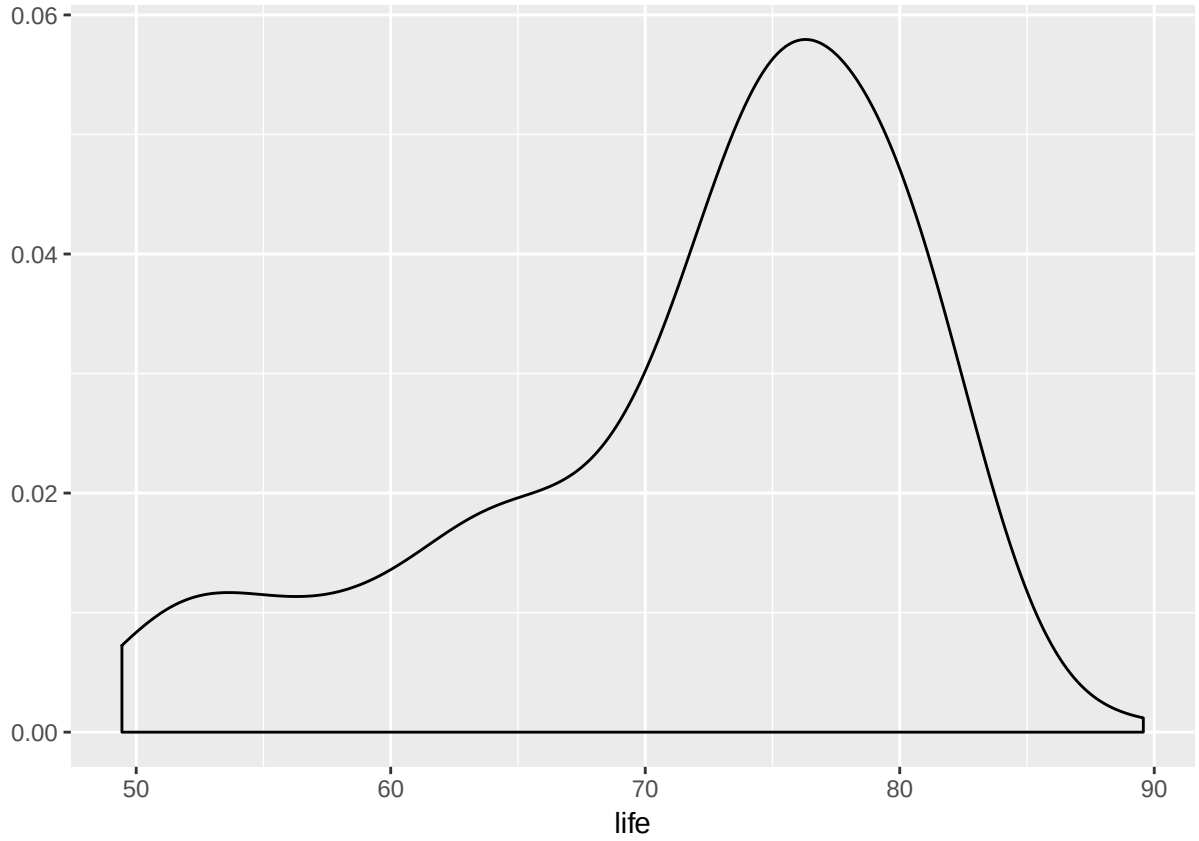
```
## Histogram ##

qplot(x = life, data = country, geom = "histogram")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 33 rows containing non-finite values (stat_bin).
```



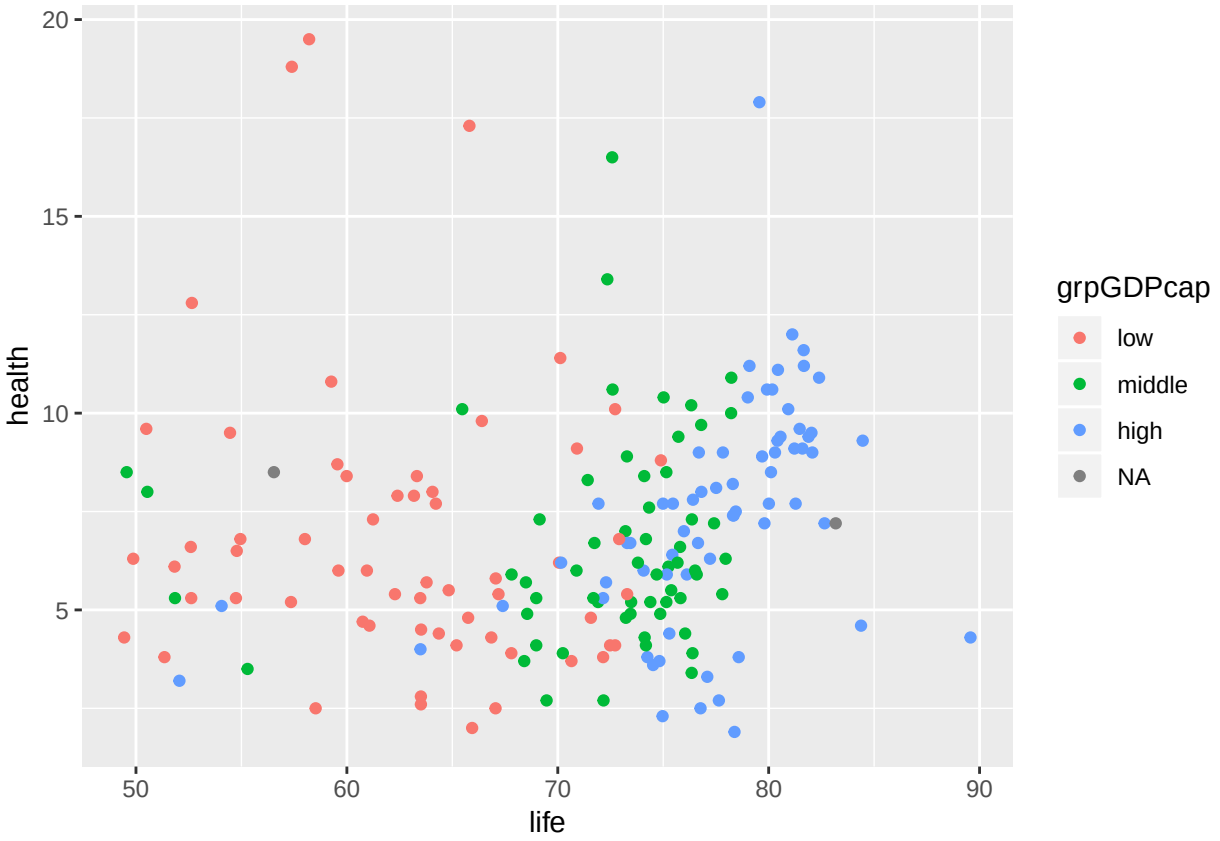
```
qplot(x = life, data = country, geom = "density")
```

```
## Warning: Removed 33 rows containing non-finite values (stat_density).
```

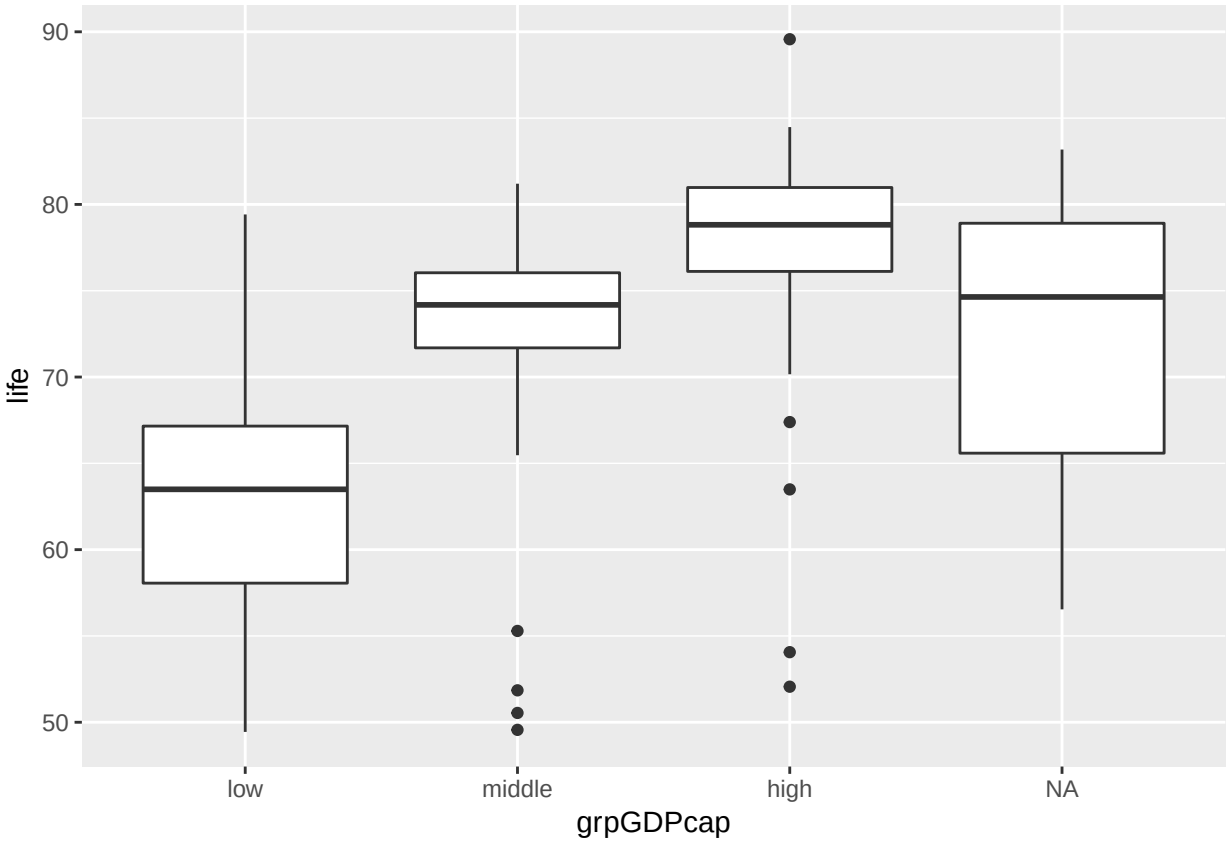



```
qplot(x = life, y=health, data = country, geom = "point", colour=grpGDPcap)
```

