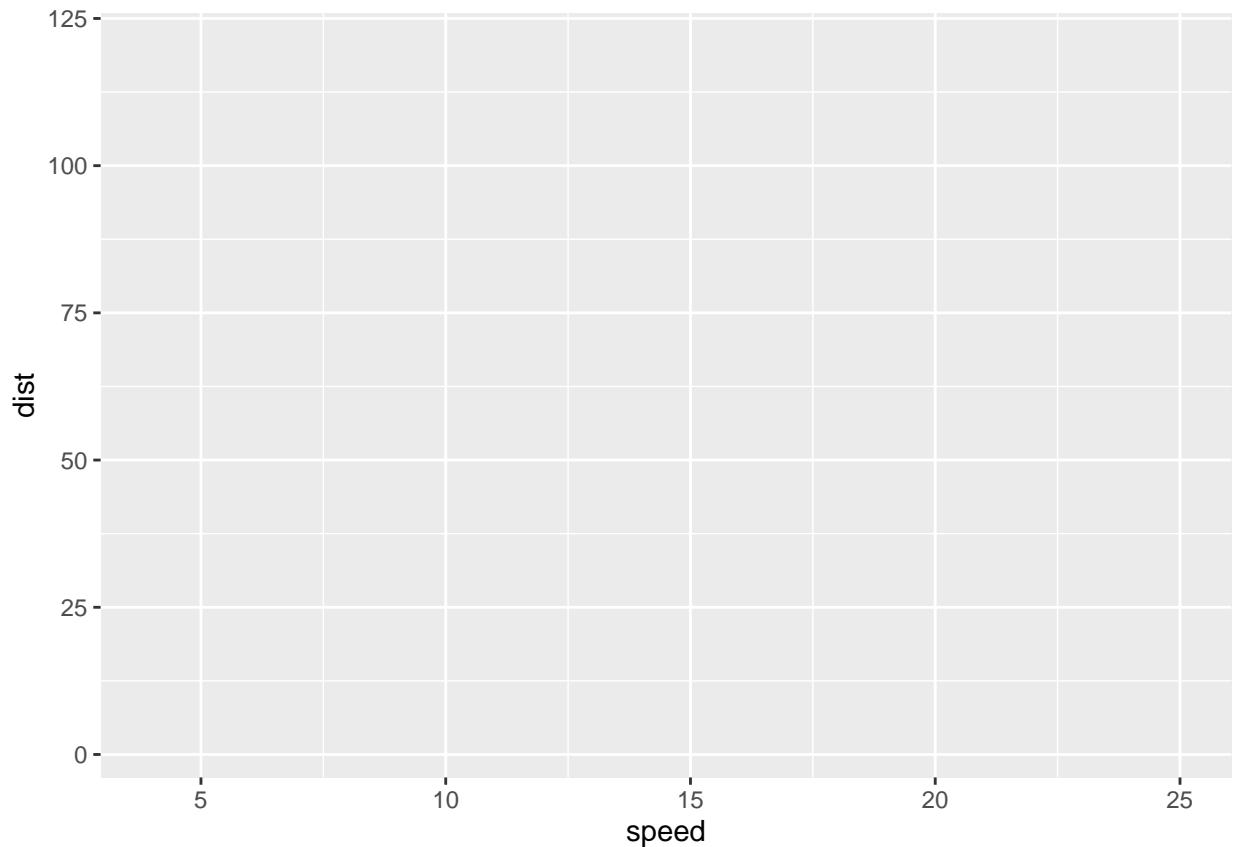


# class5.R

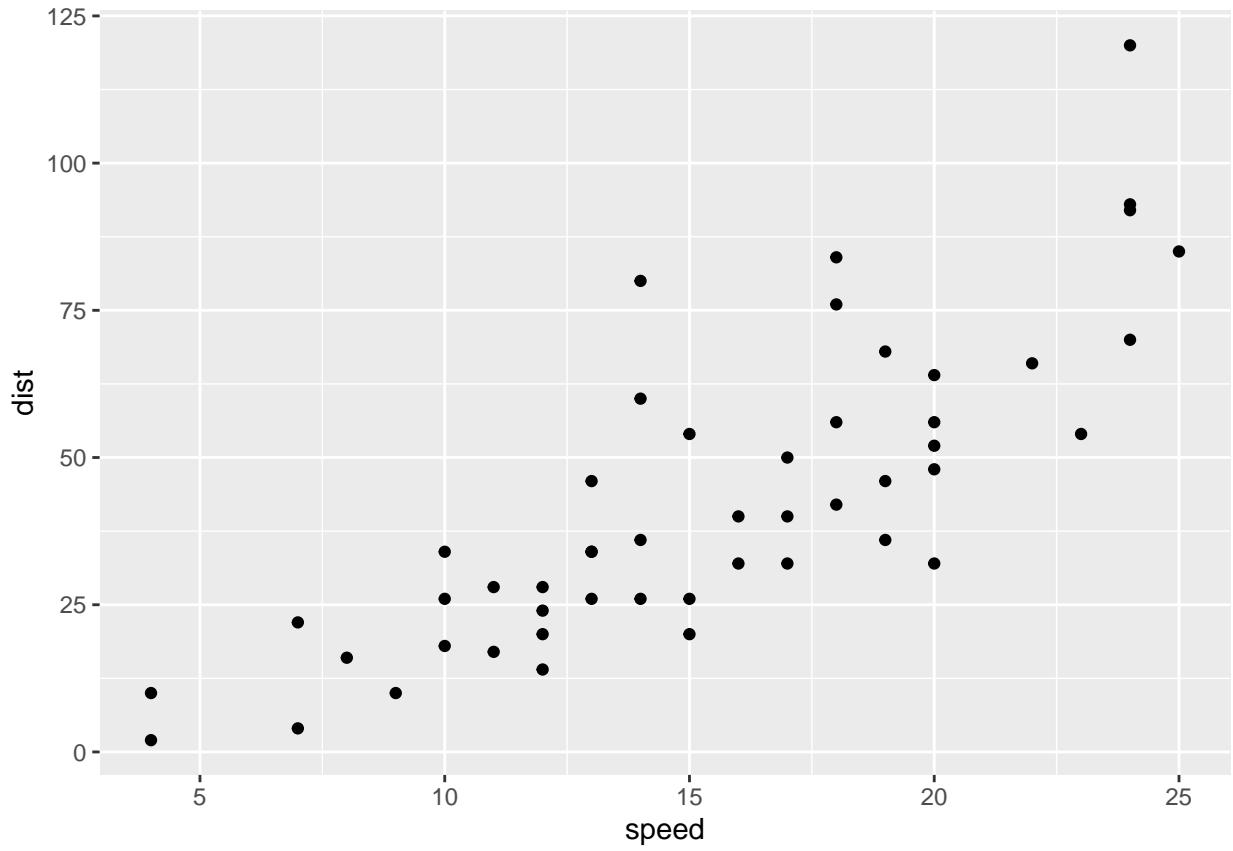
dudu

2022-02-04

```
#Class 5 ggPlot learning, always remember to library(ggplot2) first.  
library(ggplot2)  
# Open ggplot(cars) and add aes  
ggplot(cars) +  
  aes(x=speed, y=dist)
```

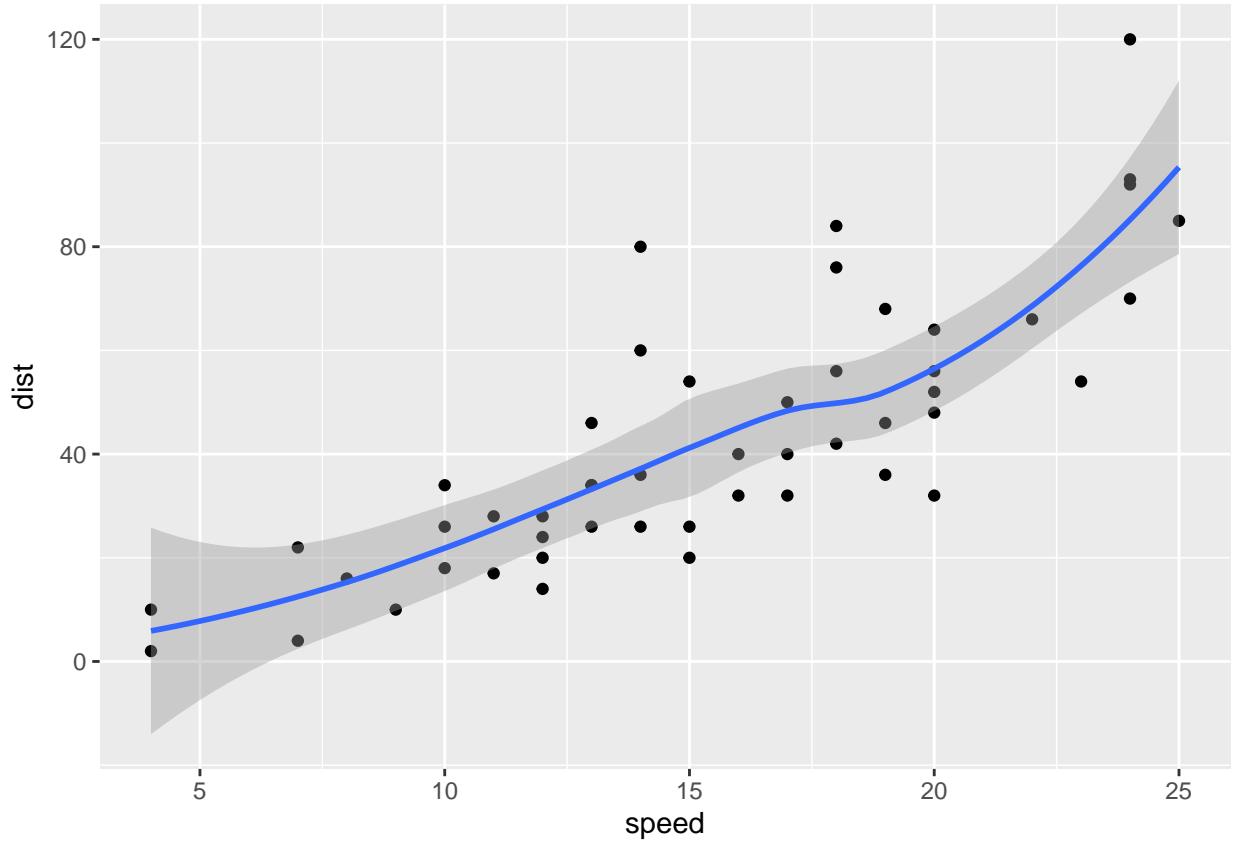


```
## we need to add one of ggplot's geometric layers (or geoms) to define how we want to visualize our data  
## We can use geom_line(), geom_col(), etc  
ggplot(cars) +  
  aes(x=speed, y=dist) +  
  geom_point()
```



```
## let's make it smooth. What is smooth? Aids the eye in seeing patterns in the presence of overplotting
ggplot(cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth()

## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```

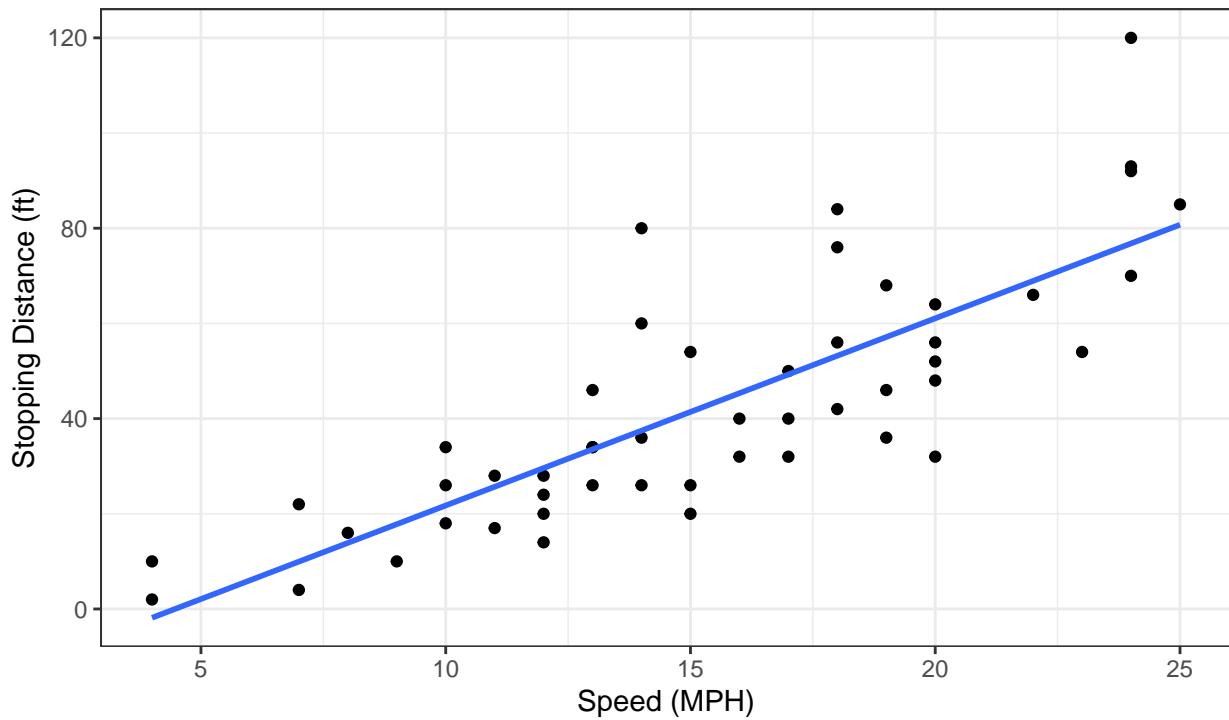


```
## add labs() function and changing the plot look to "black & white" theme by adding the theme_bw() function
ggplot(cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  labs(title="Speed and Stopping Distances of Cars",
       x="Speed (MPH)",
       y="Stopping Distance (ft)",
       subtitle = "Your informative subtitle text here",
       caption="Dataset: 'cars'") +
  geom_smooth(method="lm", se=FALSE) +
  theme_bw()

## 'geom_smooth()' using formula 'y ~ x'
```

## Speed and Stopping Distances of Cars

Your informative subtitle here



Dataset: 'cars'

```
##Adding more plot aesthetics through aes()
##Adjust the point size of a scatter plot using the size parameter.
##Change the point color of a scatter plot using the color parameter.
##Set a parameter alpha to change the transparency of all points.

#another exercise
url <- "https://bioboot.github.io/bimm143_S20/class-material/up_down_expression.txt"
genes <- read.delim(url)
head(genes)
```

```
##          Gene Condition1 Condition2      State
## 1      A4GNT -3.6808610 -3.4401355 unchanging
## 2       AAAS  4.5479580  4.3864126 unchanging
## 3      AASDH  3.7190695  3.4787276 unchanging
## 4       AATF  5.0784720  5.0151916 unchanging
## 5       AATK  0.4711421  0.5598642 unchanging
## 6 AB015752.4 -3.6808610 -3.5921390 unchanging
```

```
nrow(genes)
```

```
## [1] 5196
```

```
colnames(genes)
```

```
## [1] "Gene"          "Condition1" "Condition2" "State"  
ncol(genes)
```

```
## [1] 4
```

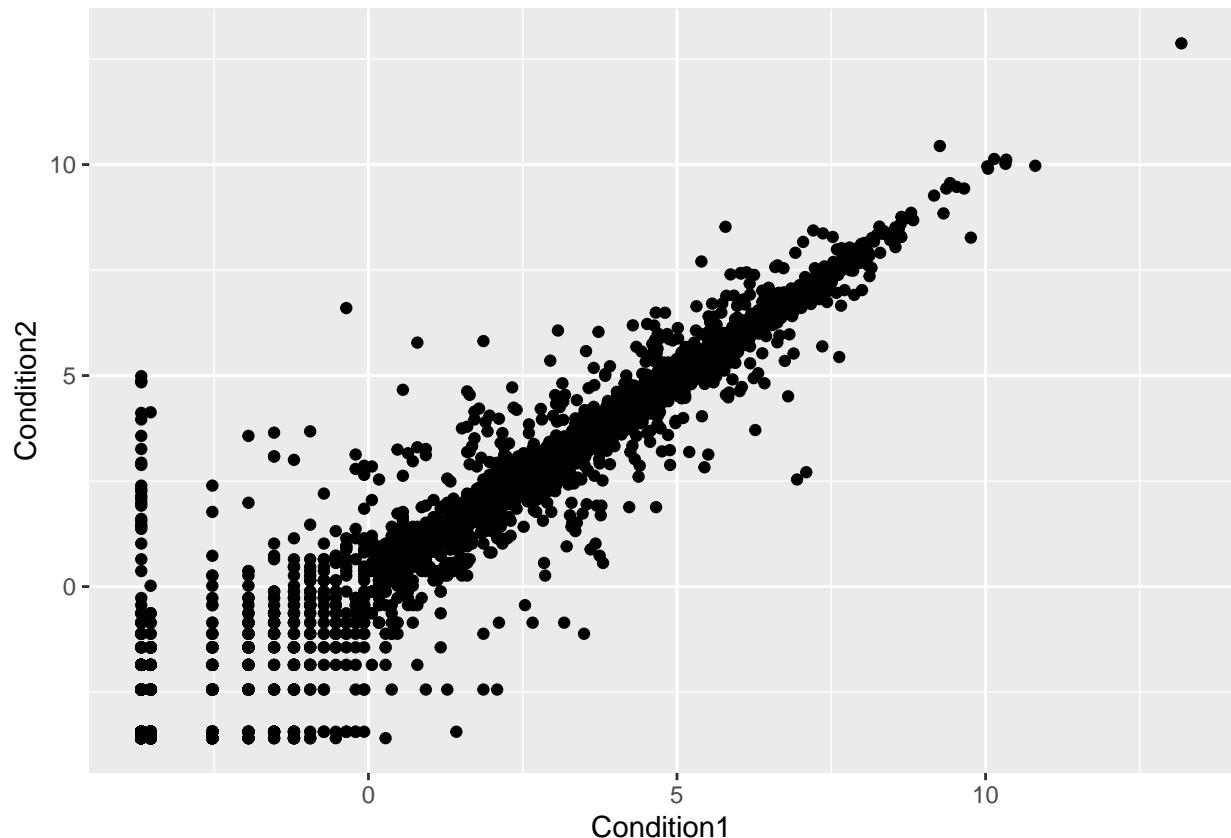
```
## table the state in the genes database  
table(genes$State)
```

```
##  
##      down unchanged      up  
##        72       4997     127
```

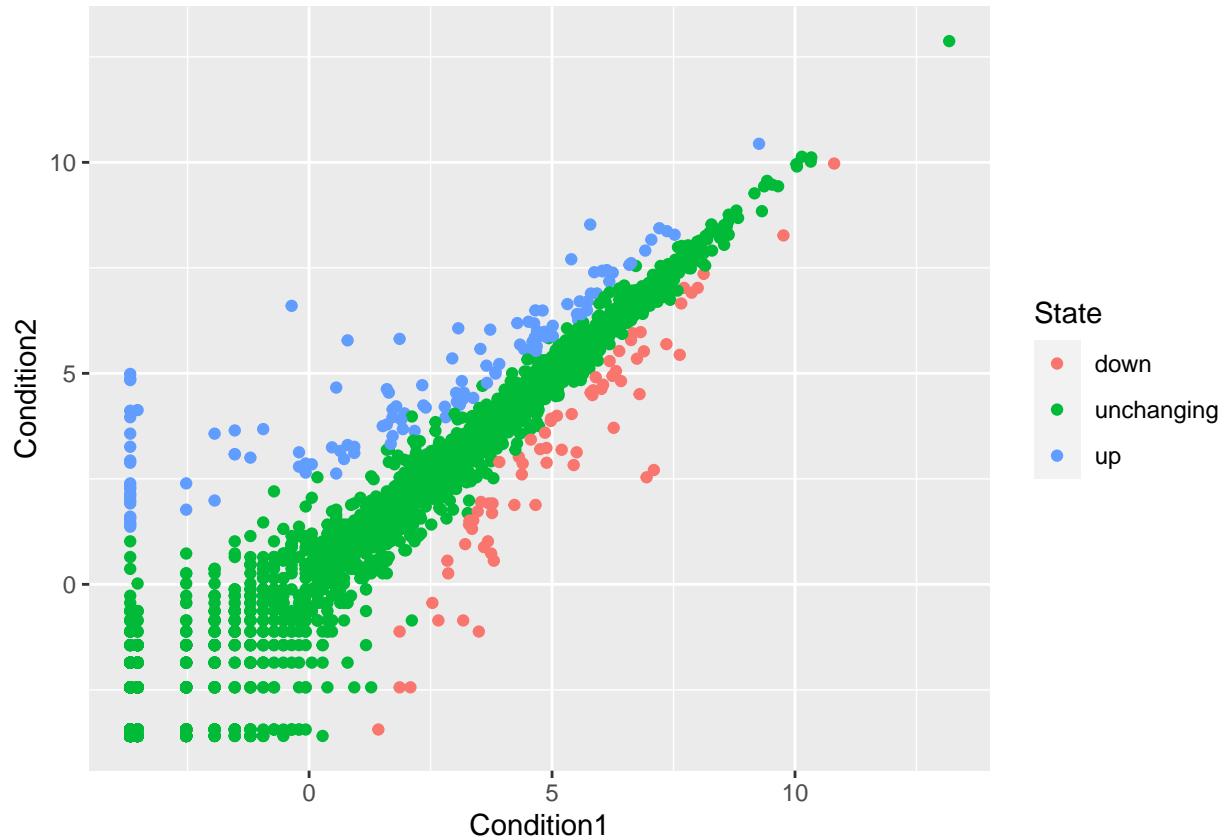
```
## the round( ,2 ) here means 2 digital after the decimal point  
round( table(genes$State)/nrow(genes) * 100, 2 )
```

```
##  
##      down unchanged      up  
##        1.39       96.17     2.44
```

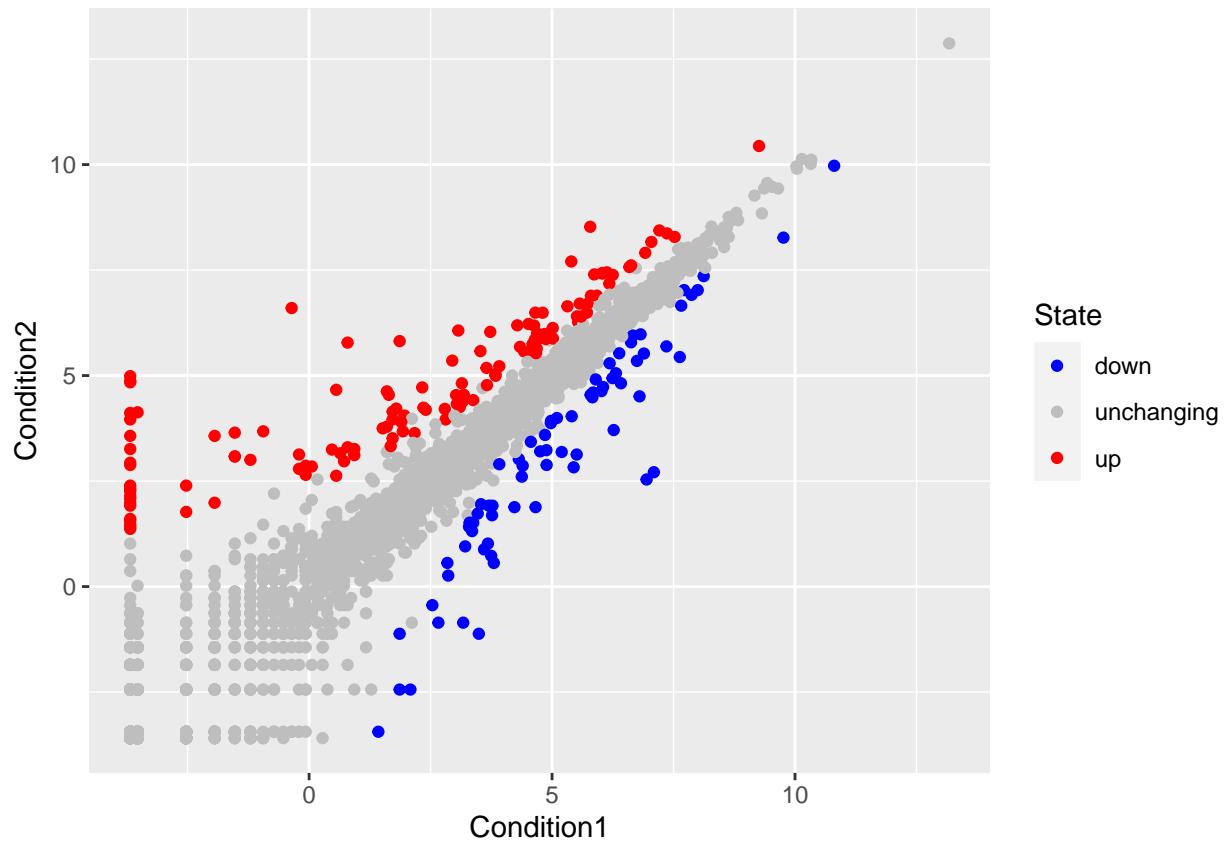
```
## make a basic plot first  
ggplot(genes) +  
  aes(x=Condition1, y=Condition2) +  
  geom_point()
```



```
## name this p, and add color for the dots according to the State
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point()
p
```

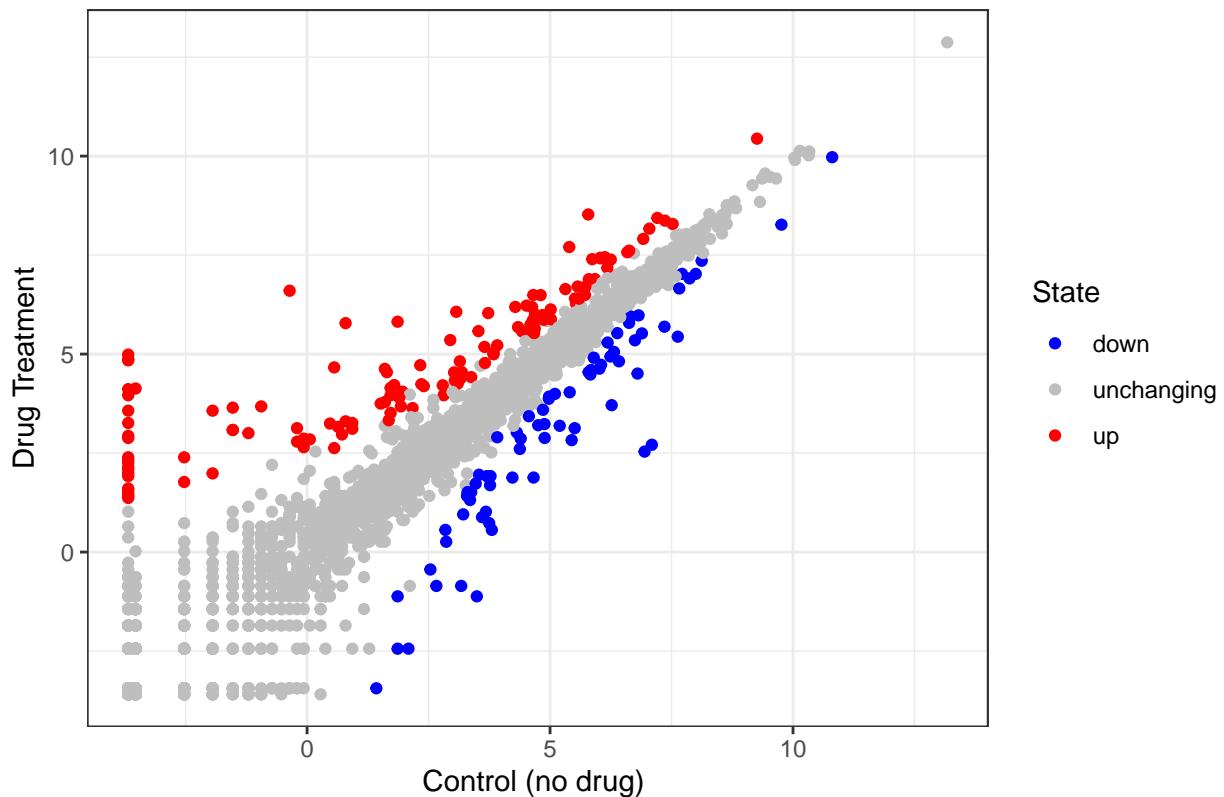


```
##change color for the dots manually
p + scale_colour_manual( values=c("blue","gray","red") )
```



```
## add labels and change color of theme
p + scale_colour_manual(values=c("blue","gray","red")) +
  labs(title="Gene Expression Changes Upon Drug Treatment",
       x="Control (no drug) ",
       y="Drug Treatment") + theme_bw()
```

## Gene Expression Changes Upon Drug Treatment



```
# what is more??
library(gapminder)
# only want 2007?
library(dplyr)

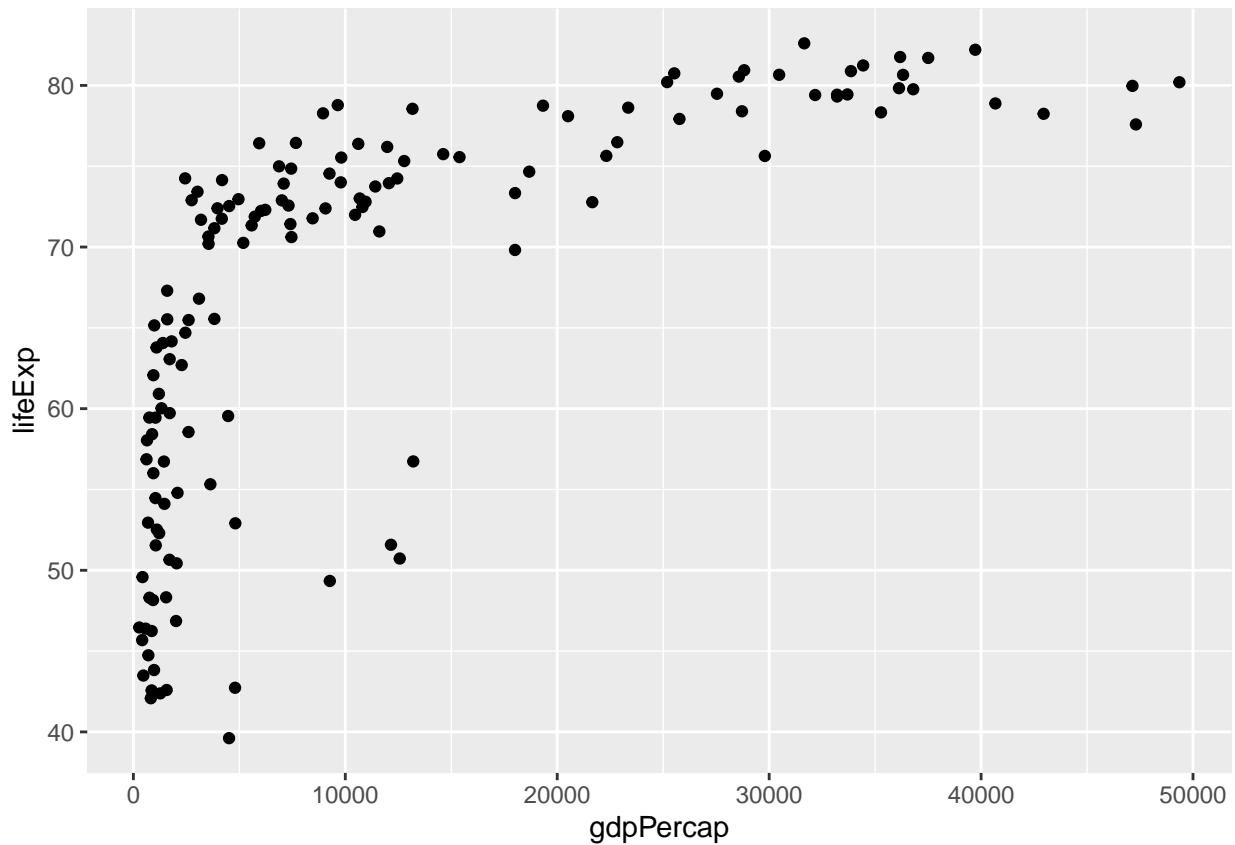
##
## Attaching package: 'dplyr'

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

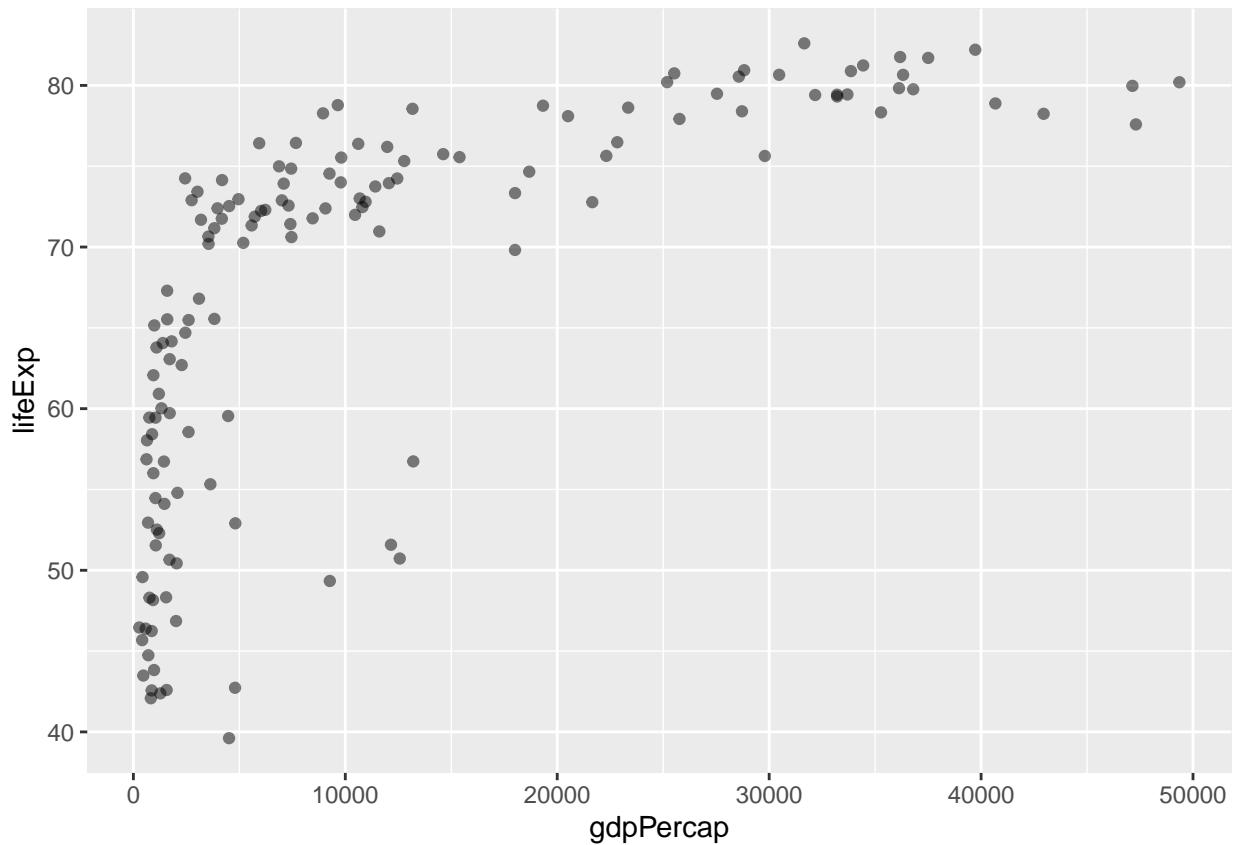
## The following objects are masked from 'package:base':
## intersect, setdiff, setequal, union

gapminder_2007 <- gapminder %>% filter(year==2007)