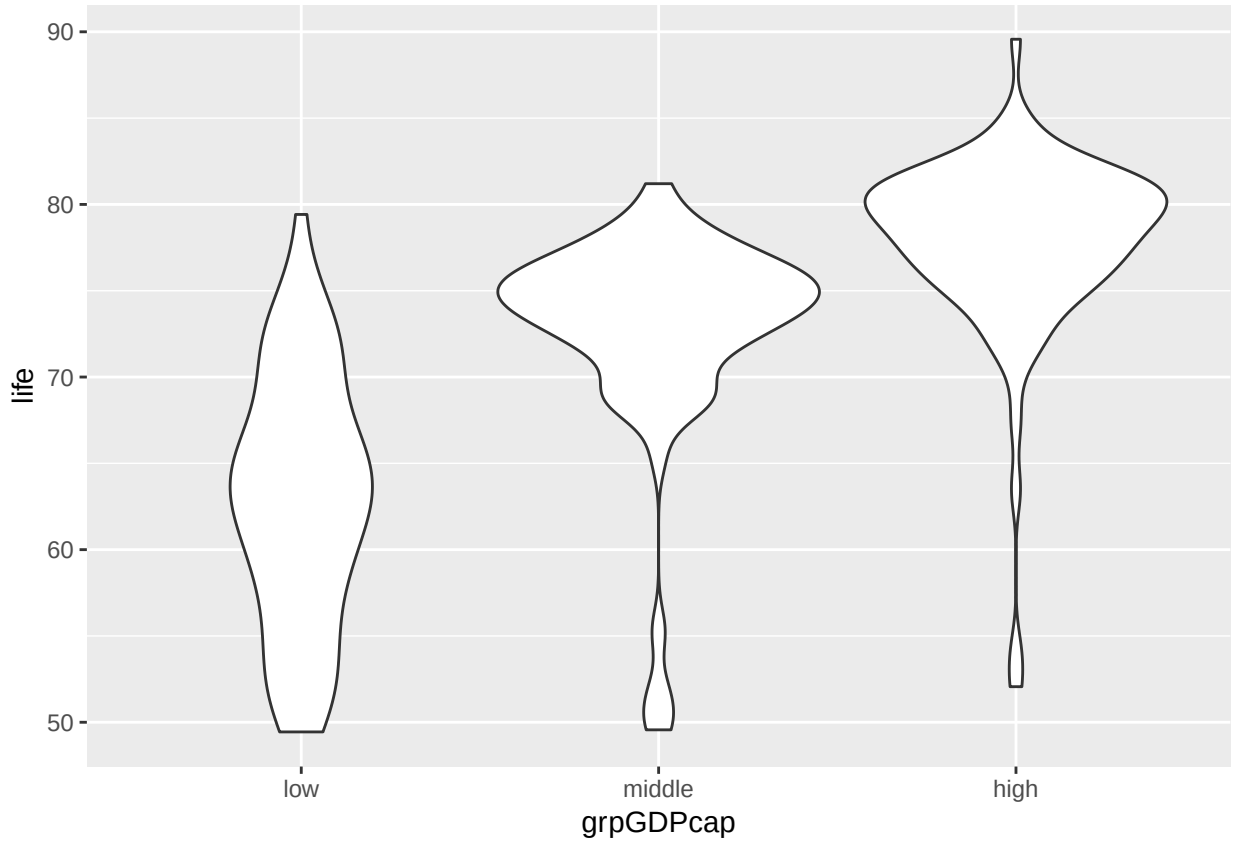
```
## Warning: Removed 68 rows containing missing values (geom_point).
```




```
qplot(x=grpGDPcap,y = life, data = country, geom = "boxplot", na.rm=T)
```

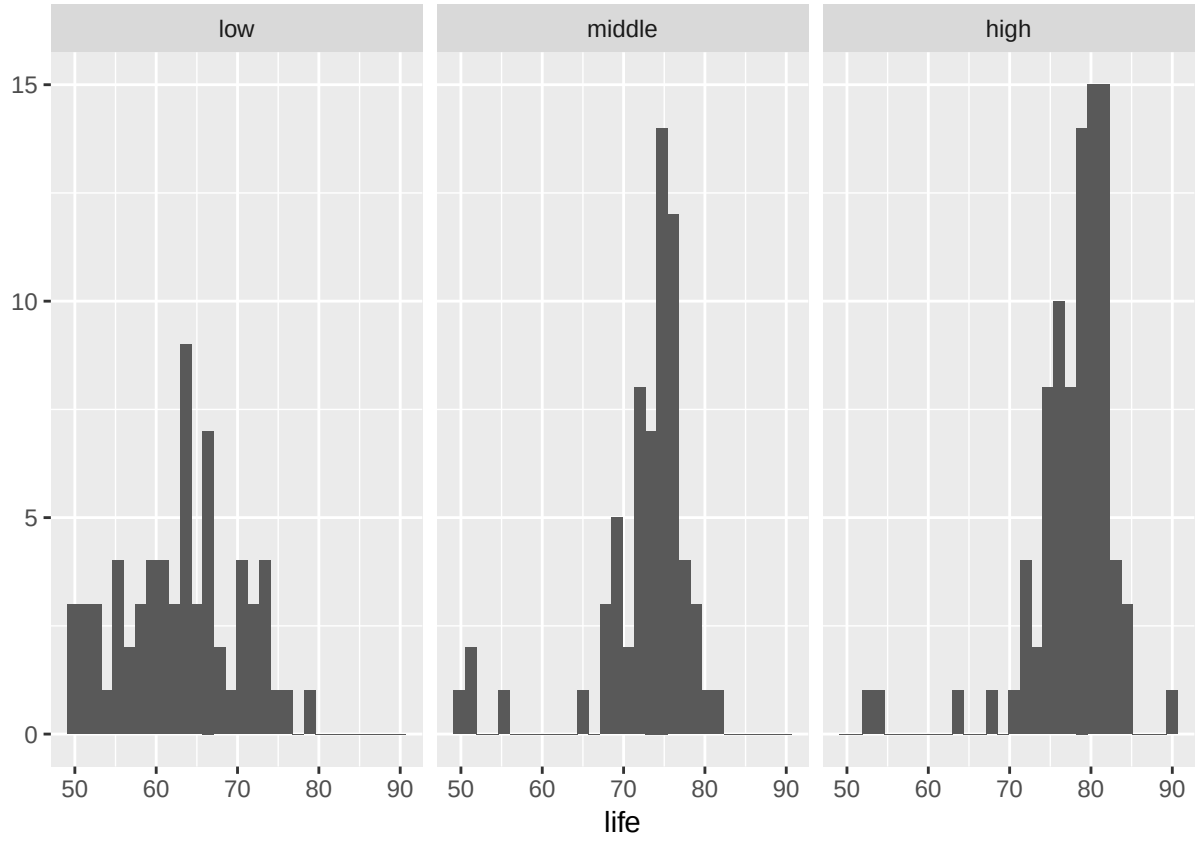



```
qplot(x=grpGDPcap,y = life, data = country[!is.na(country$grpGDPcap),], geom = "violin", na.rm=T)
```




```
# facet #
country2 <- country[!is.na(country$grpGDPcap),]

qplot(x = life, data = country2, geom = "histogram")+facet_wrap(~grpGDPcap)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 7 rows containing non-finite values (stat_bin).
```

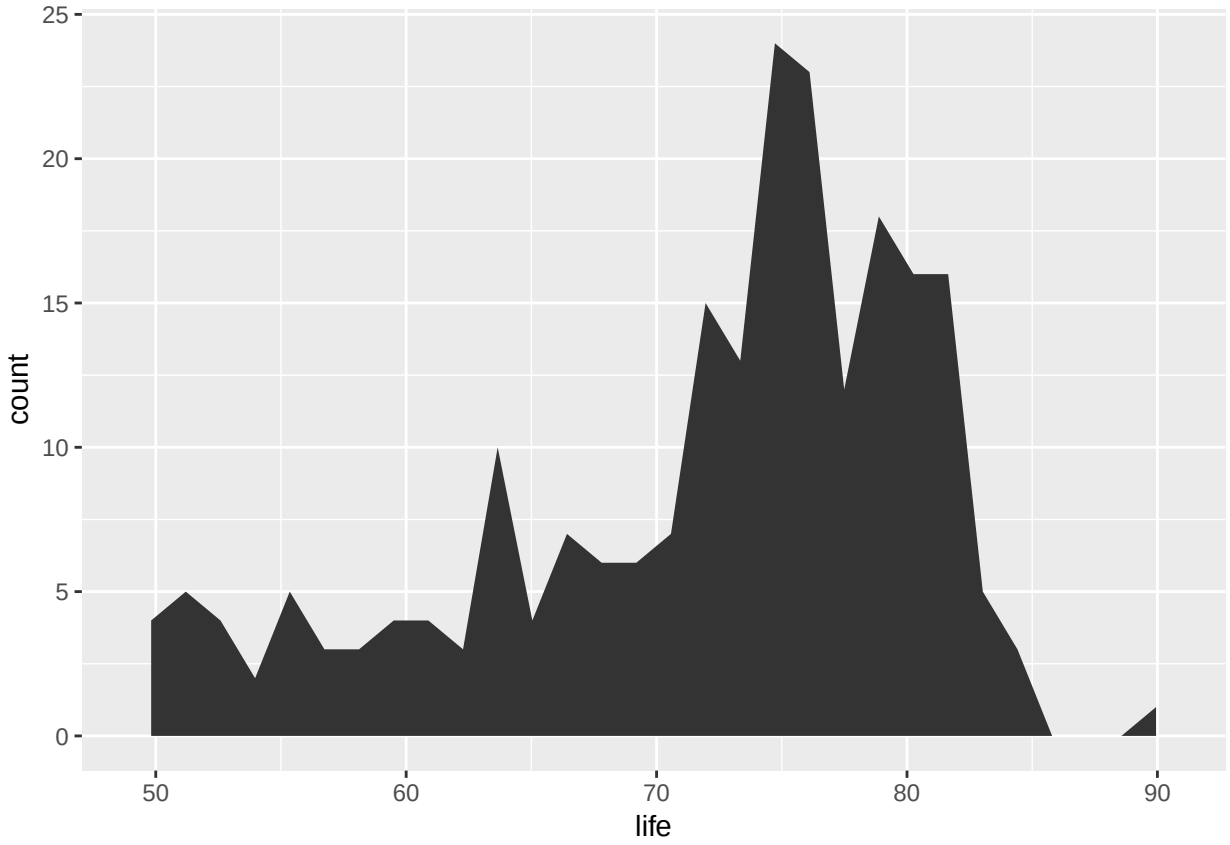


Contoh ggplots

```
## Building step by step

## One variable

## continuous var
p1 <- ggplot(country, aes(x=life))
p1 + geom_area (stat = "bin")
```

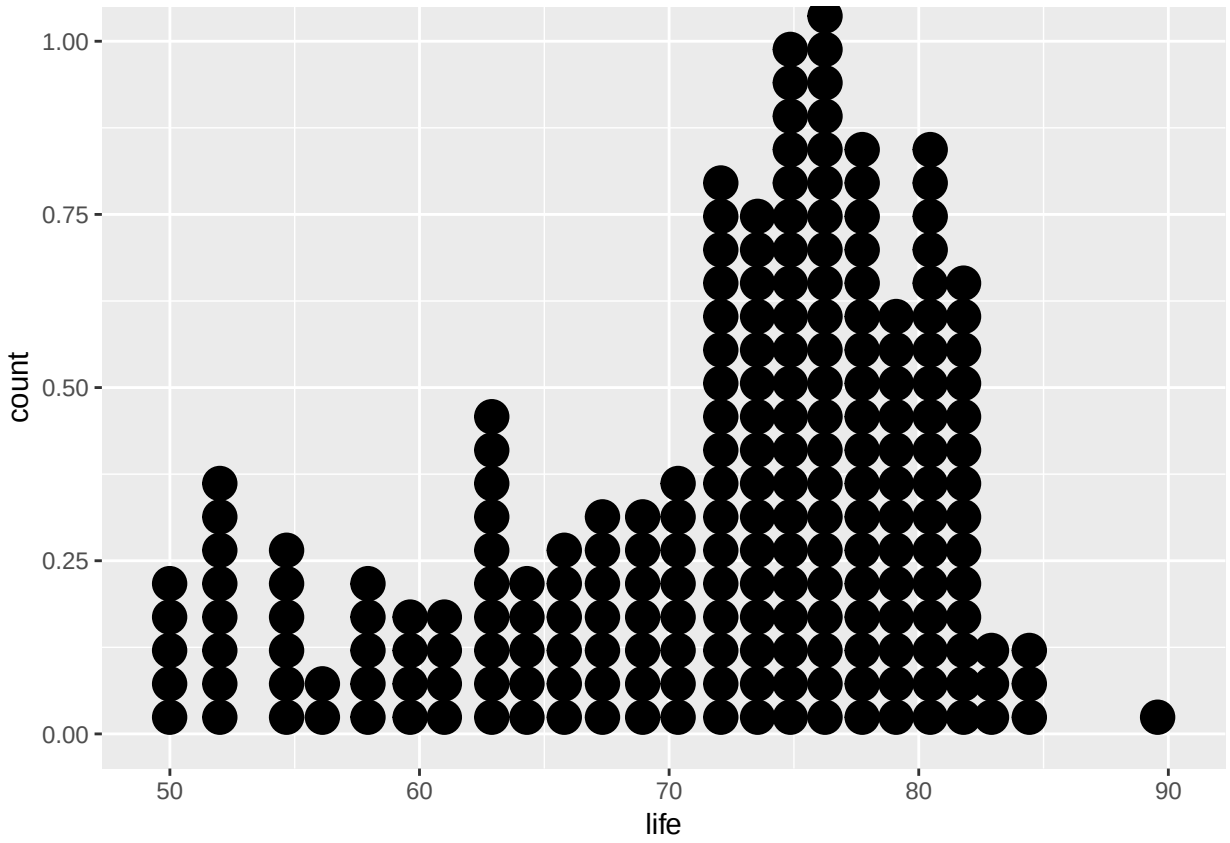
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 33 rows containing non-finite values (stat_bin).
```



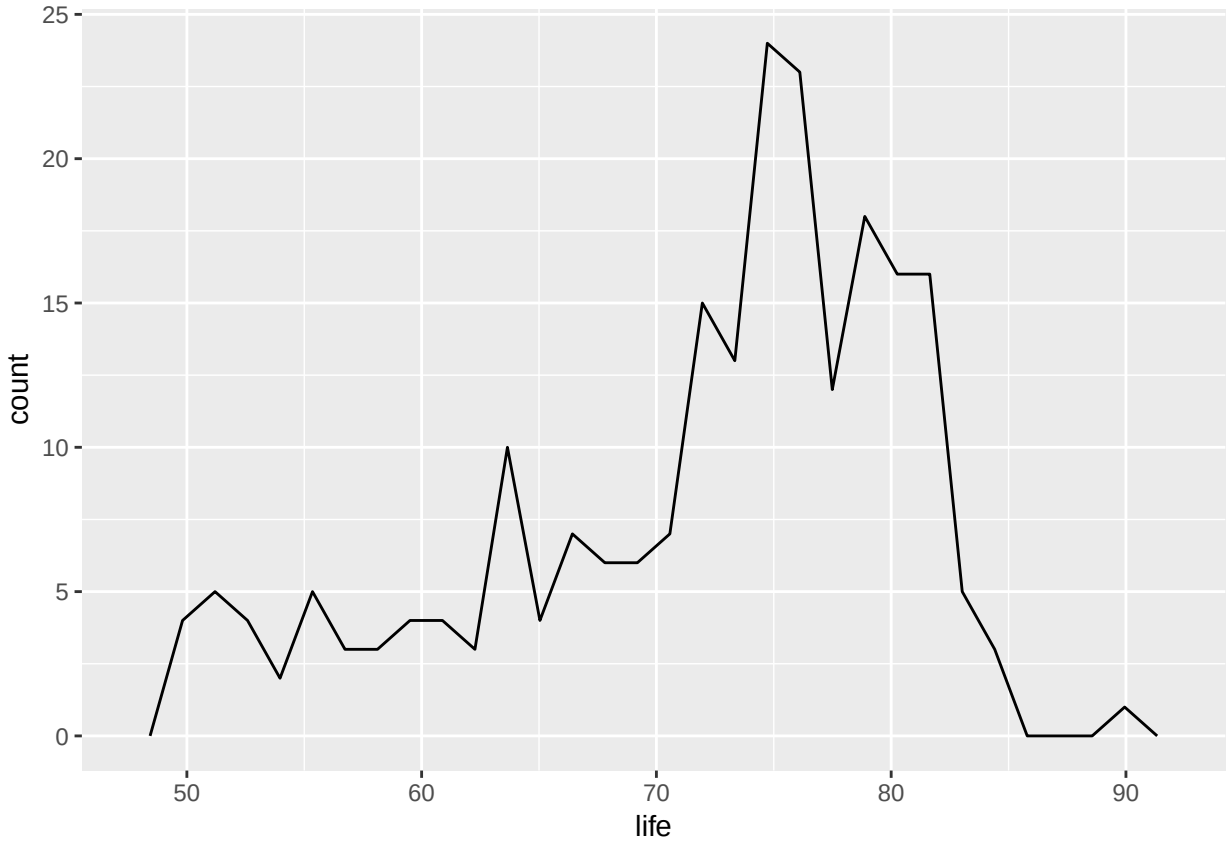

```
p1 + geom_dotplot()
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 33 rows containing non-finite values (stat_bindot).
```



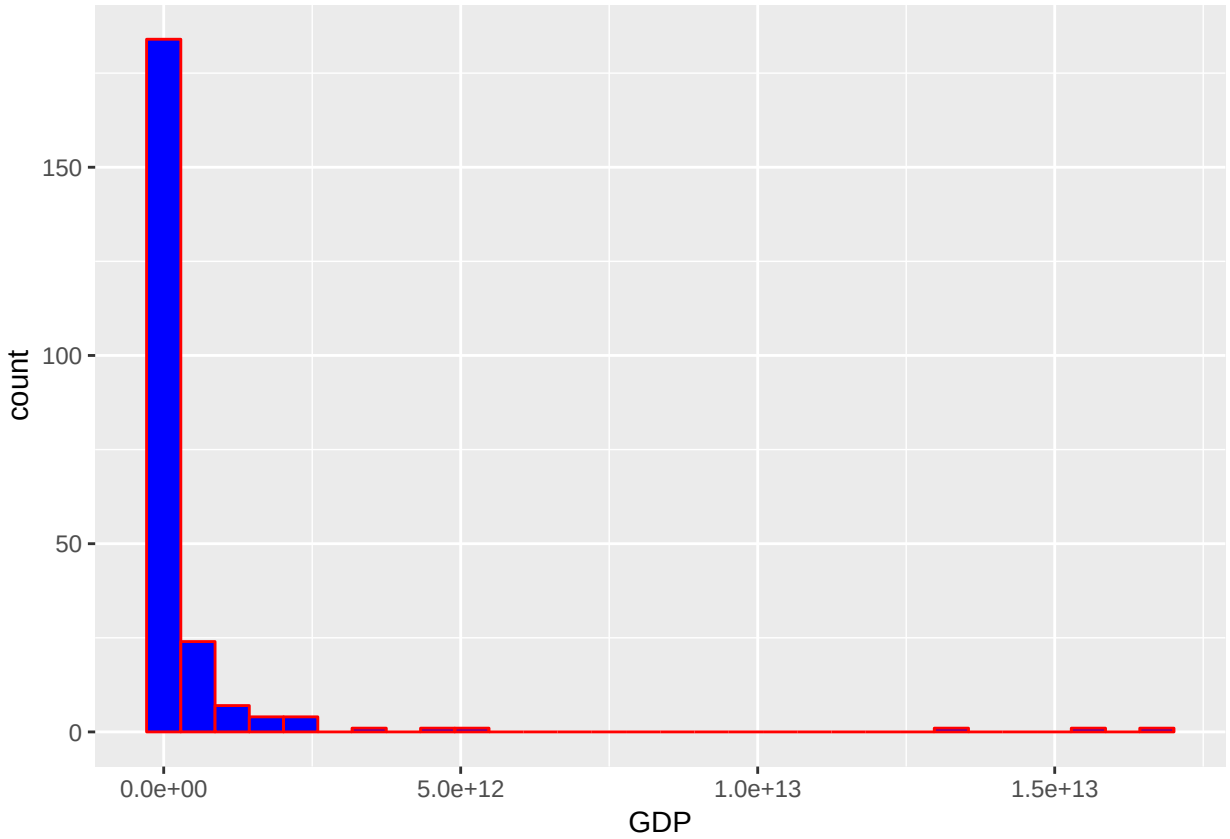

```
p1 + geom_freqpoly()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 33 rows containing non-finite values (stat_bin).
```



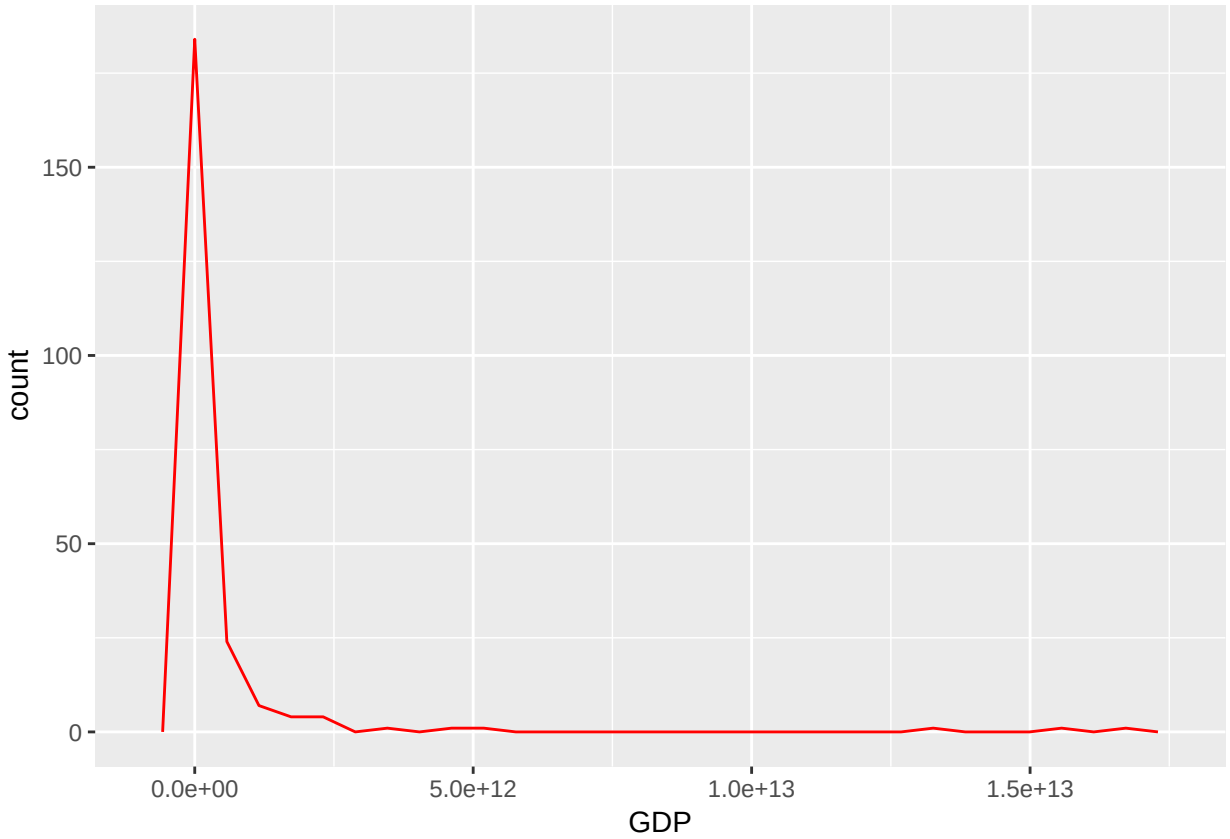