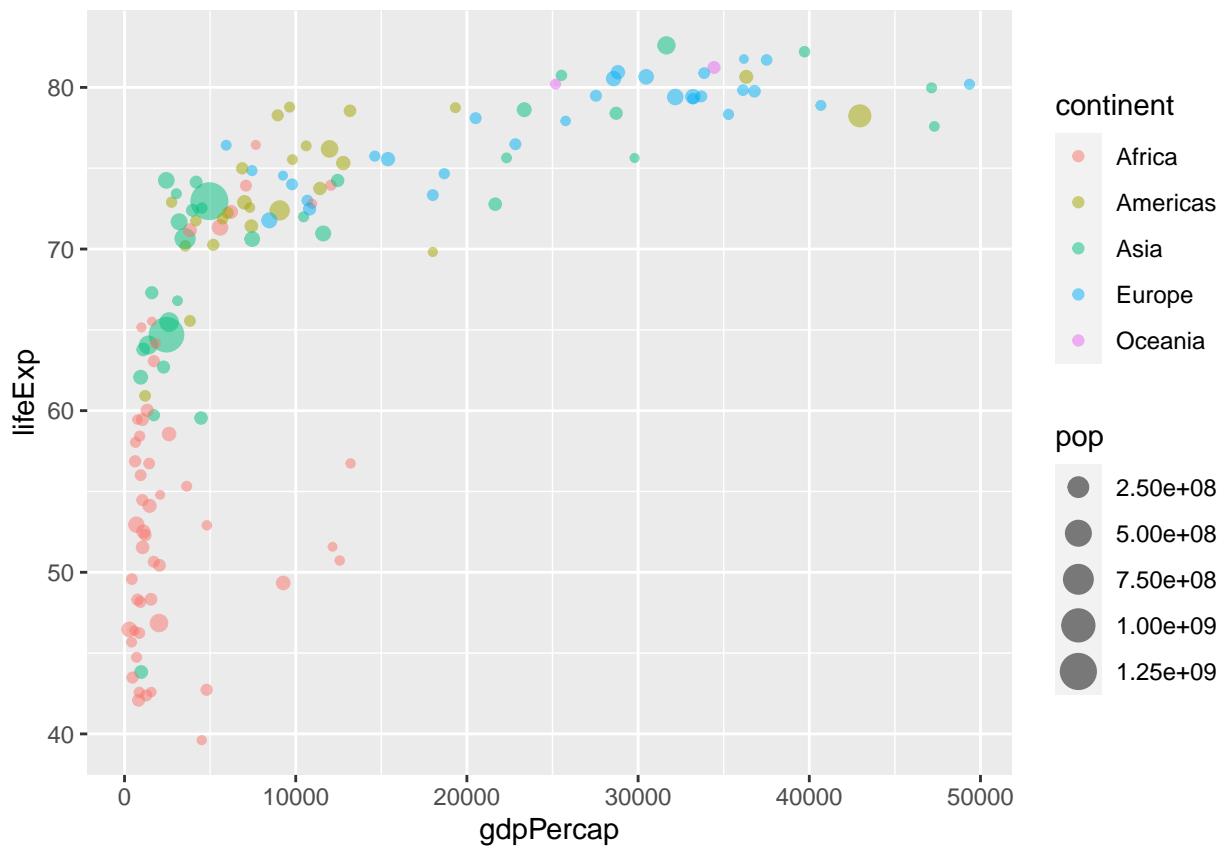
#Let's consider the gapminder_2007 dataset which contains the variables GDP per capita gdpPercap and life expectancy lifeExp
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp) +
  geom_point()
```



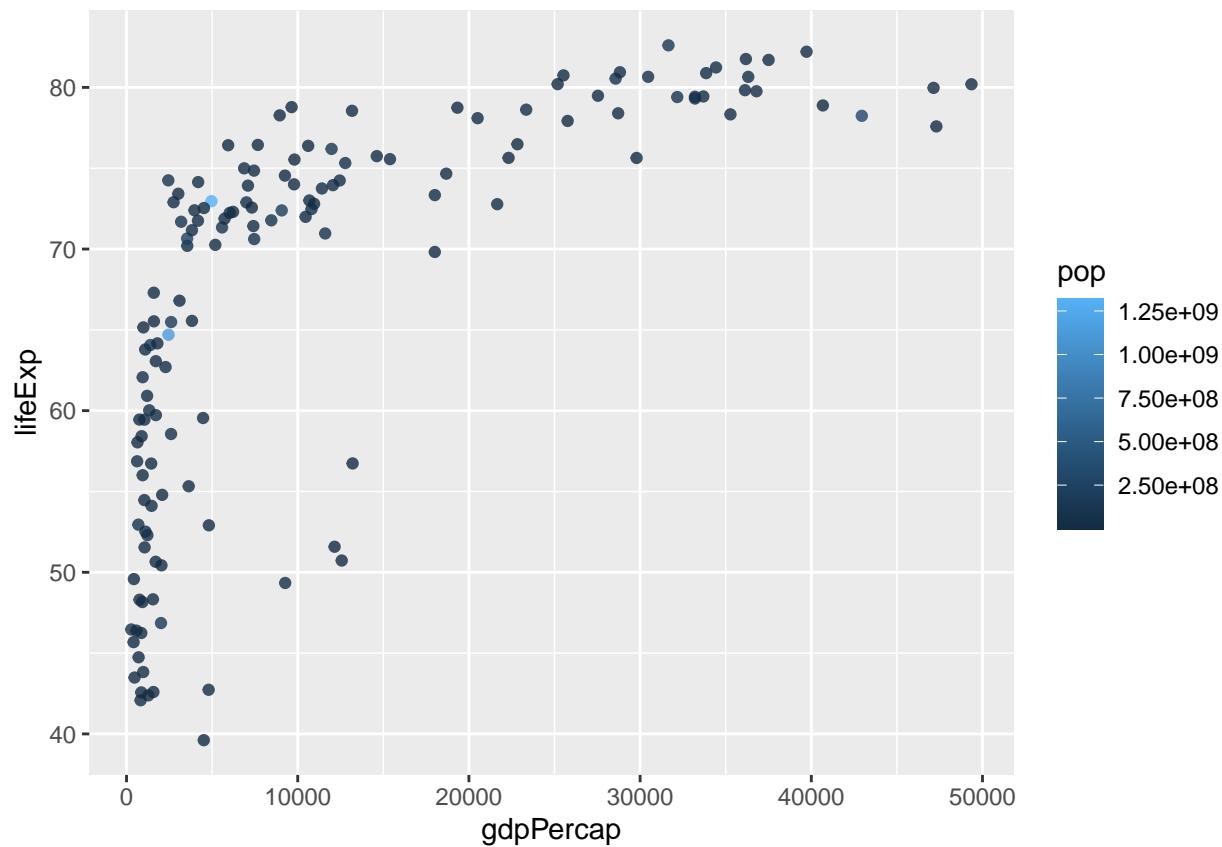
```
#on the top of each other? no worries! One useful approach here is to add an alpha=0.4 argument to your  
ggplot(gapminder_2007) +  
  aes(x=gdpPercap, y=lifeExp) +  
  geom_point(alpha=0.5)
```



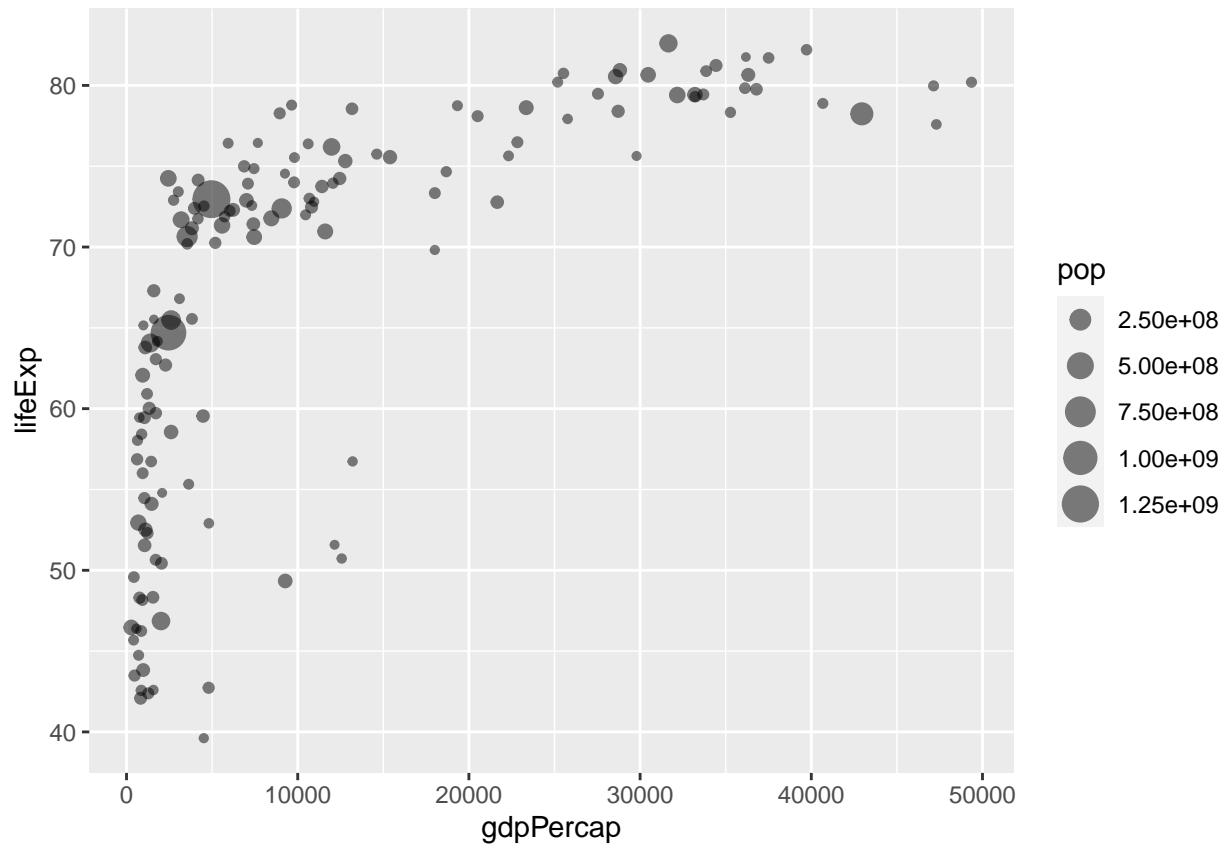
```
# add the population pop (in millions) through the point size argument to aes() we can obtain a much ri
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  geom_point(alpha=0.5)
```



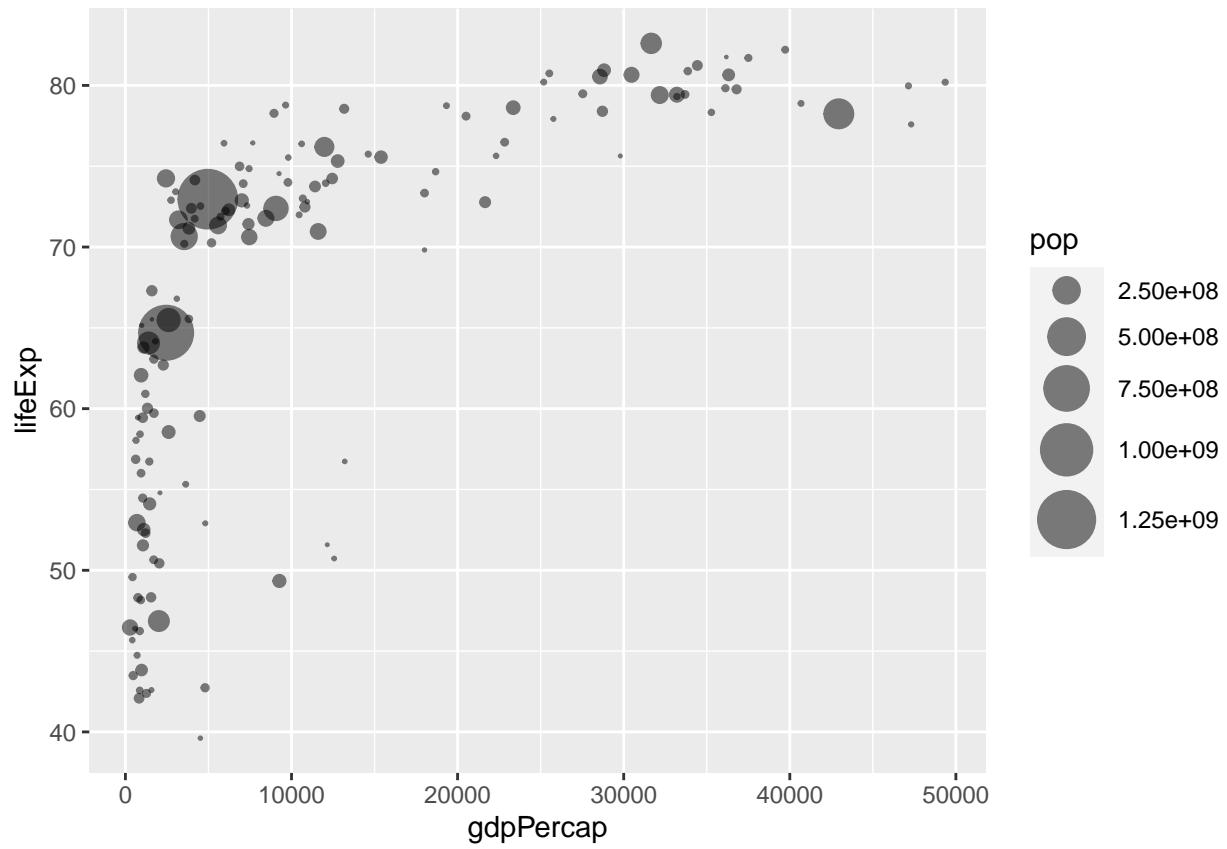
```
# we also can color the dots by the numeric variable population pop
ggplot(gapminder_2007) +
  aes(x = gdpPercap, y = lifeExp, color = pop) +
  geom_point(alpha=0.8)
```



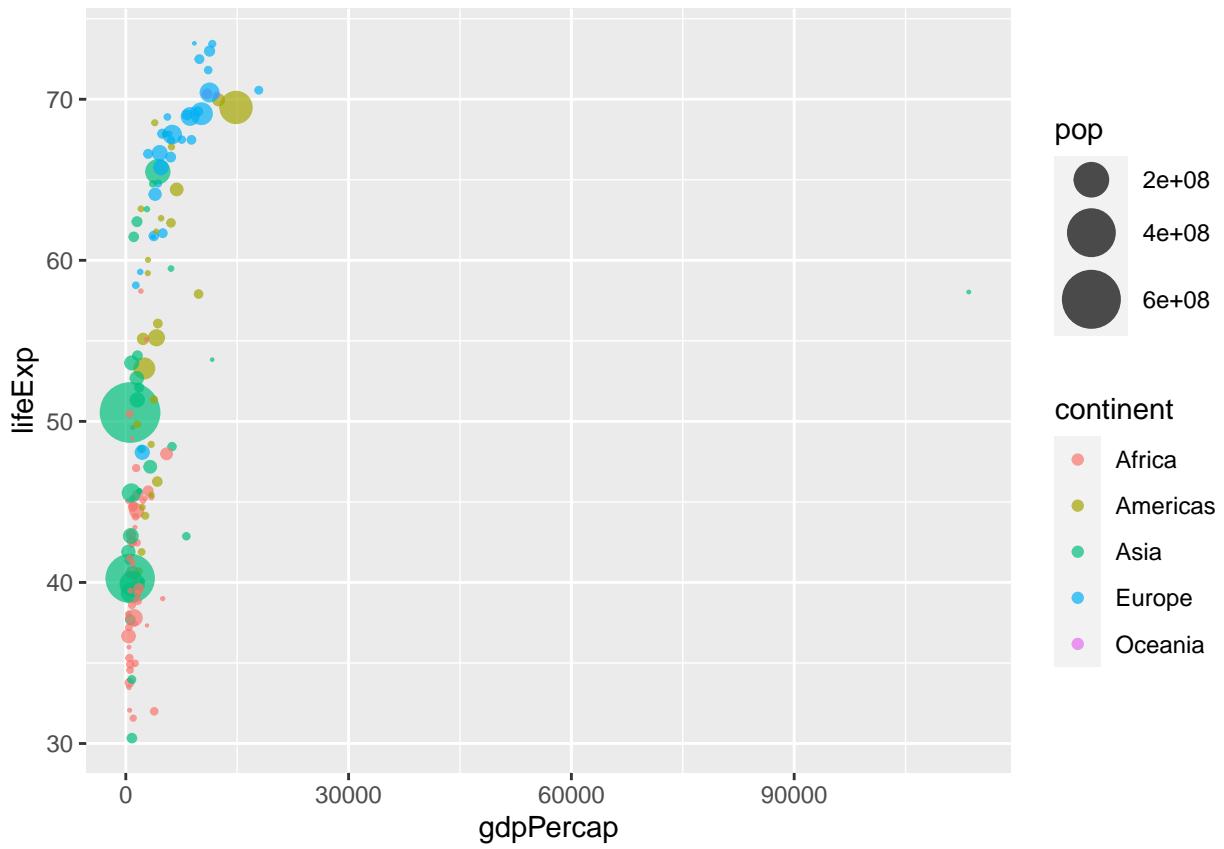
```
#I do not like using color to indicate the population in this case. let me change back to using the size
ggplot(gapminder_2007) +
  aes(x = gdpPercap, y = lifeExp, size = pop) +
  geom_point(alpha=0.5)
```



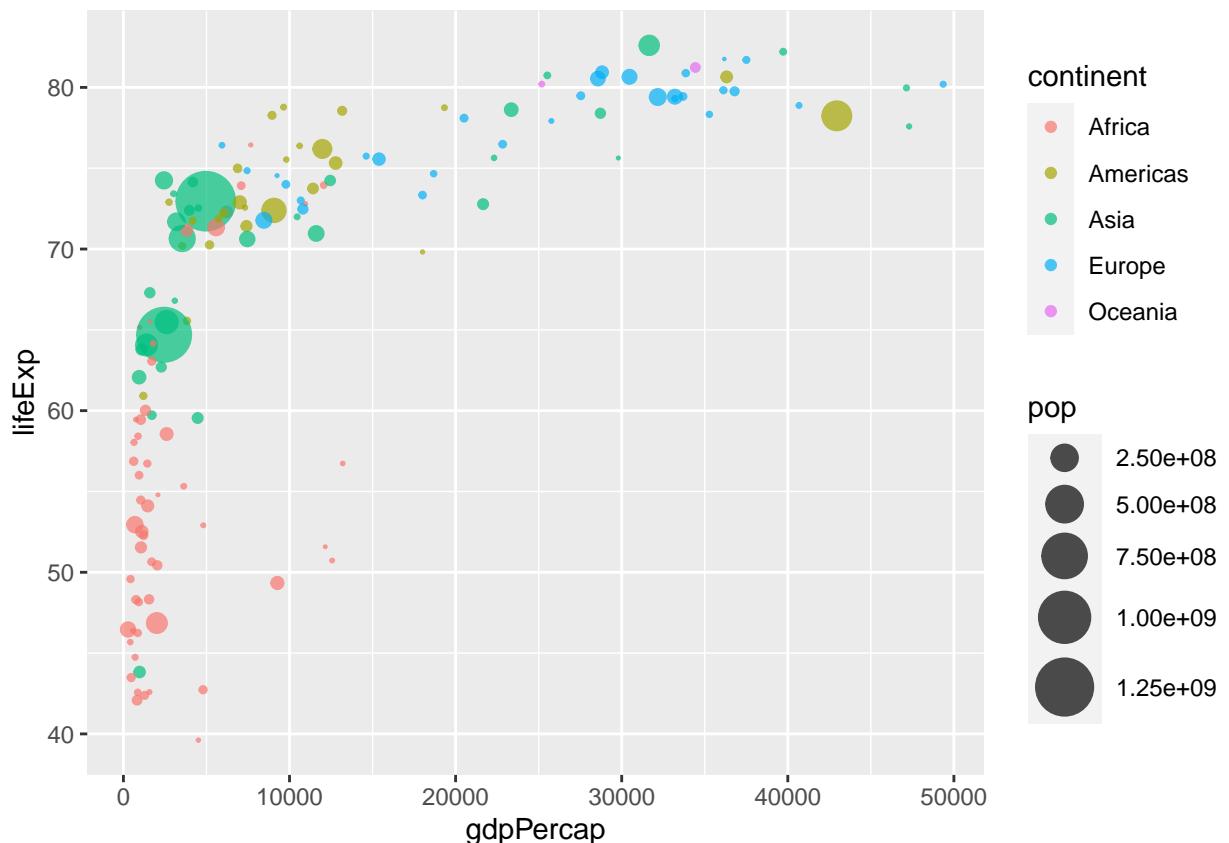
```
# This is not really reflecting the real size. let us use scale_size_area() function
ggplot(gapminder_2007) +
  geom_point(aes(x = gdpPercap, y = lifeExp,
                 size = pop), alpha=0.5) +
  scale_size_area(max_size = 10)
```



```
# let me make one for 1957
gapminder_1957 <- gapminder %>% filter(year==1957)
ggplot(gapminder_1957) +
  geom_point(aes(x = gdpPercap, y = lifeExp, color=continent,
                 size = pop), alpha=0.7) +
  scale_size_area(max_size = 10)
```

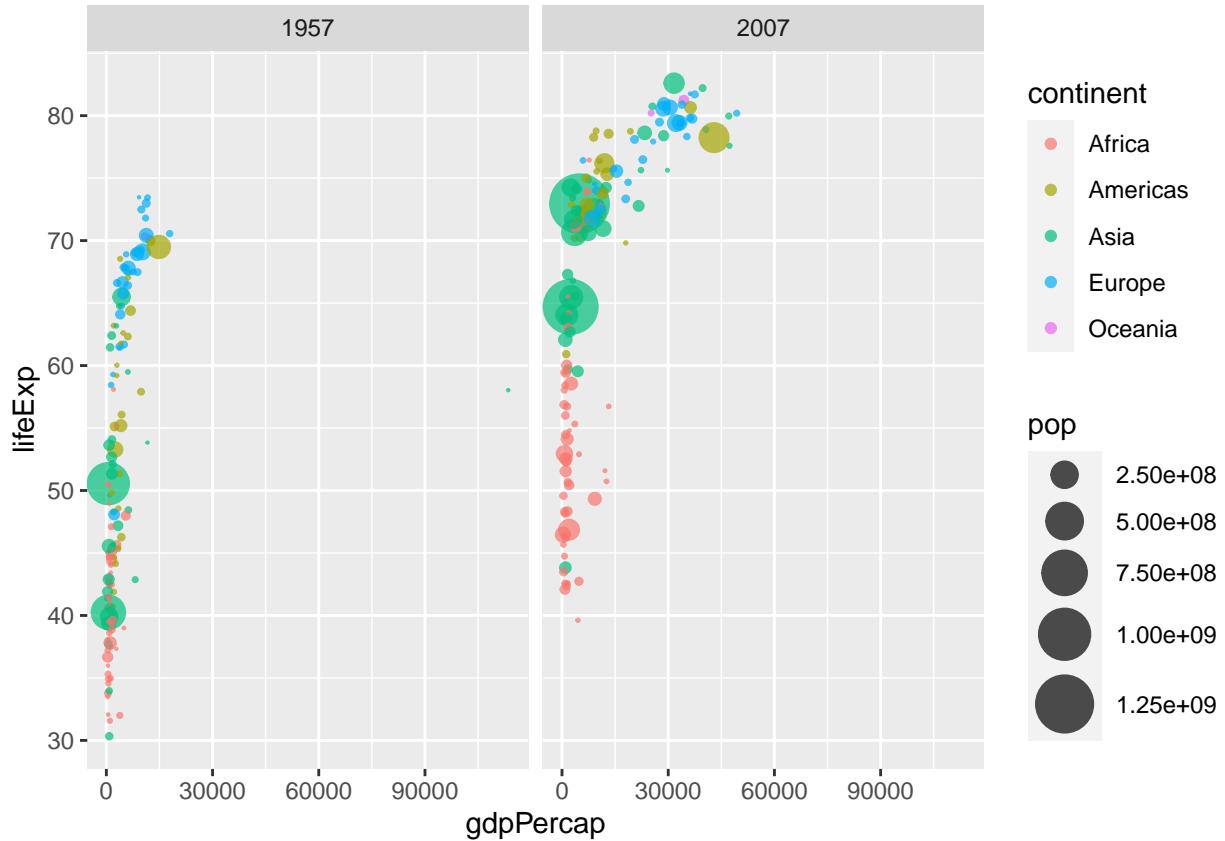


```
gapminder_top5 <- gapminder %>% filter(year==2007)
ggplot(gapminder_2007) +
  geom_point(aes(x = gdpPercap, y = lifeExp, color=continent,
                 size = pop), alpha=0.7) +
  scale_size_area(max_size = 10)
```



```
# How to put them together?? multi-panel plots in R using facet_wrap()
gapminder_1957 <- gapminder %>% filter(year==1957 | year==2007)

ggplot(gapminder_1957) +
  geom_point(aes(x = gdpPercap, y = lifeExp, color=continent,
                 size = pop), alpha=0.7) +
  scale_size_area(max_size = 10) +
  facet_wrap(~year)
```



```
# ADVANCE!!
install.packages("gifski", repos="http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
## /var/folders/9d/xssg21015fq5rb8769f22wfw0000gn/T//RtmpFFM1cS/downloaded_packages
```

```
install.packages("gganimate", repos="http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
## /var/folders/9d/xssg21015fq5rb8769f22wfw0000gn/T//RtmpFFM1cS/downloaded_packages
```

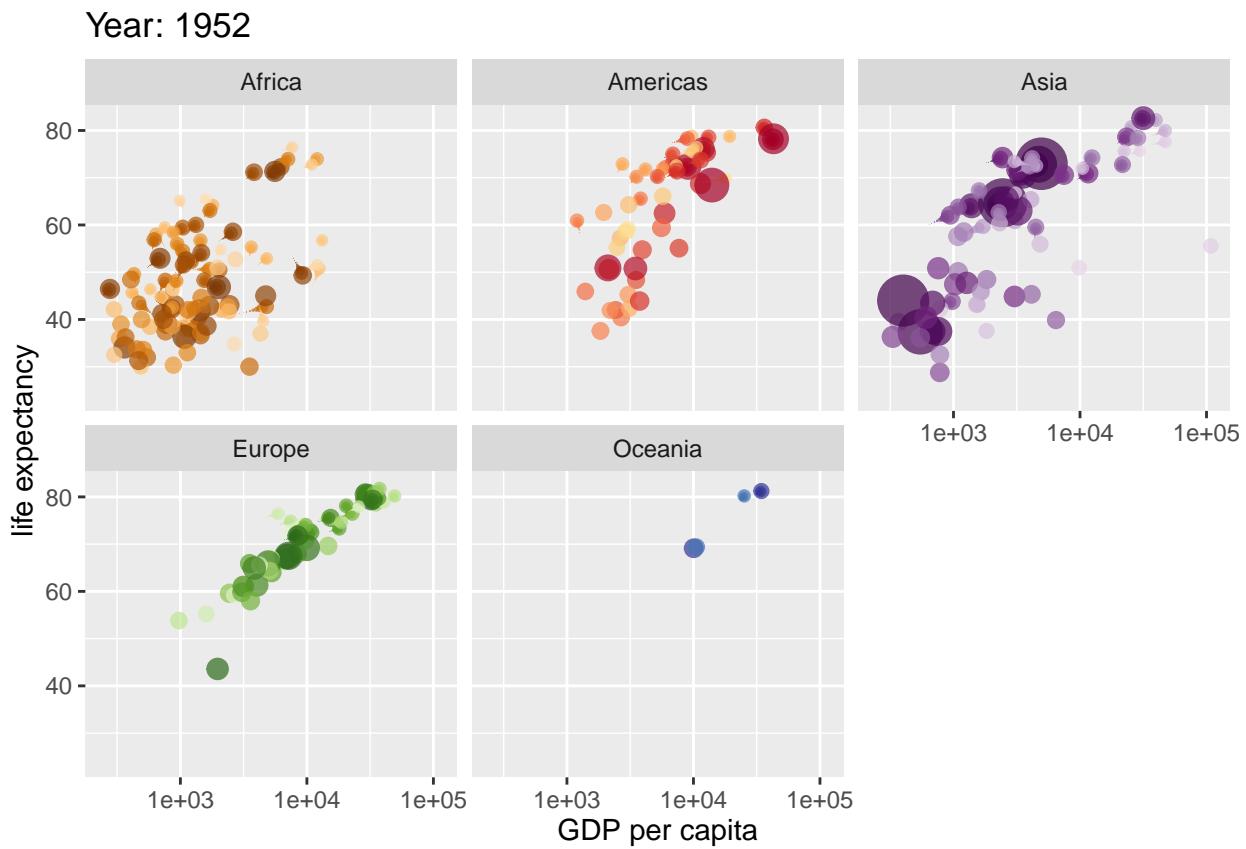
```
library(ggplot2)
library(gapminder)
library(gganimate)

# Setup nice regular ggplot of the gapminder data
ggplot(gapminder, aes(gdpPercap, lifeExp, size = pop, colour = country)) +
  geom_point(alpha = 0.7, show.legend = FALSE) +
  scale_colour_manual(values = country_colors) +
  scale_size(range = c(2, 12)) +
  scale_x_log10() +
# Facet by continent
```

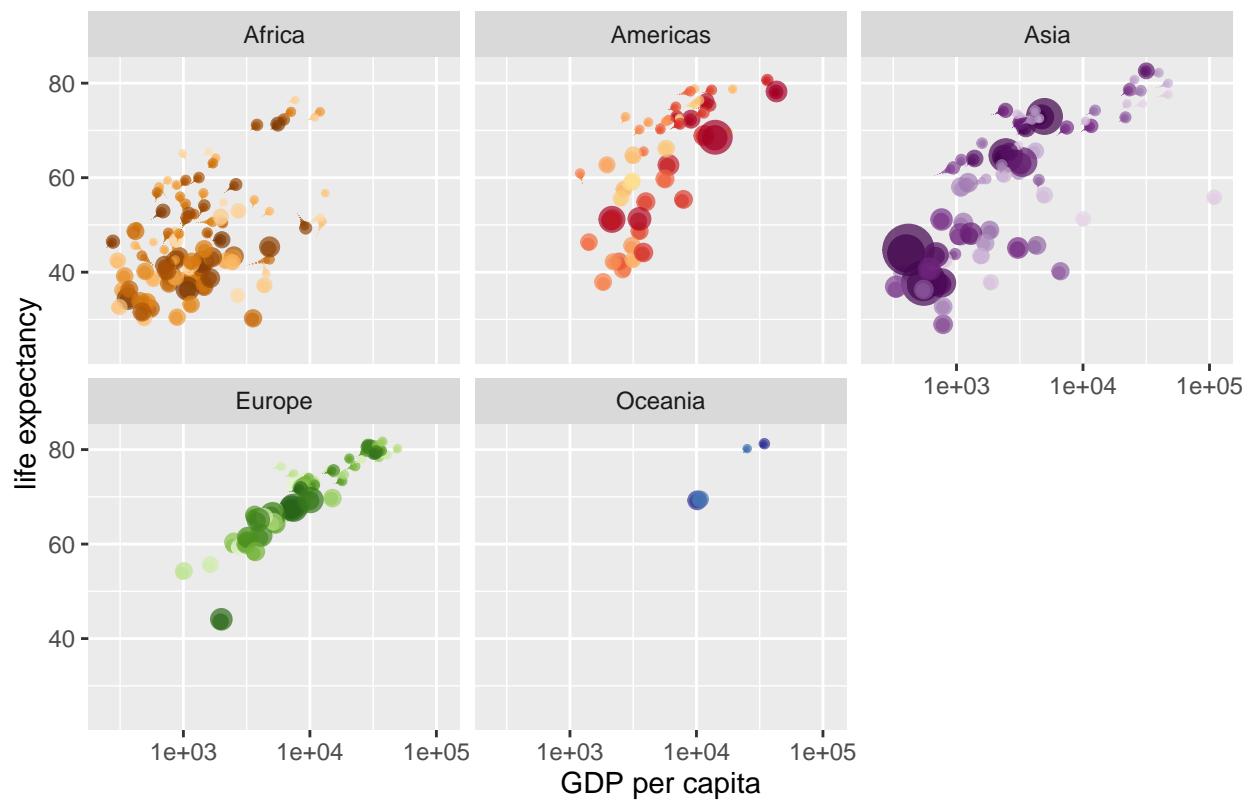
```

facet_wrap(~continent) +
# Here comes the gganimate specific bits
labs(title = 'Year: {frame_time}', x = 'GDP per capita', y = 'life expectancy') +
transition_time(year) +
shadow_wake(wake_length = 0.1, alpha = FALSE)

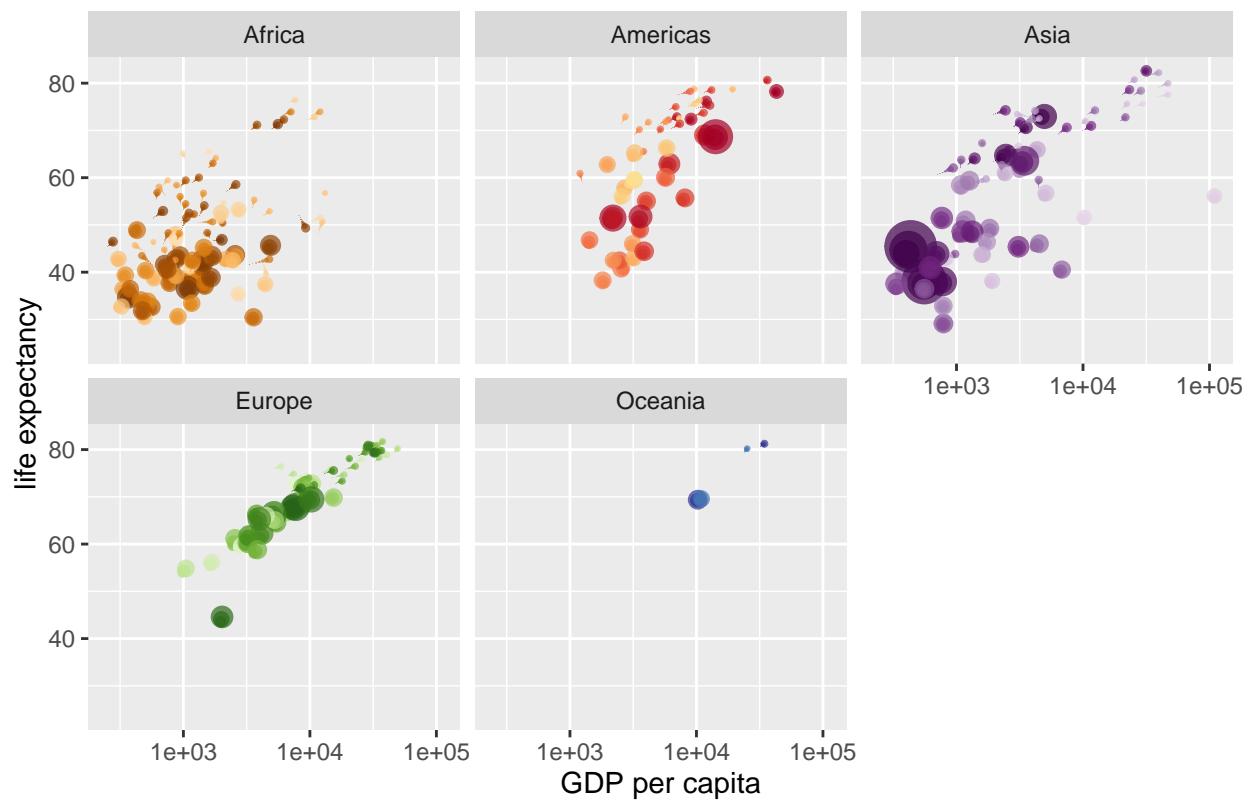
```



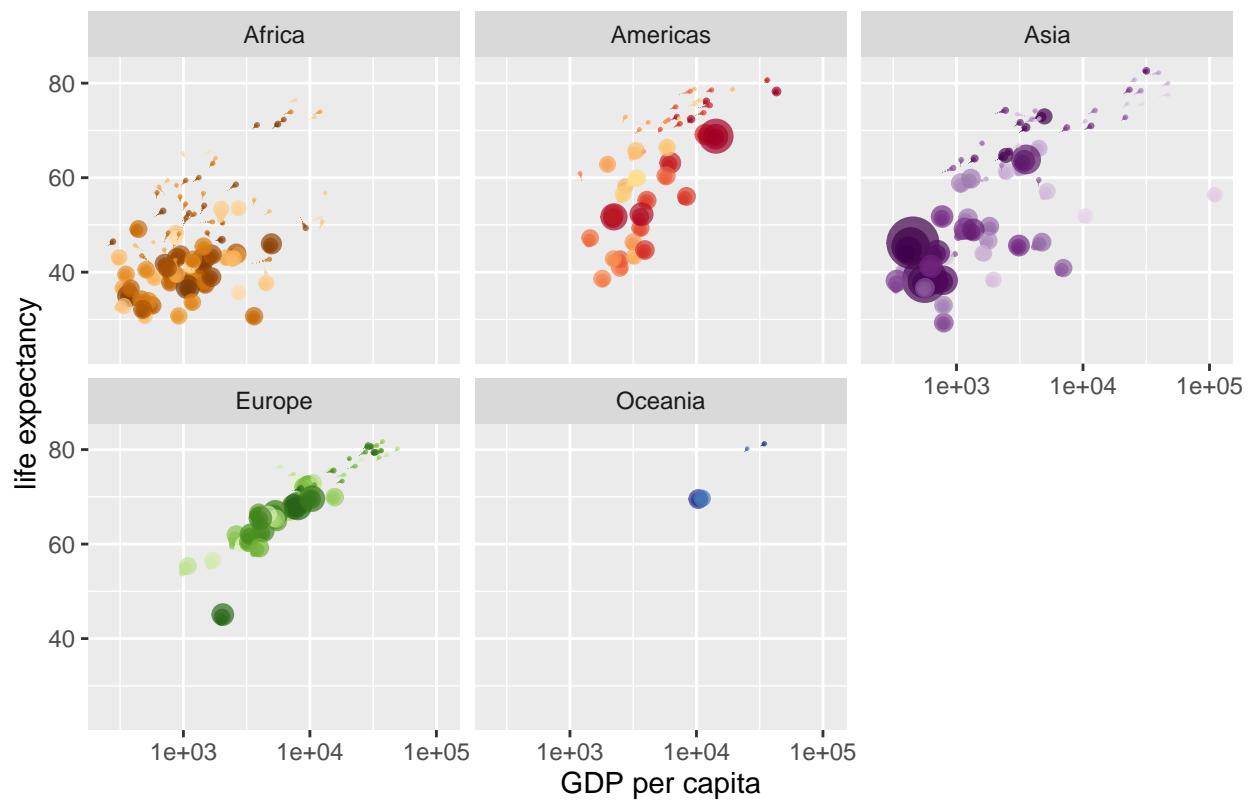
Year: 1953



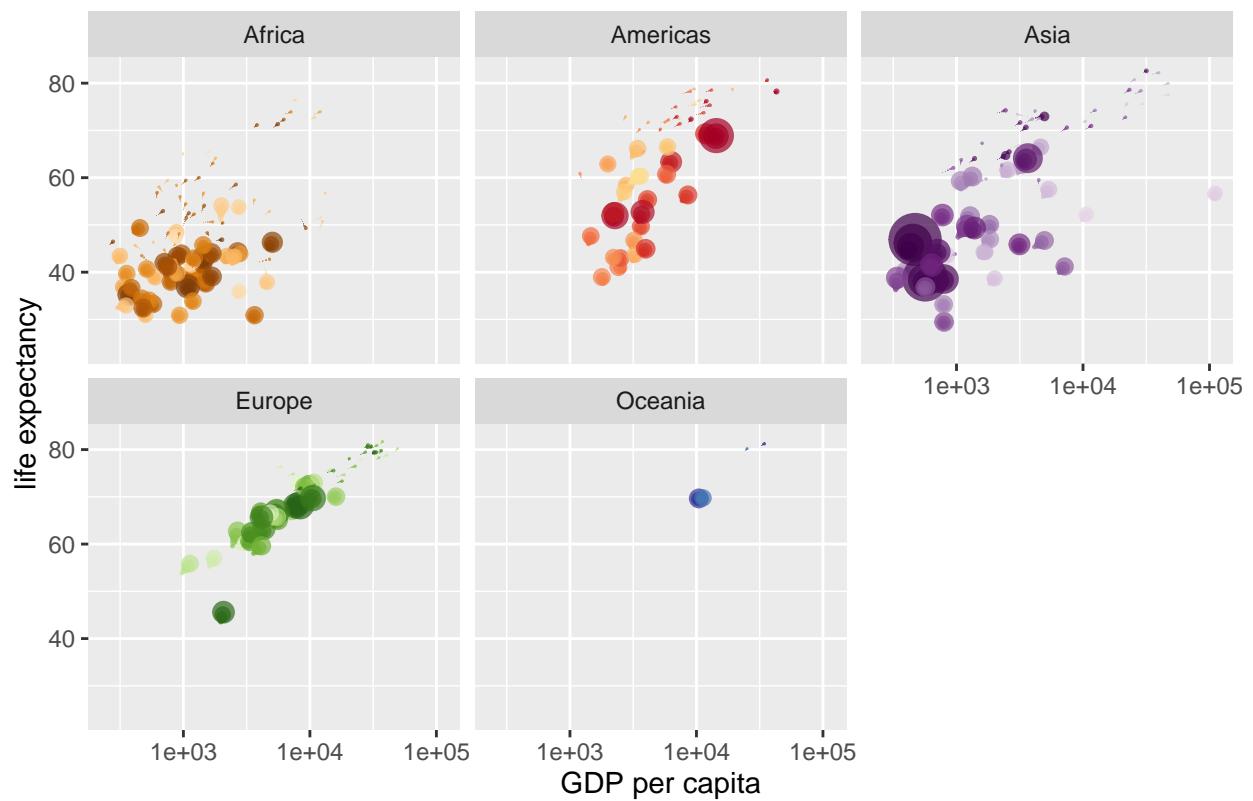
Year: 1953



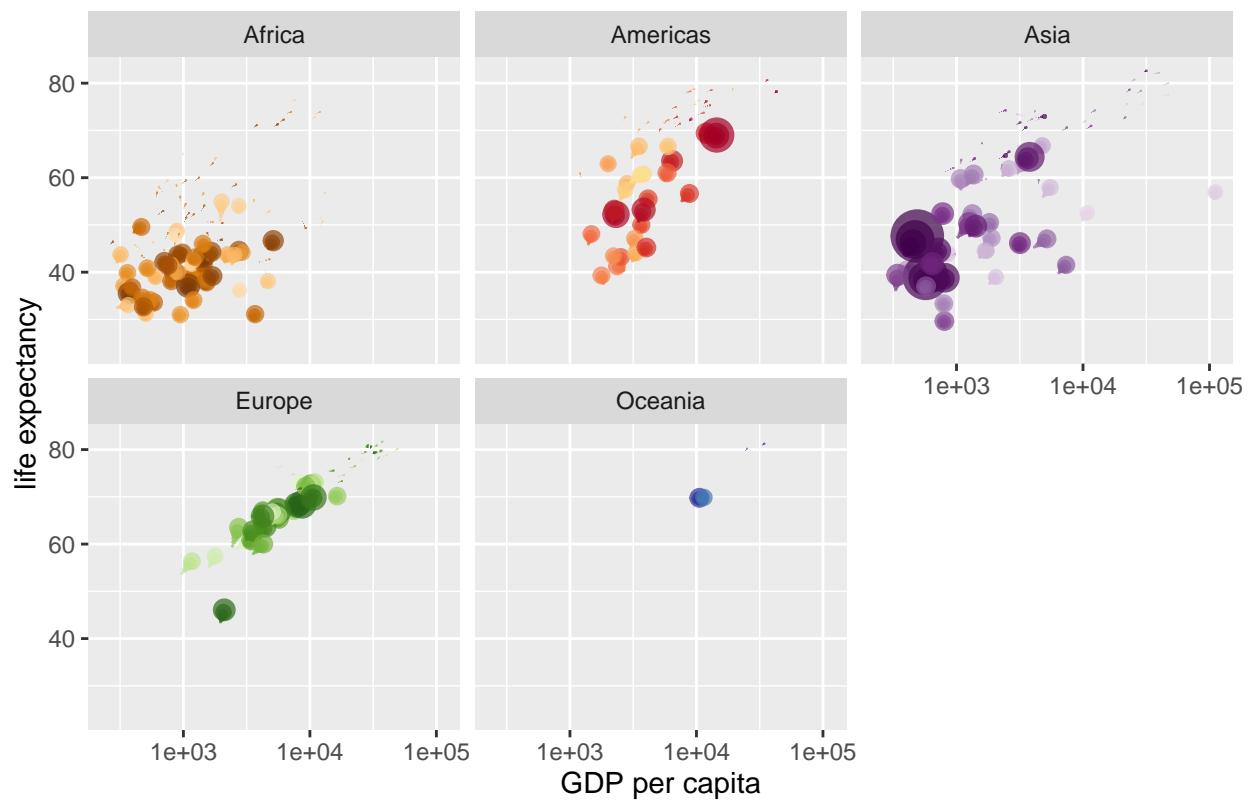
Year: 1954



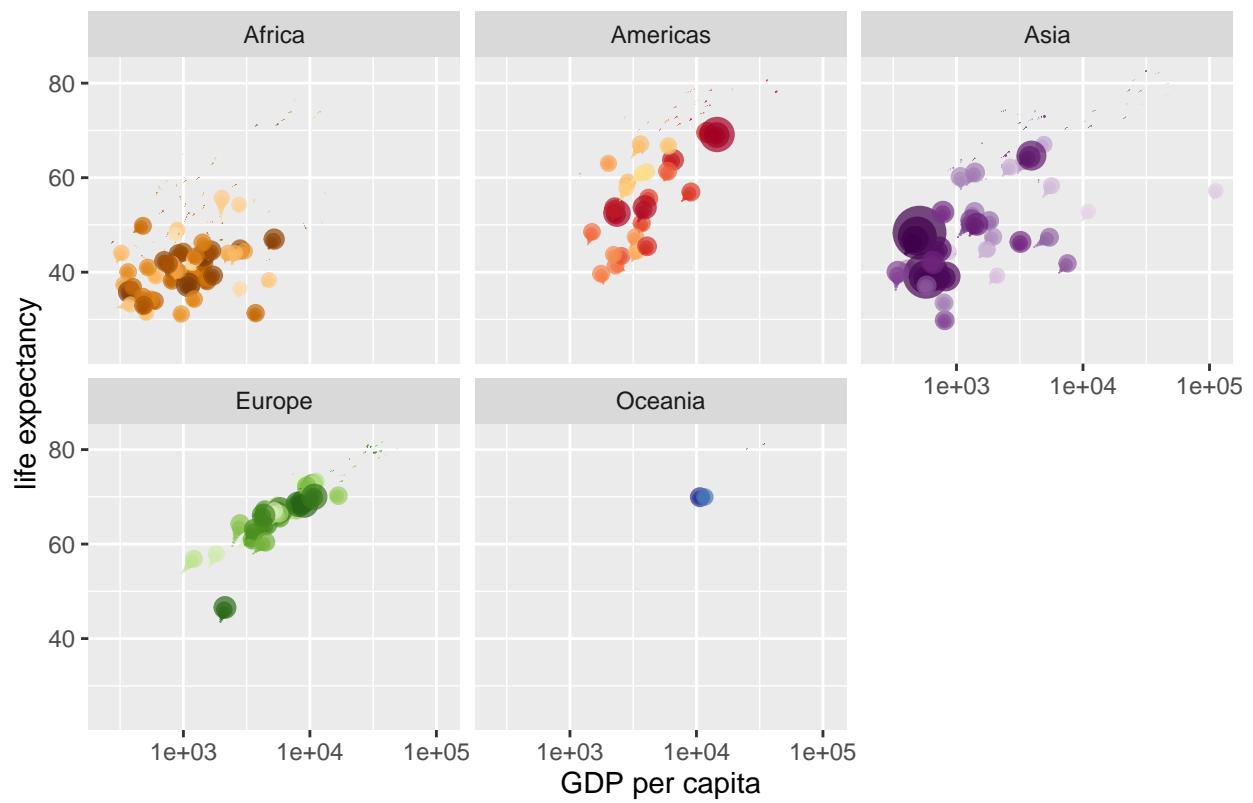
Year: 1954



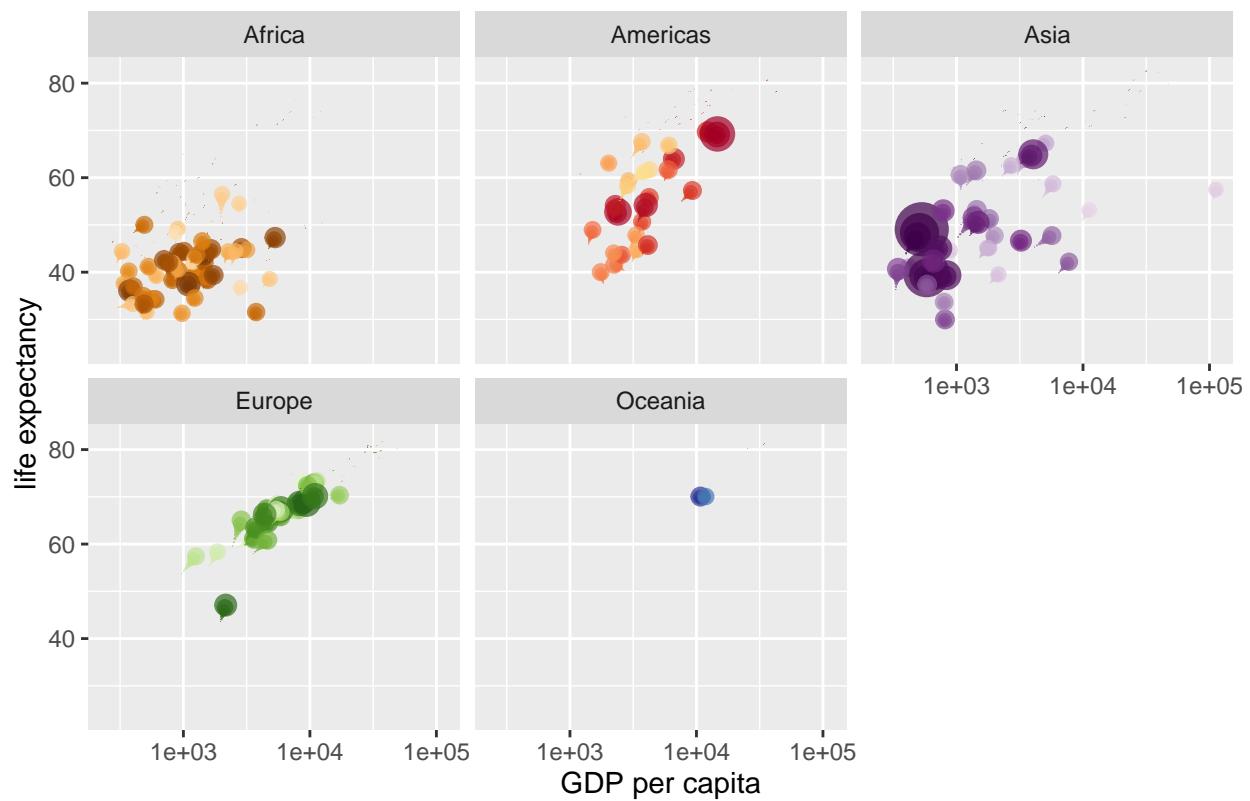
Year: 1955



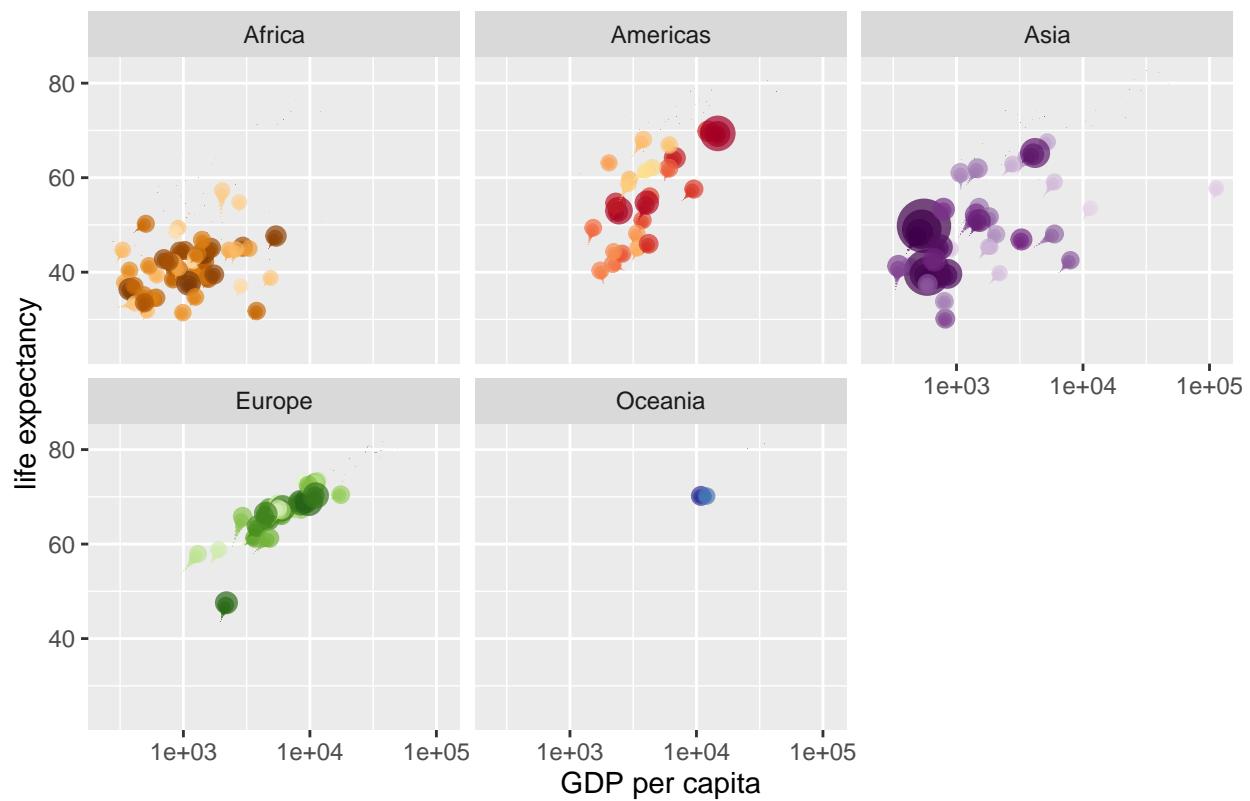
Year: 1955



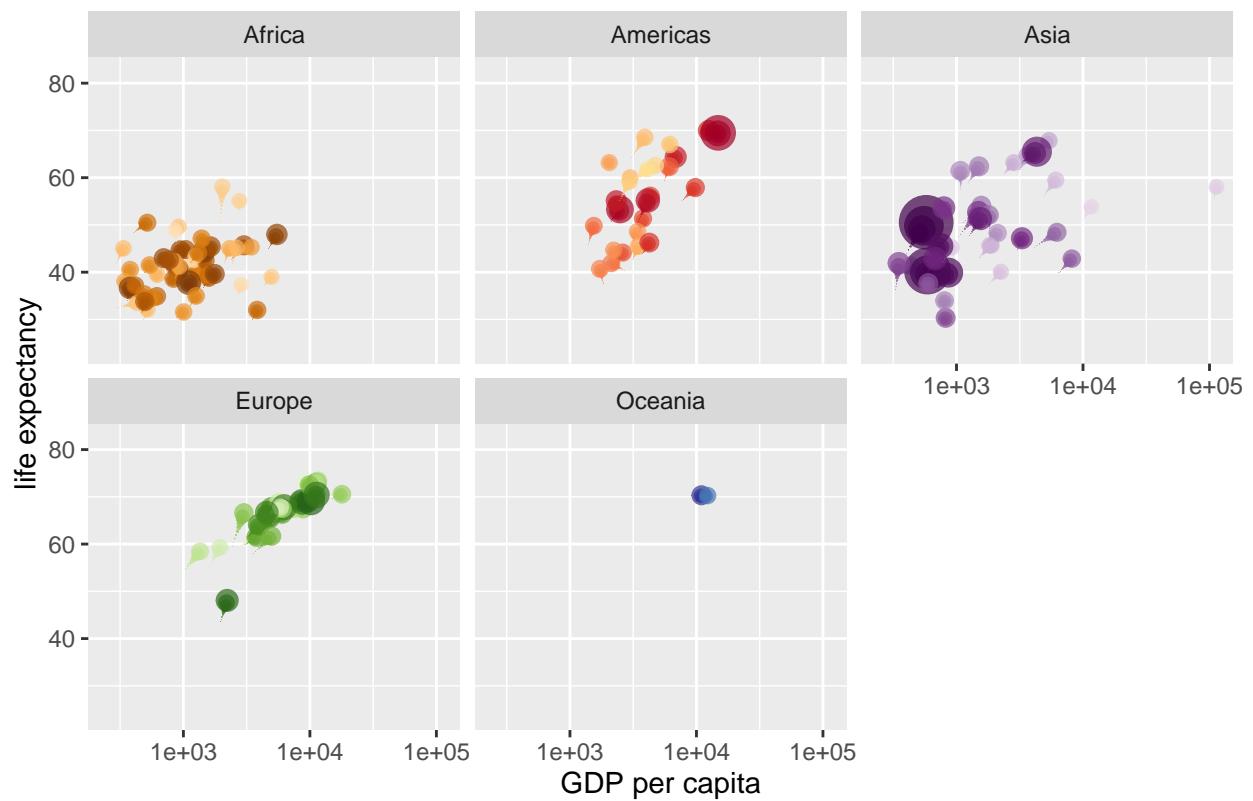
Year: 1956



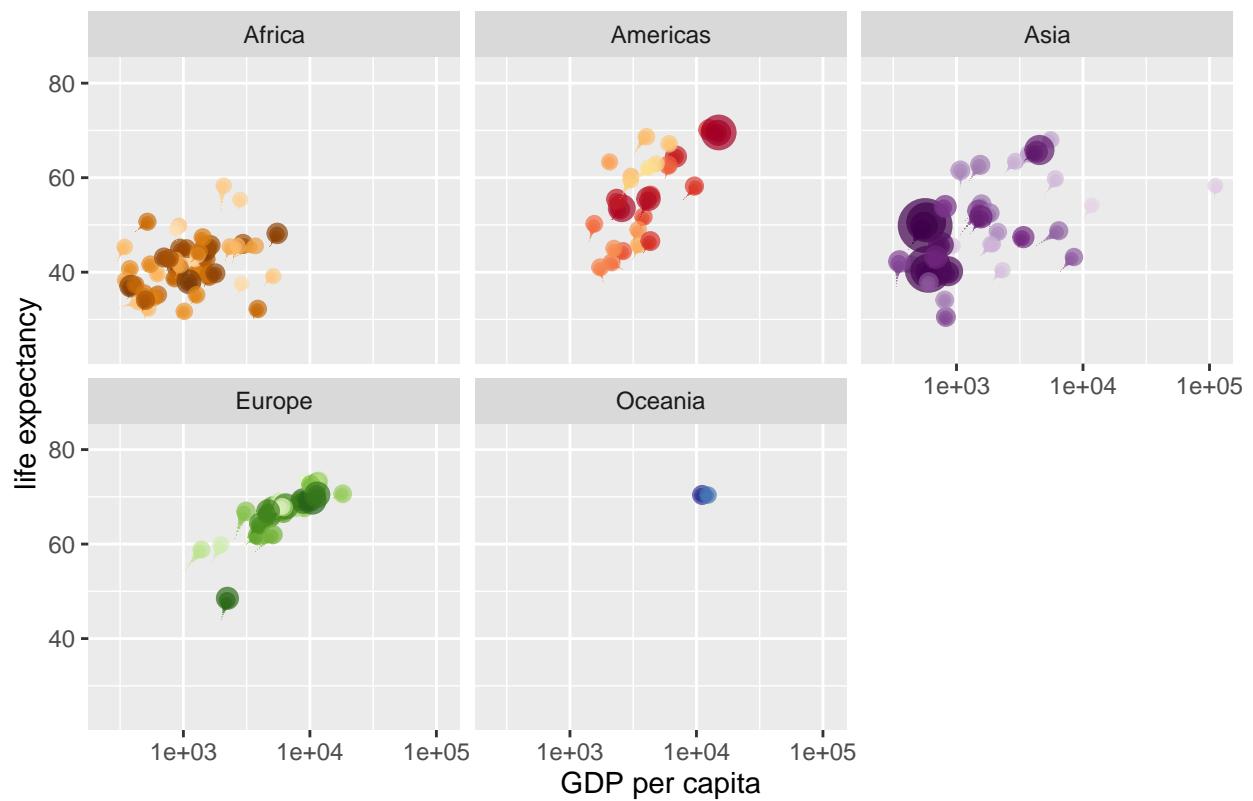
Year: 1956



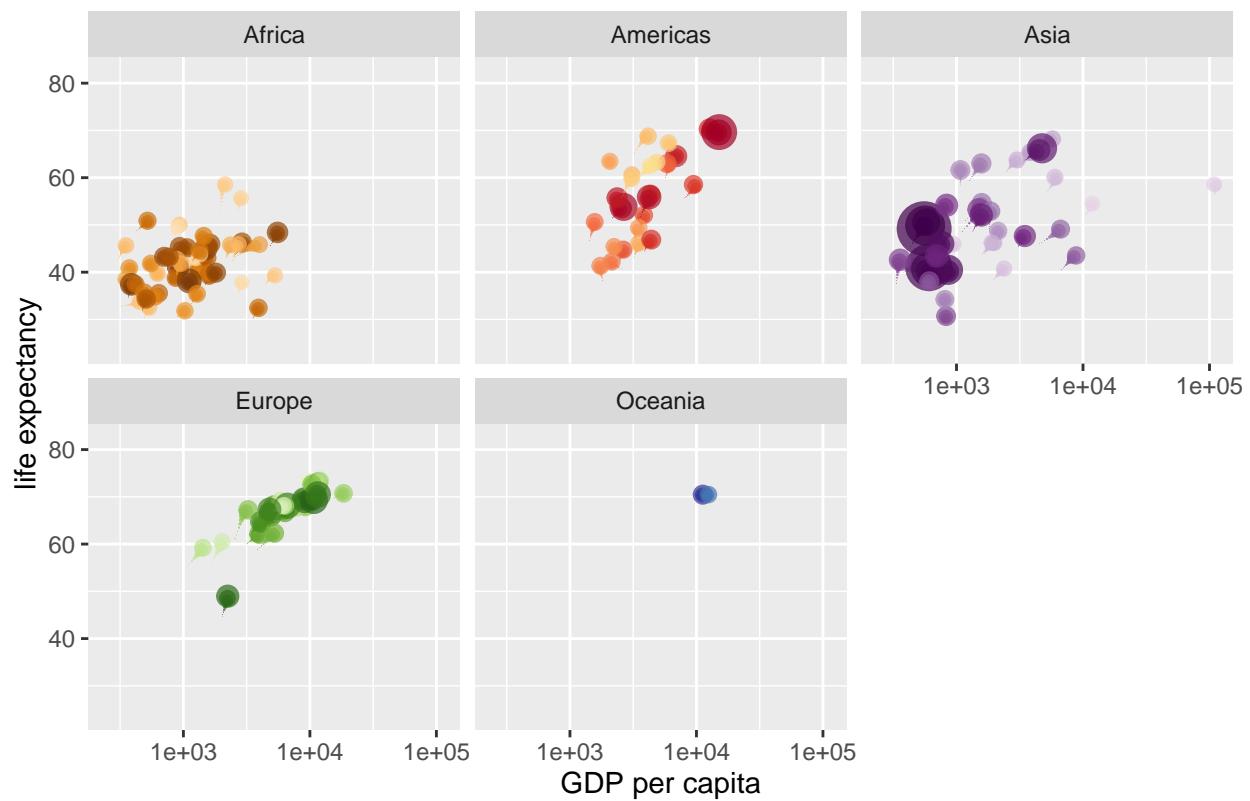
Year: 1957



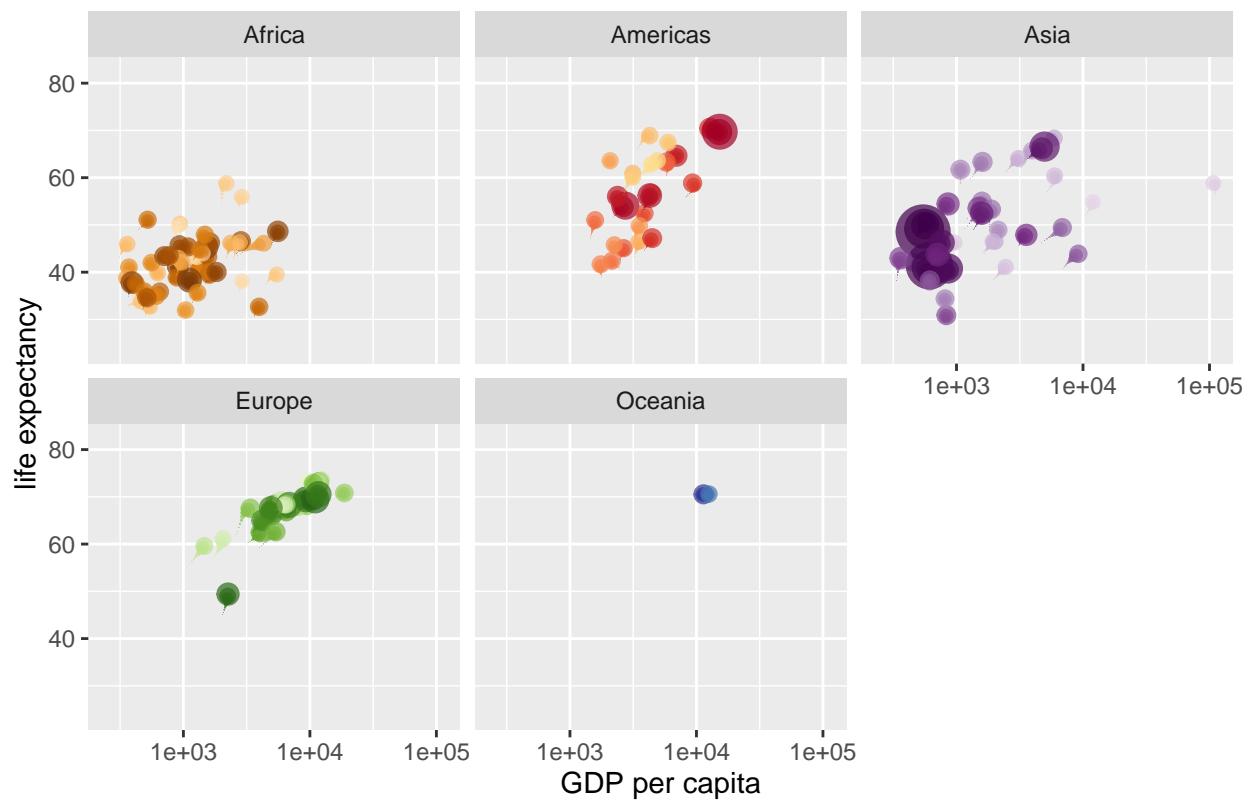
Year: 1958



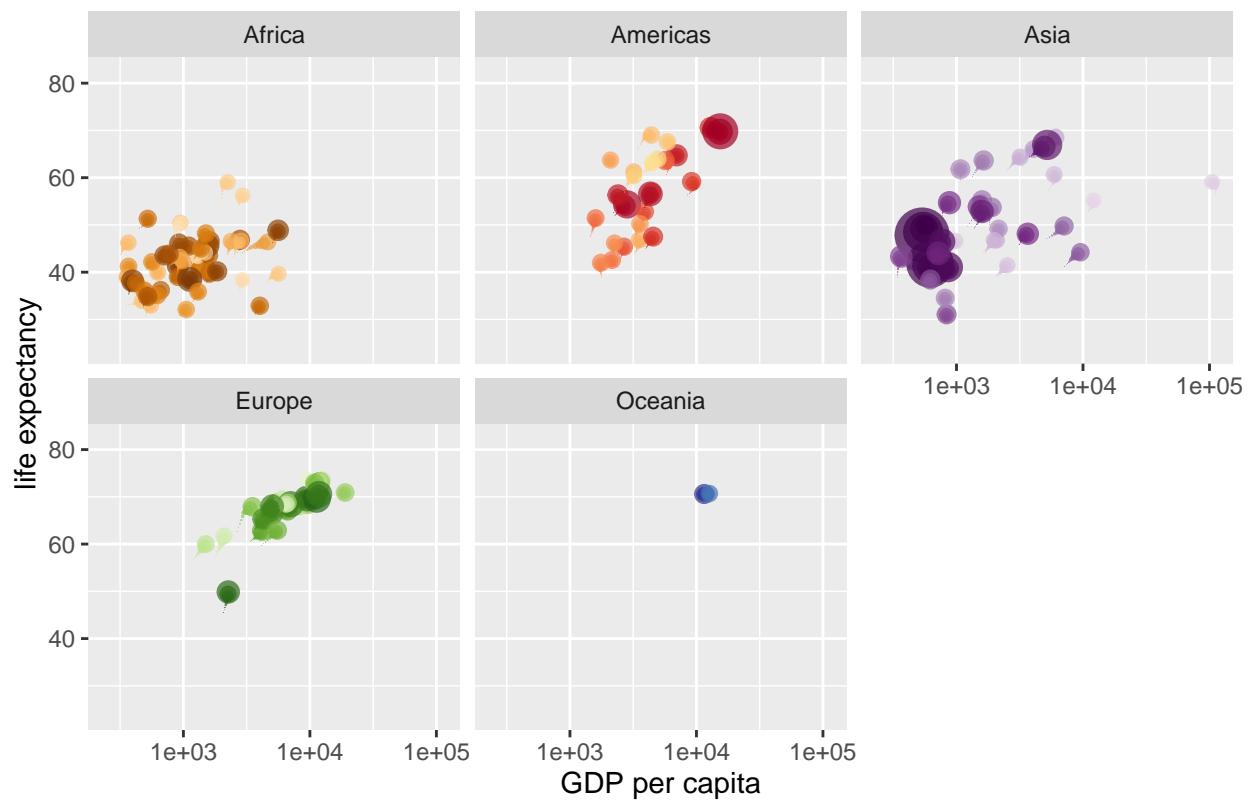
Year: 1958



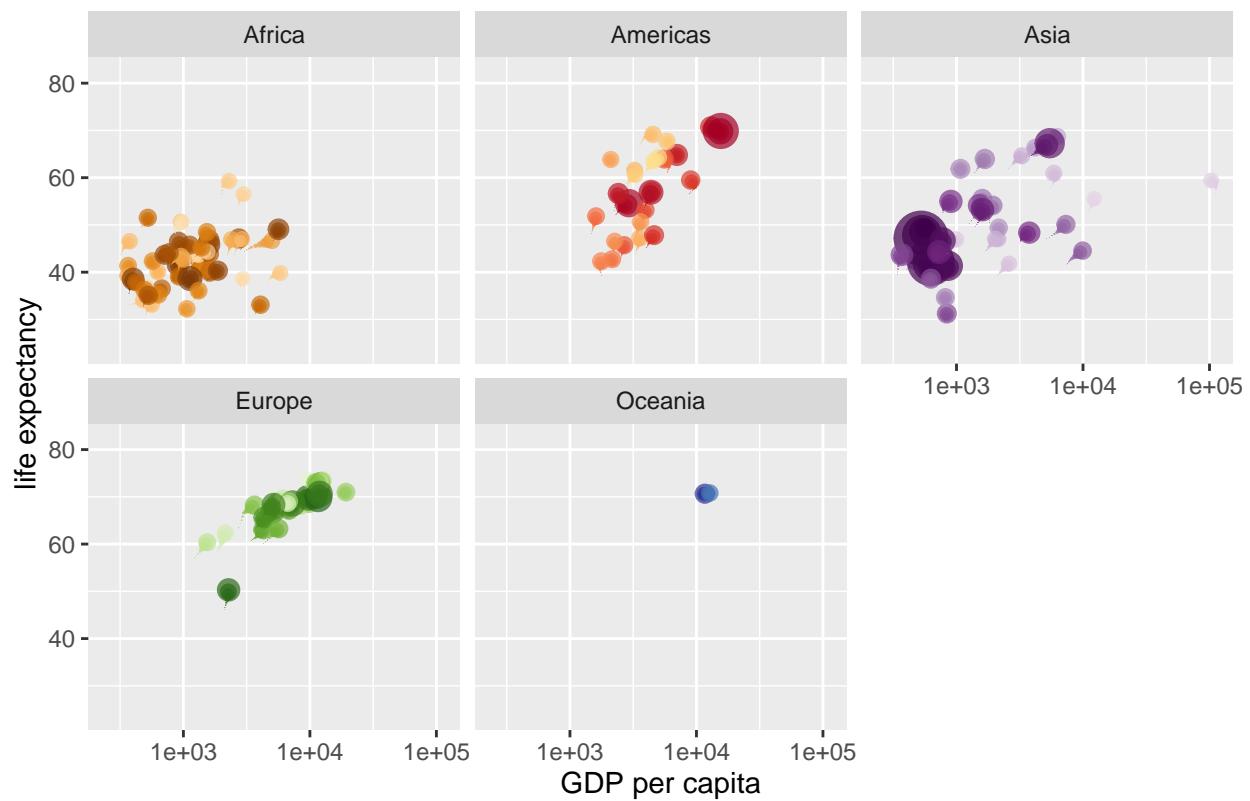
Year: 1959



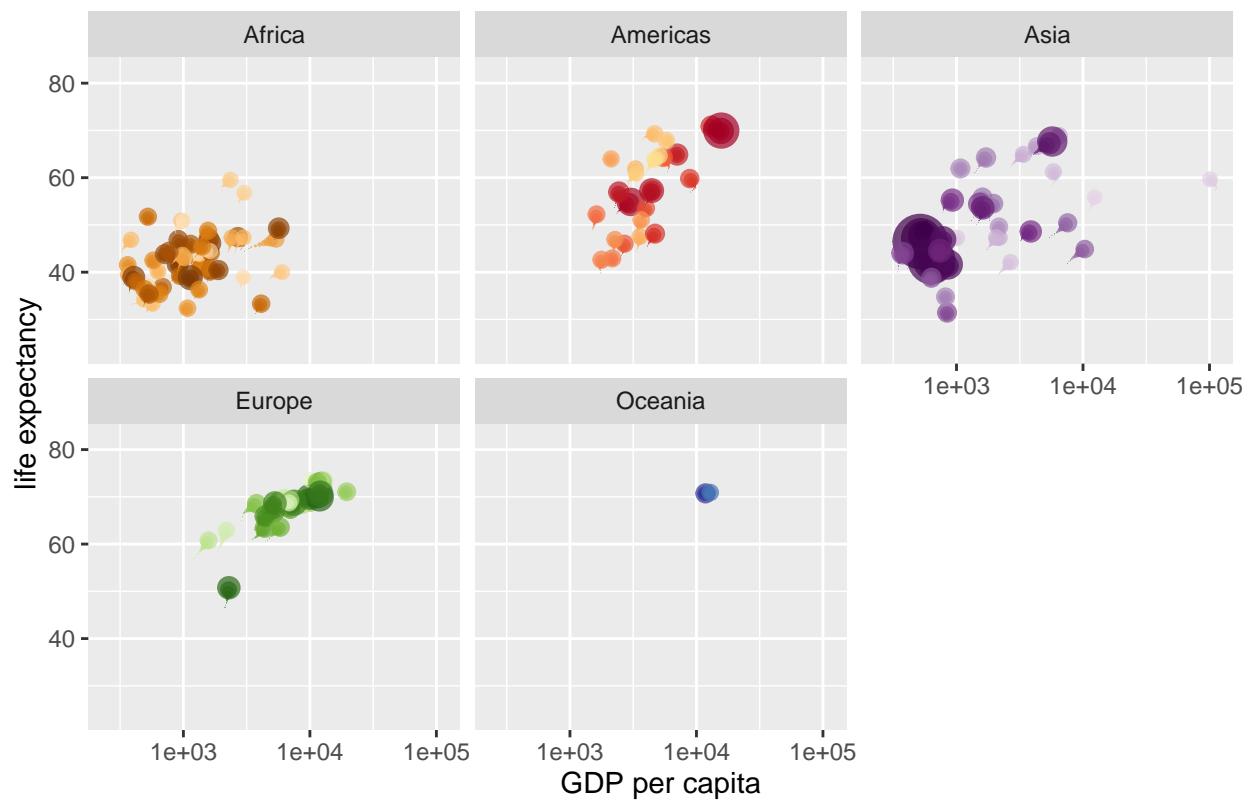
Year: 1959



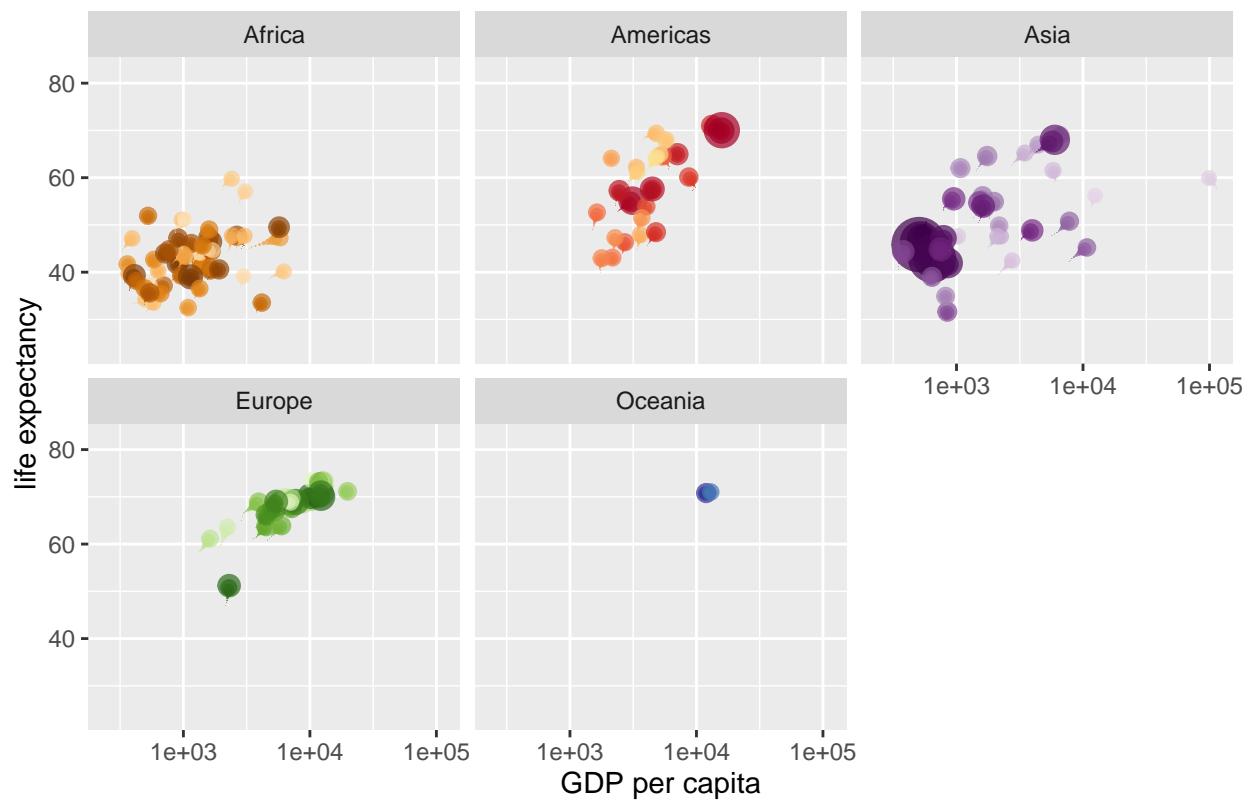