```
ggplot(country, aes(GDP)) + geom_histogram(fill="blue", color="red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 27 rows containing non-finite values (stat_bin).
```




```
ggplot(country, aes(GDP)) + geom_freqpoly(colour="red")
```

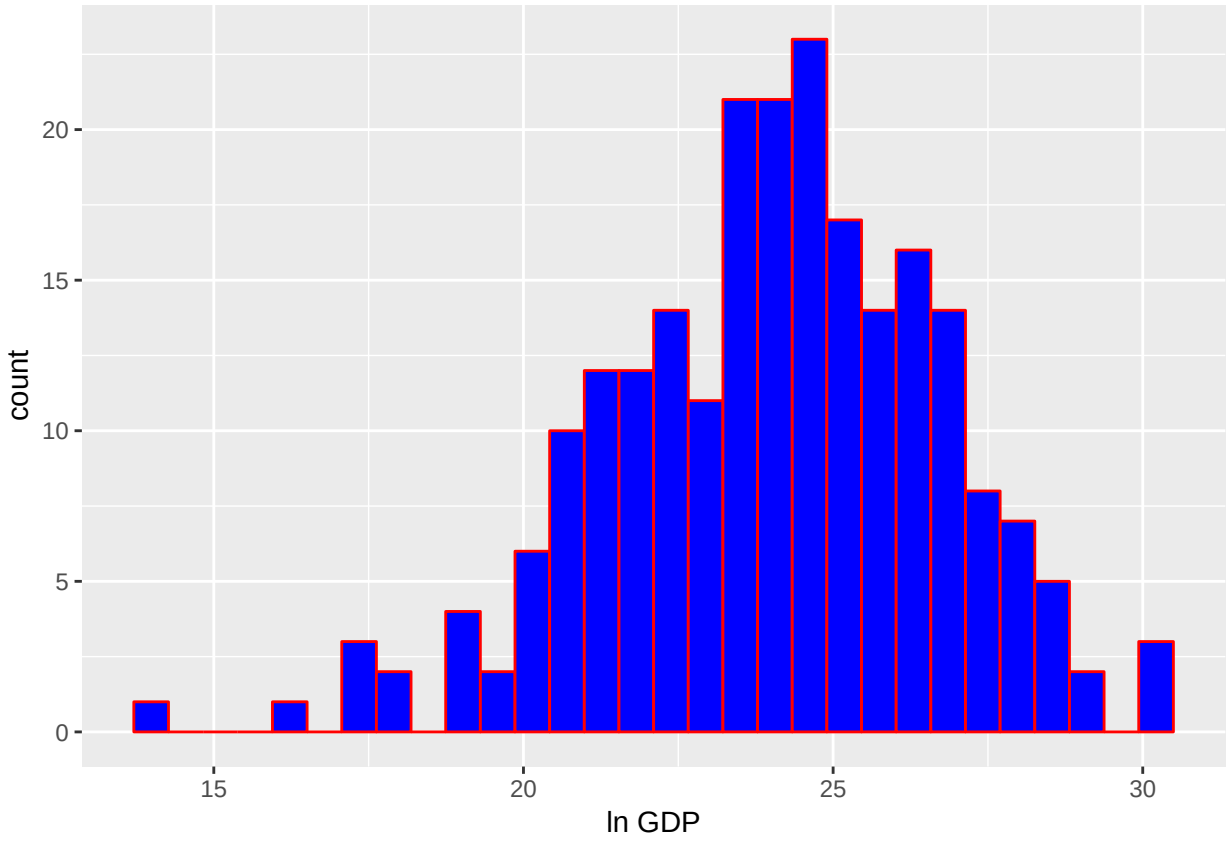
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 27 rows containing non-finite values (stat_bin).
```




```
ggplot(country, aes(lGDP)) + geom_histogram(fill="blue", color="red")+
  xlab("ln GDP")
```

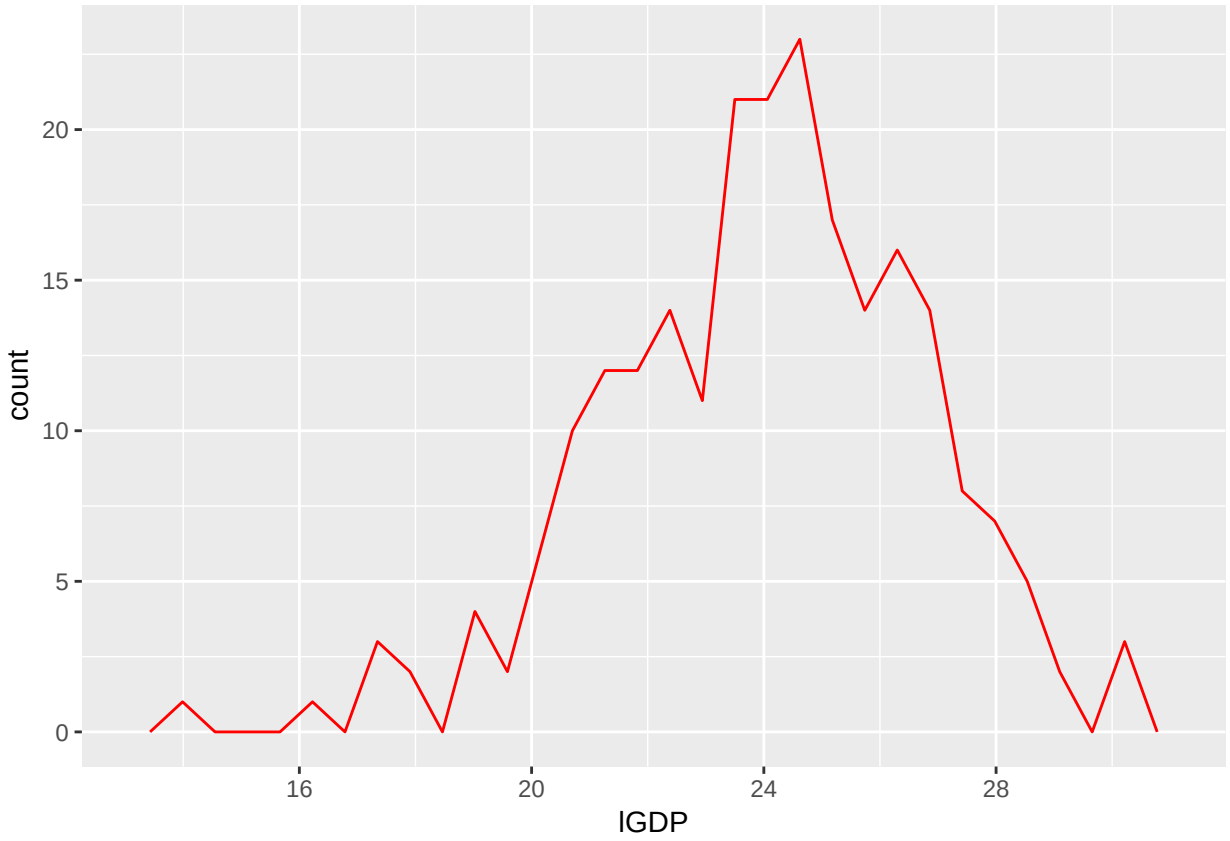
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 27 rows containing non-finite values (stat_bin).
```

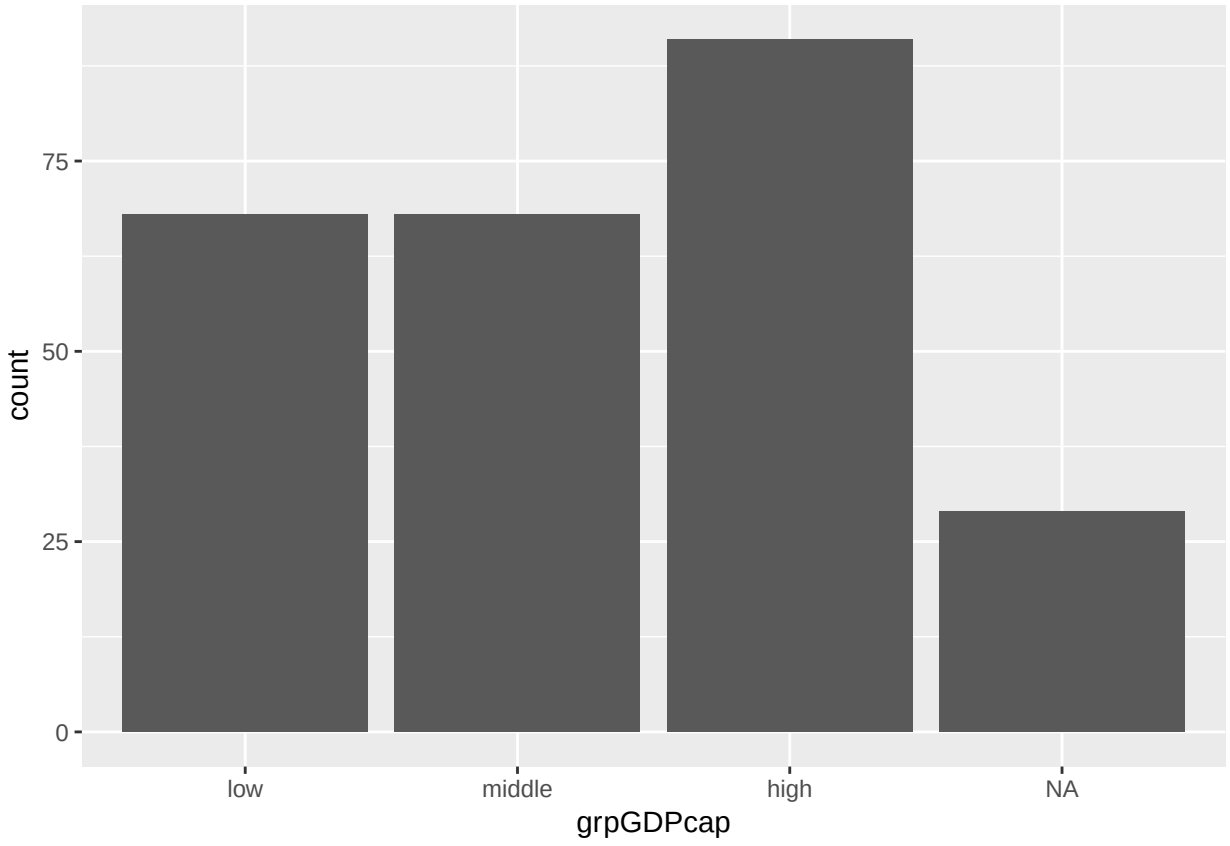



```
ggplot(country, aes(lGDP)) + geom_freqpoly(colour="red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 27 rows containing non-finite values (stat_bin).
```

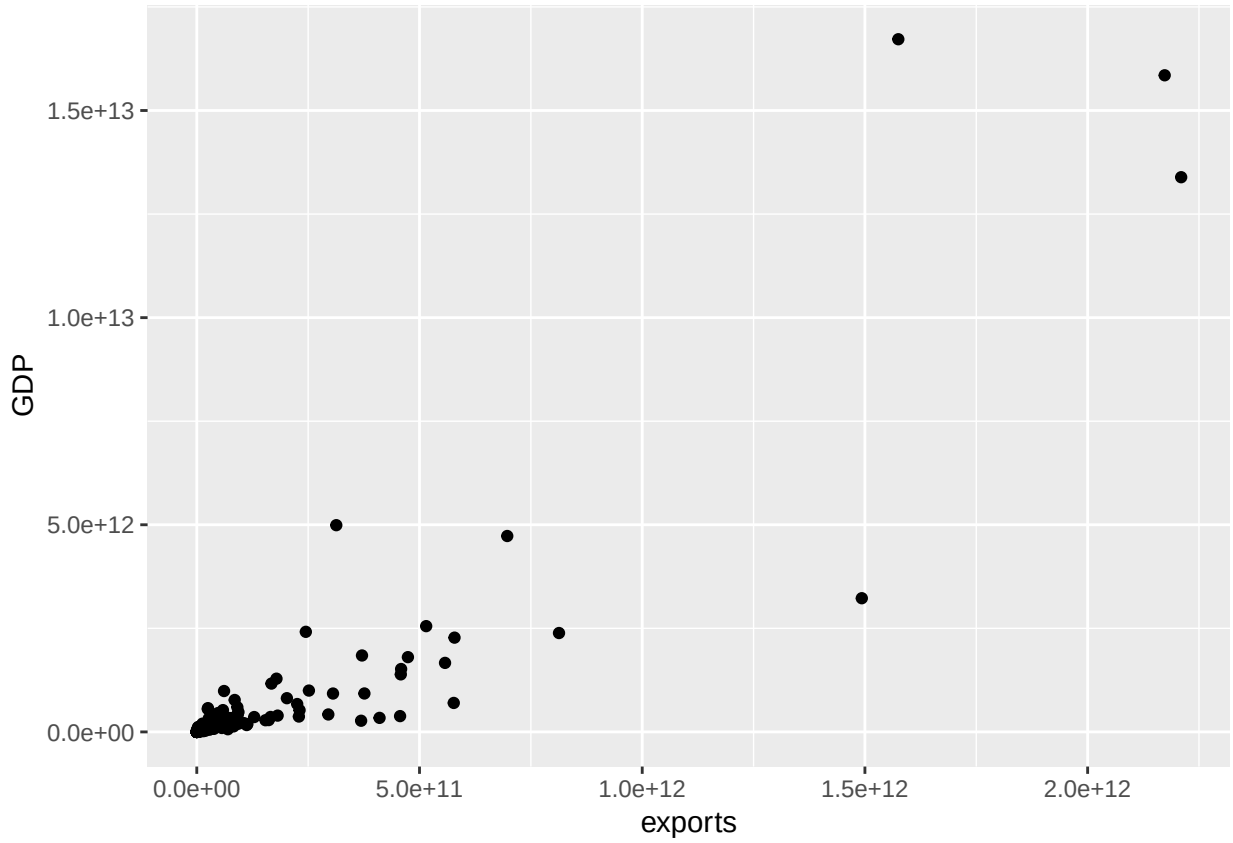



```
## Barplot
p2 <-  ggplot(country, aes(x=grpGDPcap))
p2 + geom_bar()
```



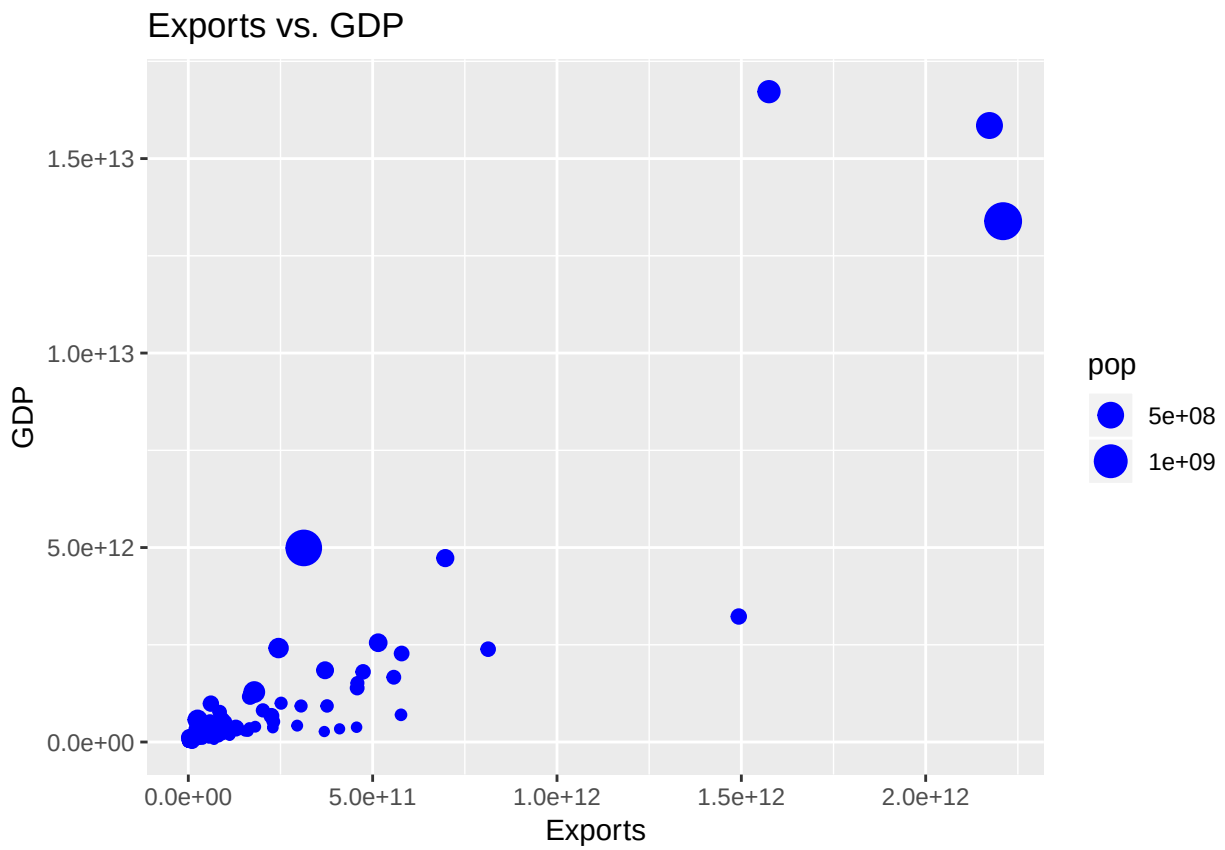

```
## Scatter Plot
ggplot(country,aes(x=exports, y=GDP)) + geom_point()
```

```
## Warning: Removed 33 rows containing missing values (geom_point).
```




```
## Adding Color, label and title
ggplot(country,aes(x=exports, y=GDP)) + geom_point(colour="blue",aes(size = pop)) +  xlab("Exports") +
  ylab("GDP") +
  ggtitle("Exports vs. GDP")
```

```
## Warning: Removed 33 rows containing missing values (geom_point).
```



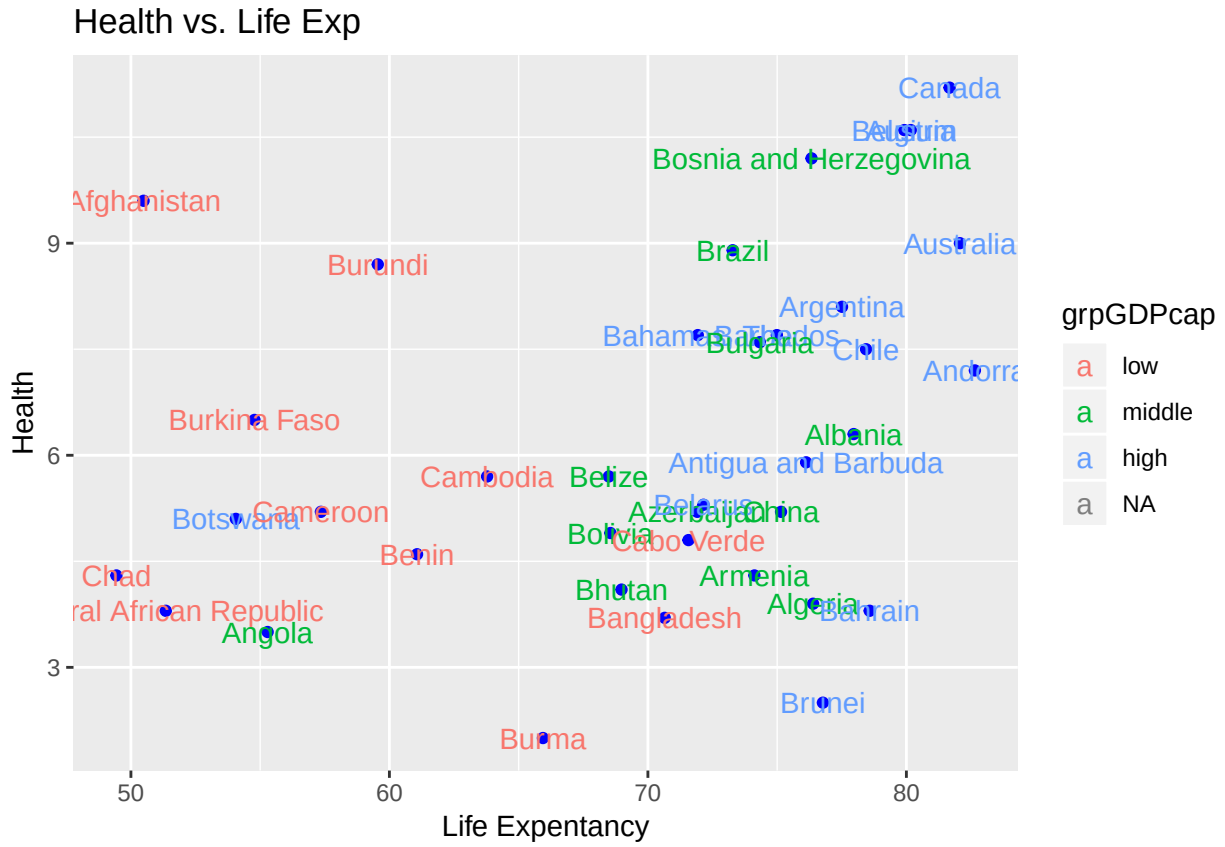