Year: 1960



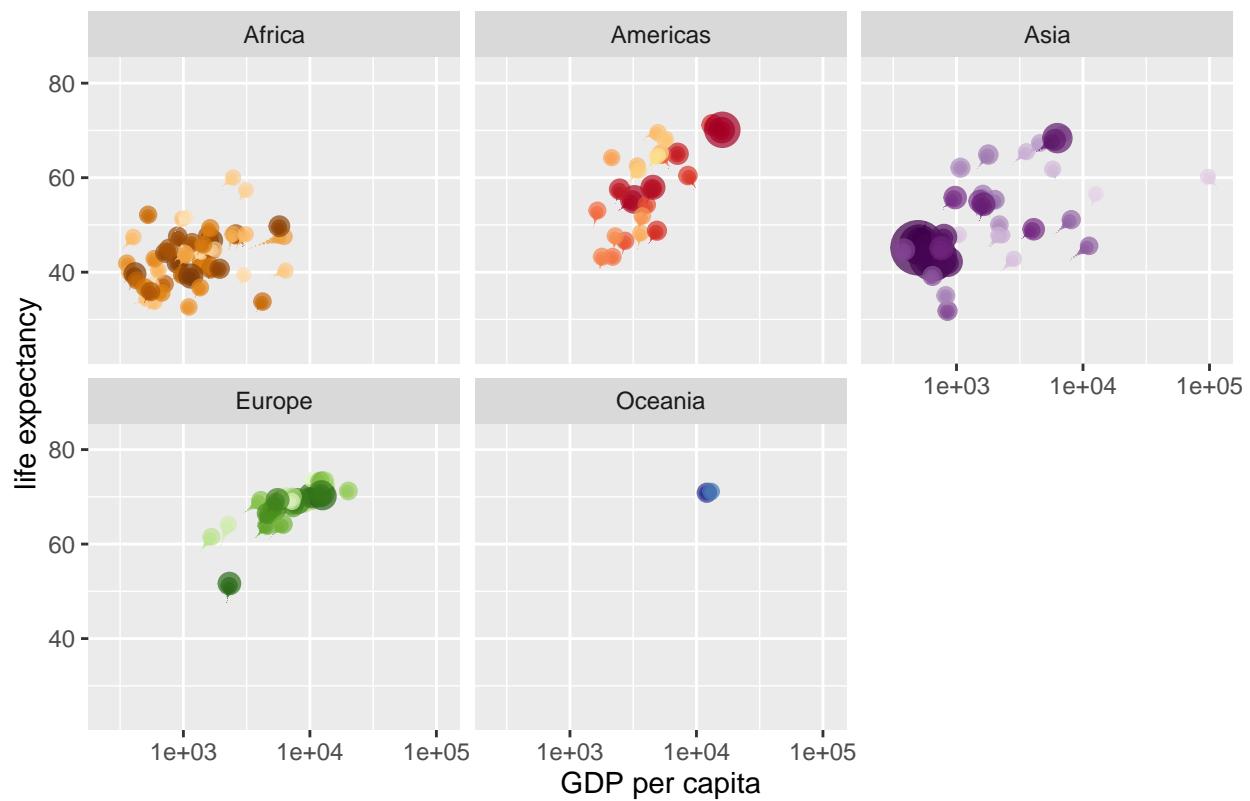
Year: 1960



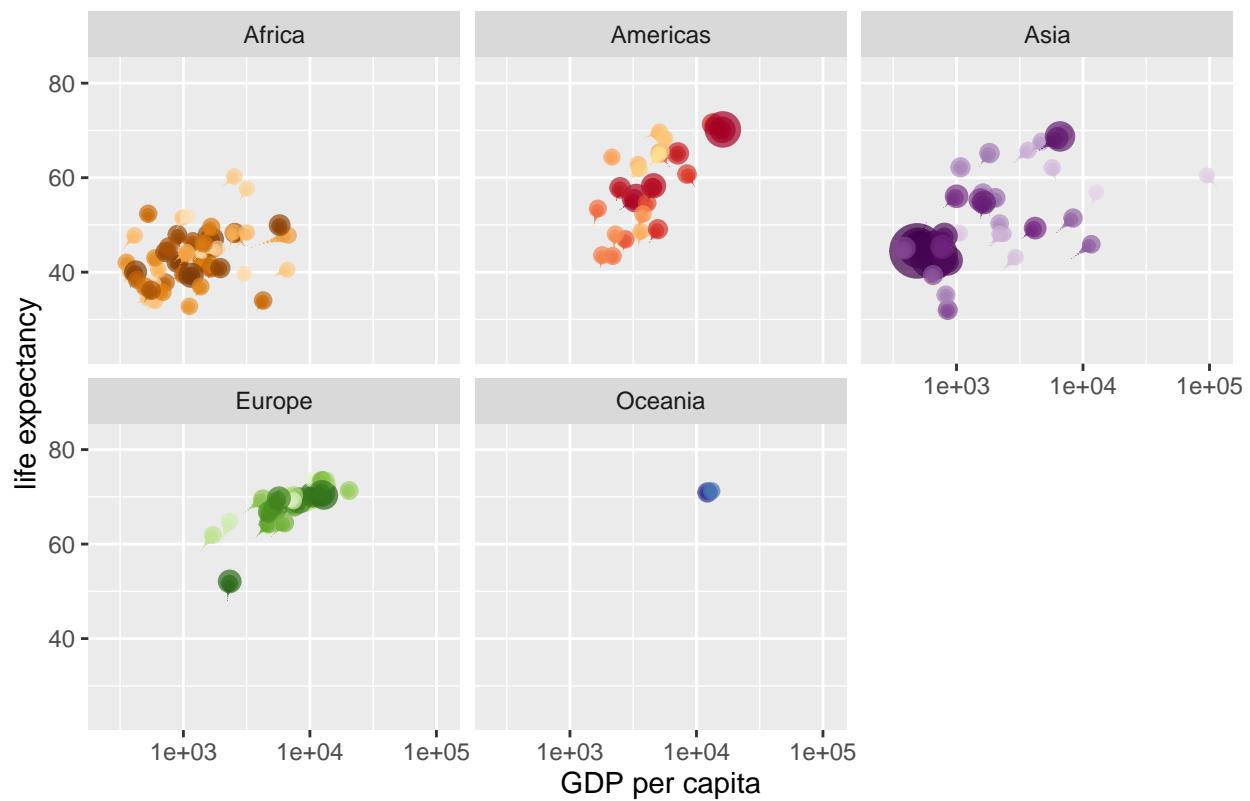
Year: 1961



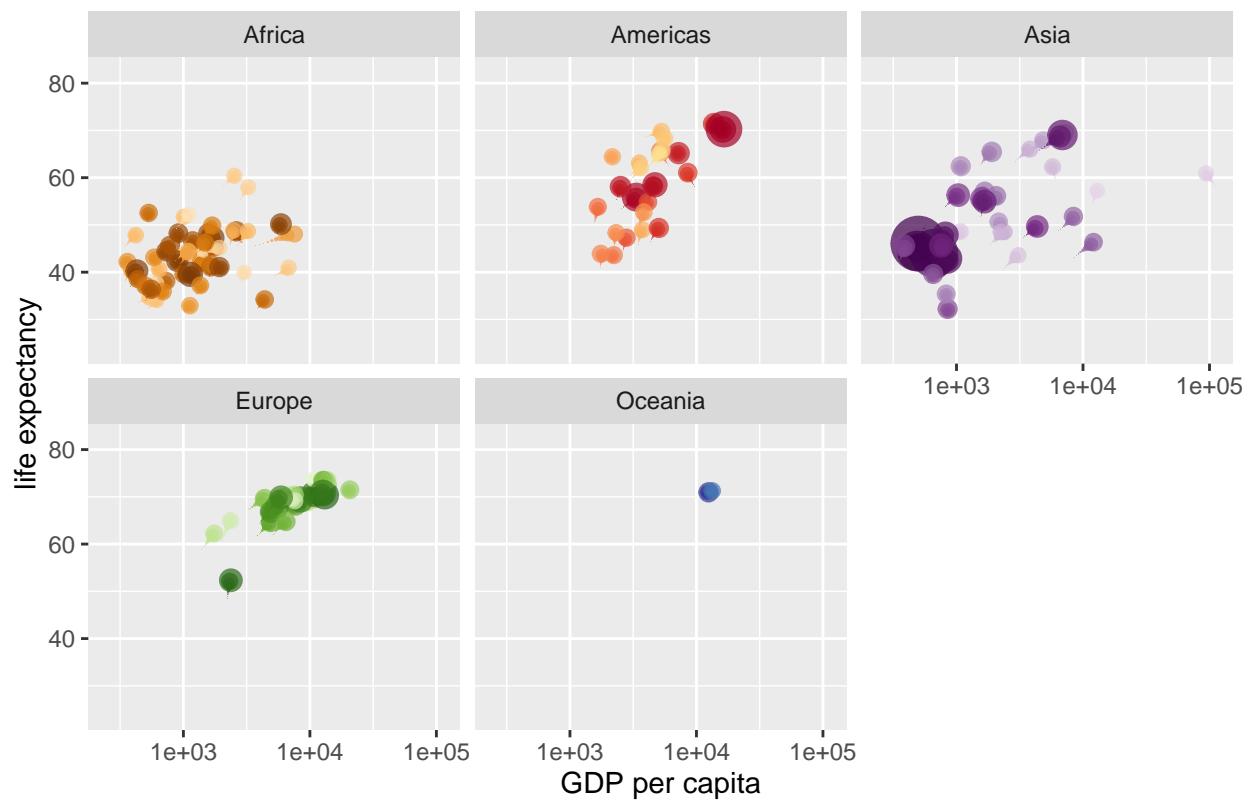
Year: 1961



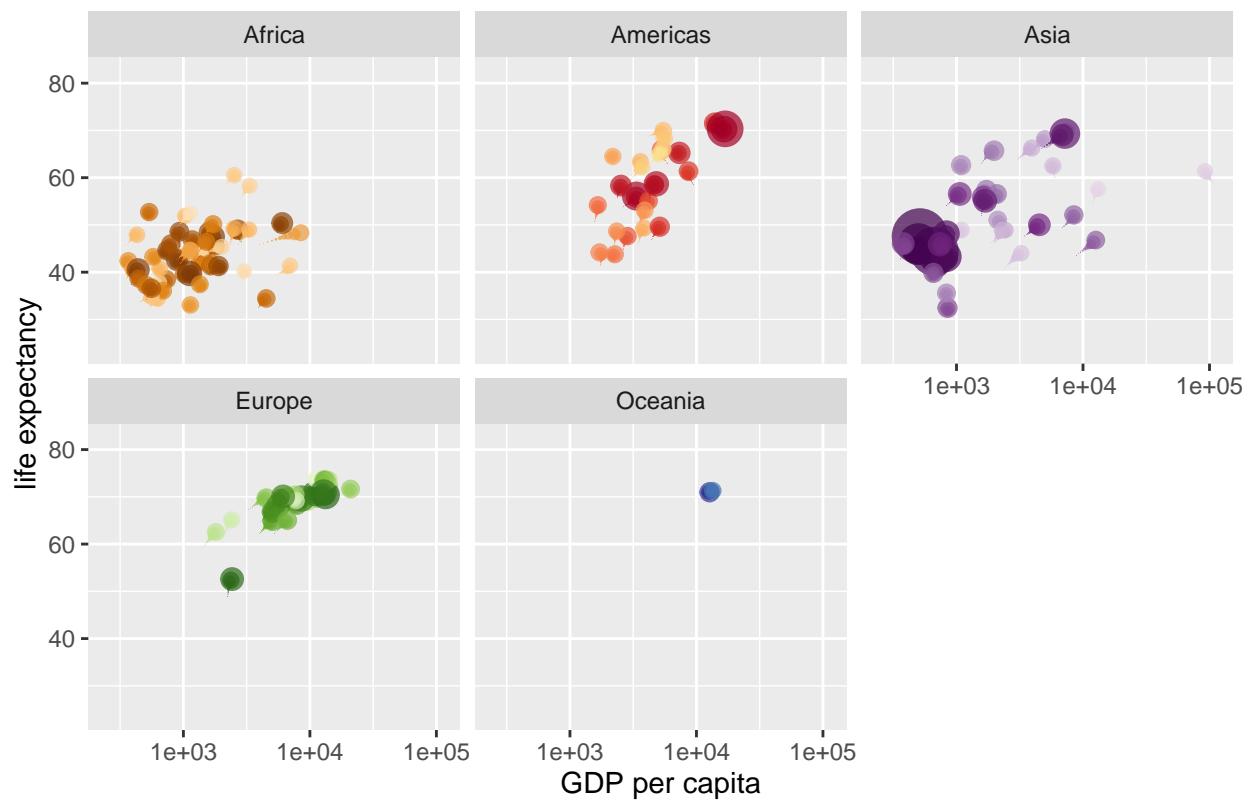
Year: 1962



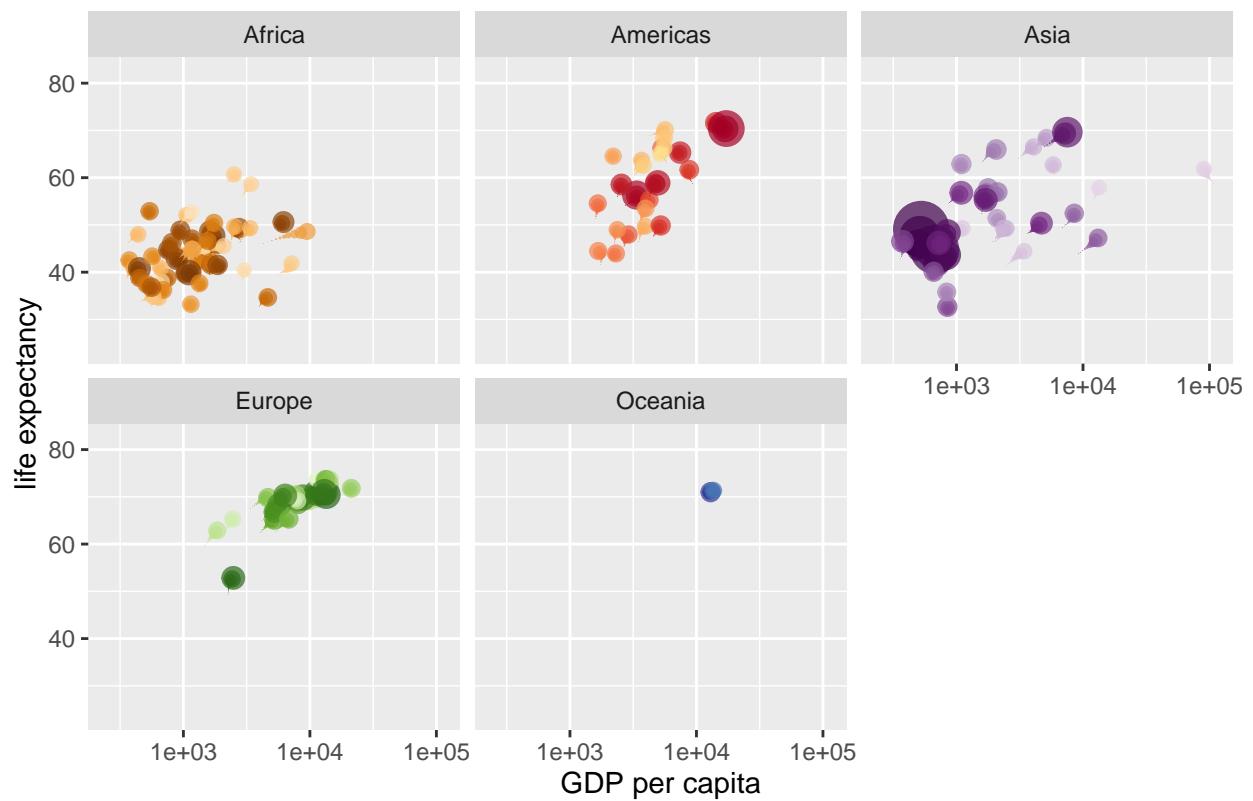
Year: 1963



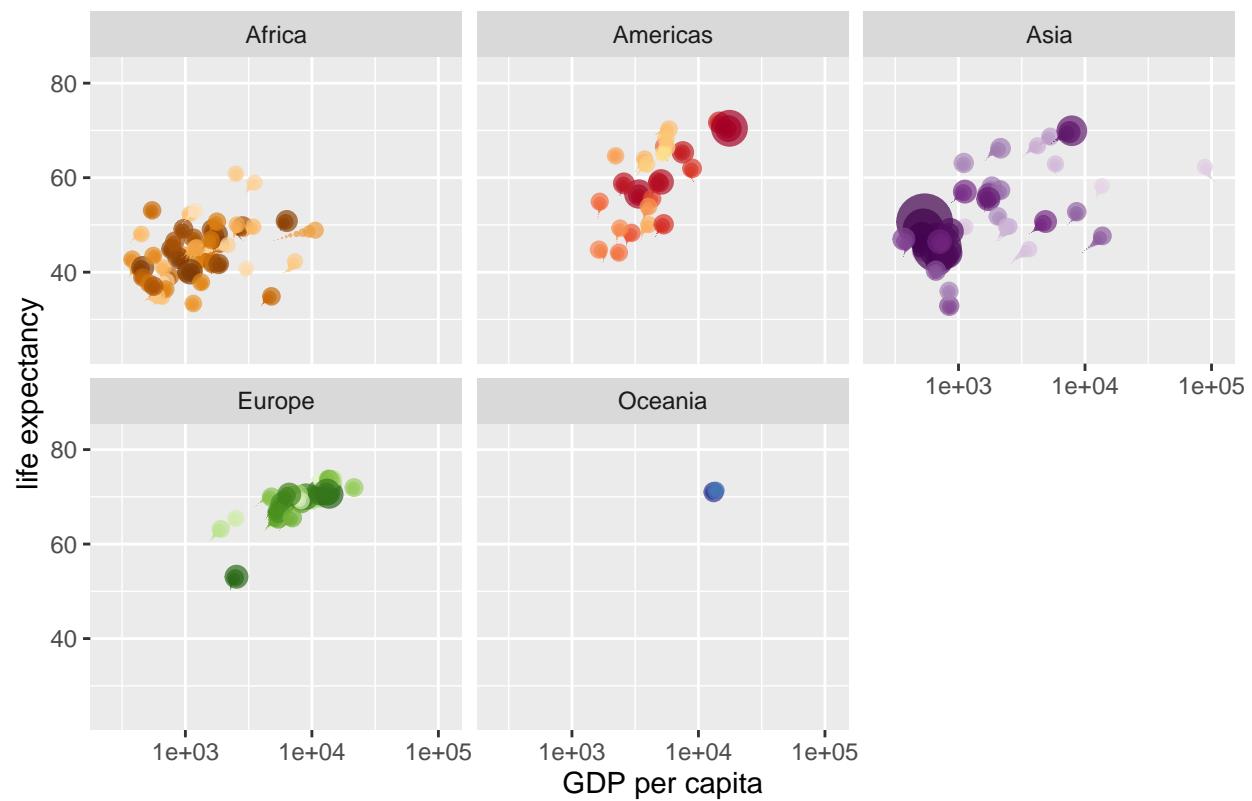
Year: 1963



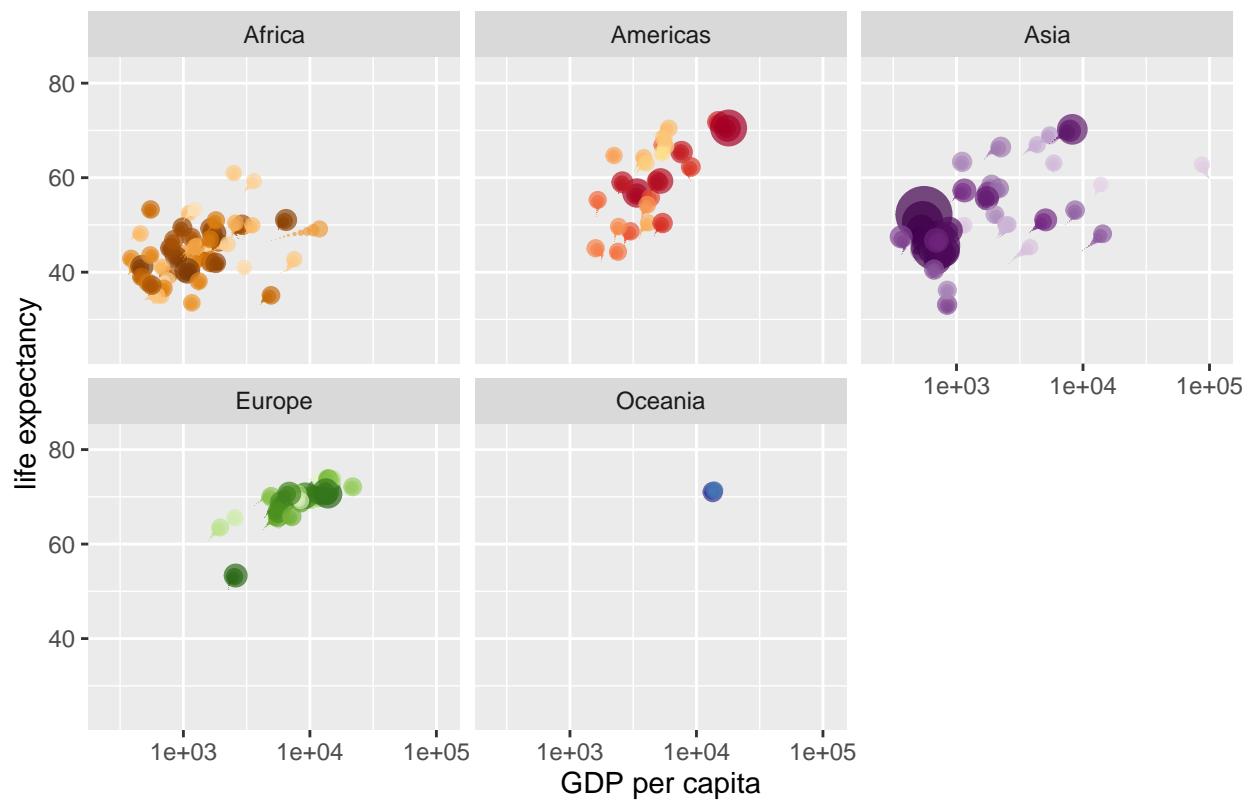
Year: 1964



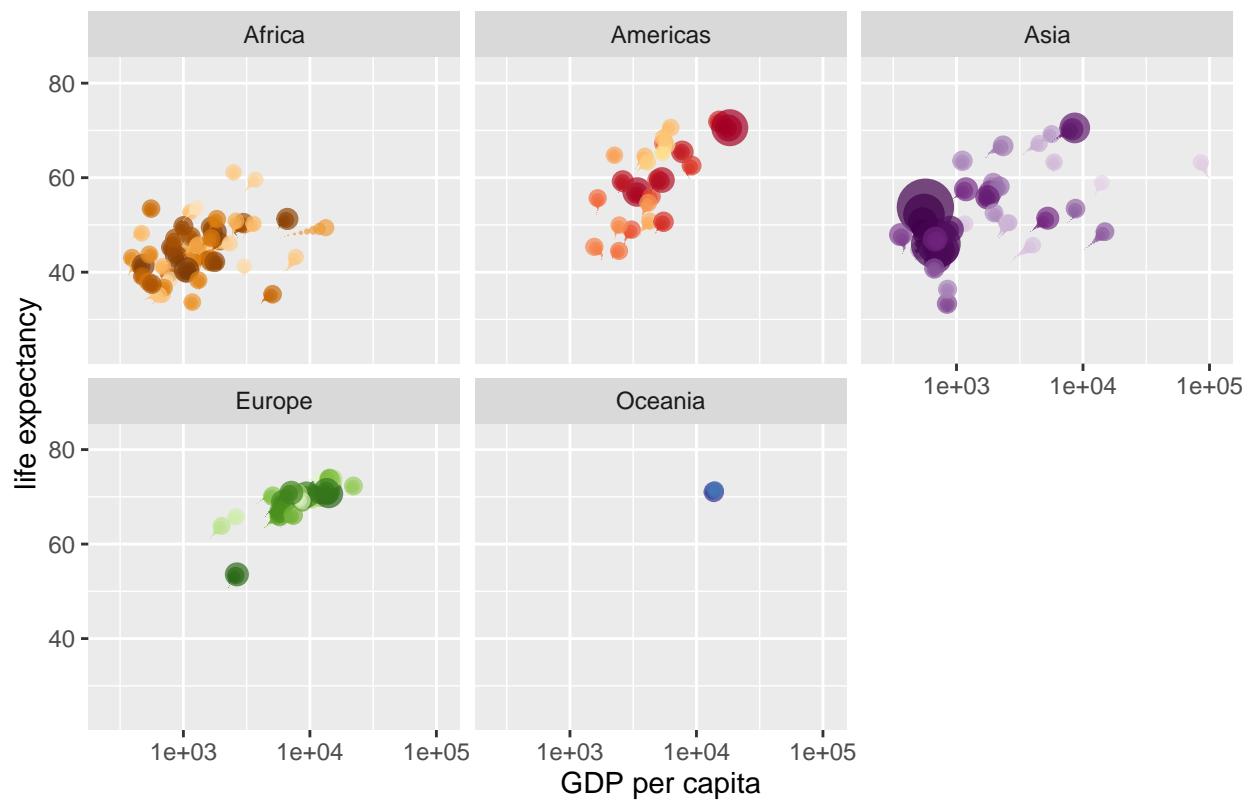
Year: 1964



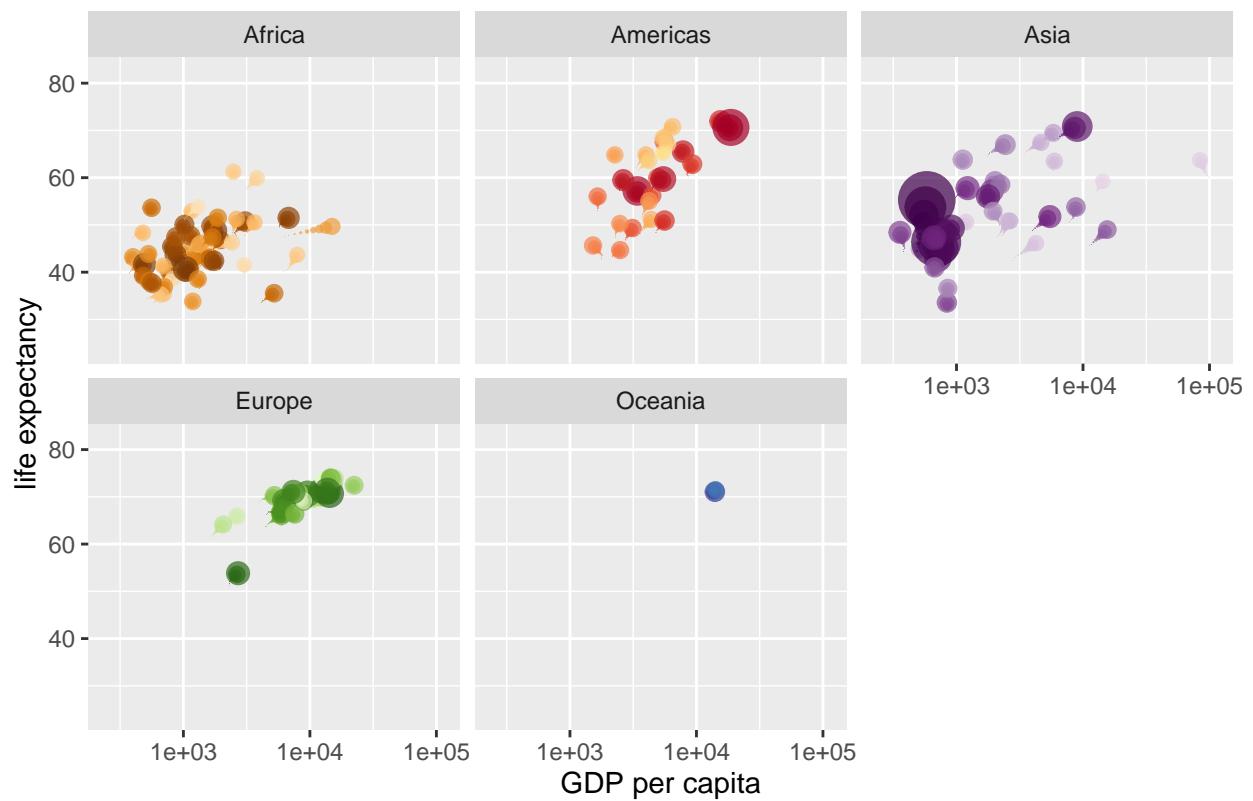
Year: 1965



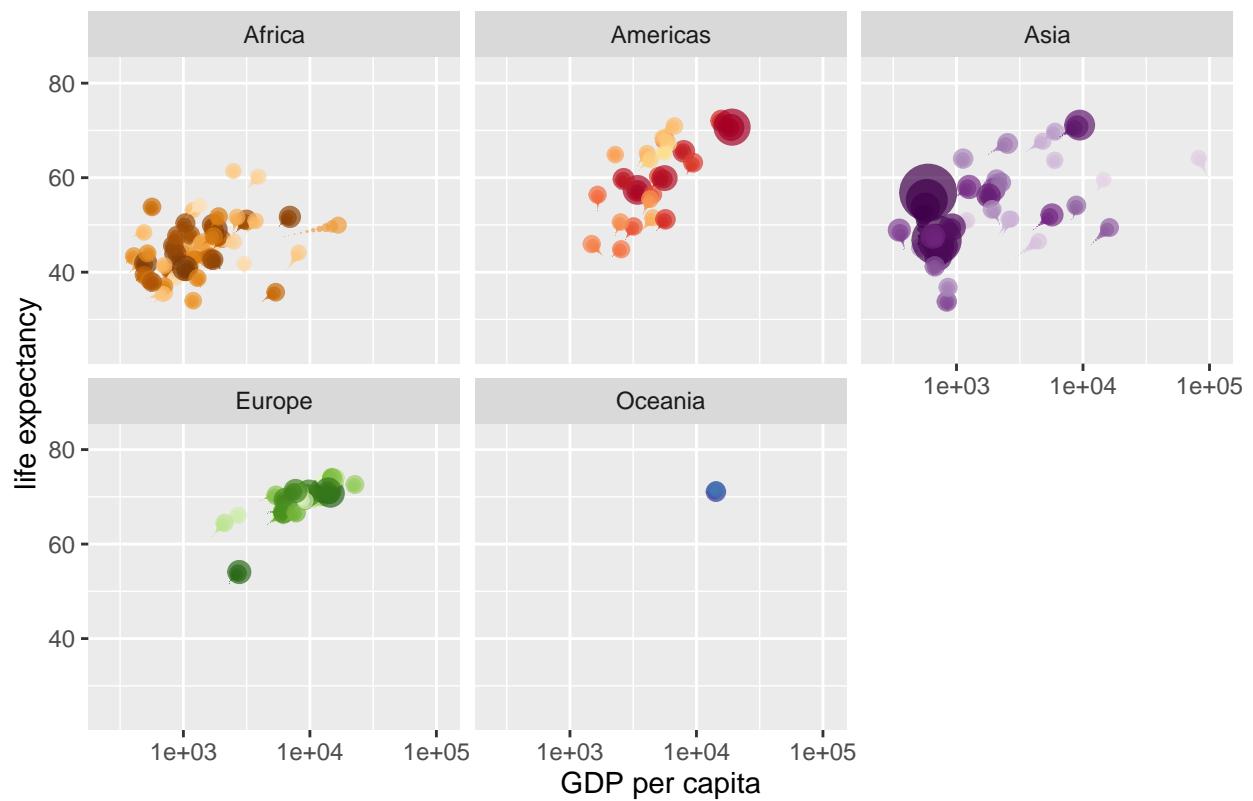
Year: 1965



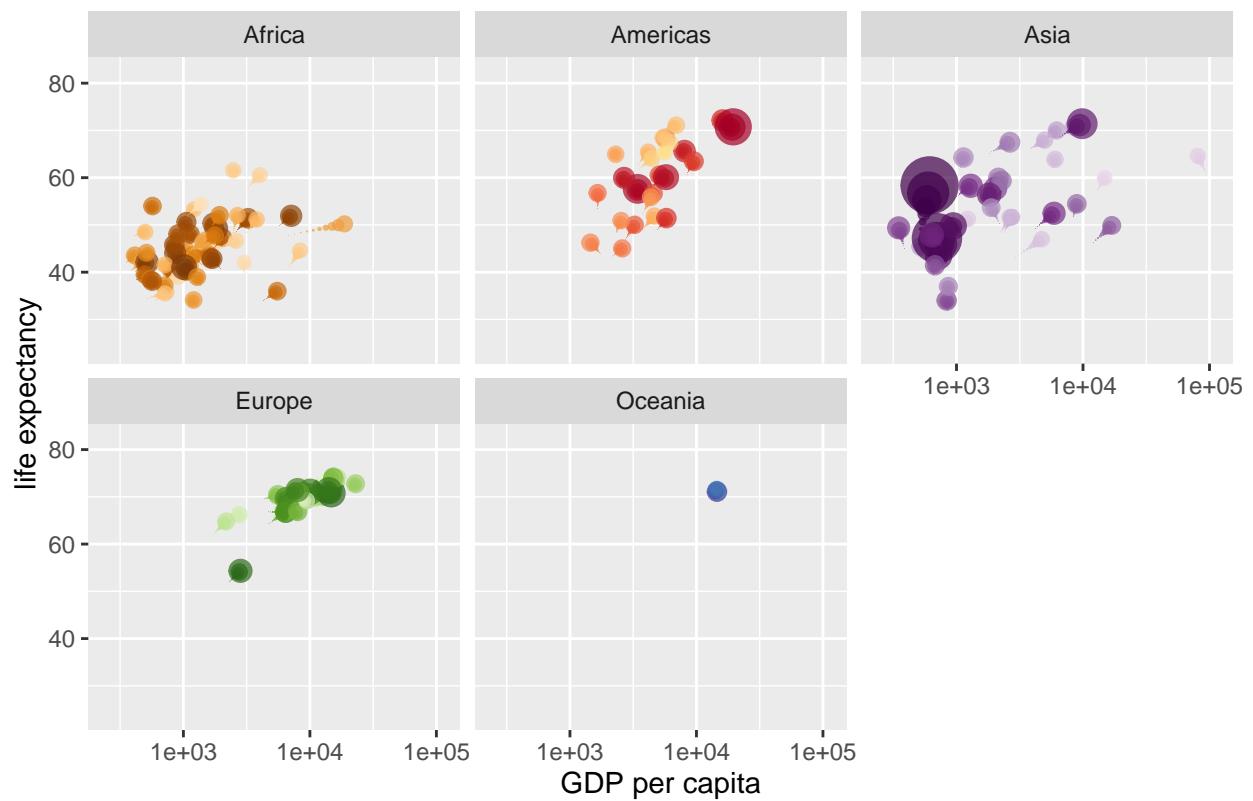
Year: 1966



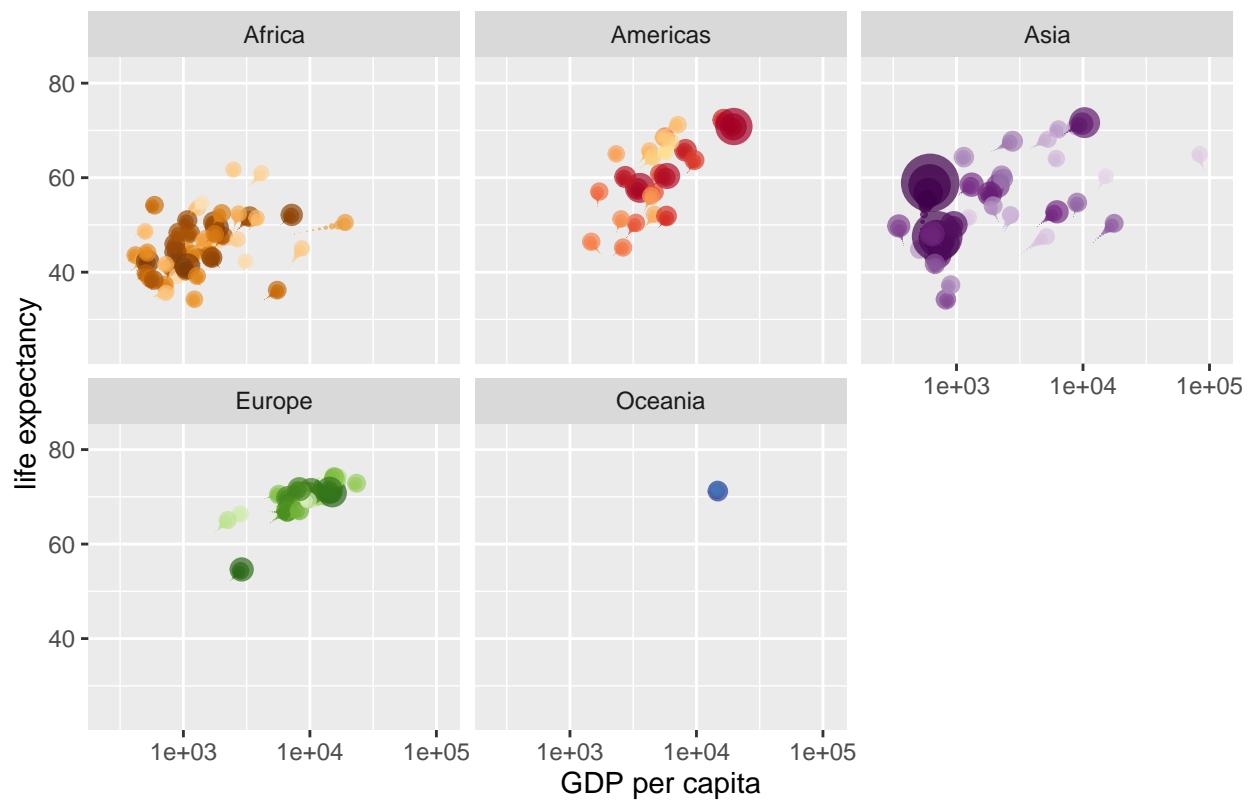
Year: 1966



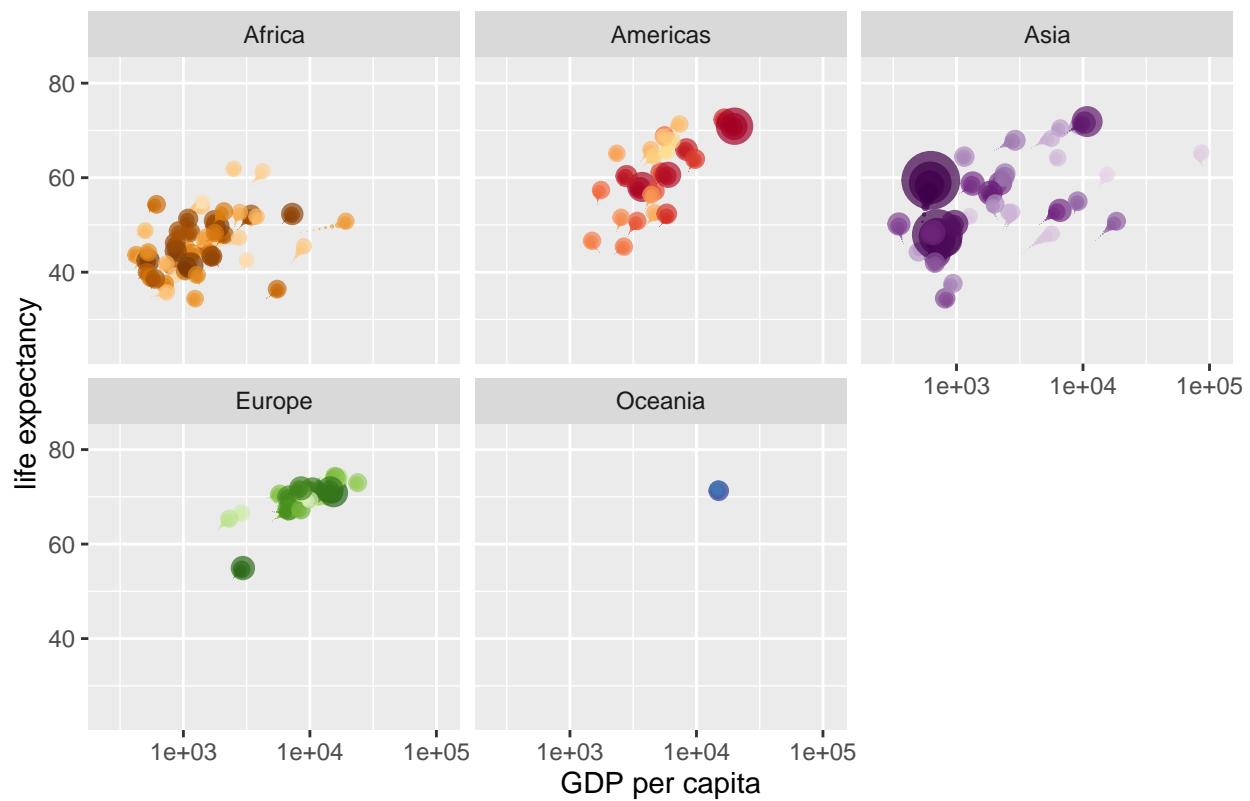
Year: 1967



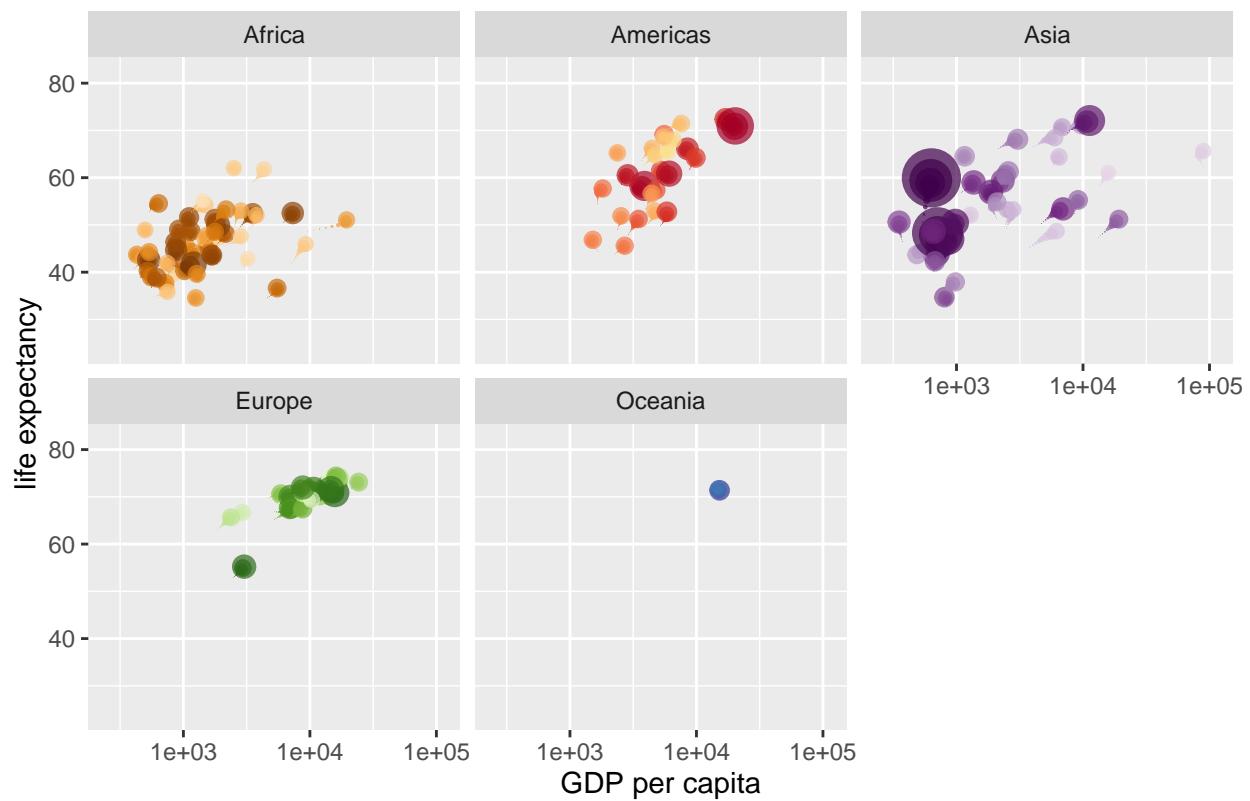
Year: 1968



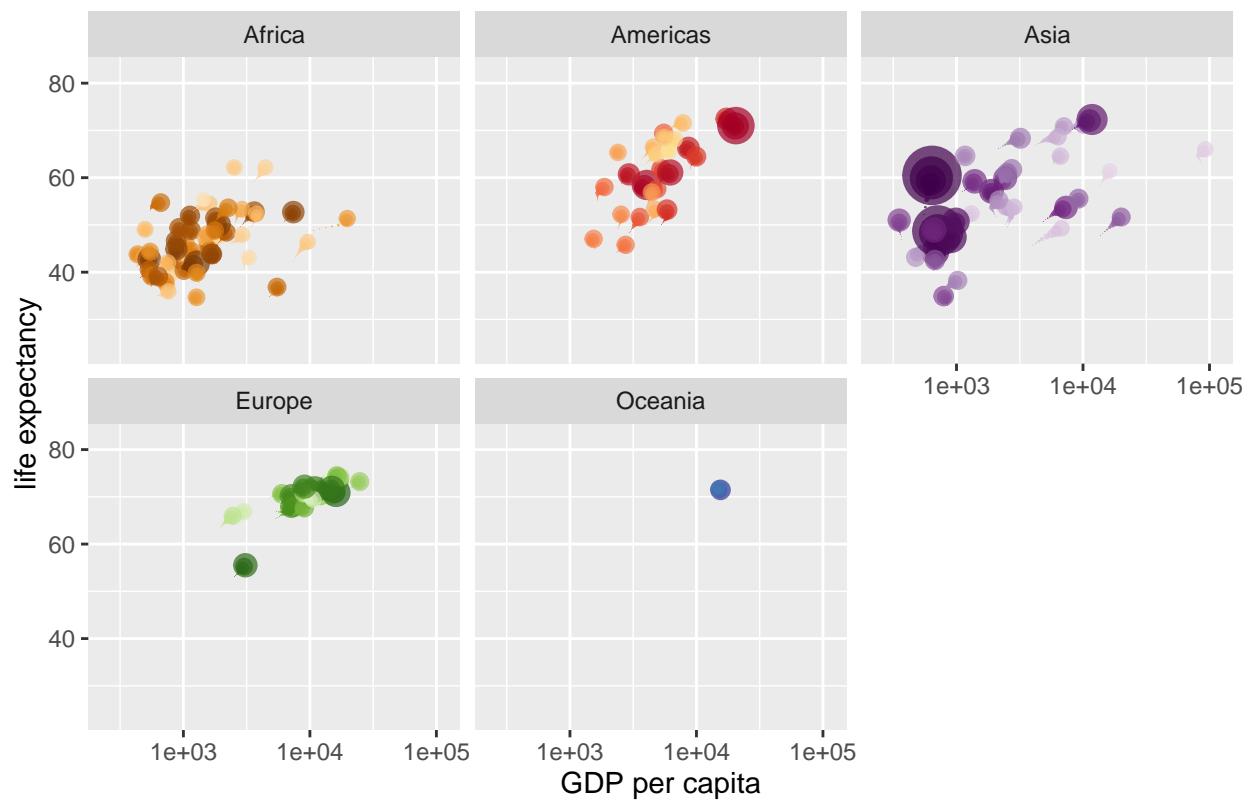
Year: 1968



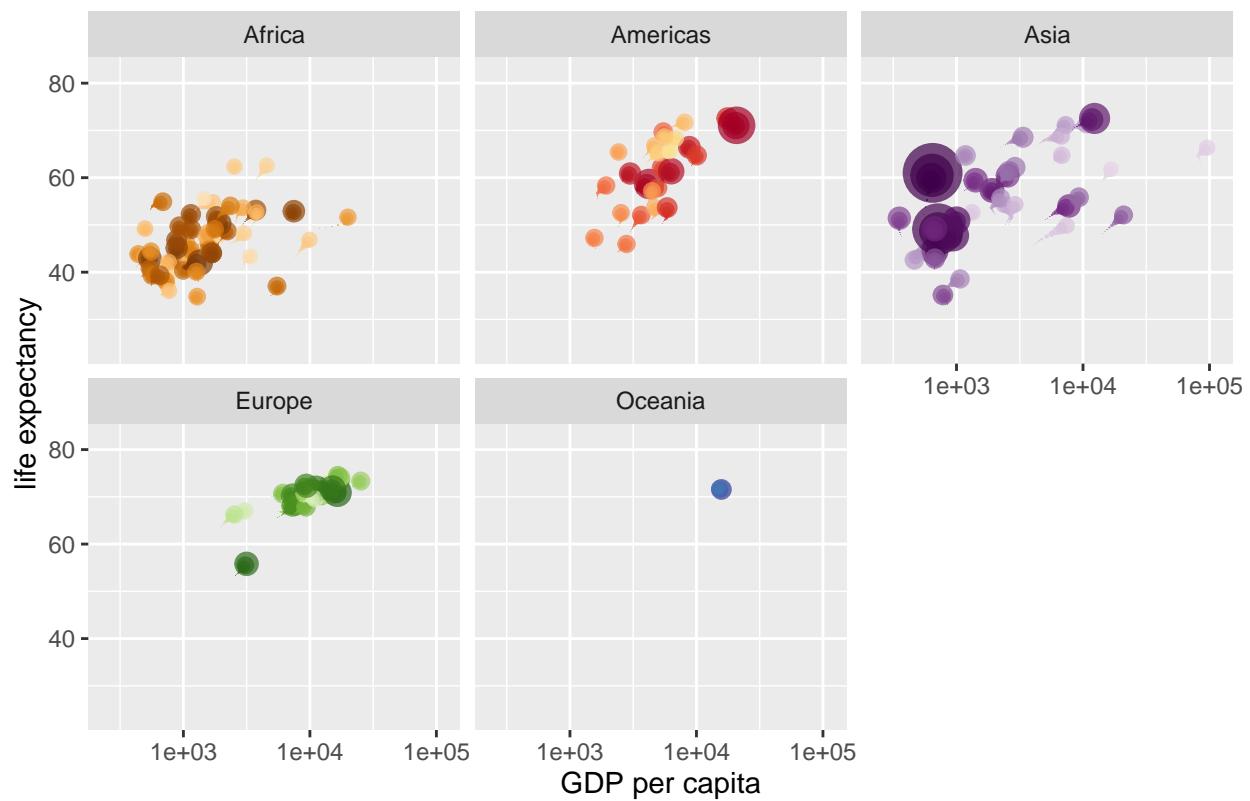
Year: 1969



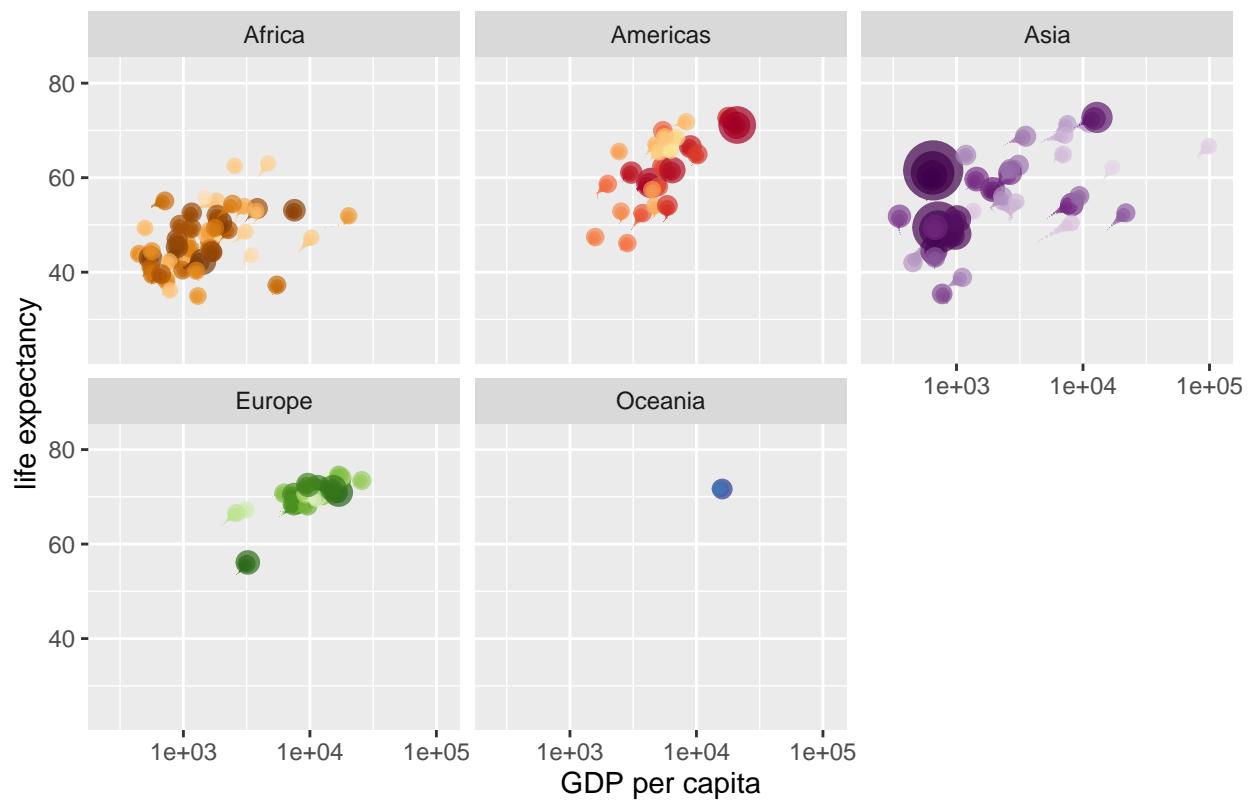
Year: 1969



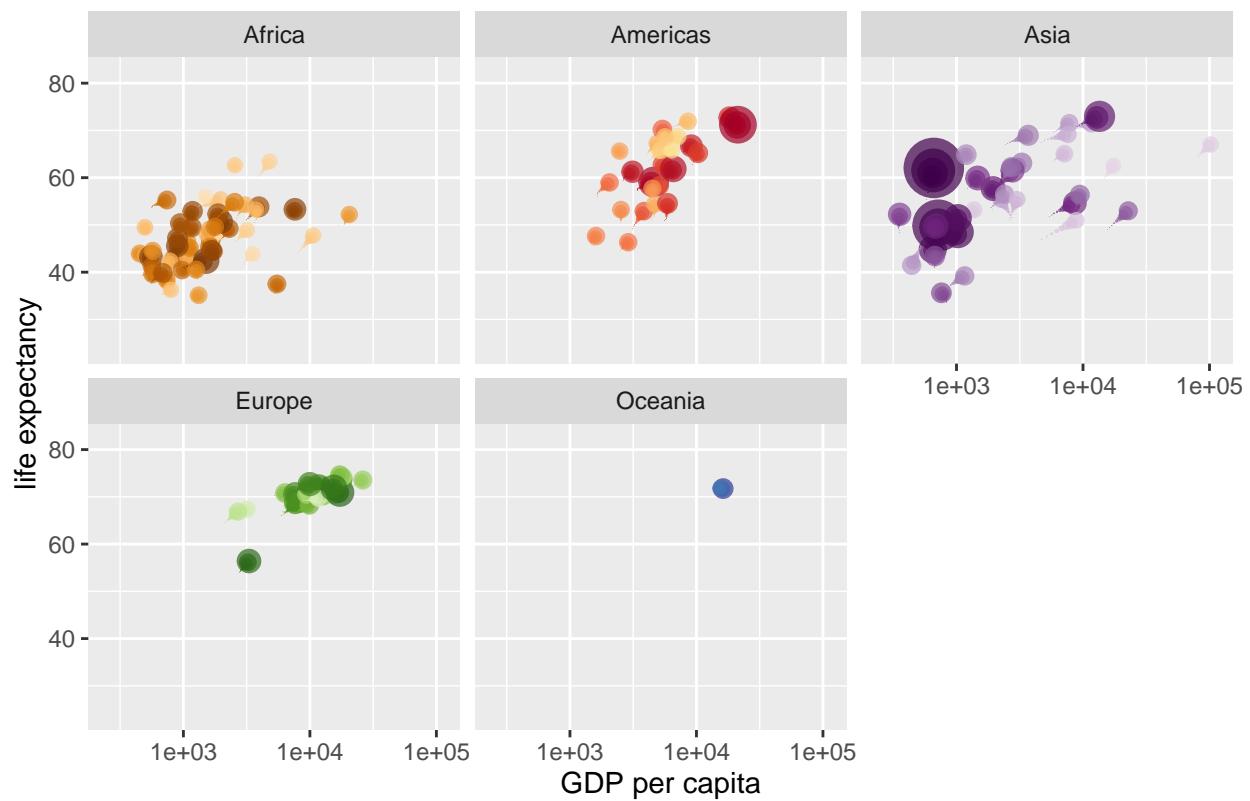
Year: 1970



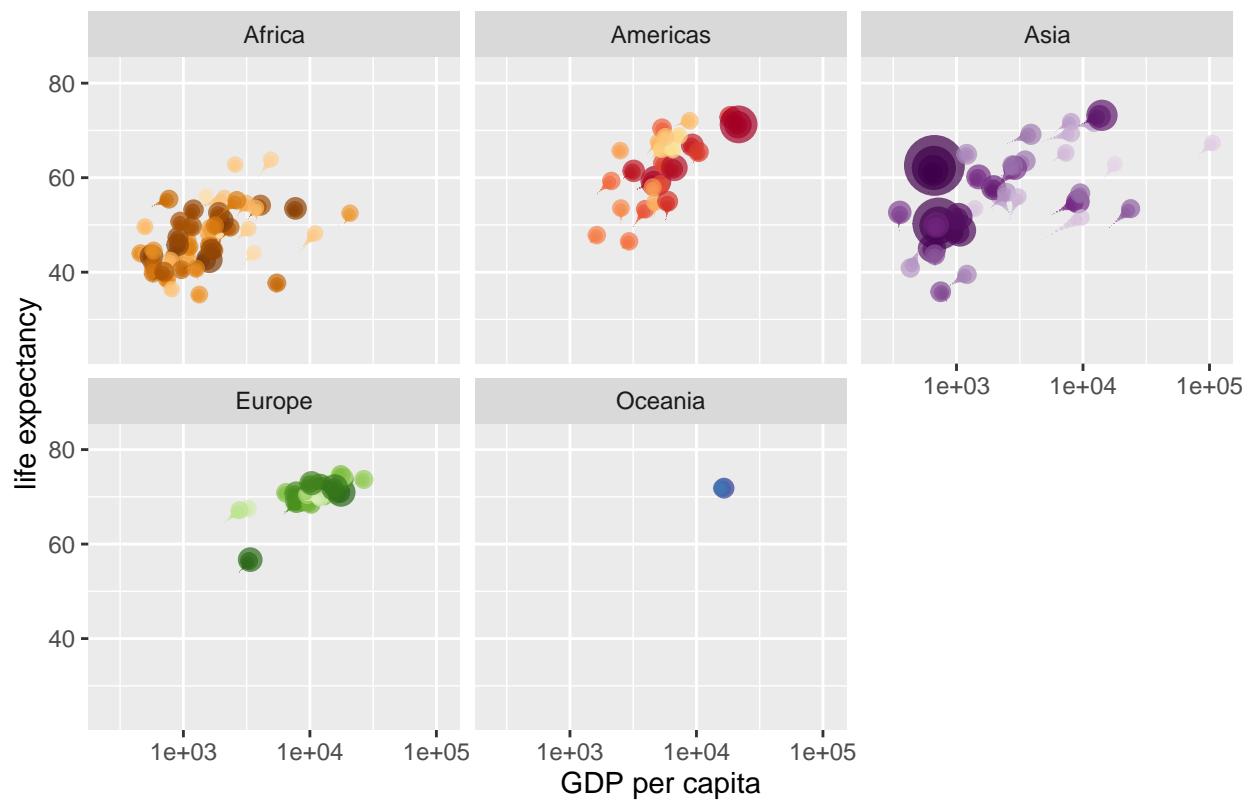
Year: 1970



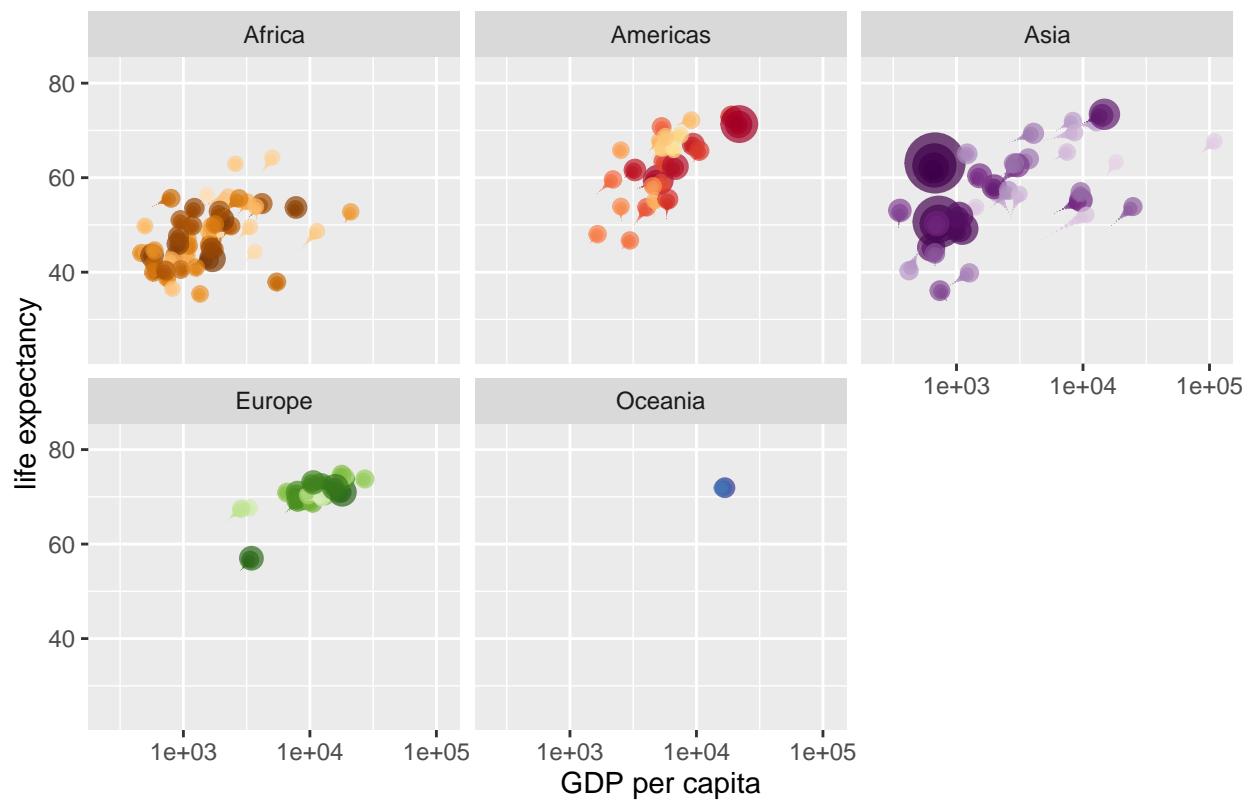
Year: 1971



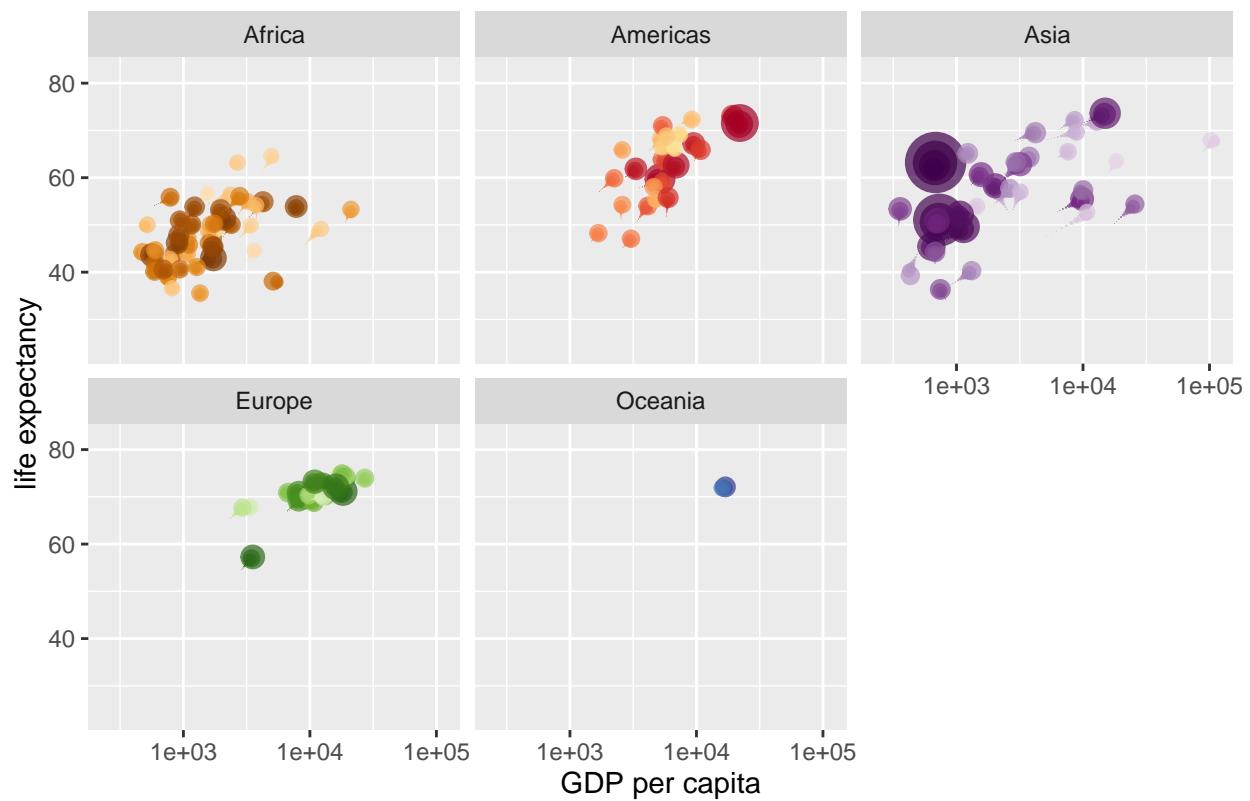
Year: 1971



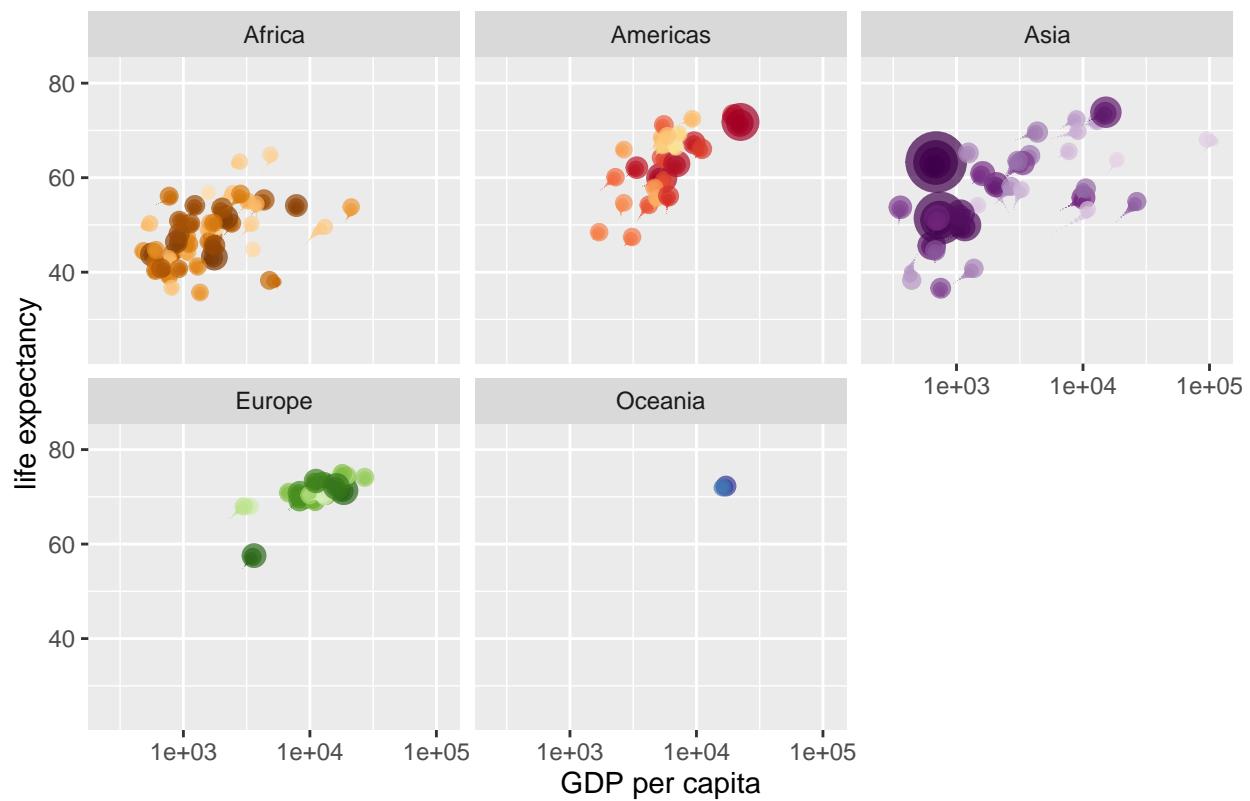
Year: 1972



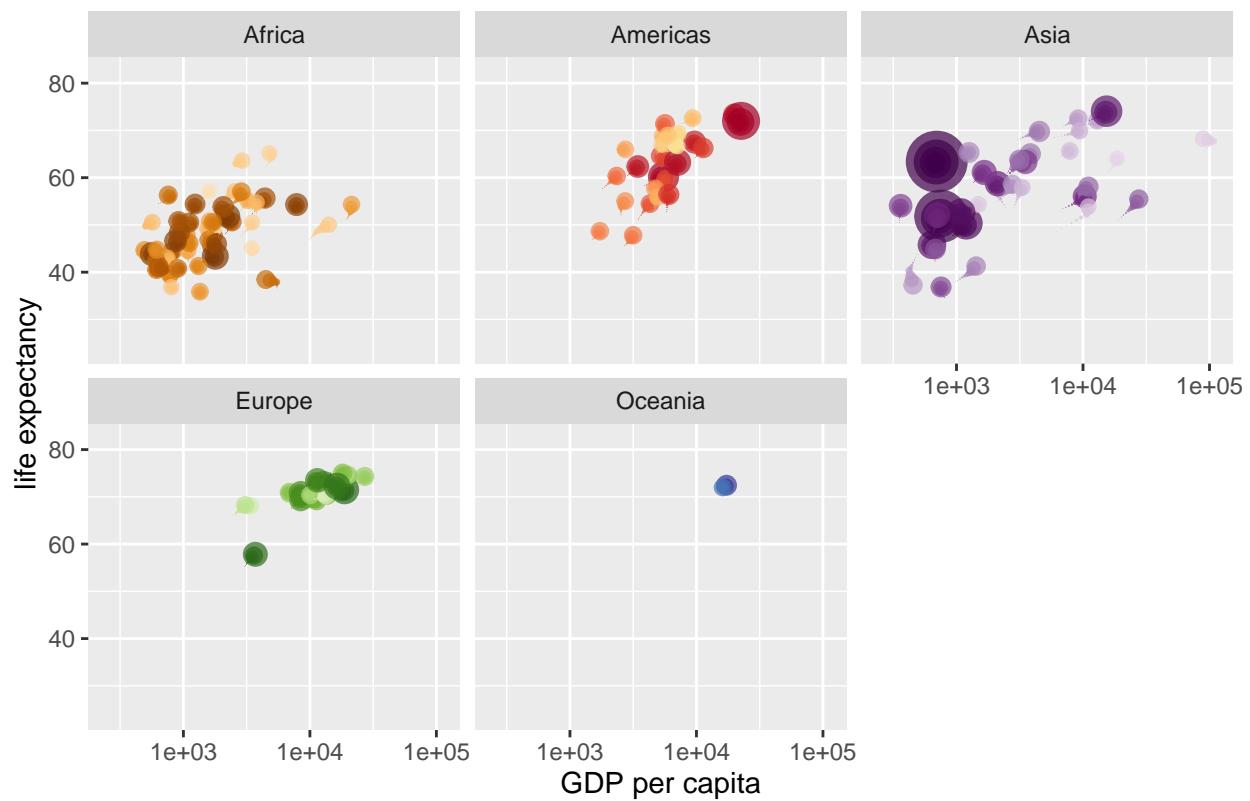
Year: 1973



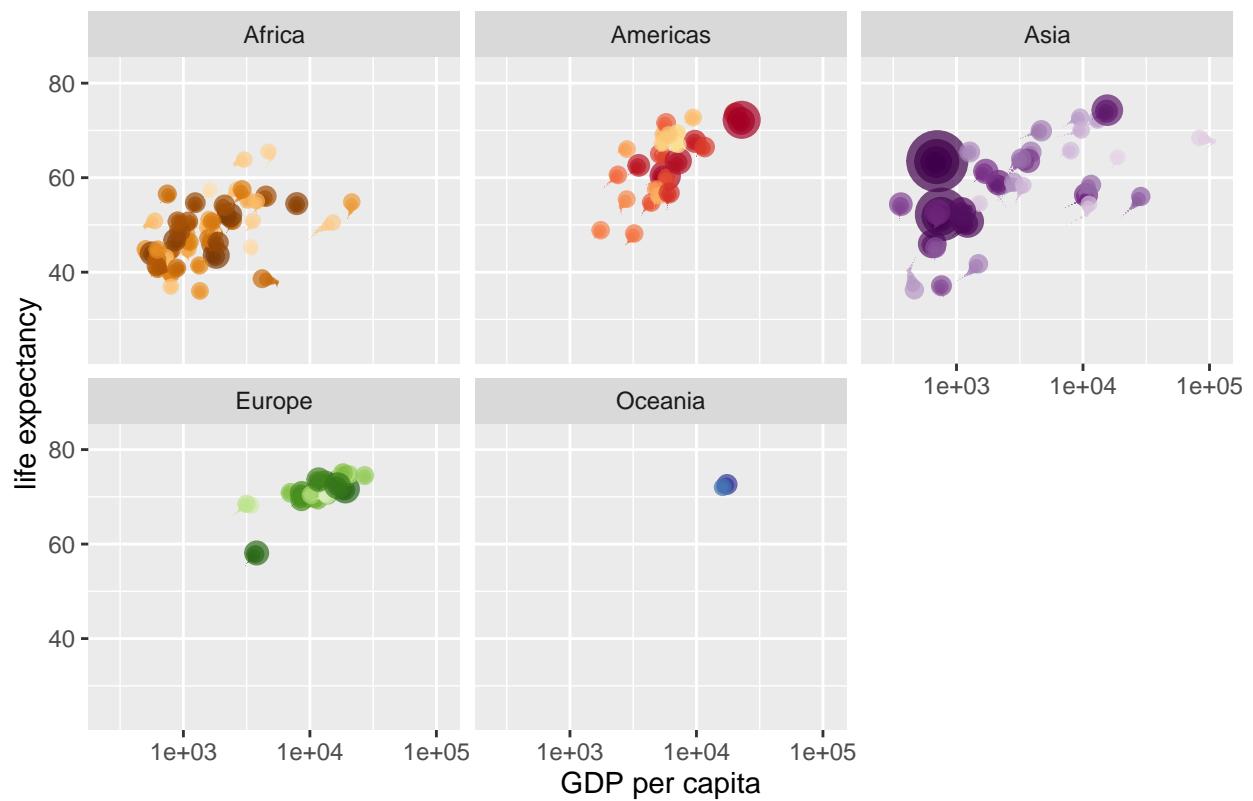
Year: 1973



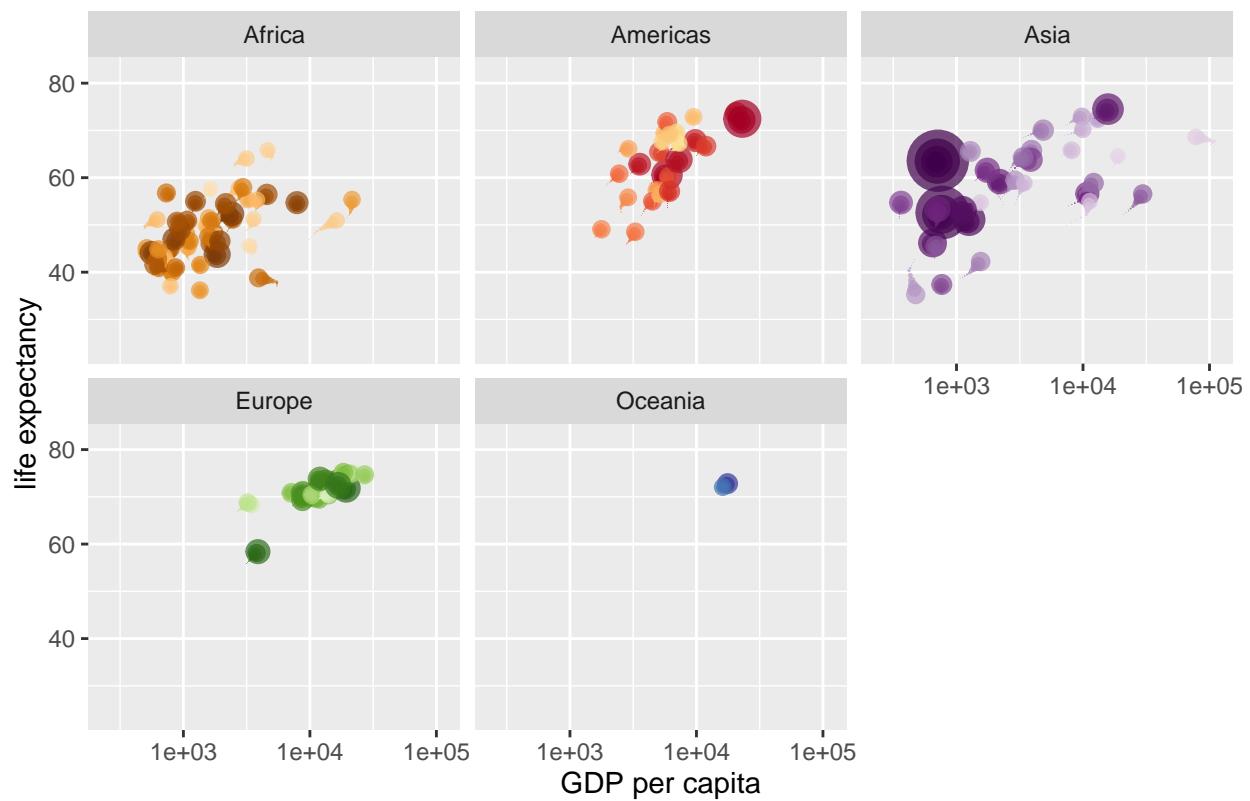
Year: 1974



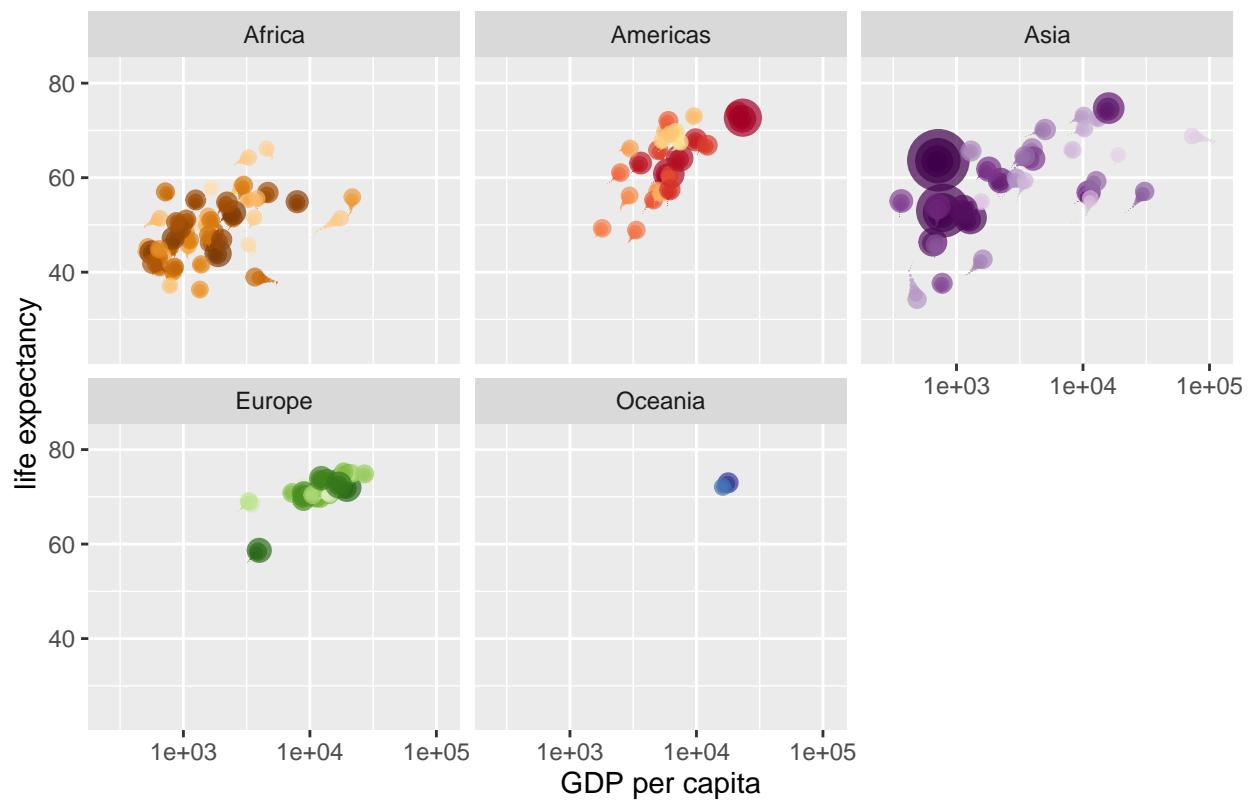
Year: 1974



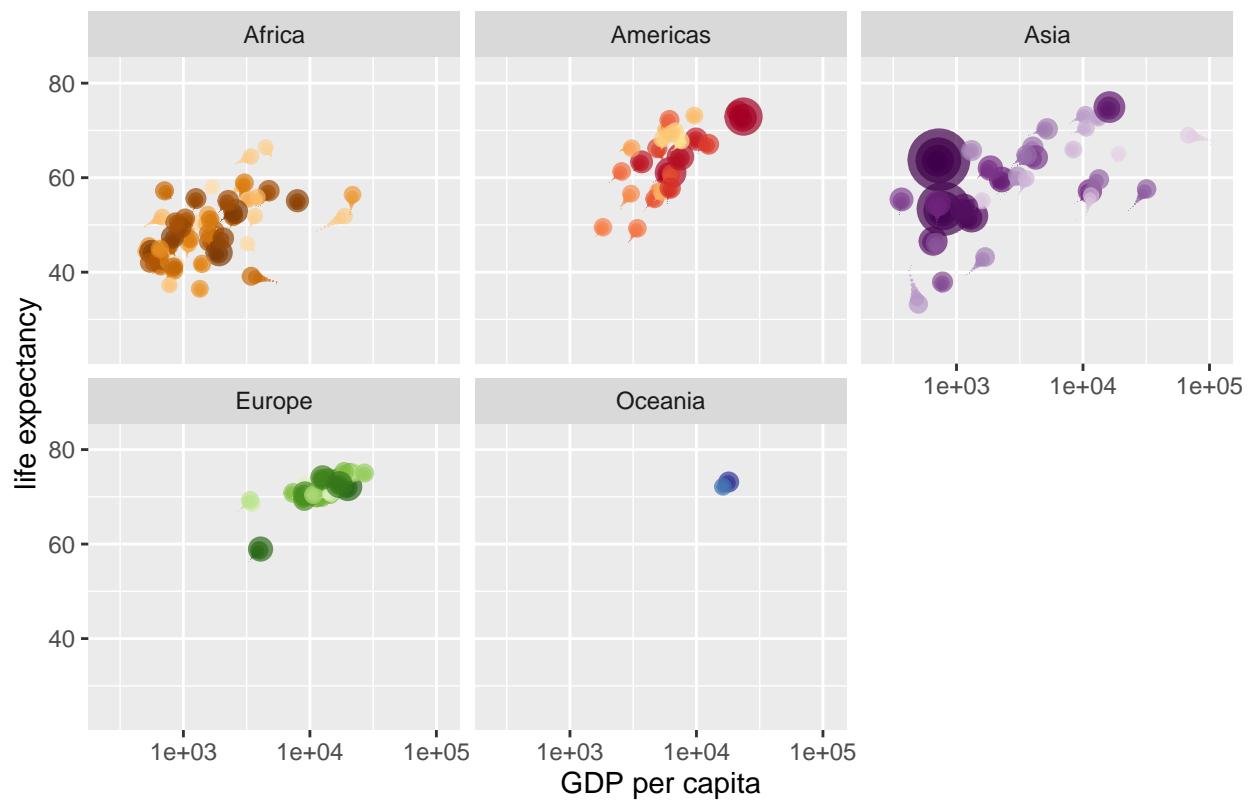
Year: 1975



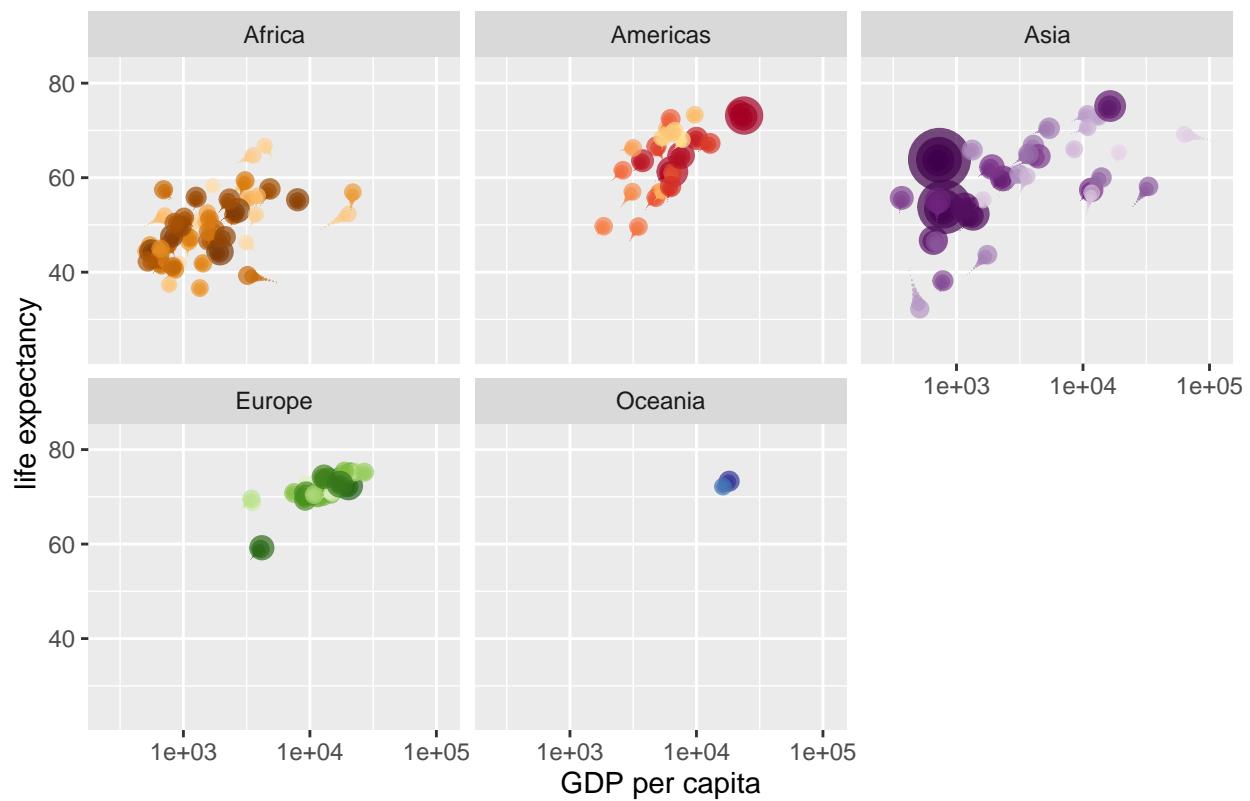
Year: 1975



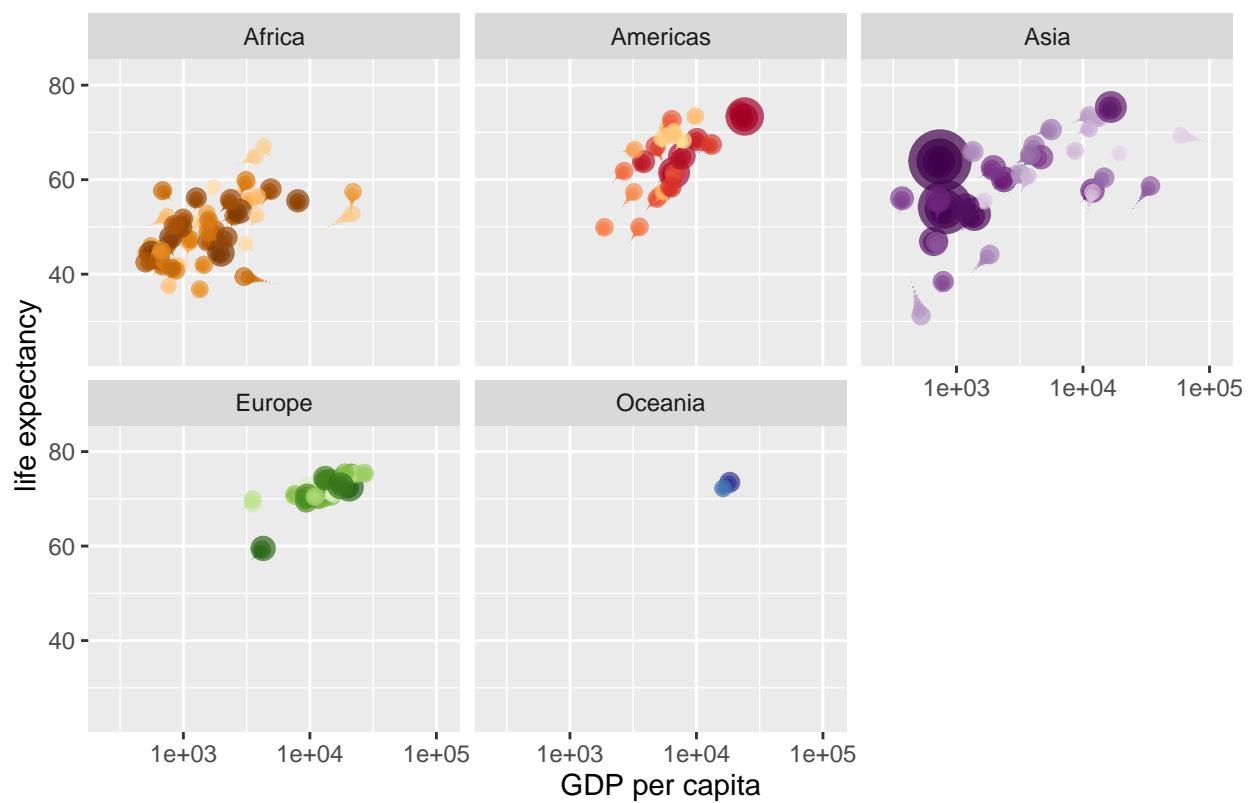