```
p3 <- ggplot(country[1:50,],aes(x=life, y=health)) +
  geom_point(colour="blue") +  xlab("Life Expentancy") +
  ylab("Health") +
  ggtitle("Health vs. Life Exp")

p3
```

```
## Warning: Removed 13 rows containing missing values (geom_point).
```

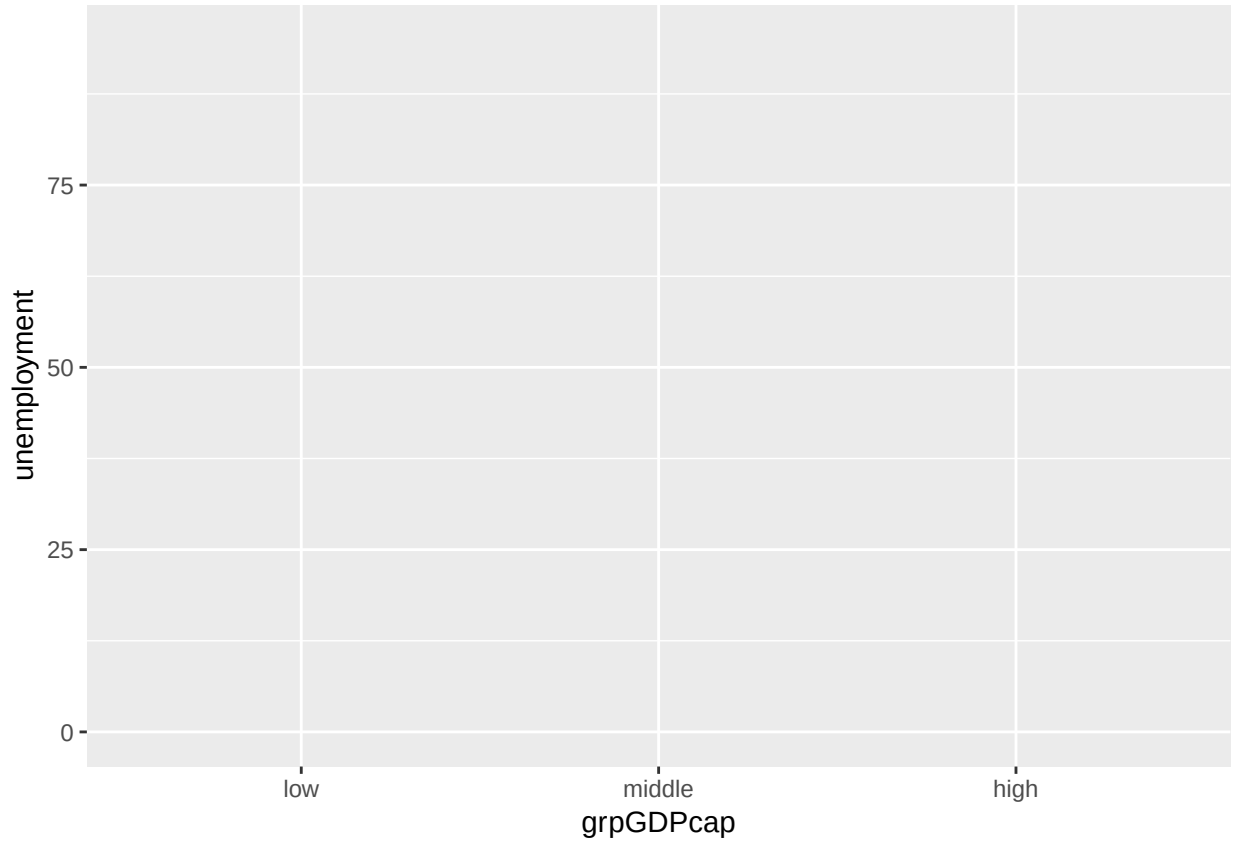

```
p3 + geom_text(aes(label = country, col=grpGDPcap))
```

```
## Warning: Removed 13 rows containing missing values (geom_point).
```

```
## Warning: Removed 13 rows containing missing values (geom_text).
```

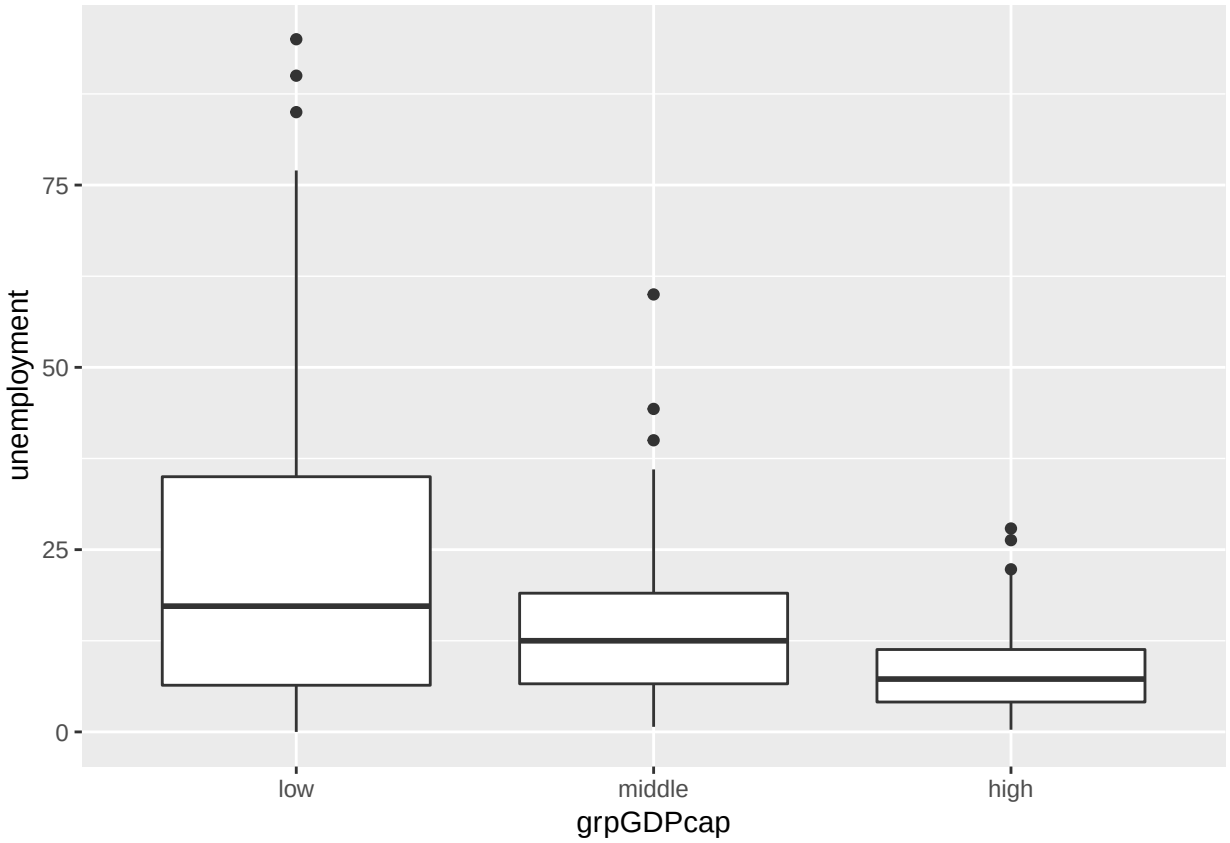



```
## Box Plot
p4 <- ggplot(country2, aes(x = grpGDPcap, y = unemployment))
p4
```



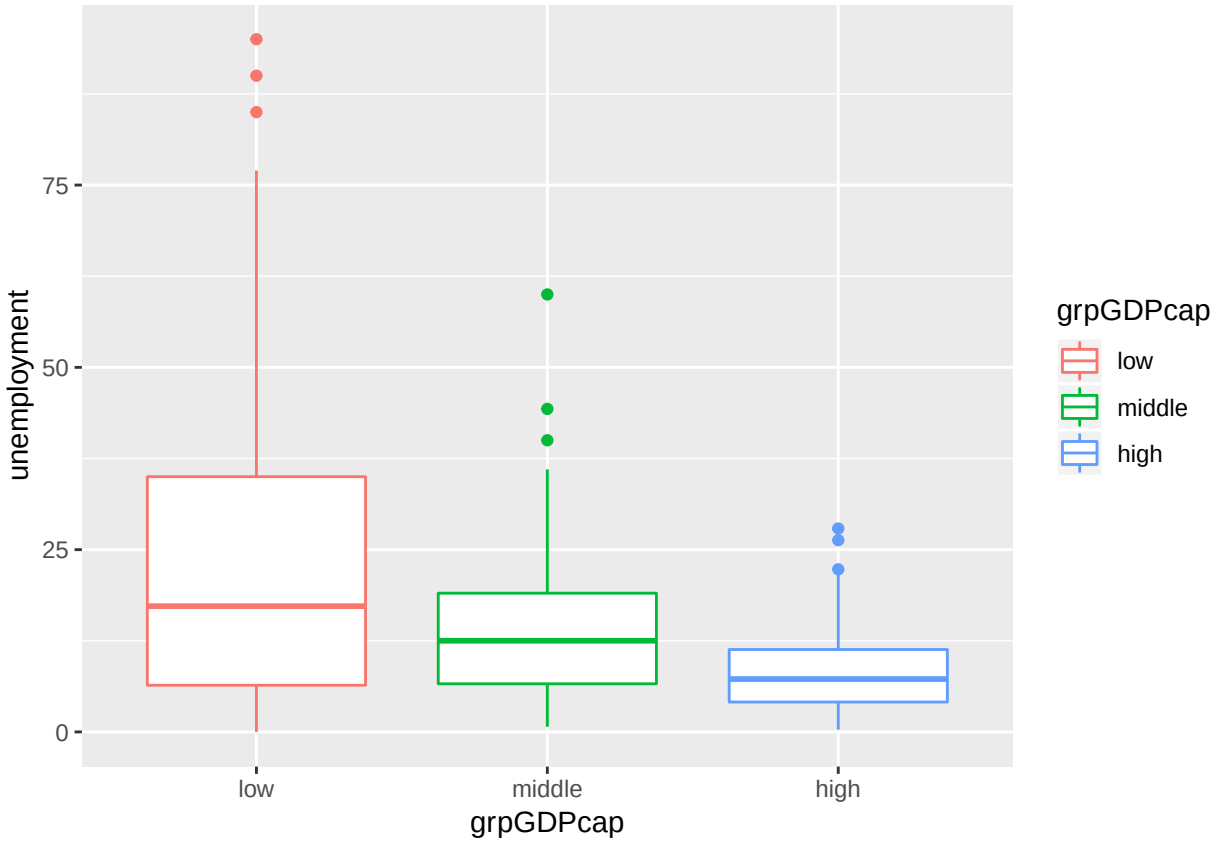