Year: 1976



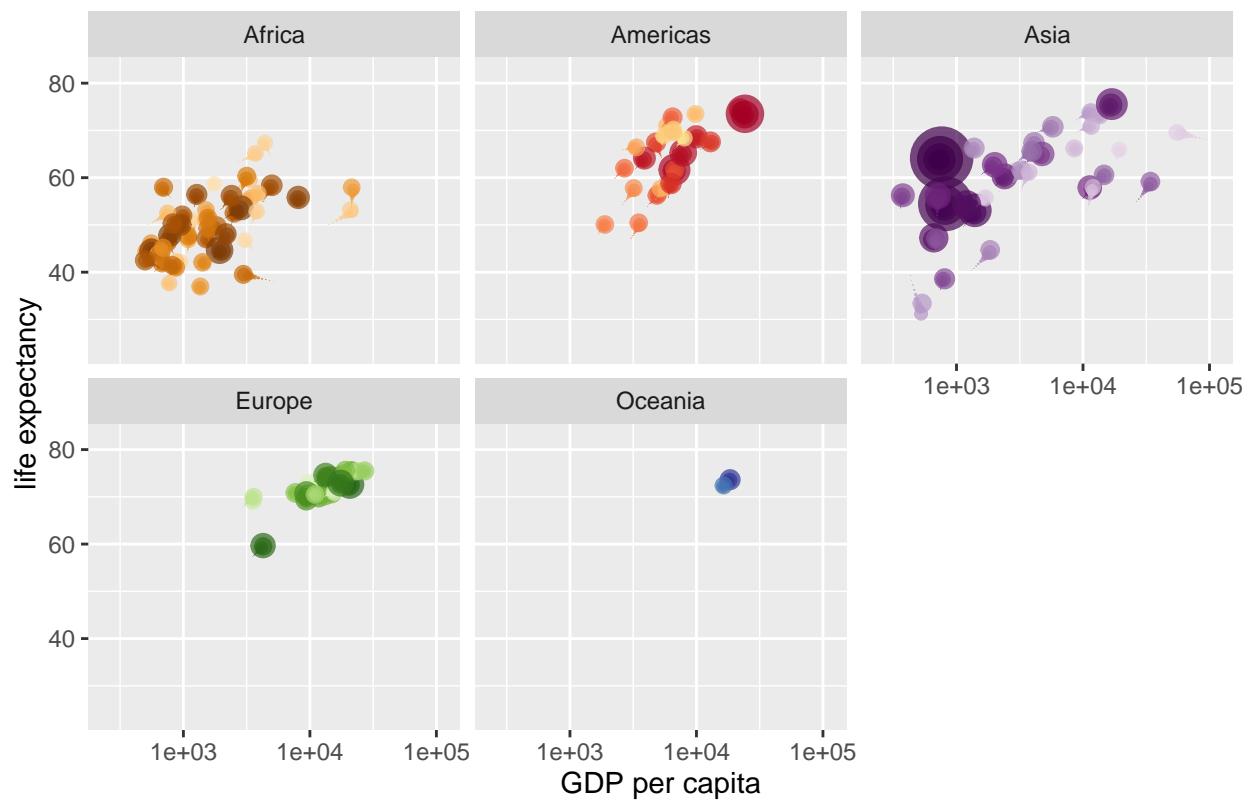
Year: 1976



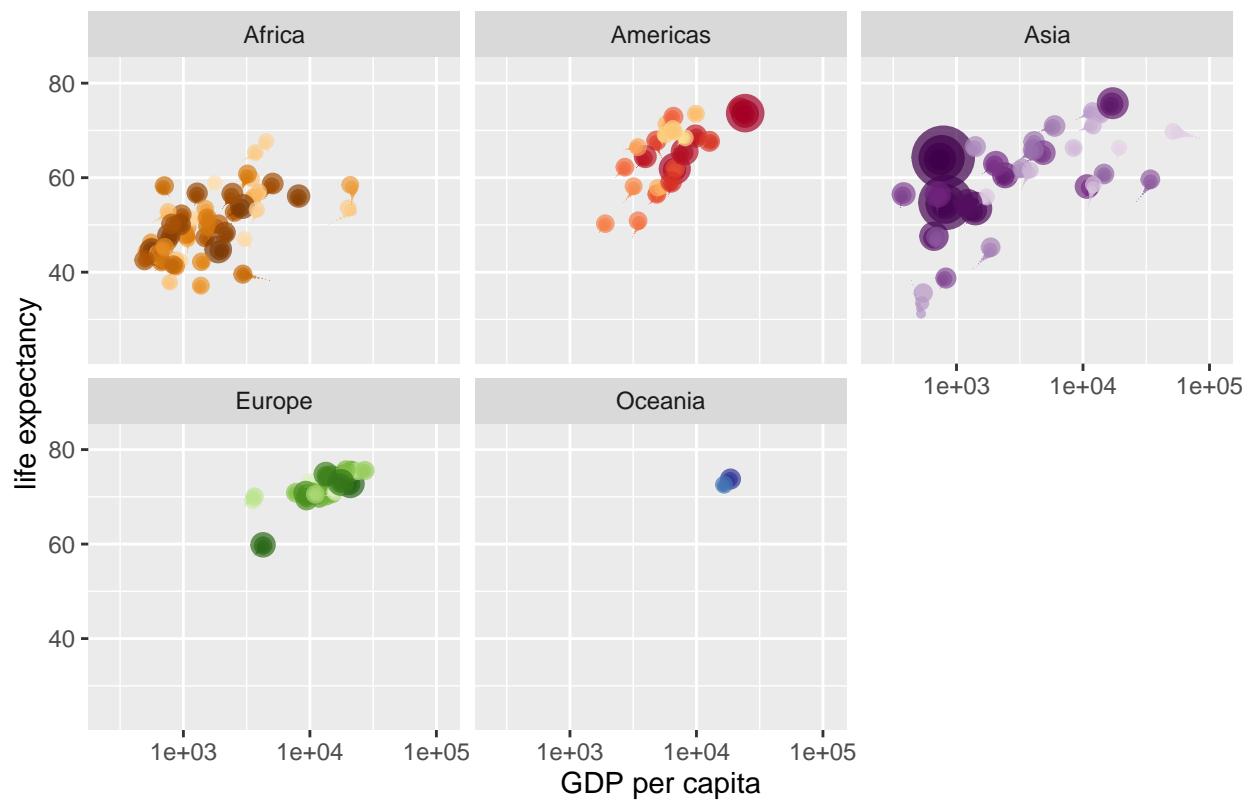
Year: 1977



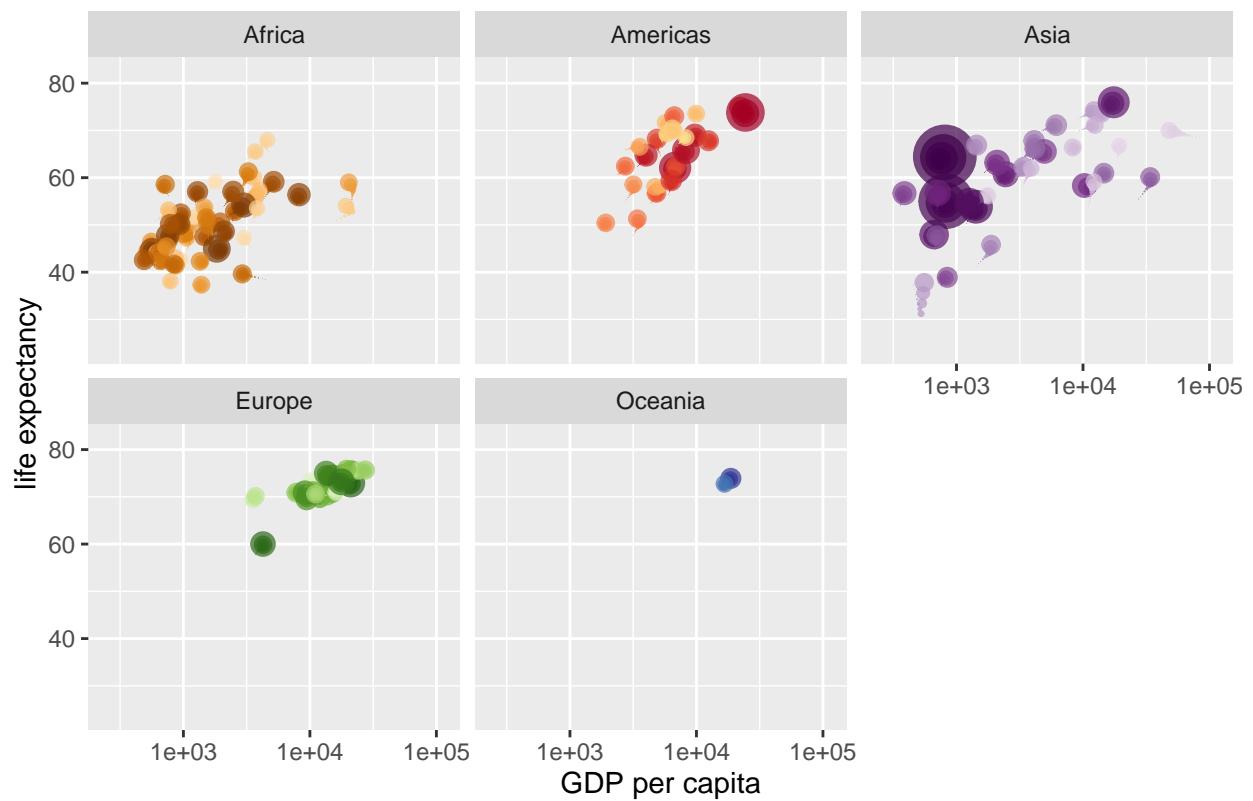
Year: 1978



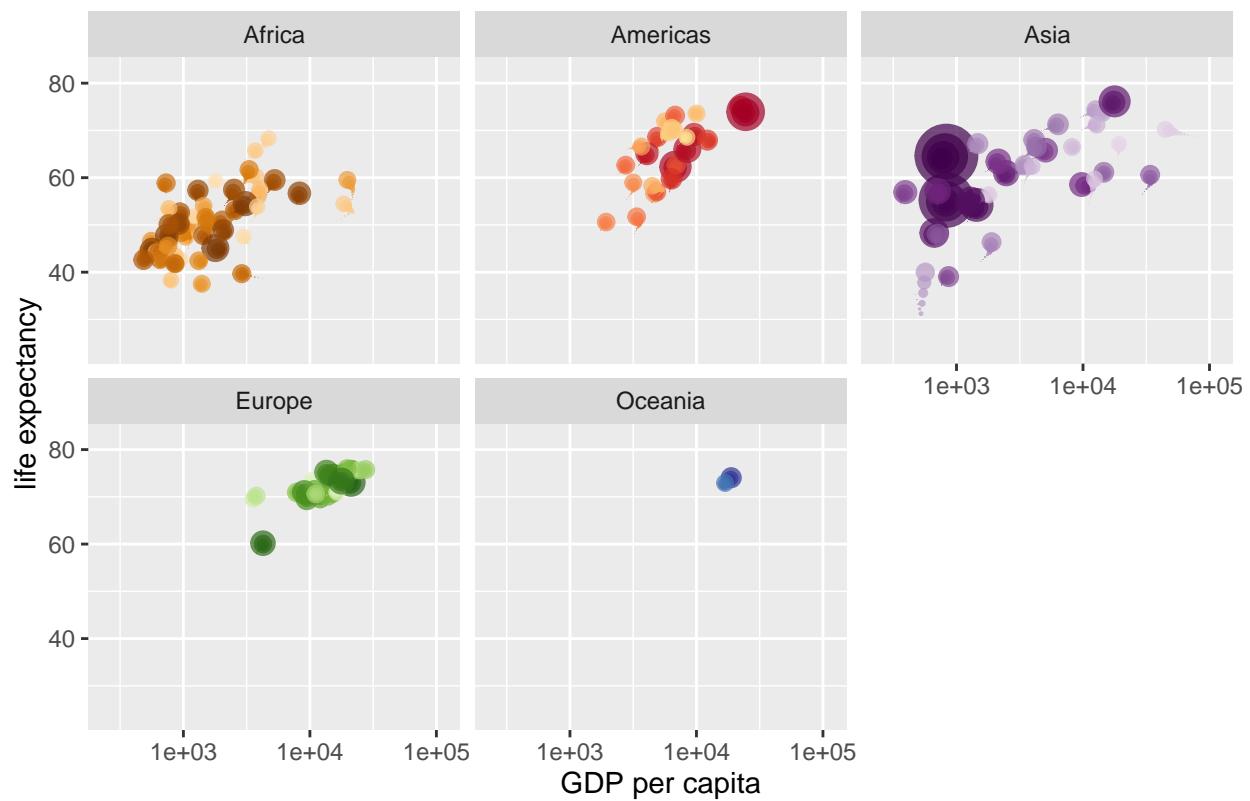
Year: 1978



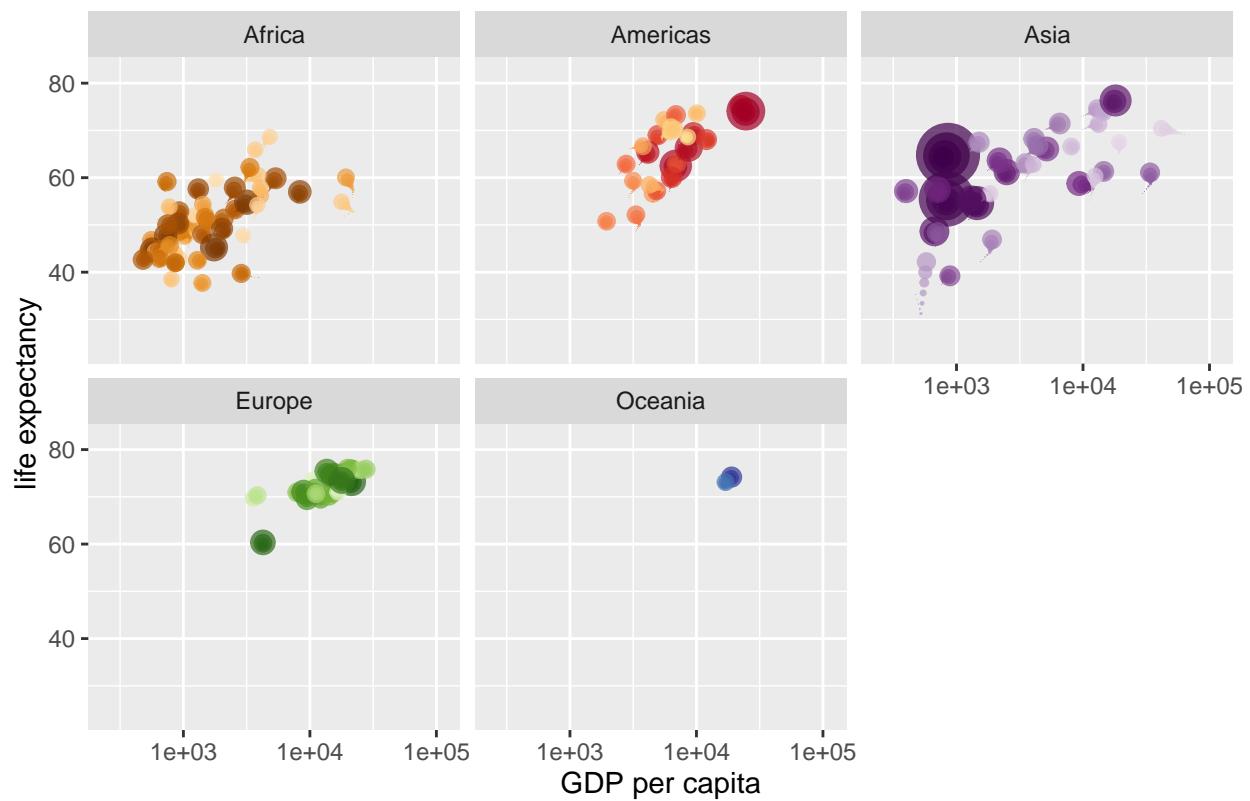
Year: 1979



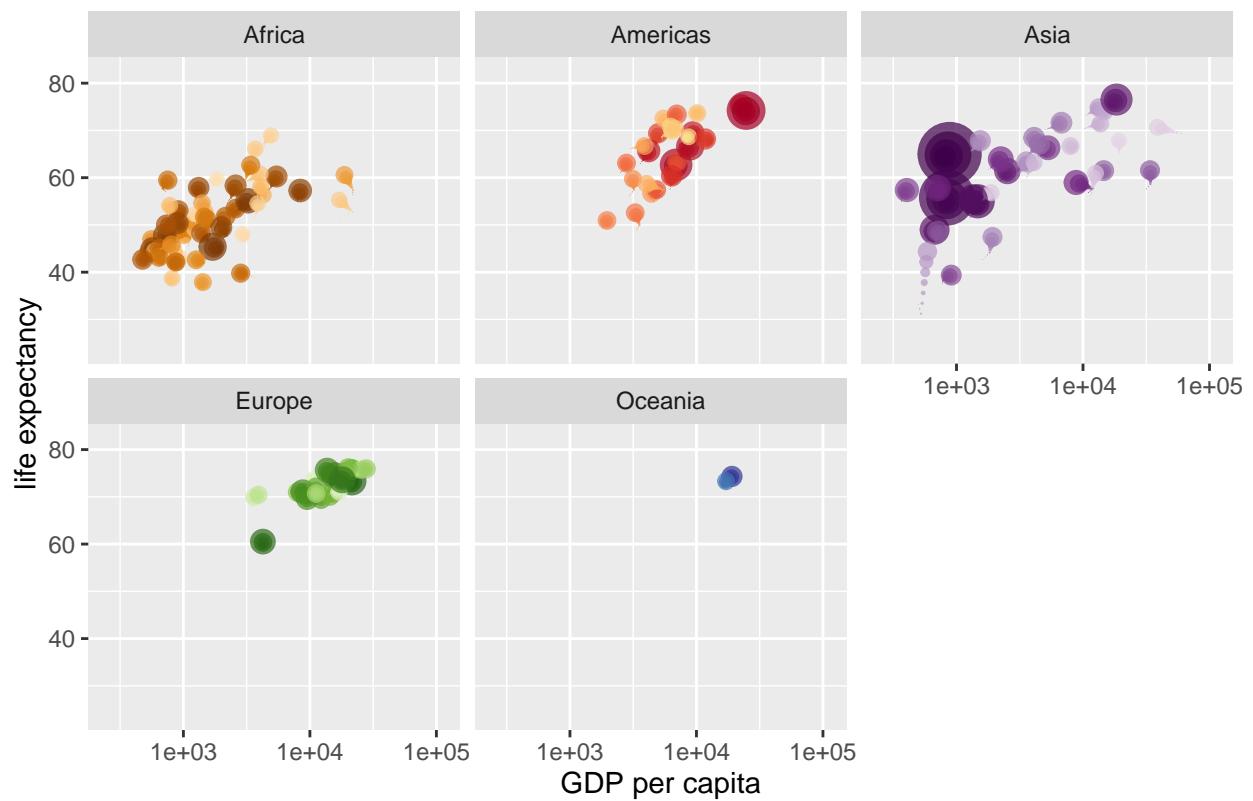
Year: 1979



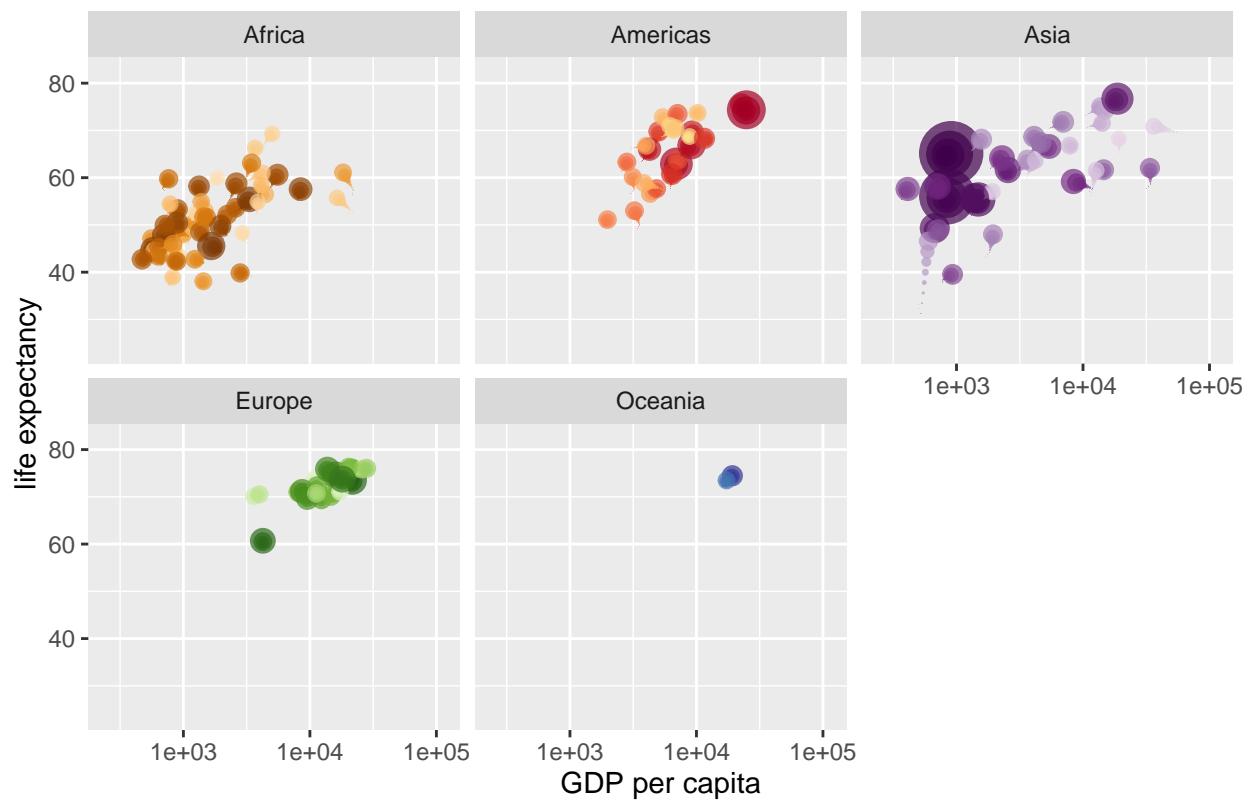
Year: 1980



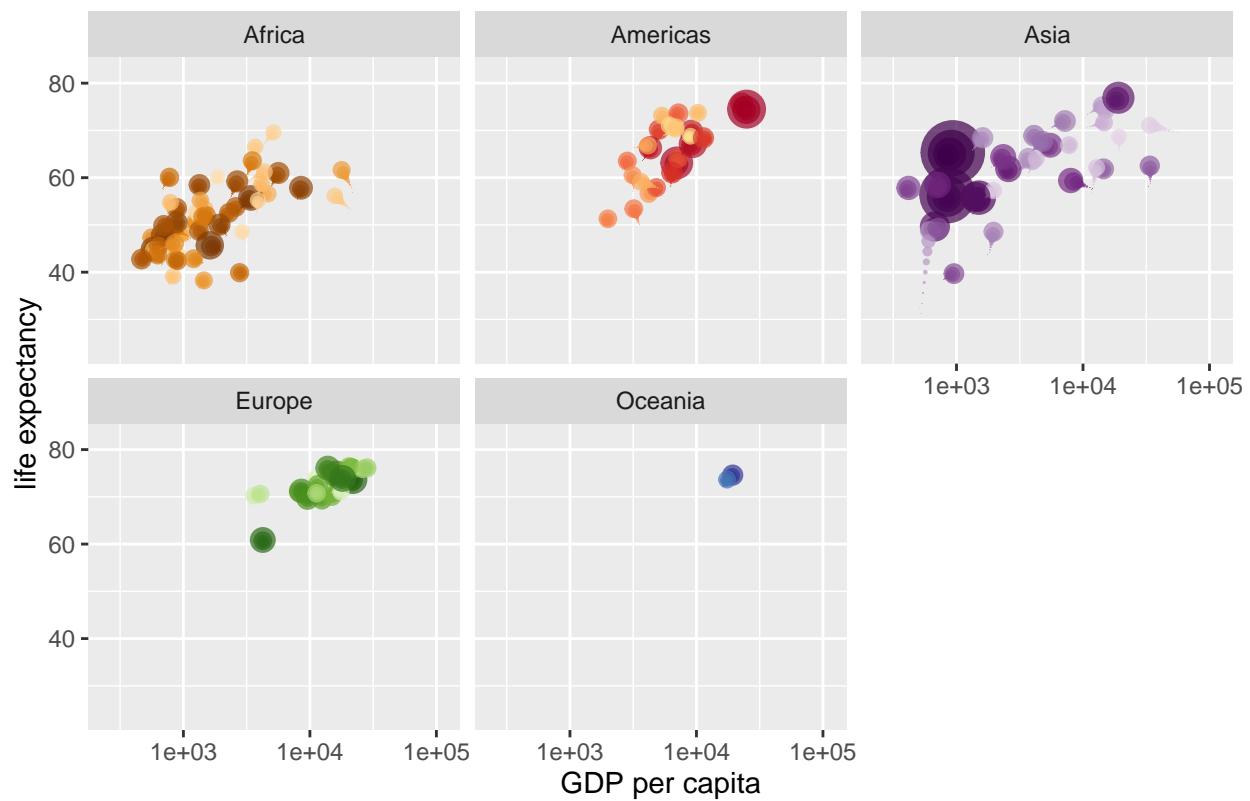
Year: 1980



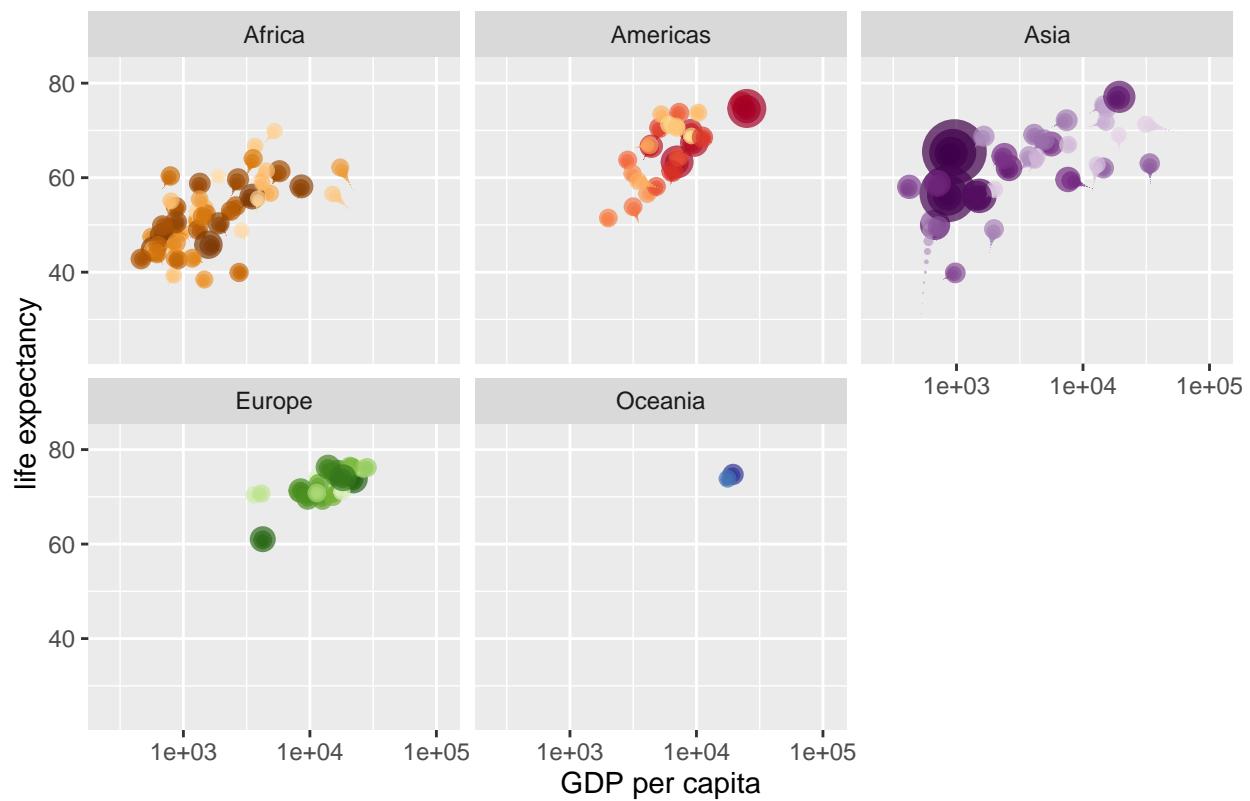
Year: 1981



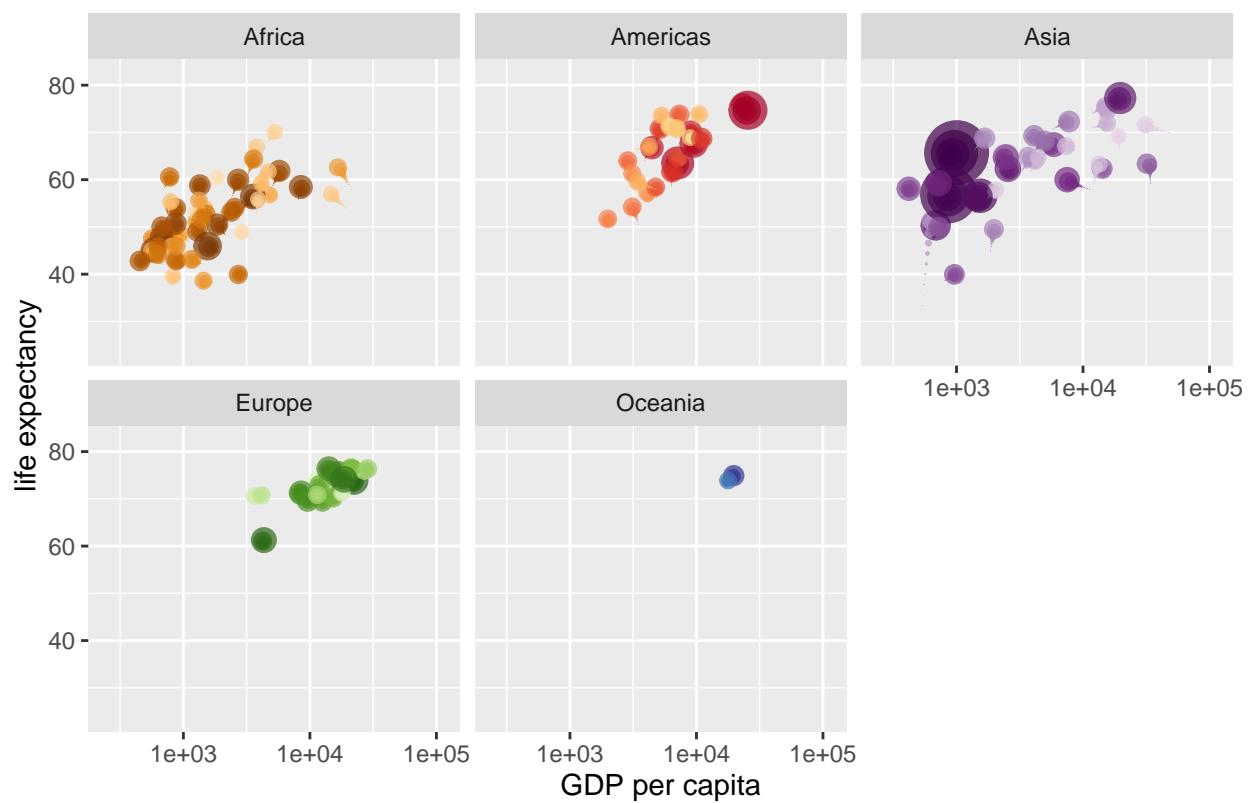
Year: 1981



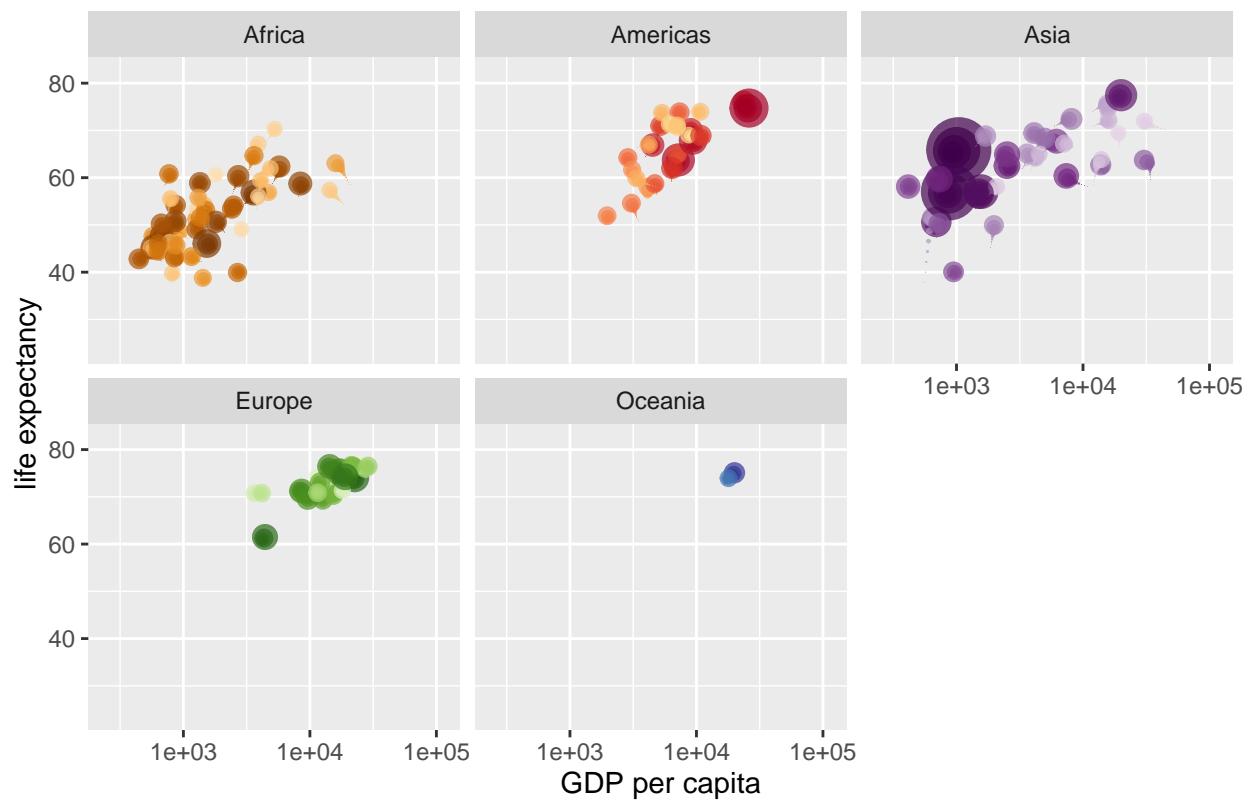
Year: 1982



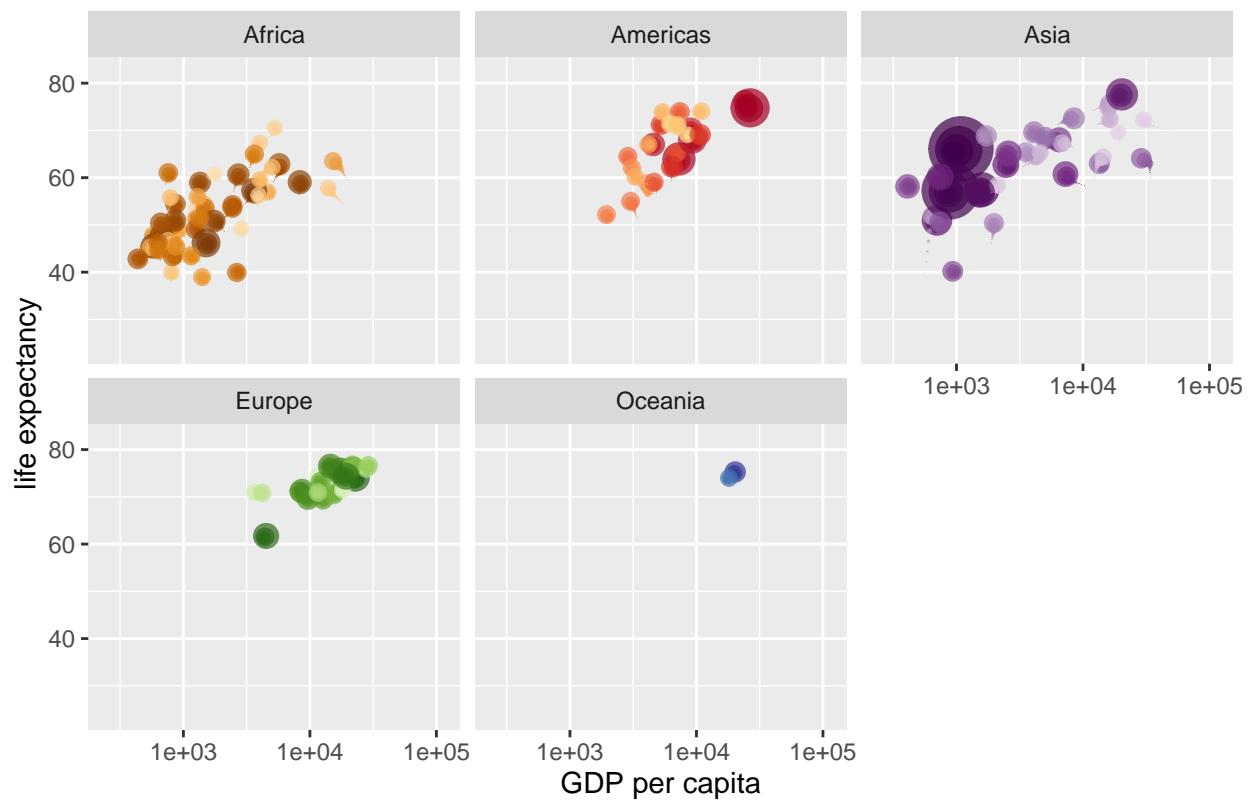
Year: 1983



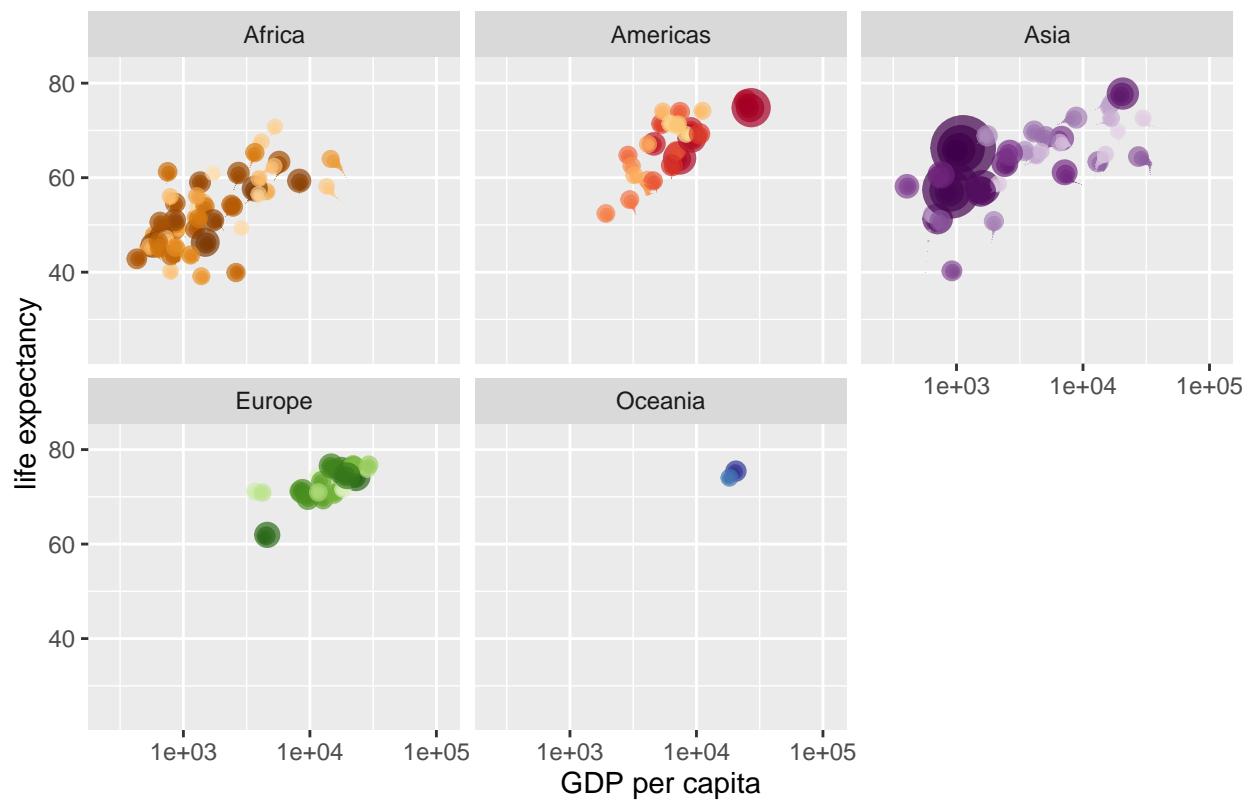
Year: 1983



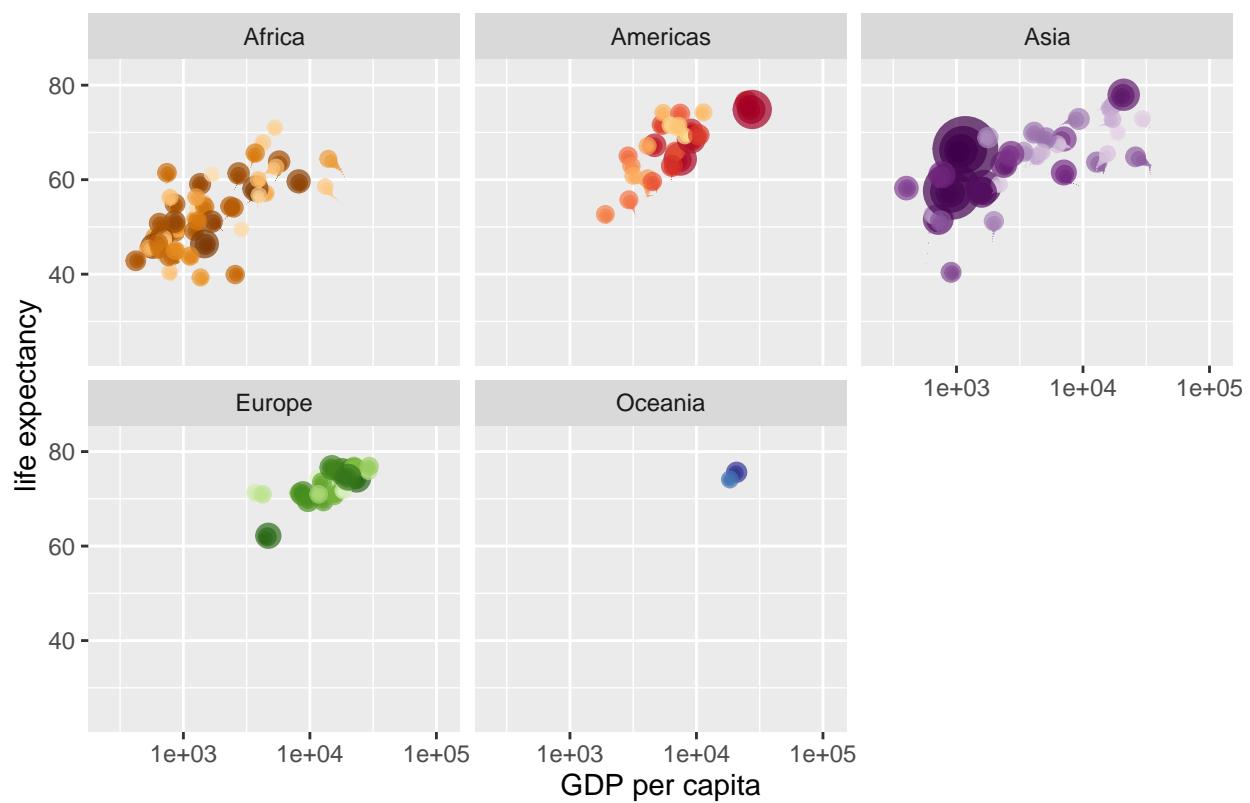
Year: 1984



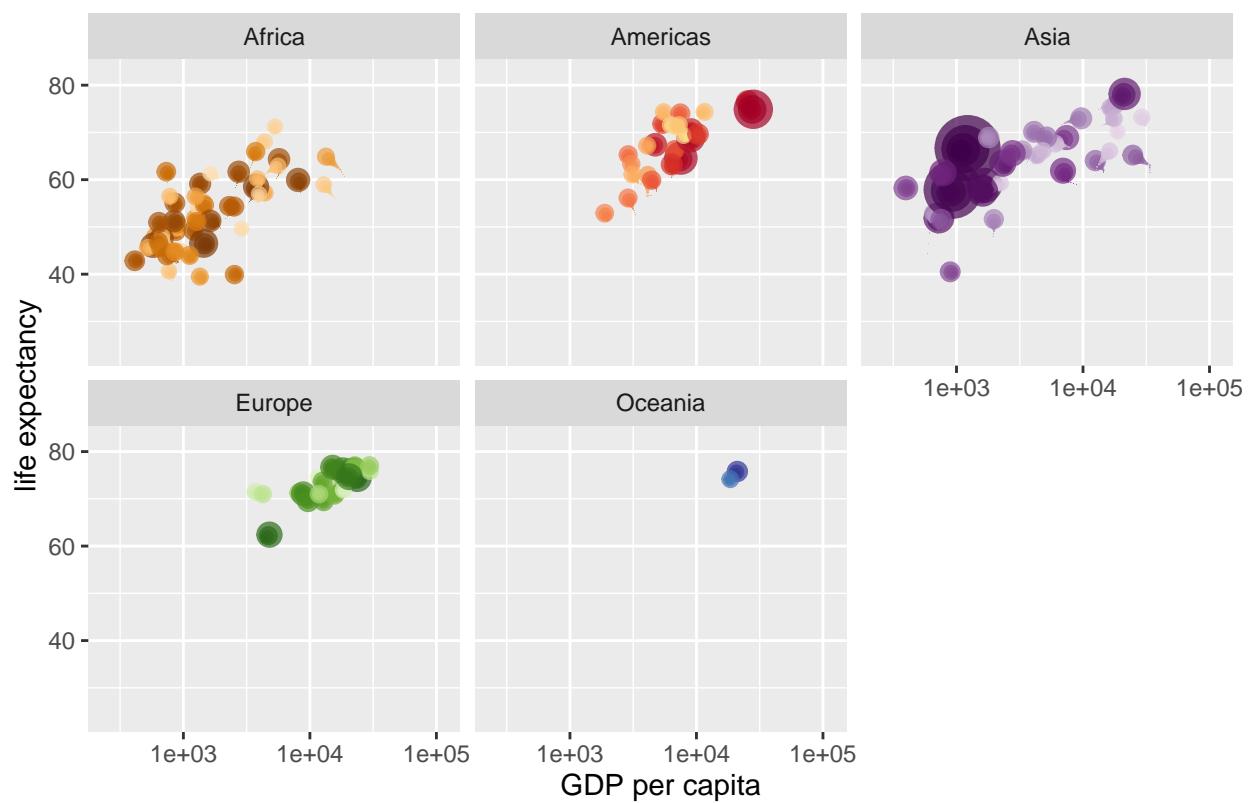
Year: 1984



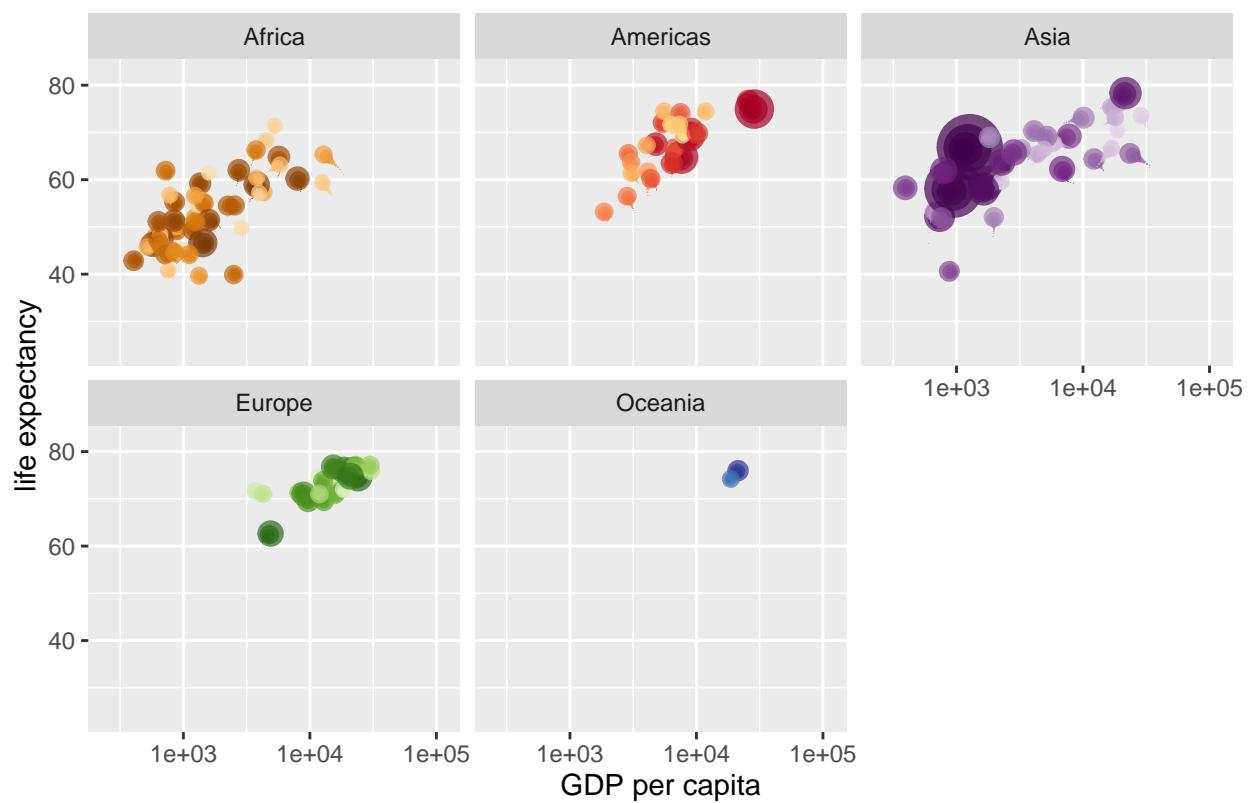
Year: 1985



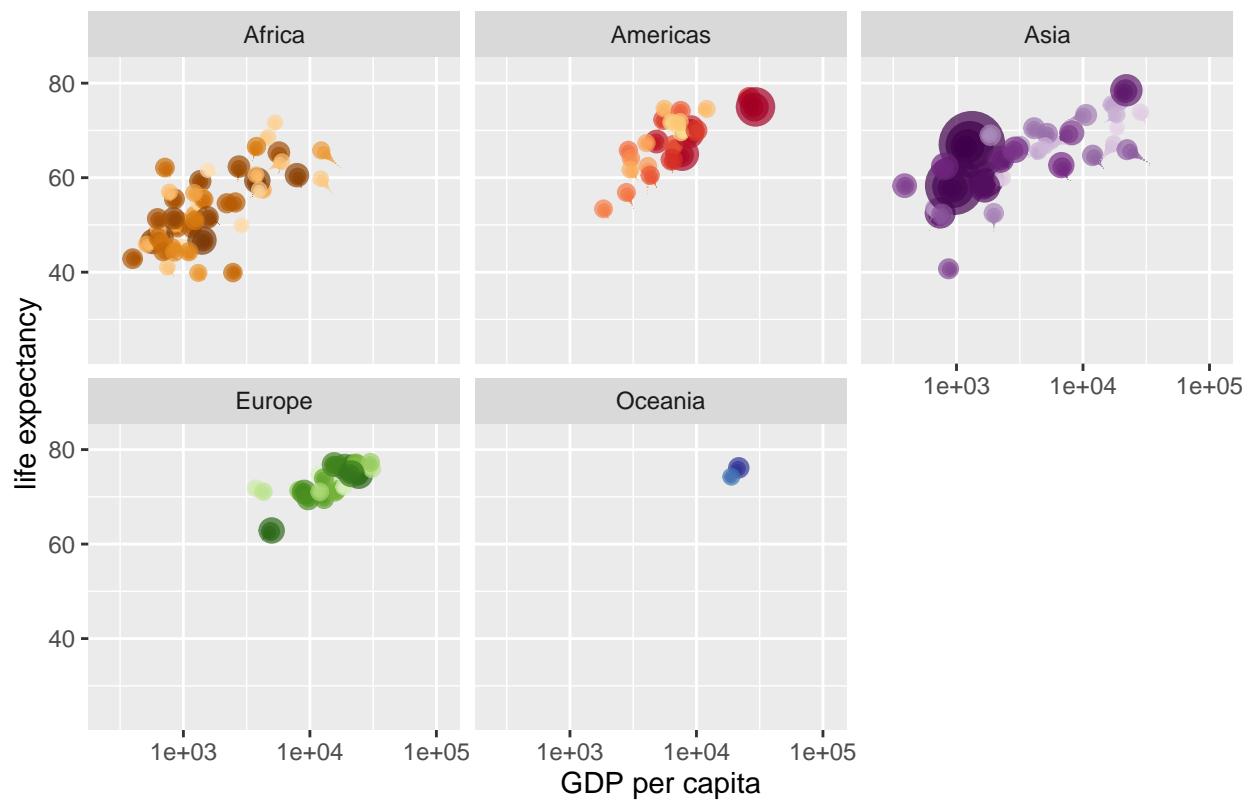
Year: 1985



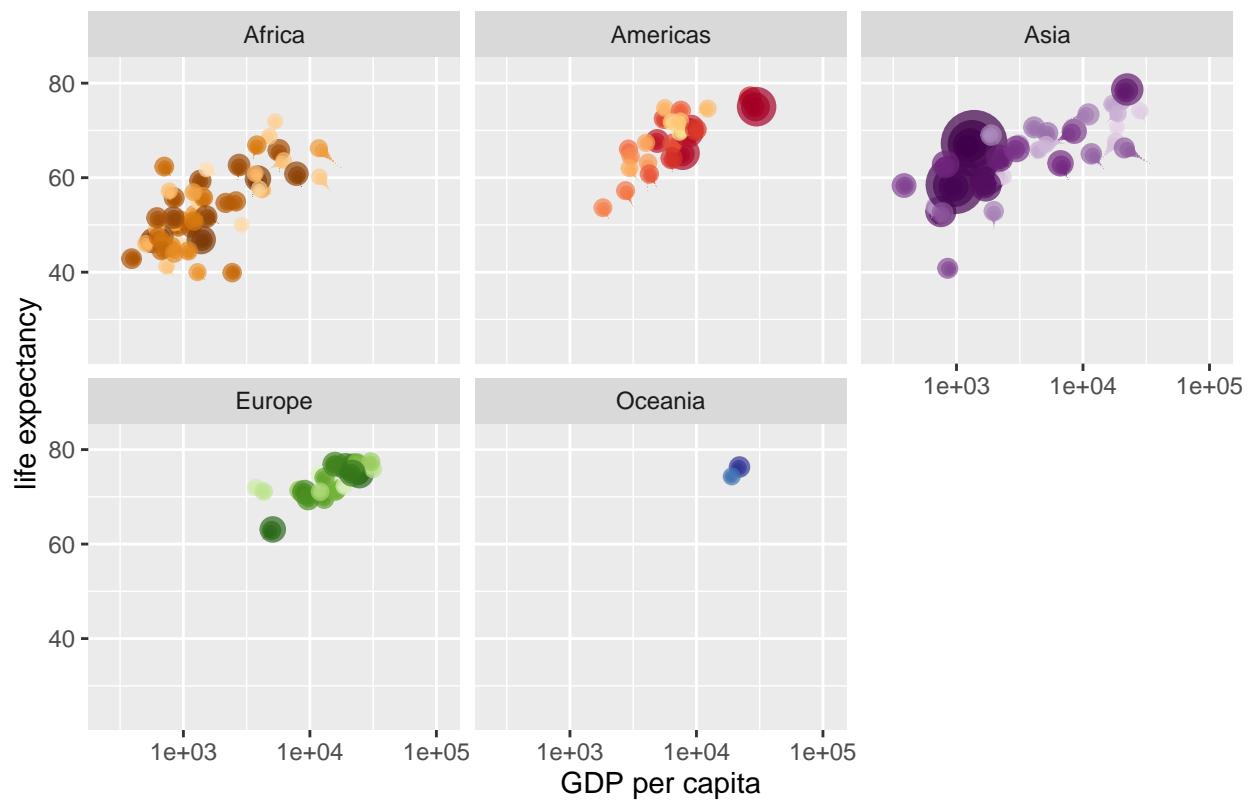
Year: 1986



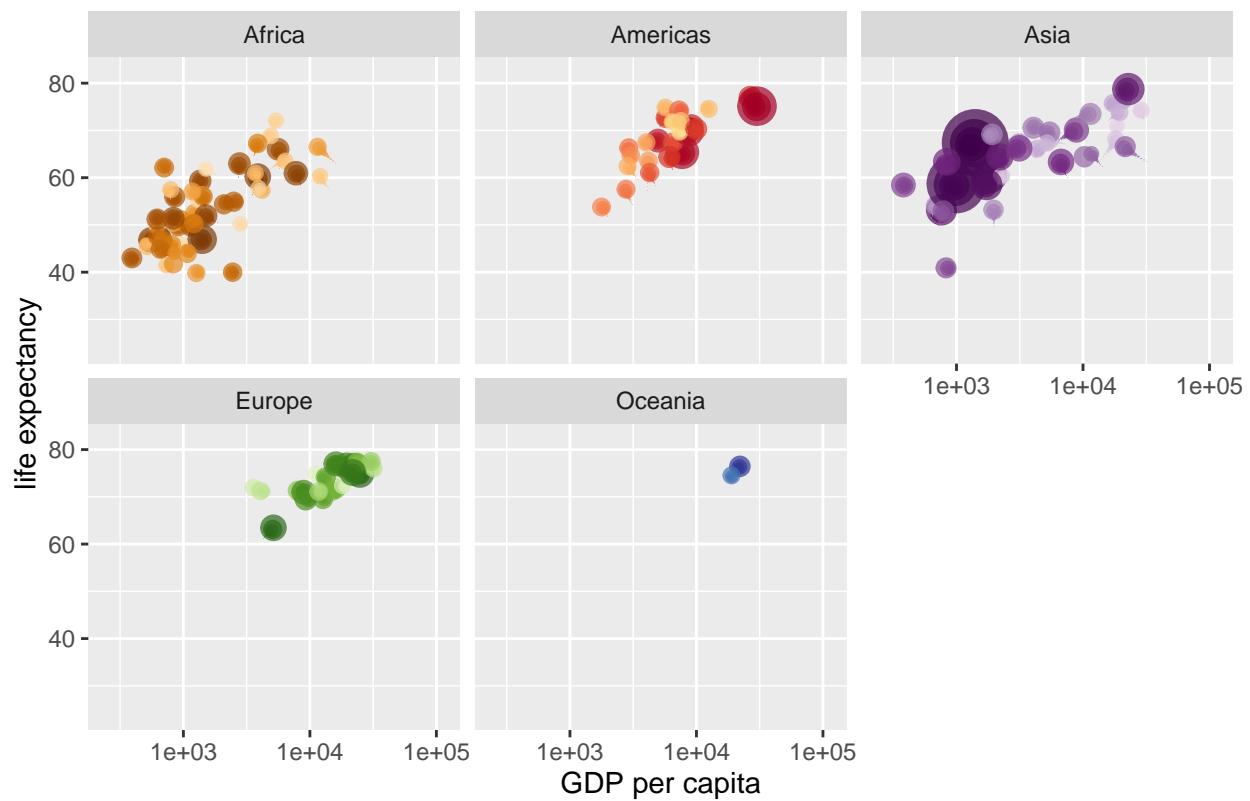
Year: 1986



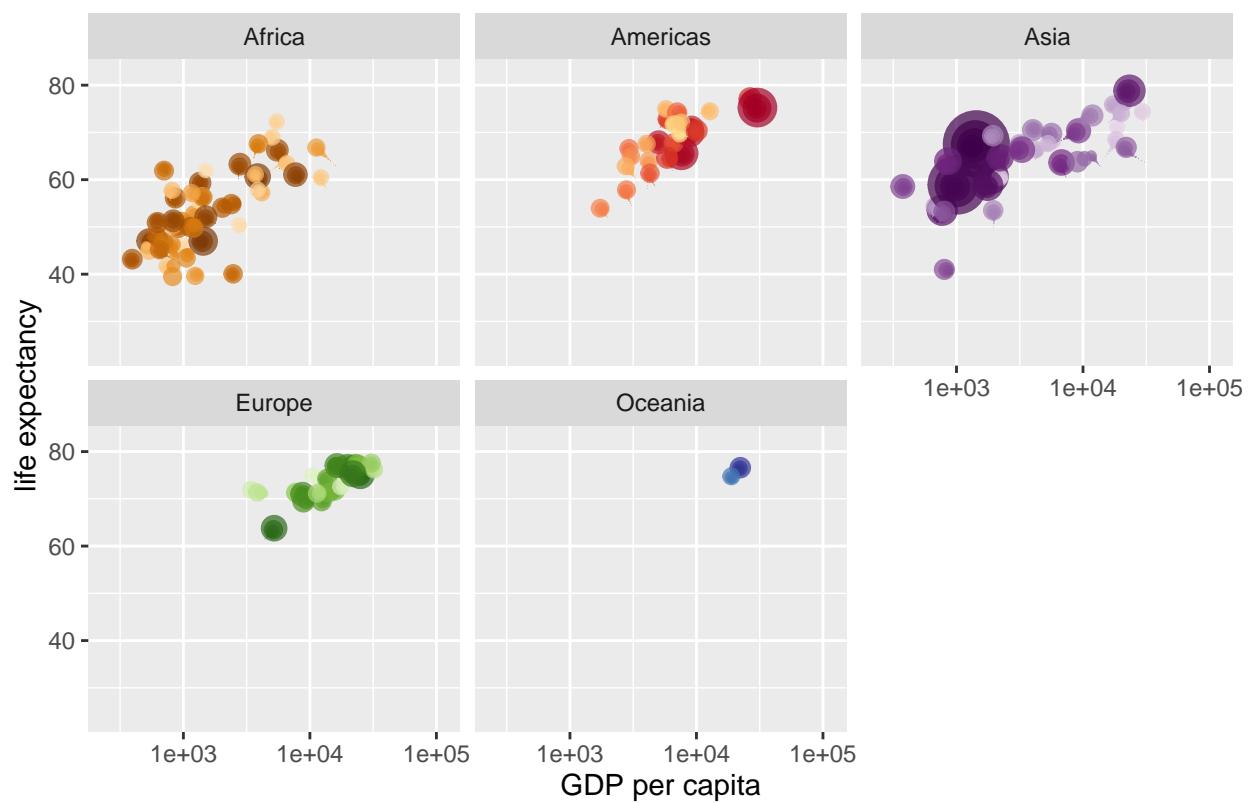
Year: 1987



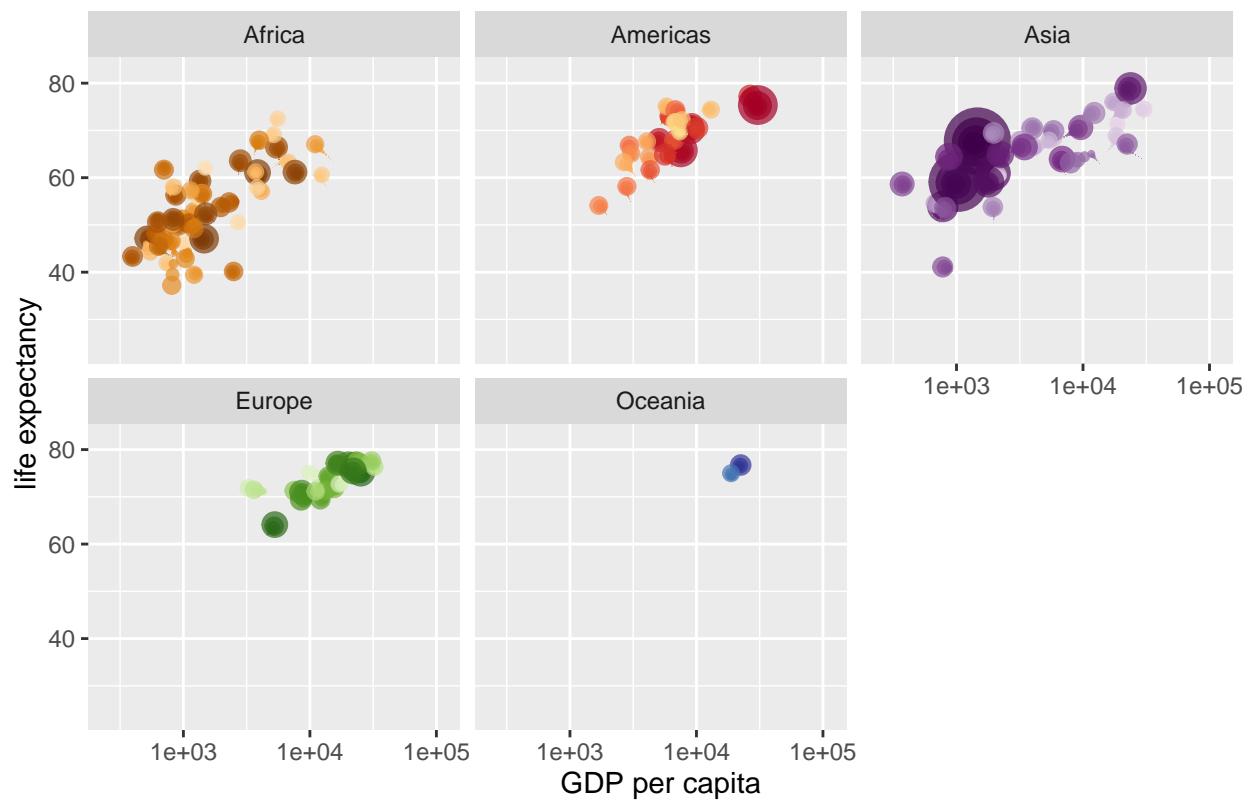
Year: 1988



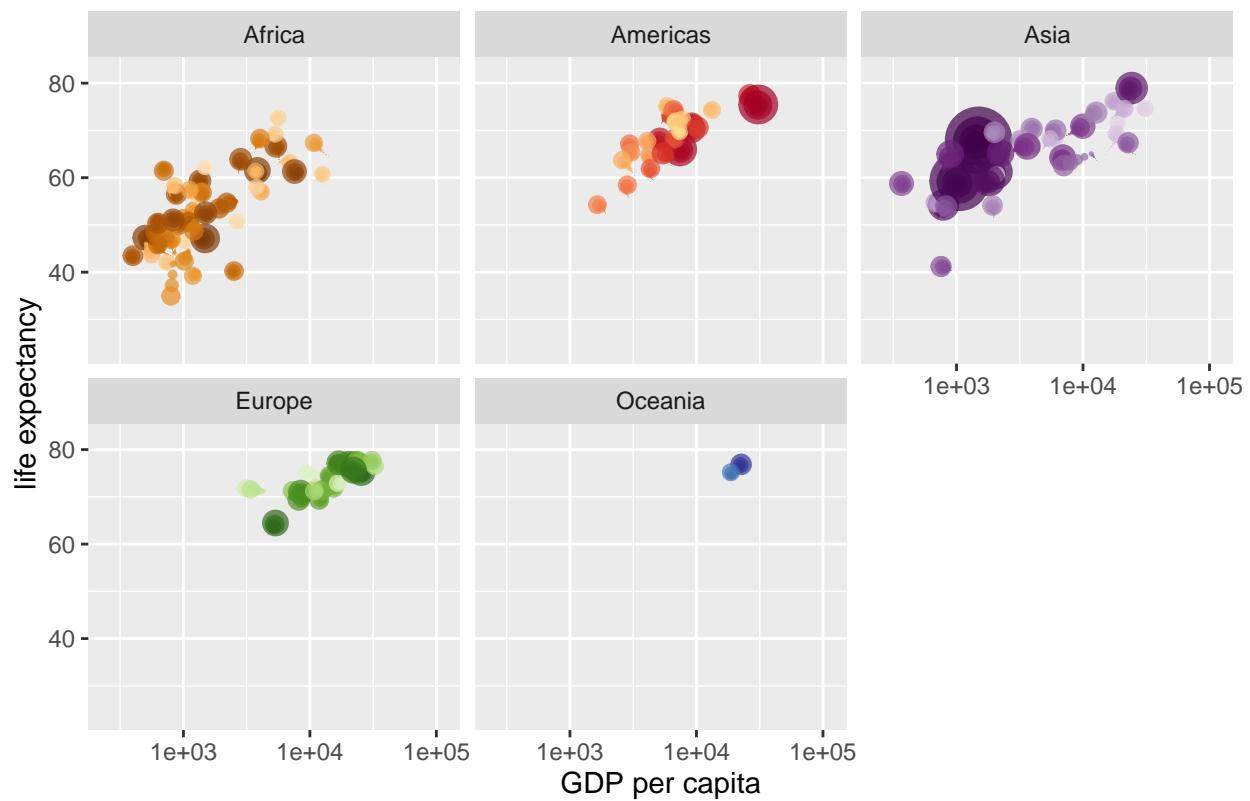
Year: 1988



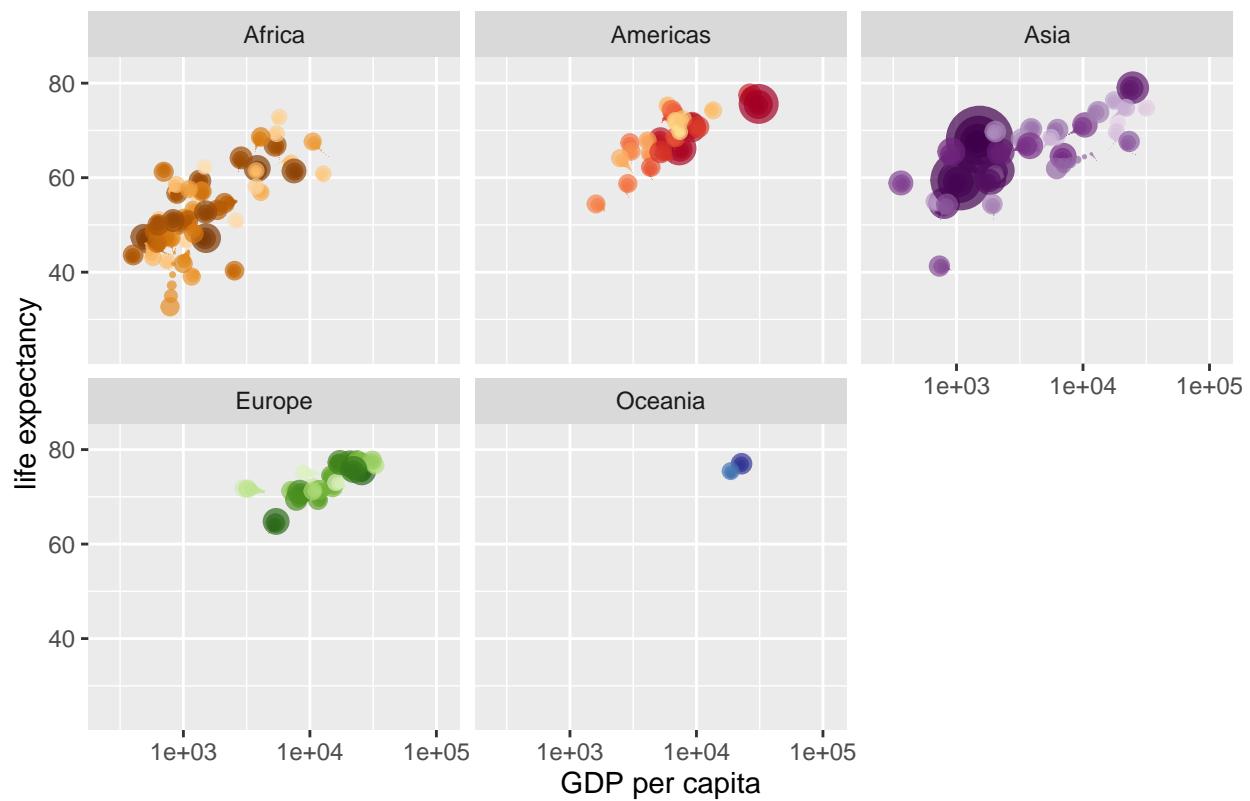
Year: 1989



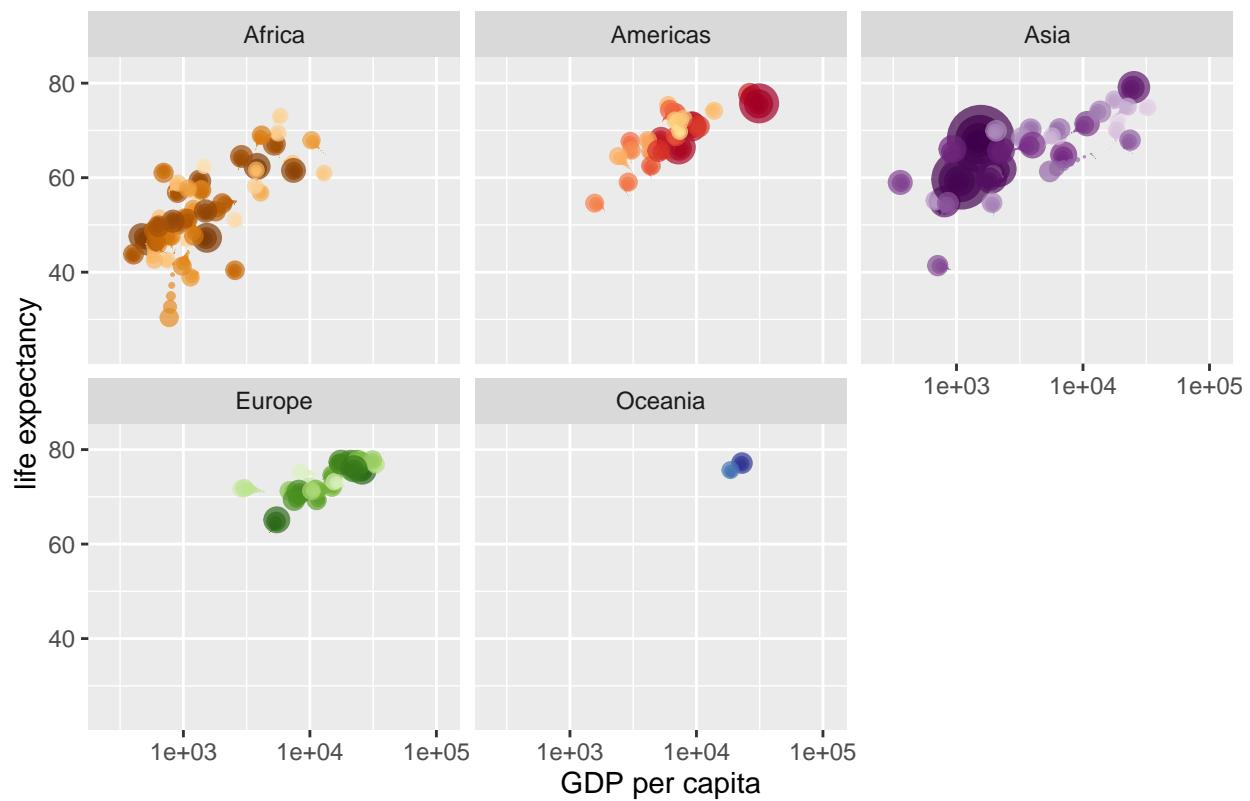
Year: 1989



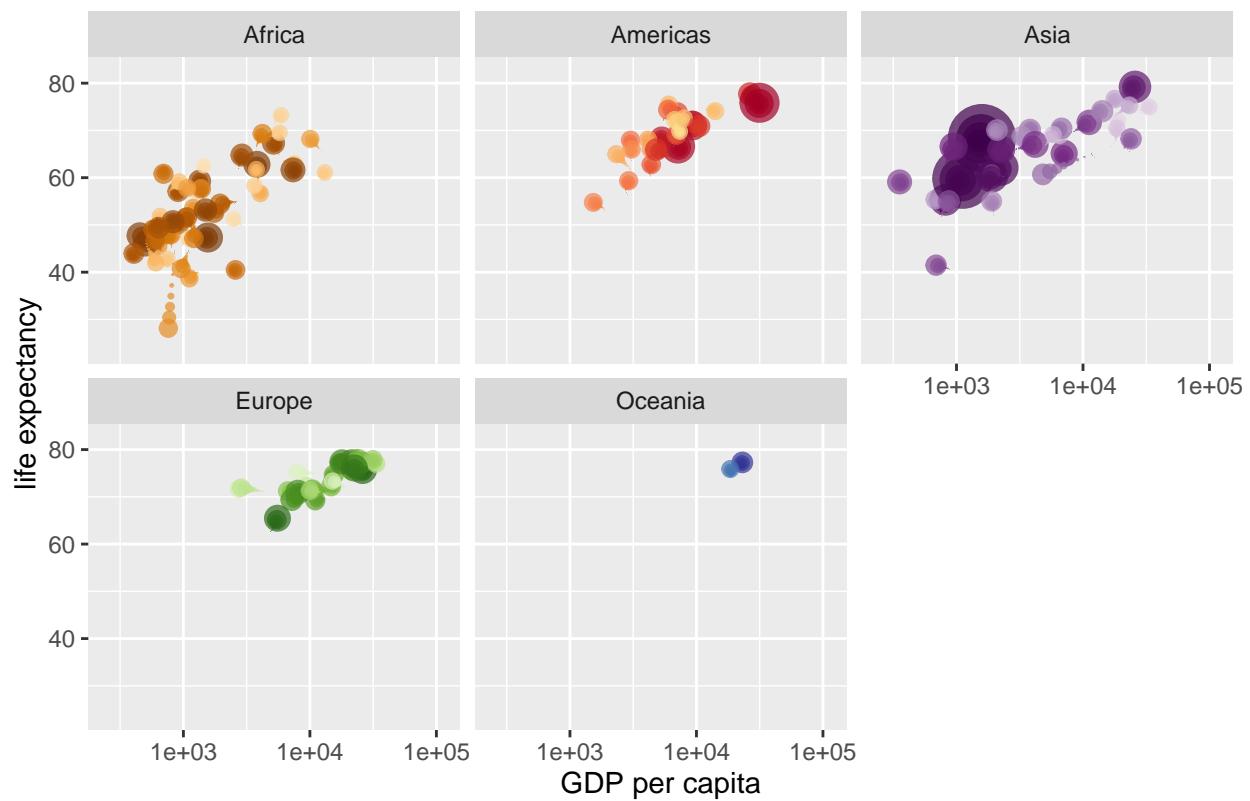
Year: 1990



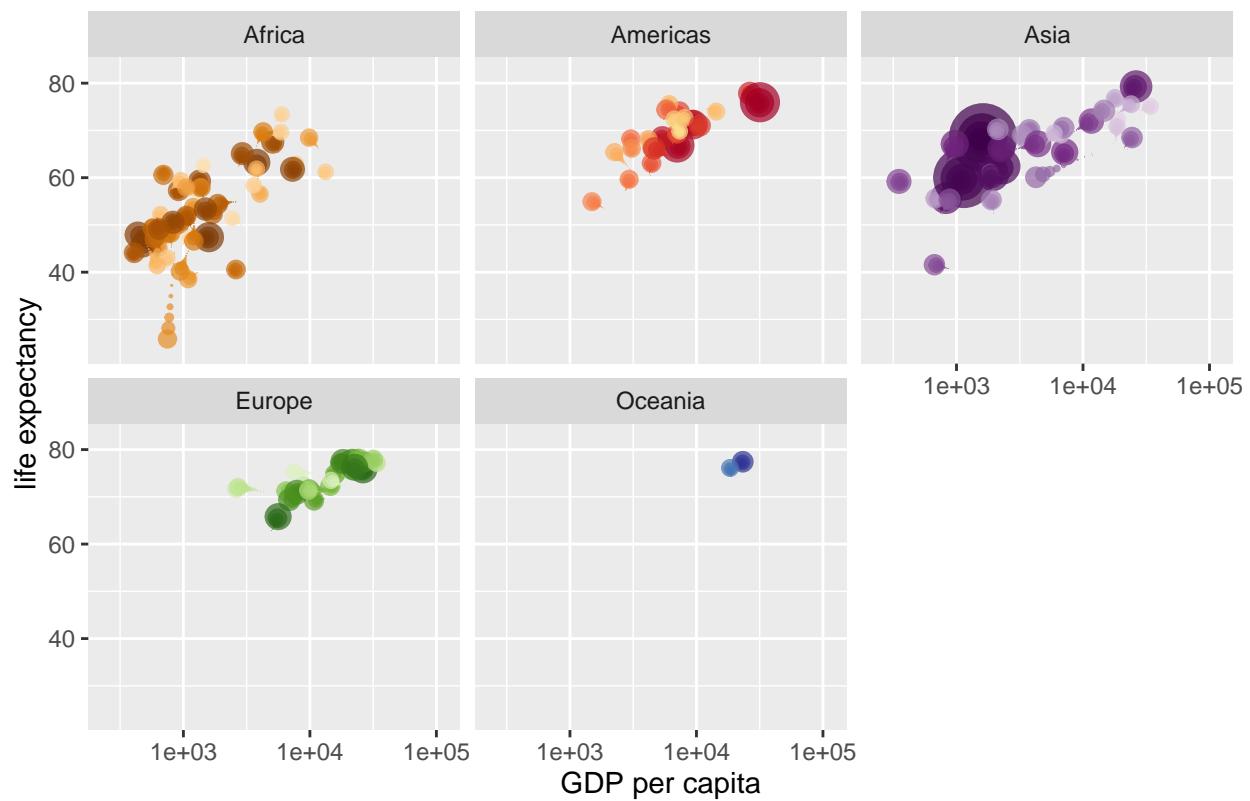
Year: 1990



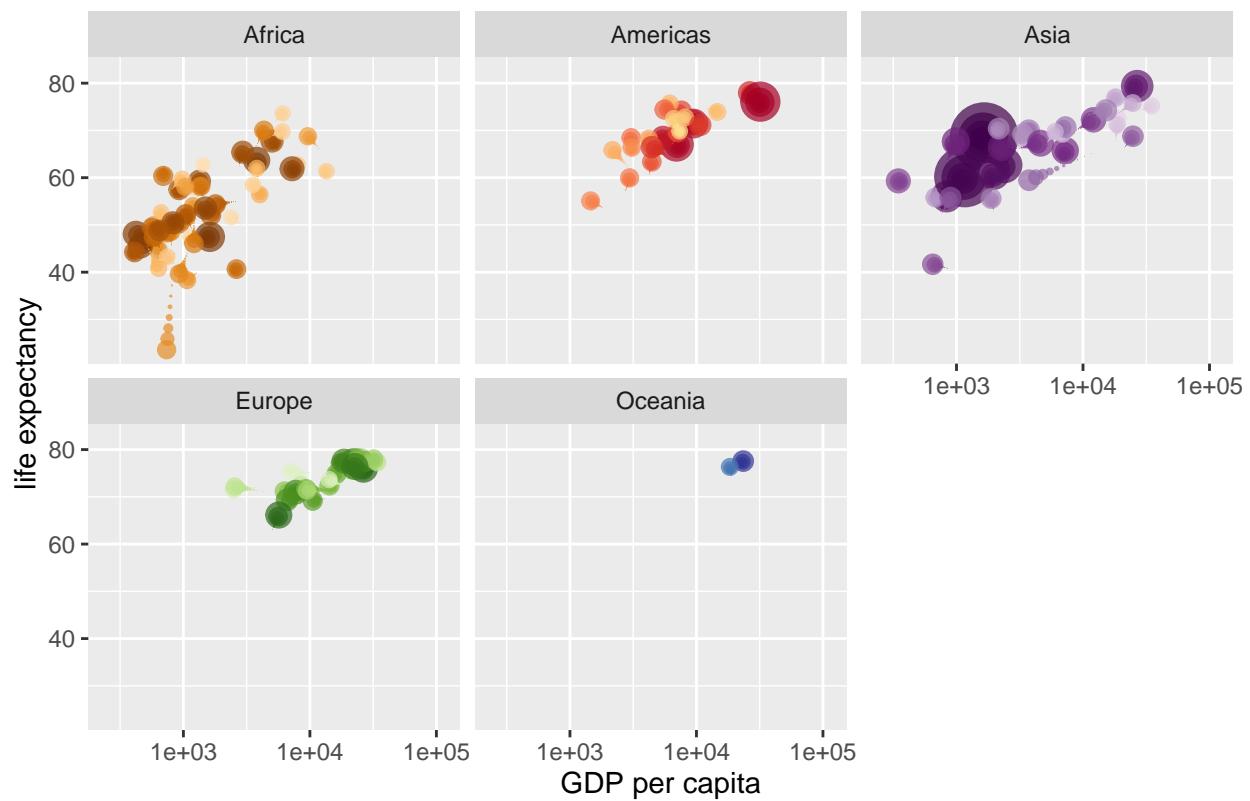
Year: 1991



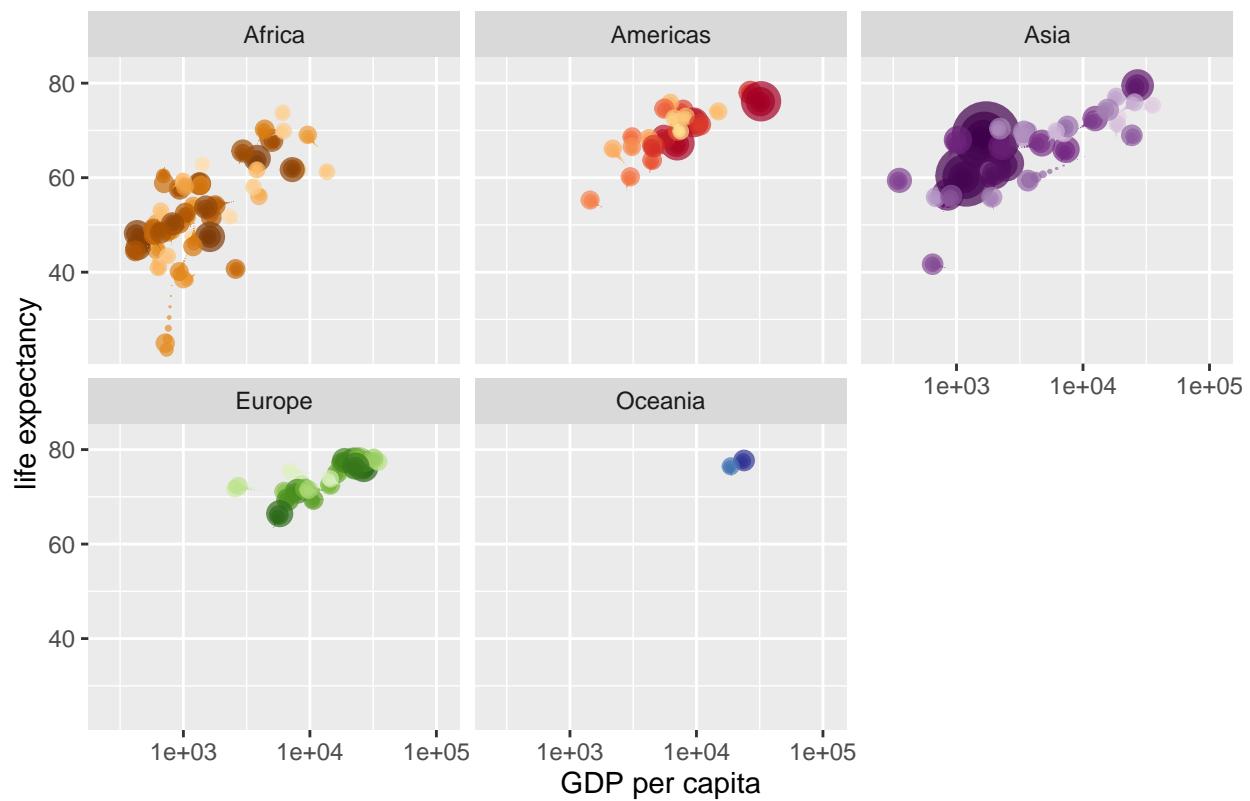
Year: 1991



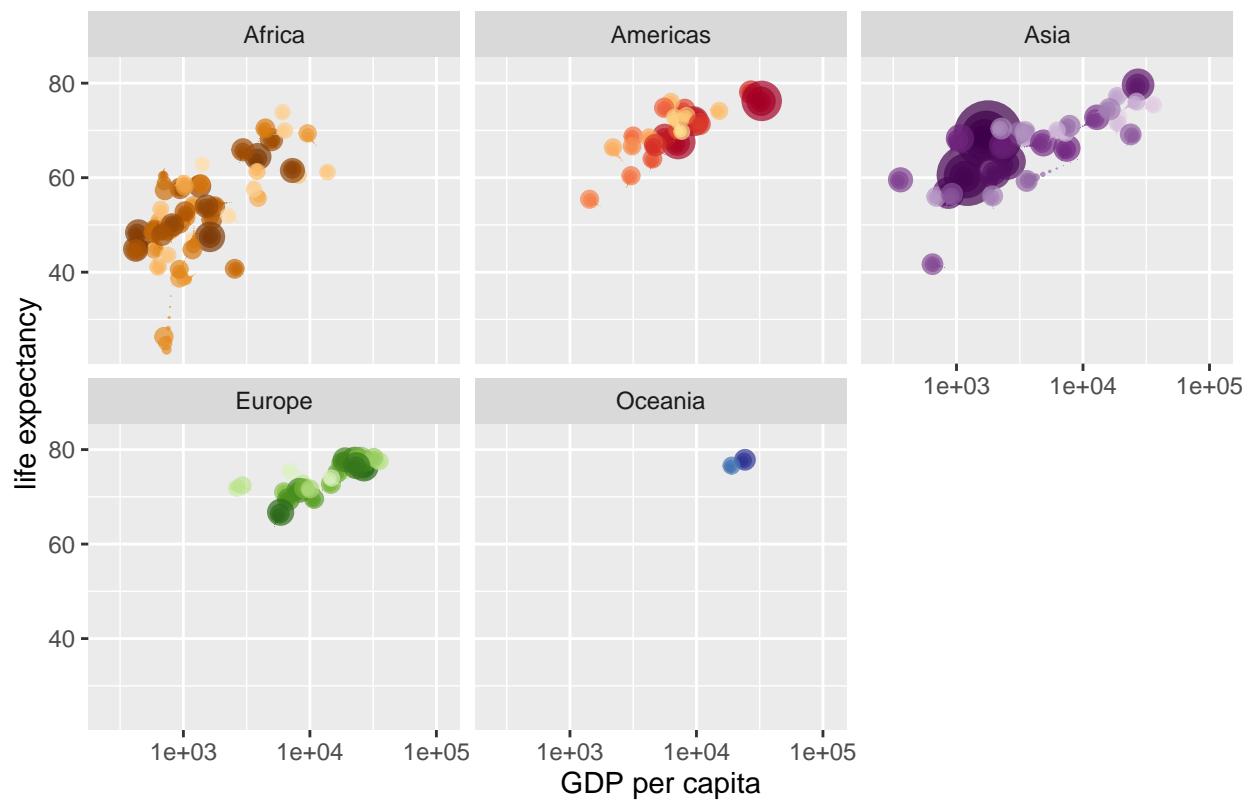