```
# Default plot
p4 + geom_boxplot()
```

```
## Warning: Removed 27 rows containing non-finite values (stat_boxplot).
```



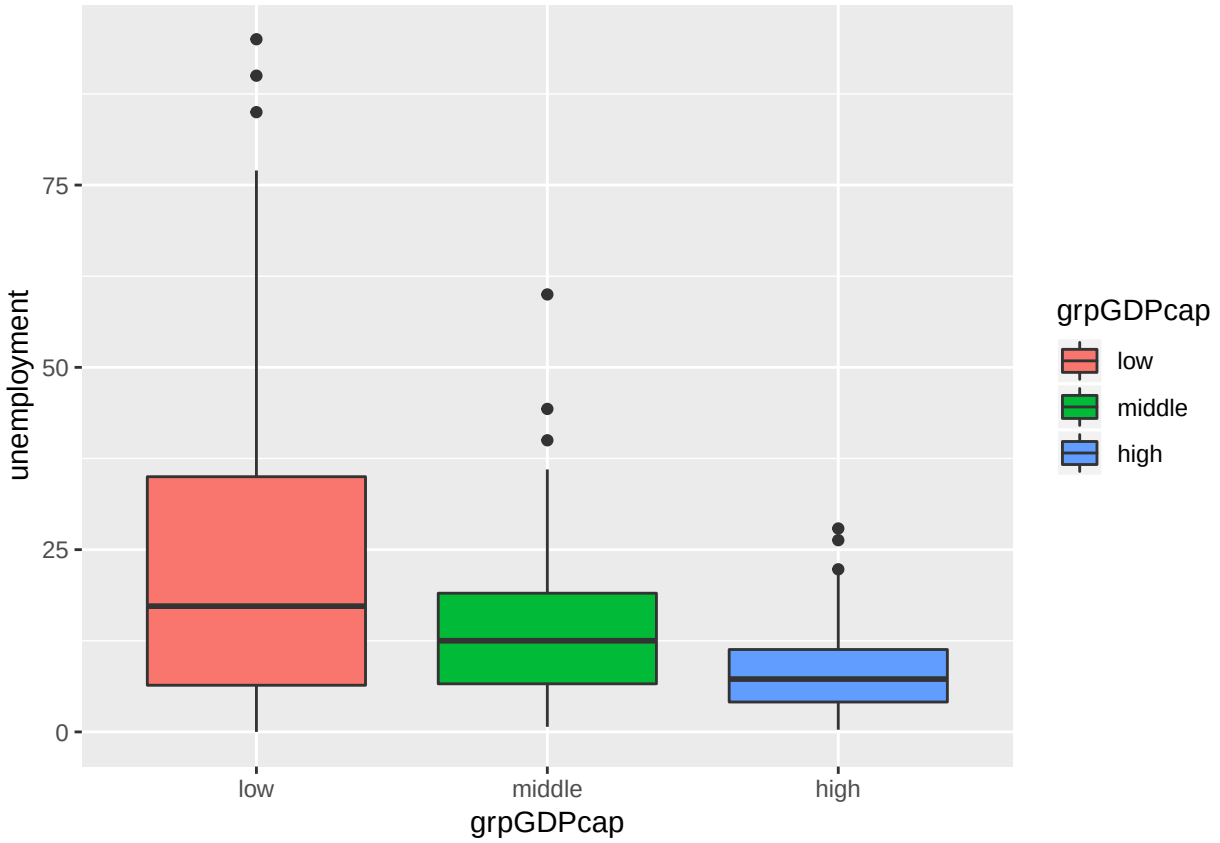

```
# Color by group
p4 + geom_boxplot(aes(color = grpGDPcap))
```

```
## Warning: Removed 27 rows containing non-finite values (stat_boxplot).
```



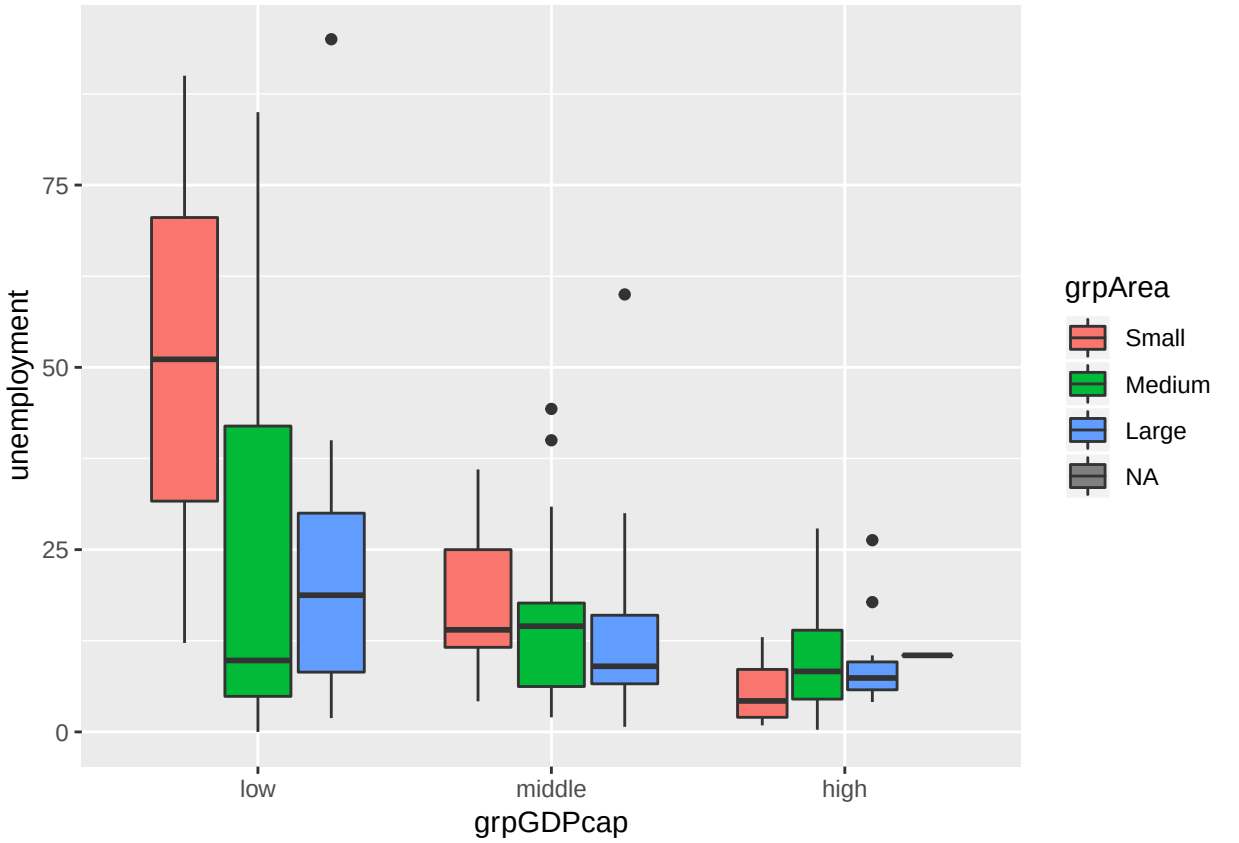

```
# Change fill color by group
p4 + geom_boxplot(aes(fill = grpGDPcap))
```

```
## Warning: Removed 27 rows containing non-finite values (stat_boxplot).
```



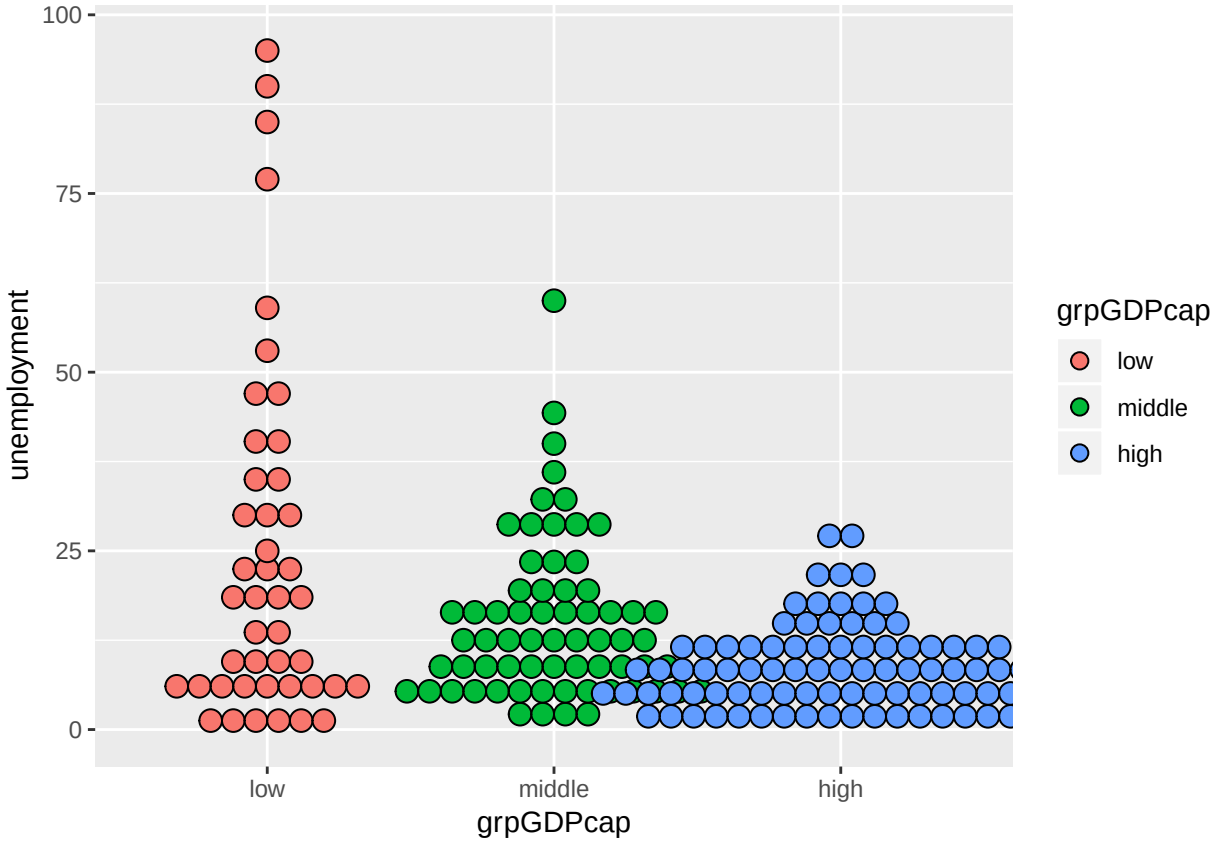

```
ggplot(country2, aes(x = grpGDPcap, y = unemployment, fill=grpArea))+
  geom_boxplot()
```

```
## Warning: Removed 27 rows containing non-finite values (stat_boxplot).
```



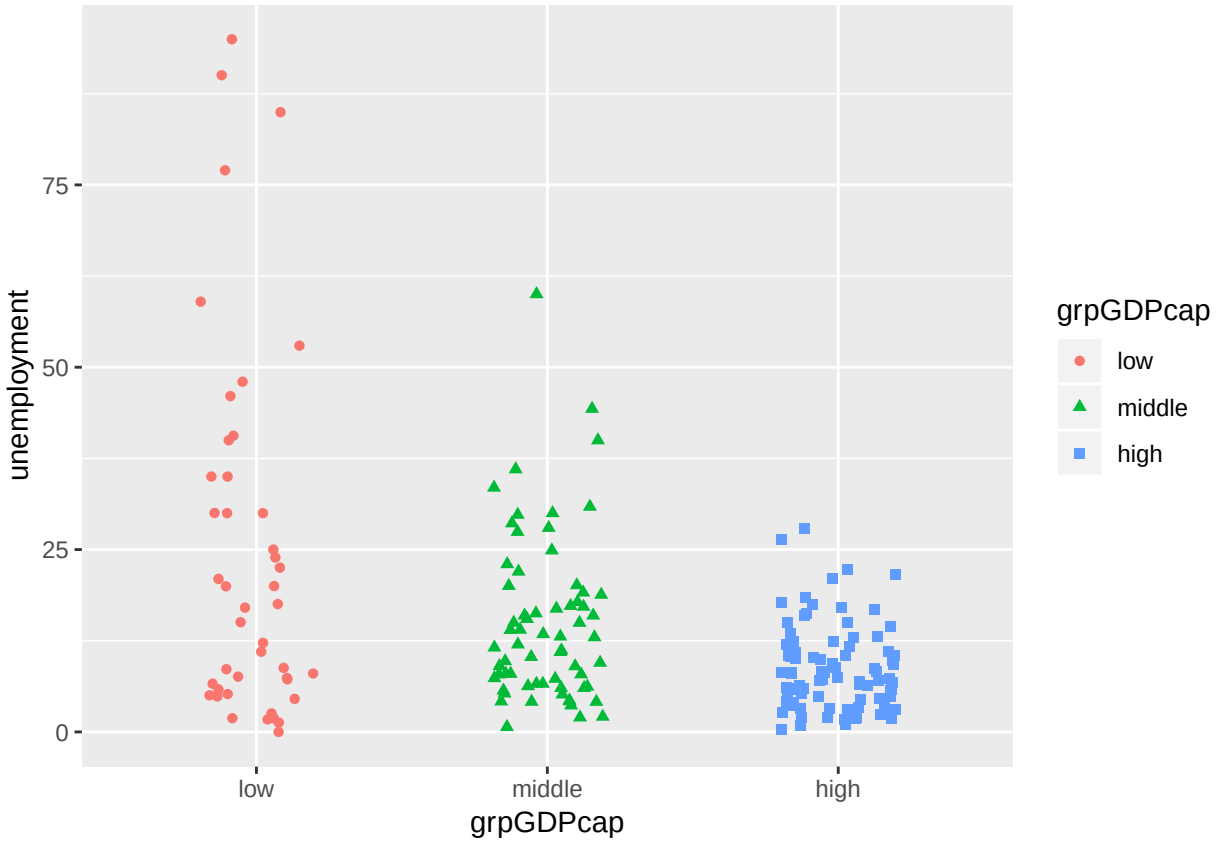

```
p4 + geom_dotplot(binaxis = "y", stackdir = "center",aes(shape=grpGDPcap,
                                                         fill=grpGDPcap))
```

```
## Warning: Ignoring unknown aesthetics: shape
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 27 rows containing non-finite values (stat_bindot).
```

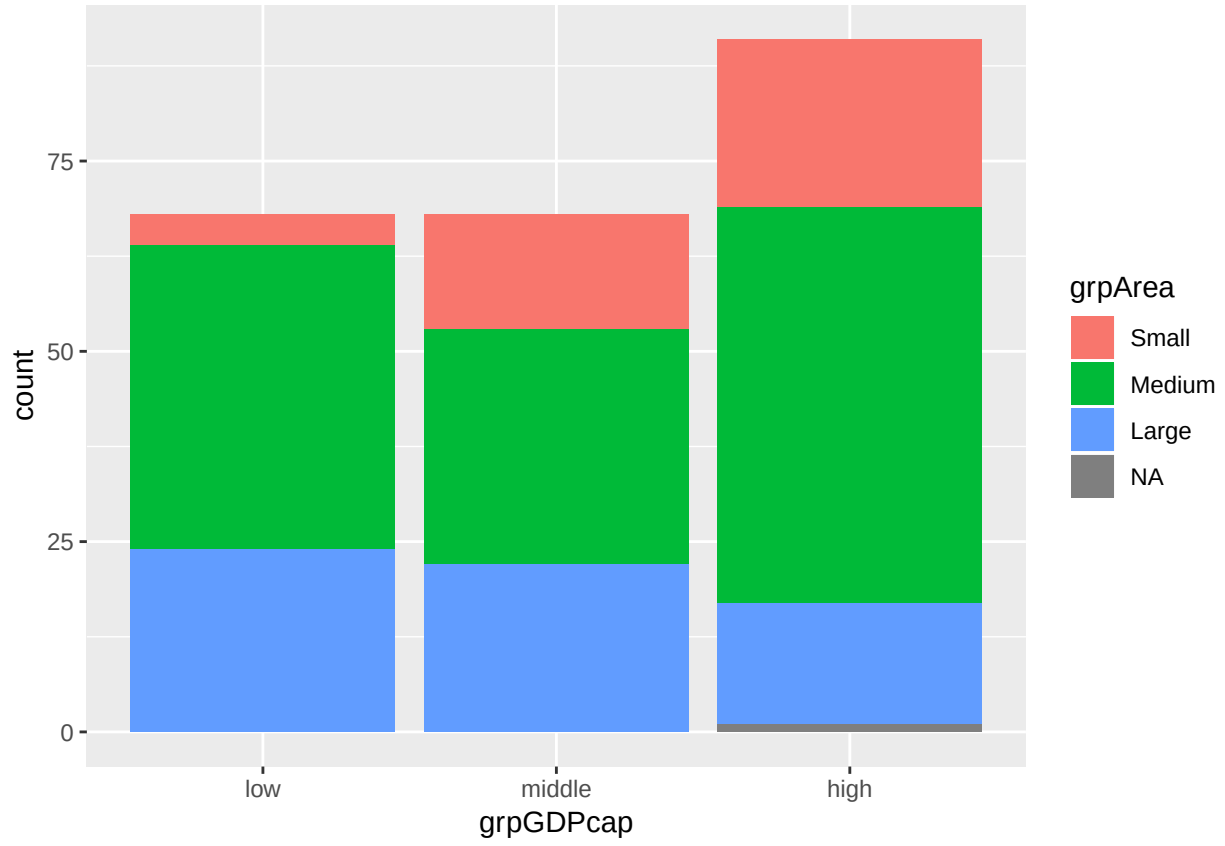



```
p4 +  geom_jitter(aes(color = grpGDPcap, shape = grpGDPcap),
                  position=position_jitter(0.2))
```

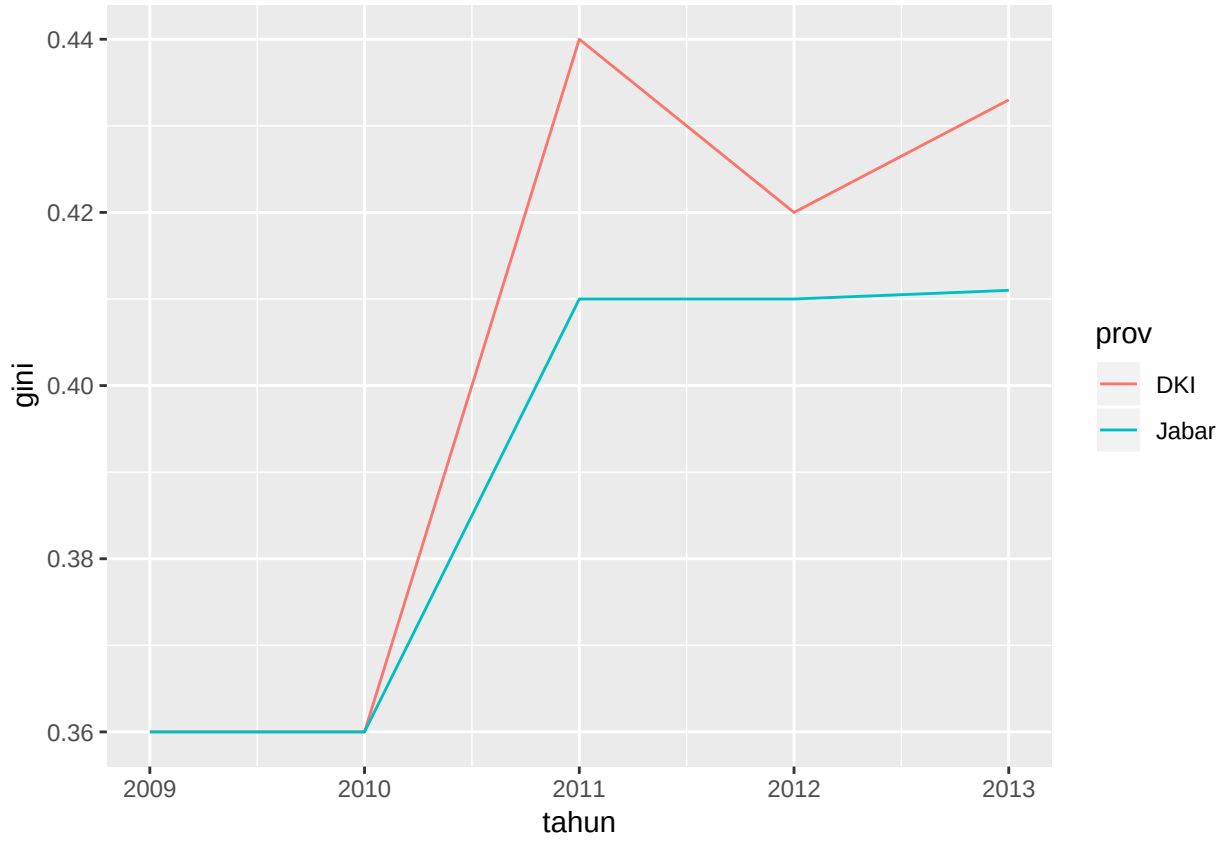
```
## Warning: Removed 27 rows containing missing values (geom_point).
```




```
ggplot(country2, aes(x=grpGDPcap, fill=grpArea)) + geom_bar()
```




```
## Data Gini Ratio ##

tahun <-  rep(c(2009, 2010, 2011, 2012, 2013),2)
gini <- c(0.36, 0.36, 0.44, 0.42, 0.433, 0.36, 0.36, 0.41, 0.41, 0.411)
prov <- rep(c("DKI", "Jabar"),each=5)
giniData <- data.frame(tahun, gini, prov)

p10 <- ggplot(giniData, aes(x = tahun, y = gini))
p10 + geom_line(aes(color = prov))
```


0.44
0.42
0.40
0.38
0.36
gini
2009
2010
2011
2012
2013
tahun
prov
DKI
Jabar