Year: 1992



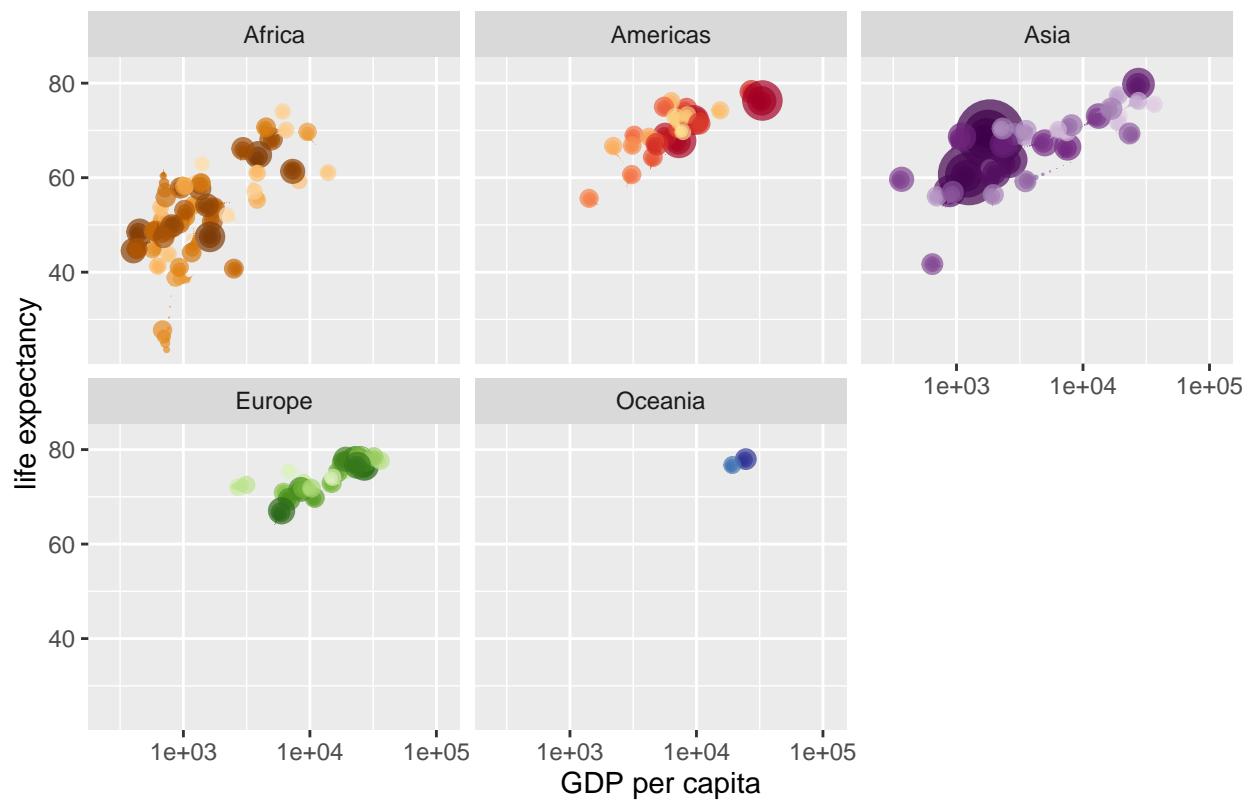
Year: 1993



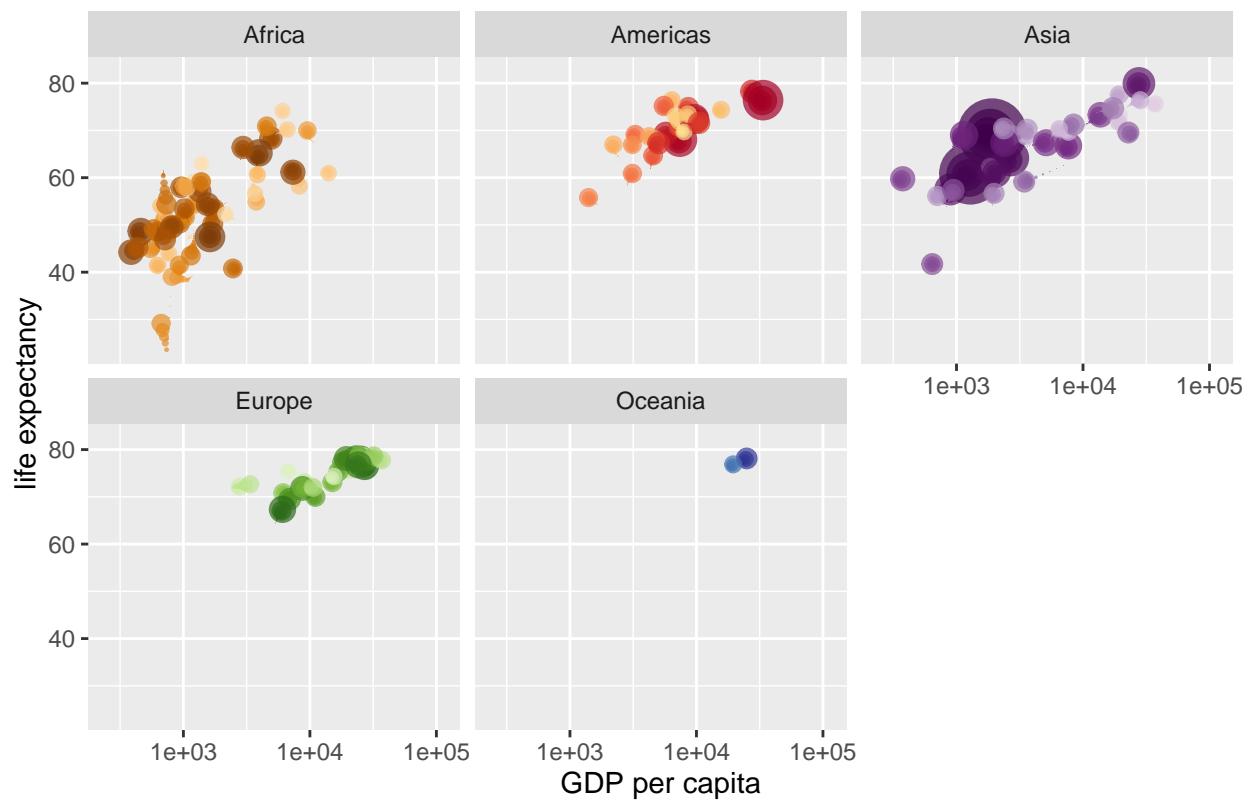
Year: 1993



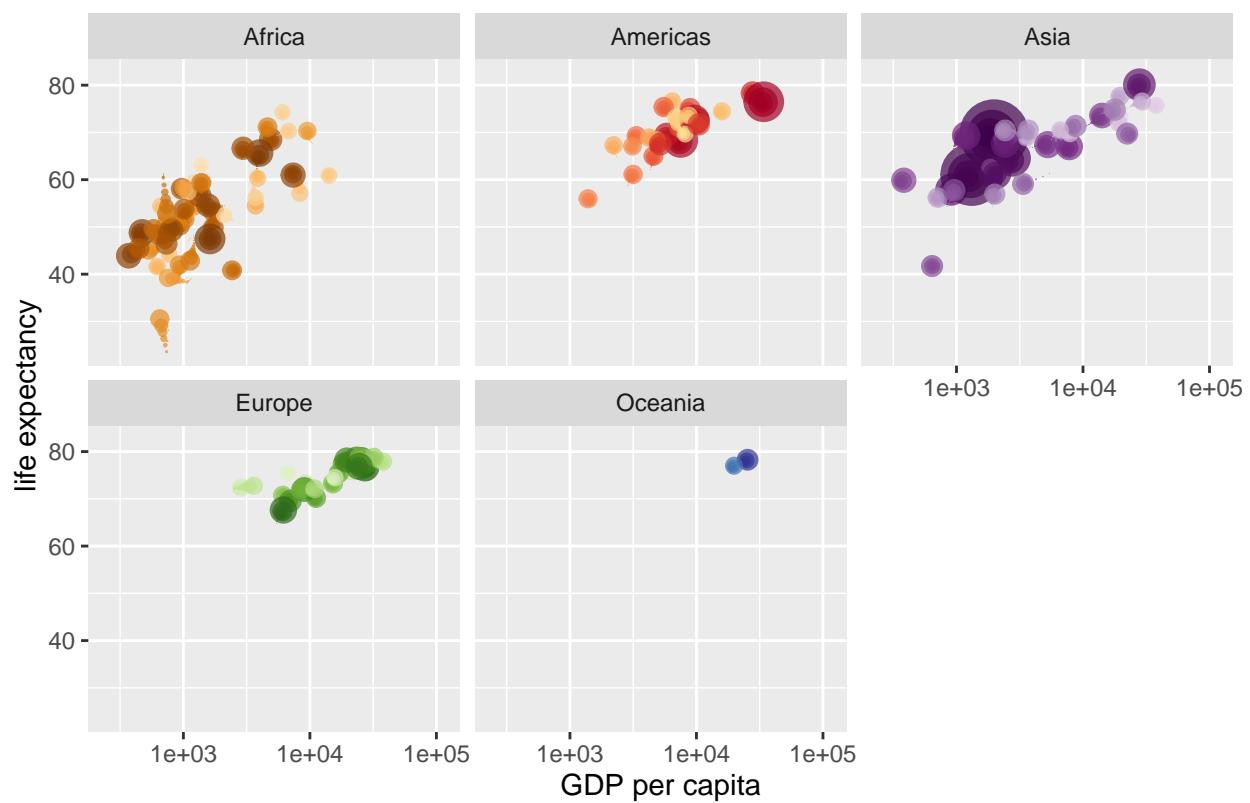
Year: 1994



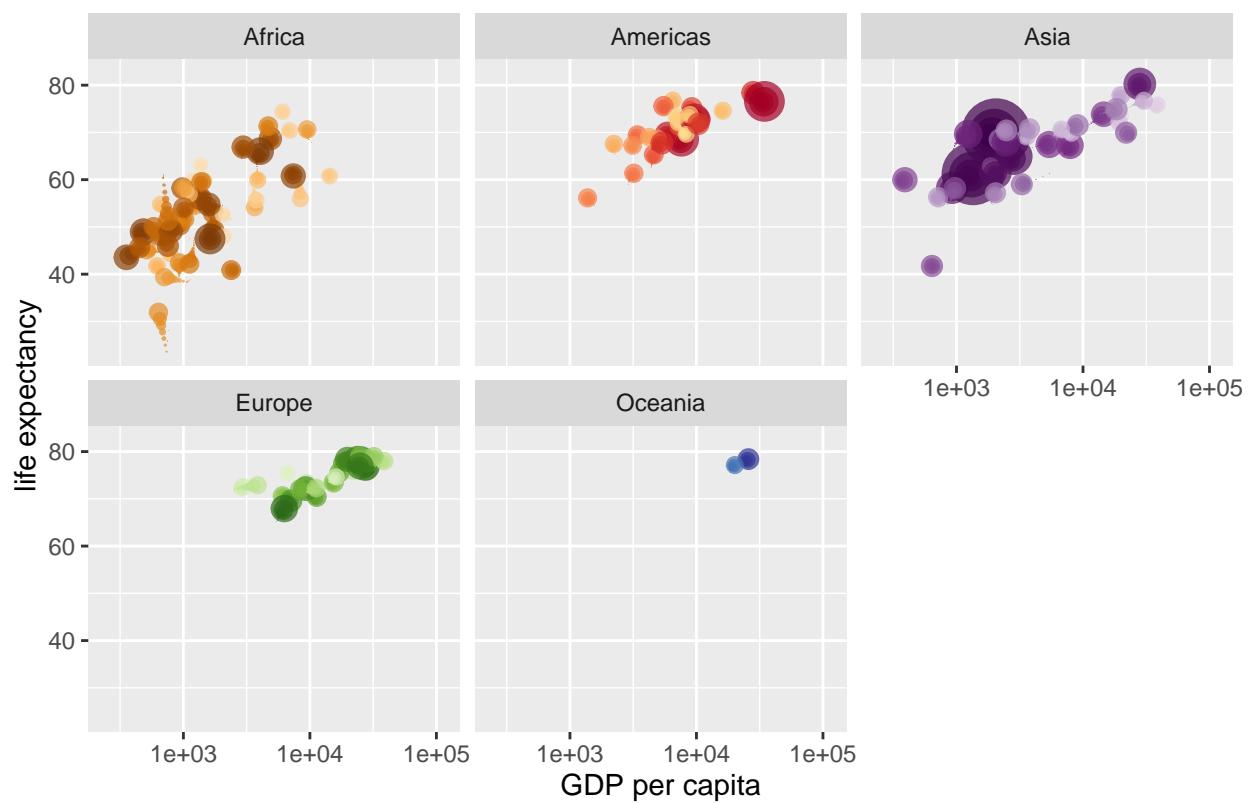
Year: 1994



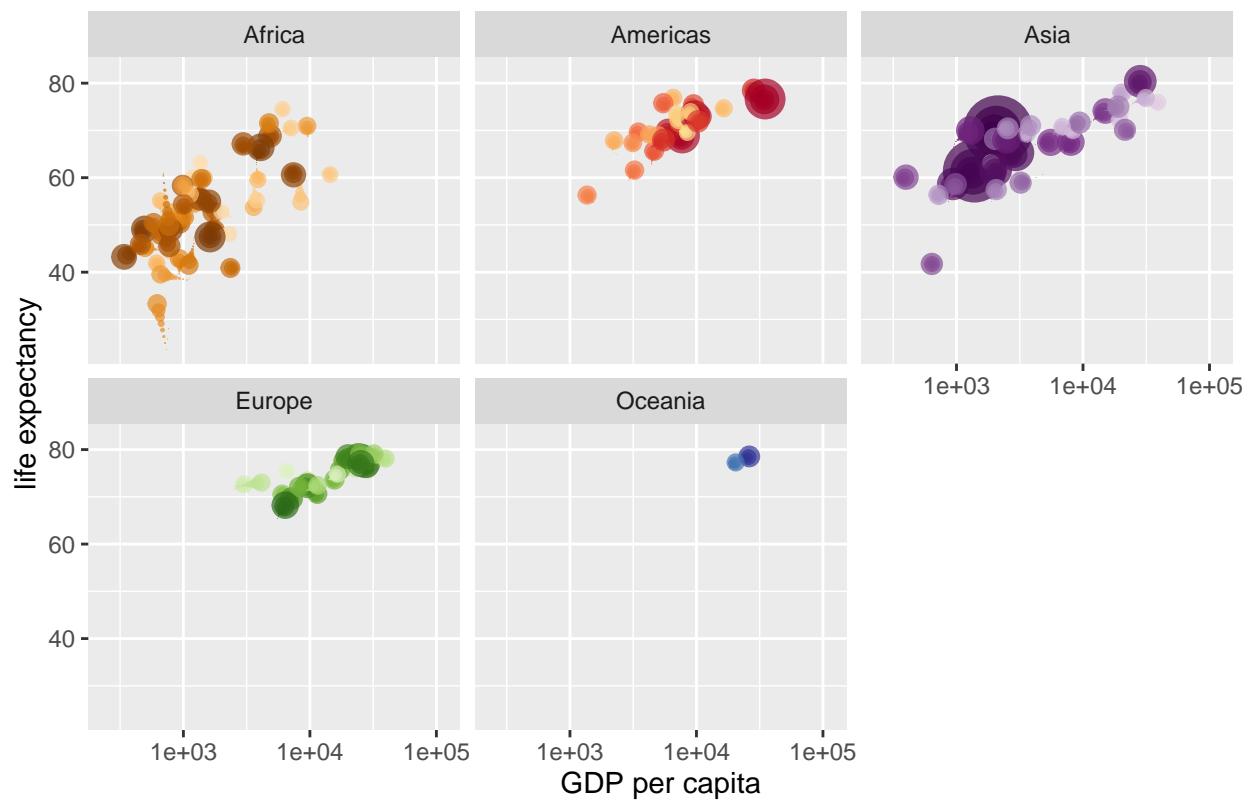
Year: 1995



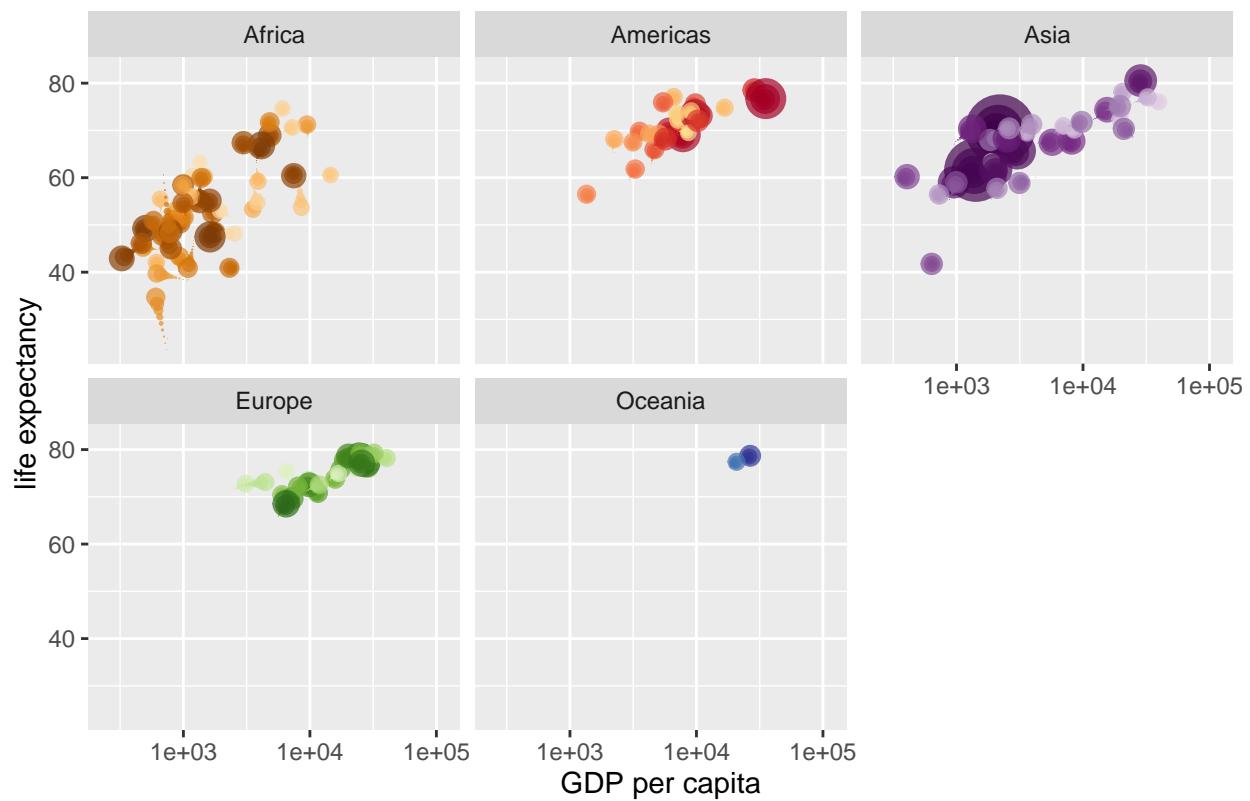
Year: 1995



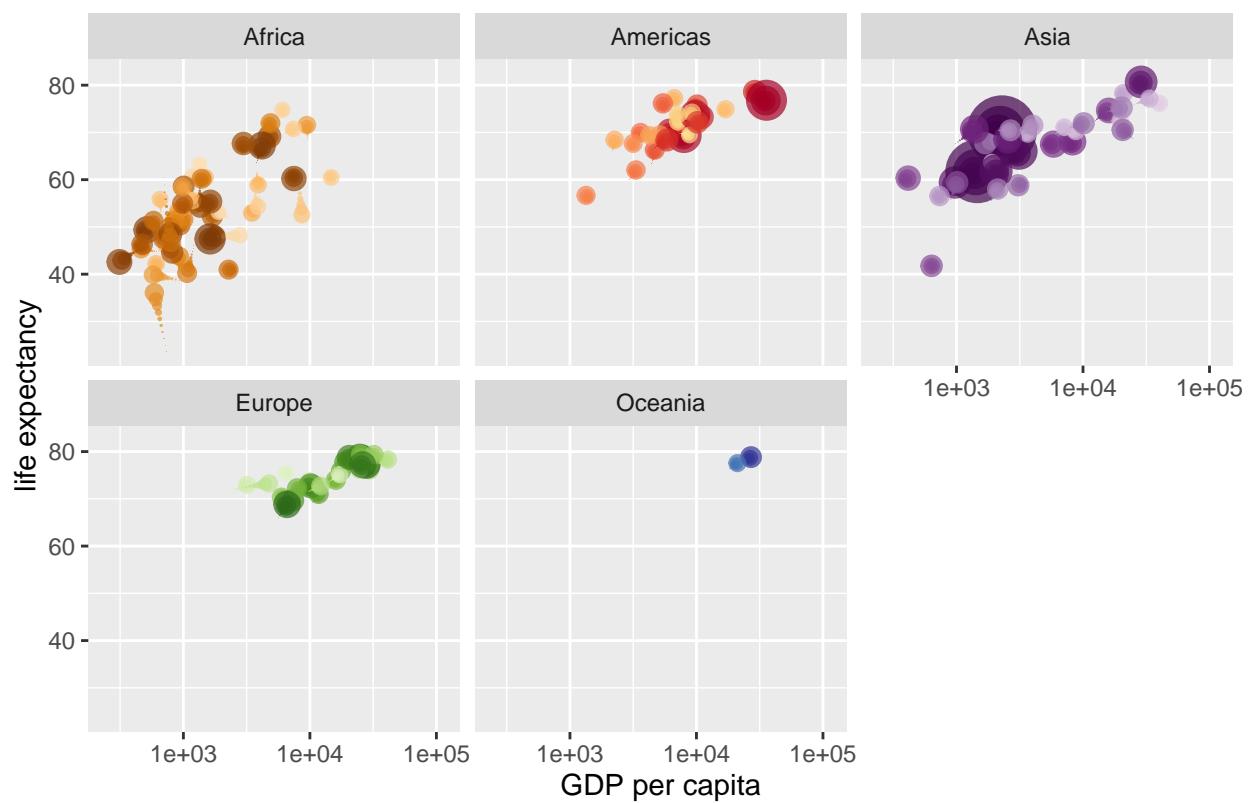
Year: 1996



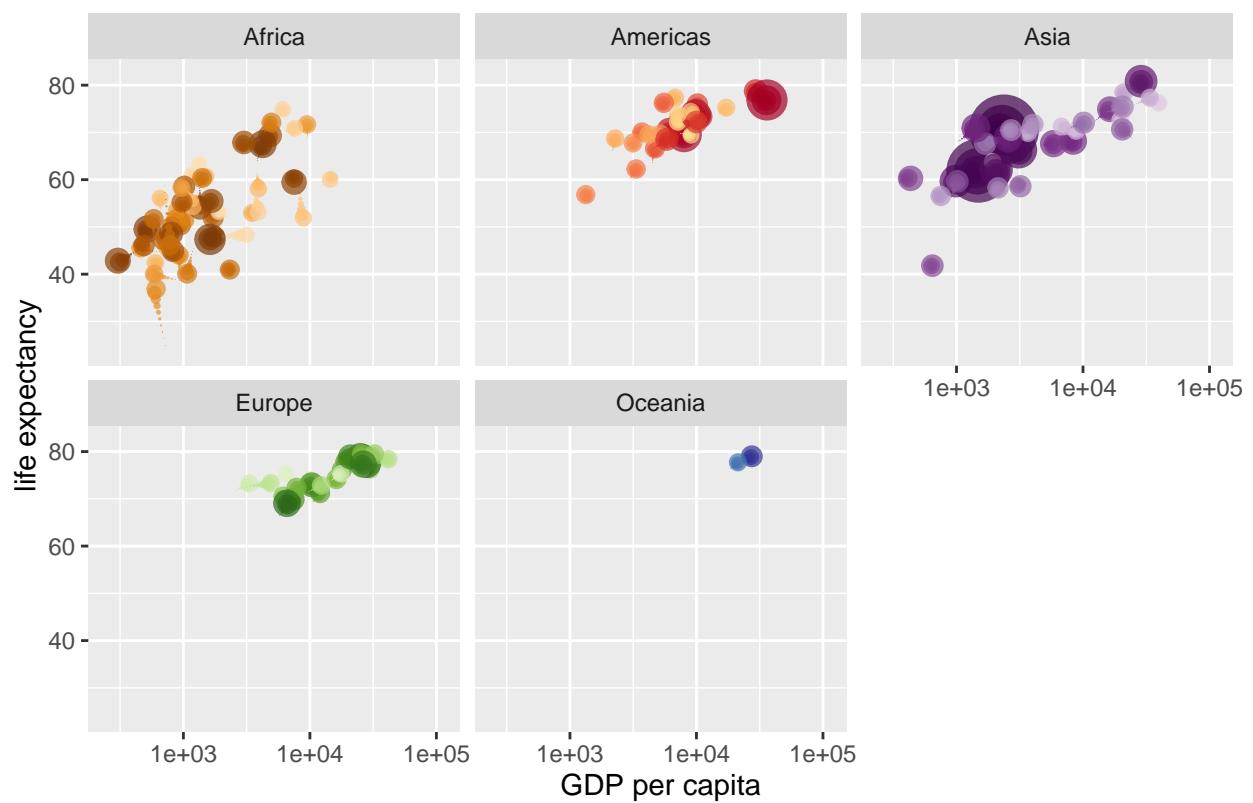
Year: 1996



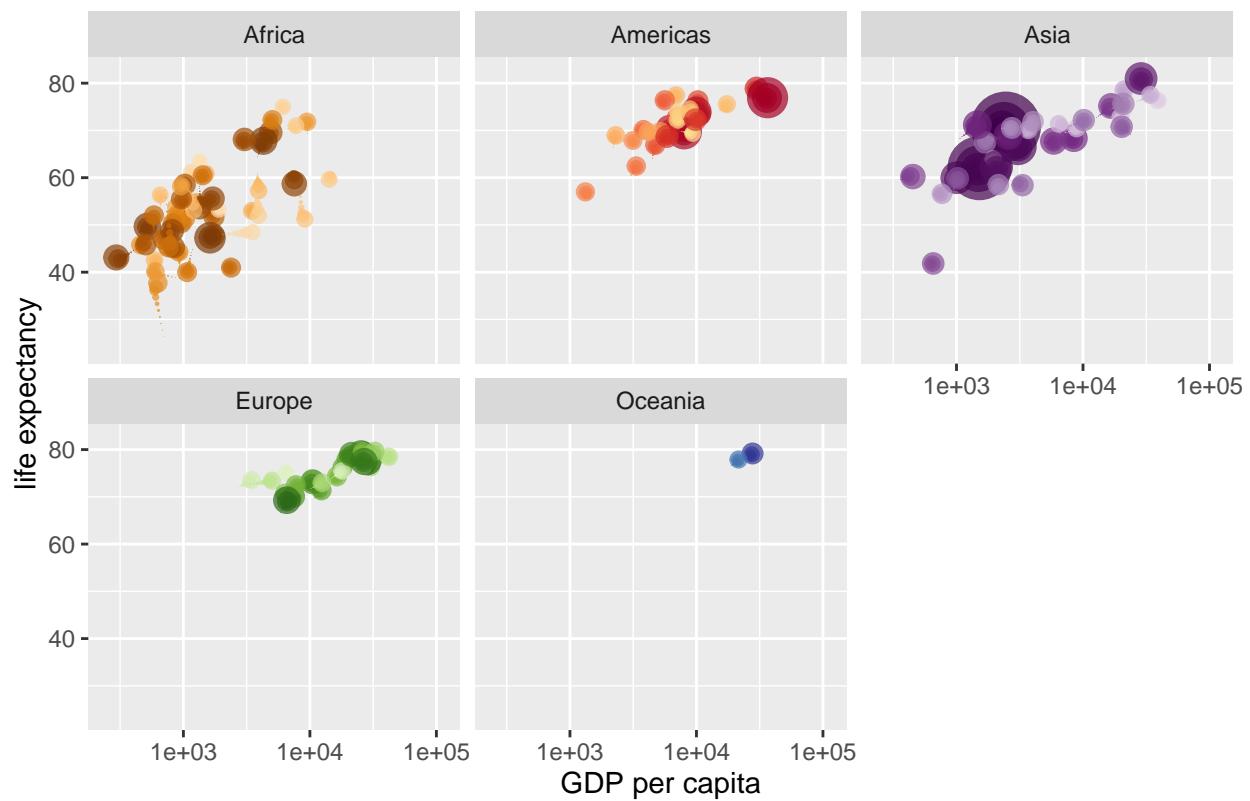
Year: 1997



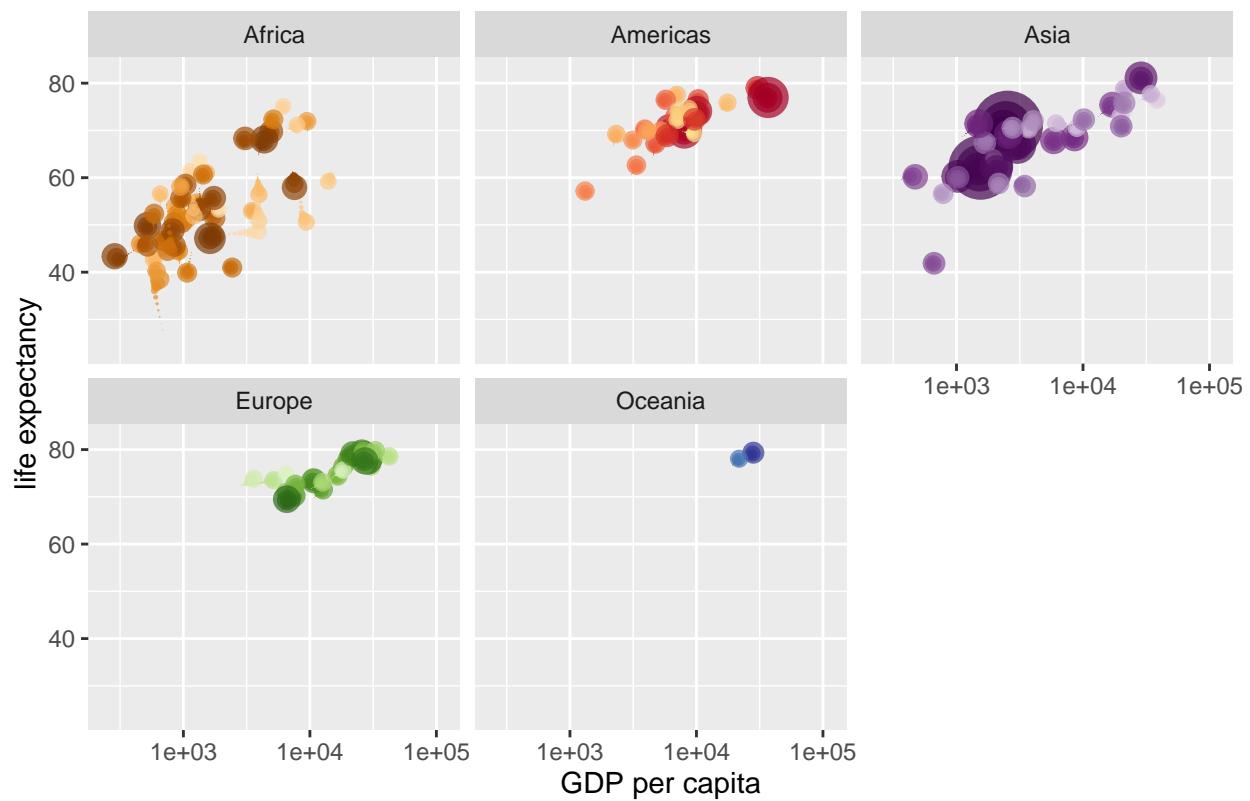
Year: 1998



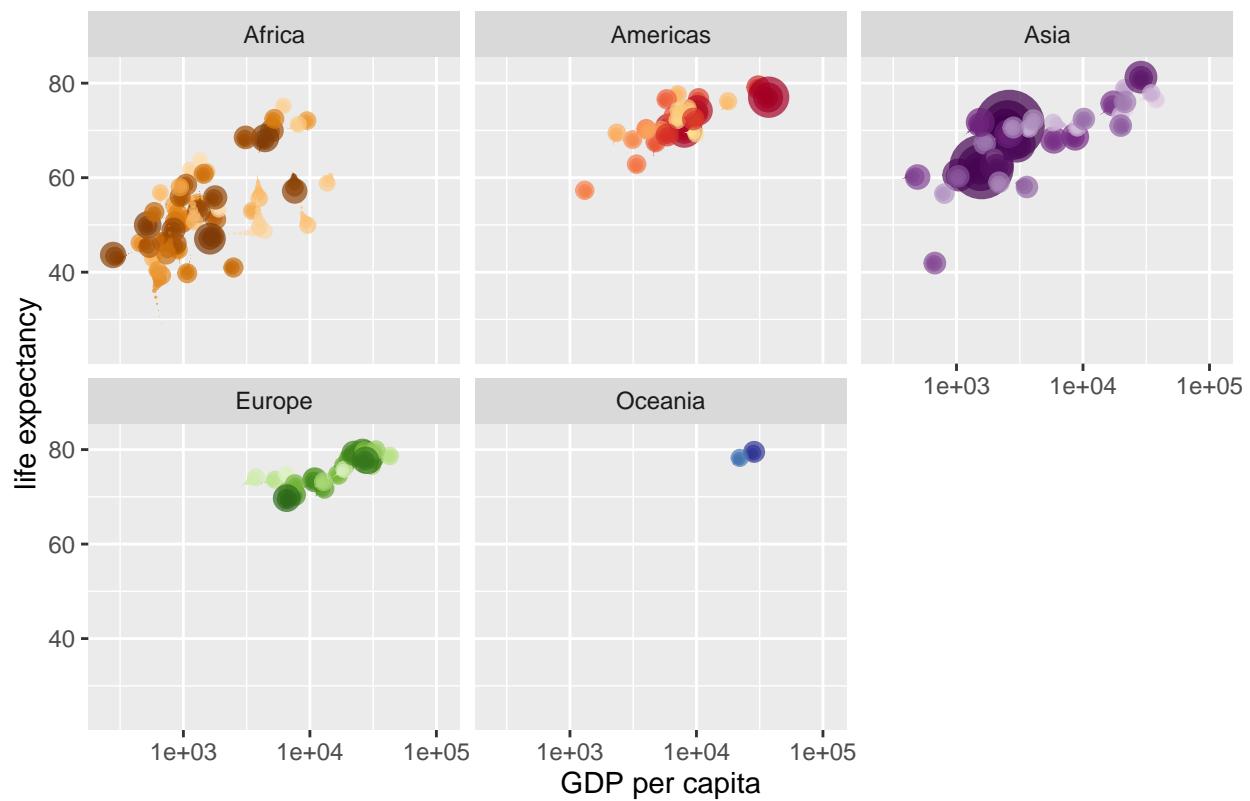
Year: 1998



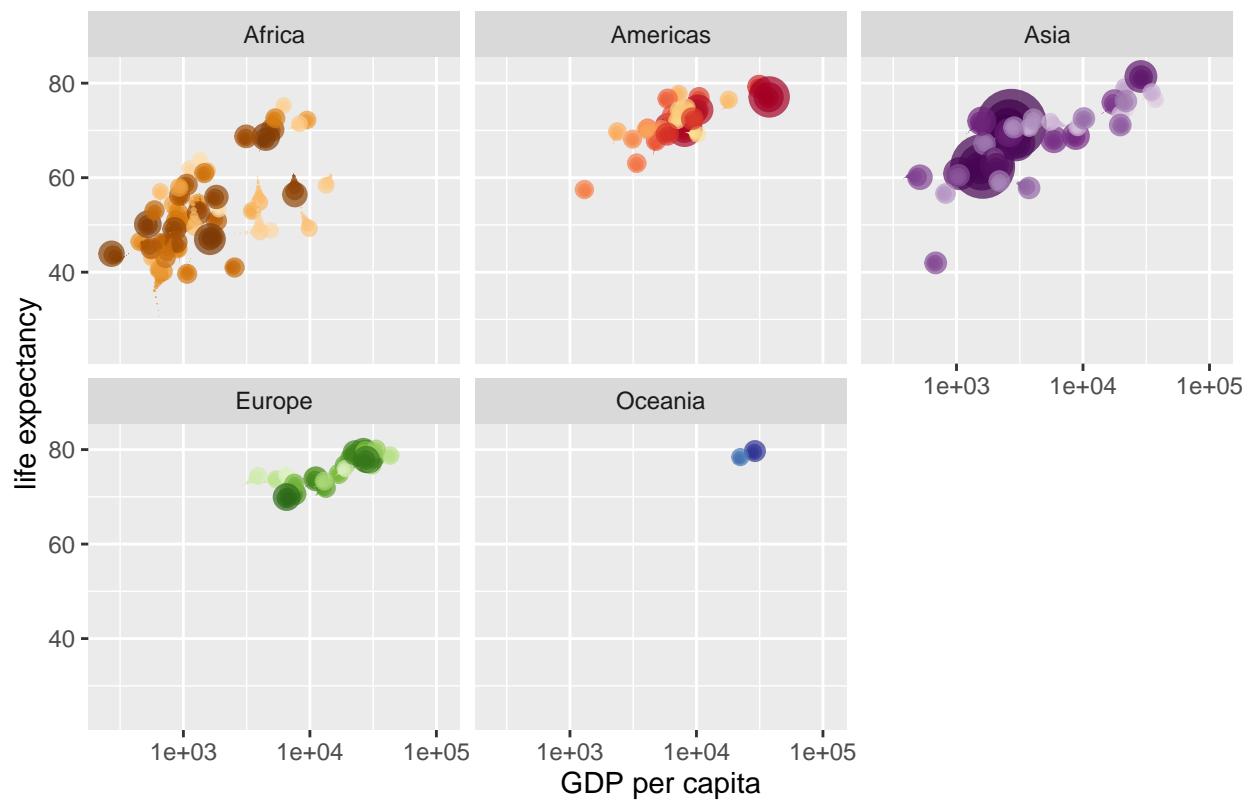
Year: 1999



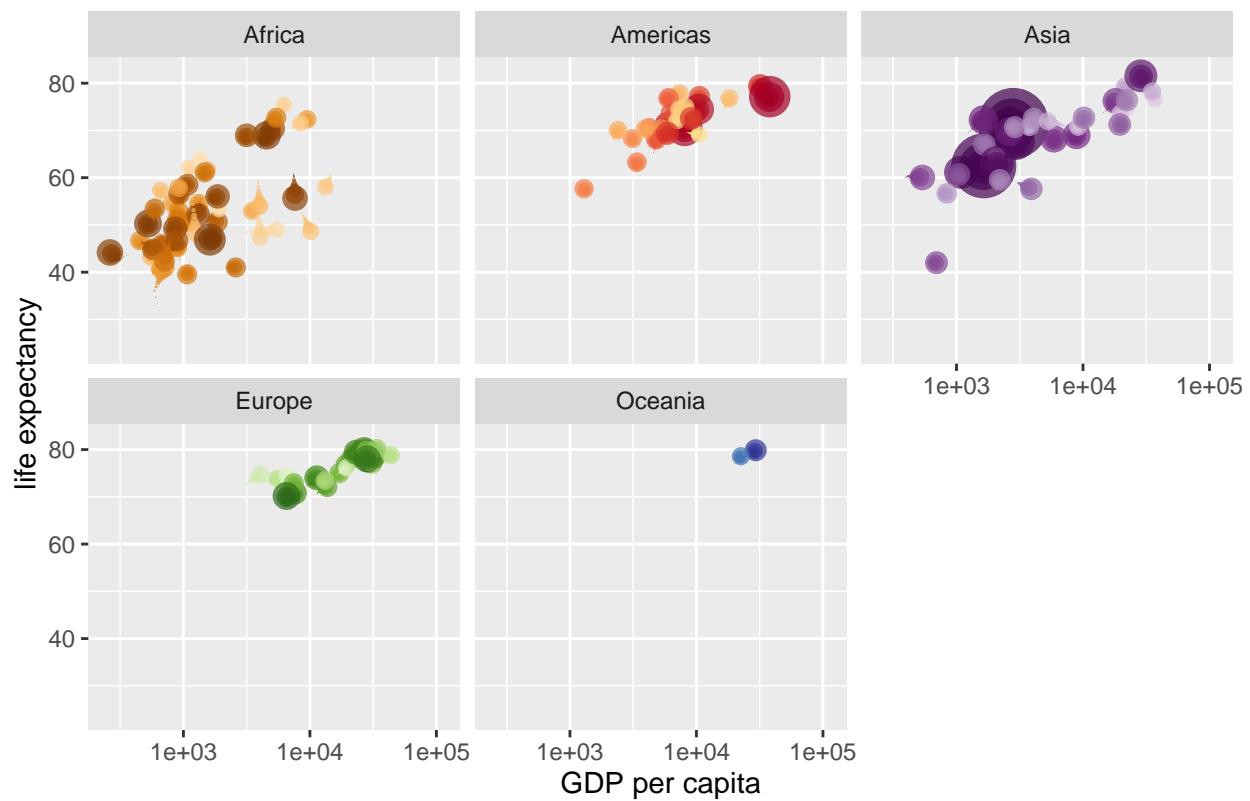
Year: 1999



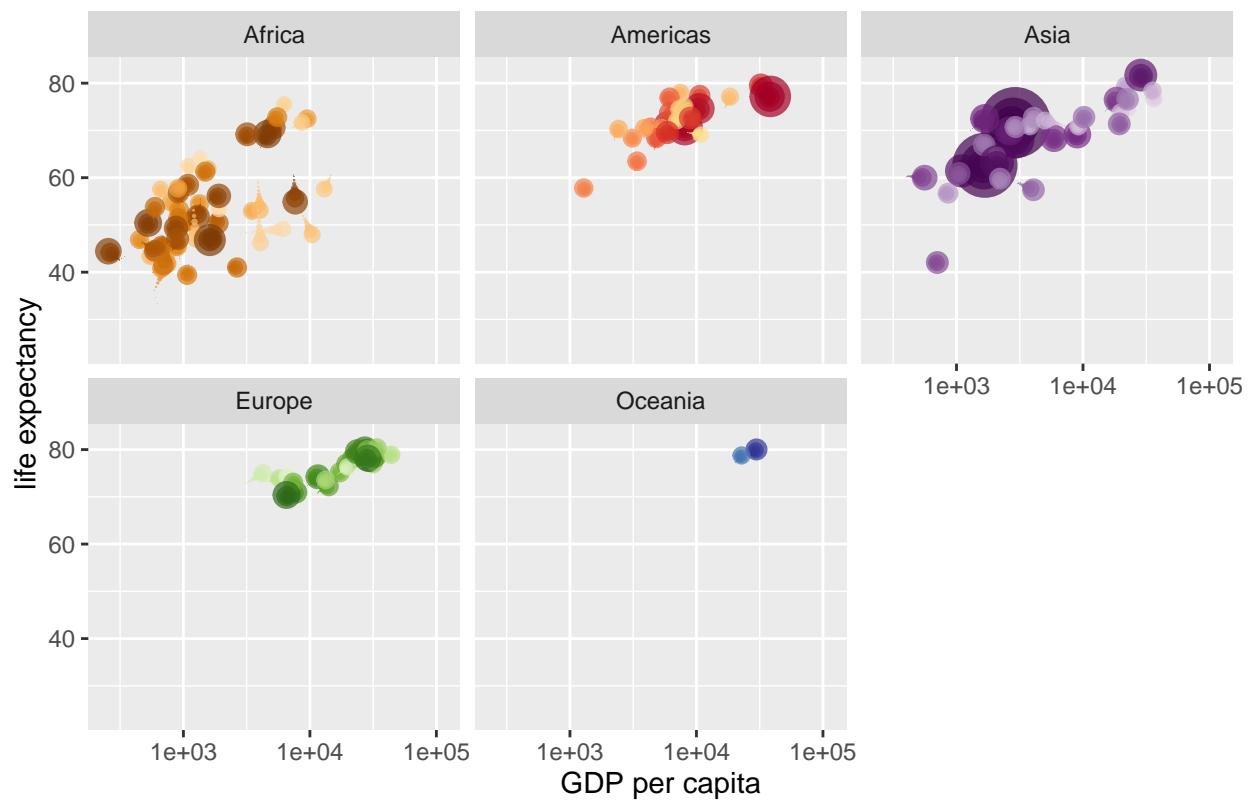
Year: 2000



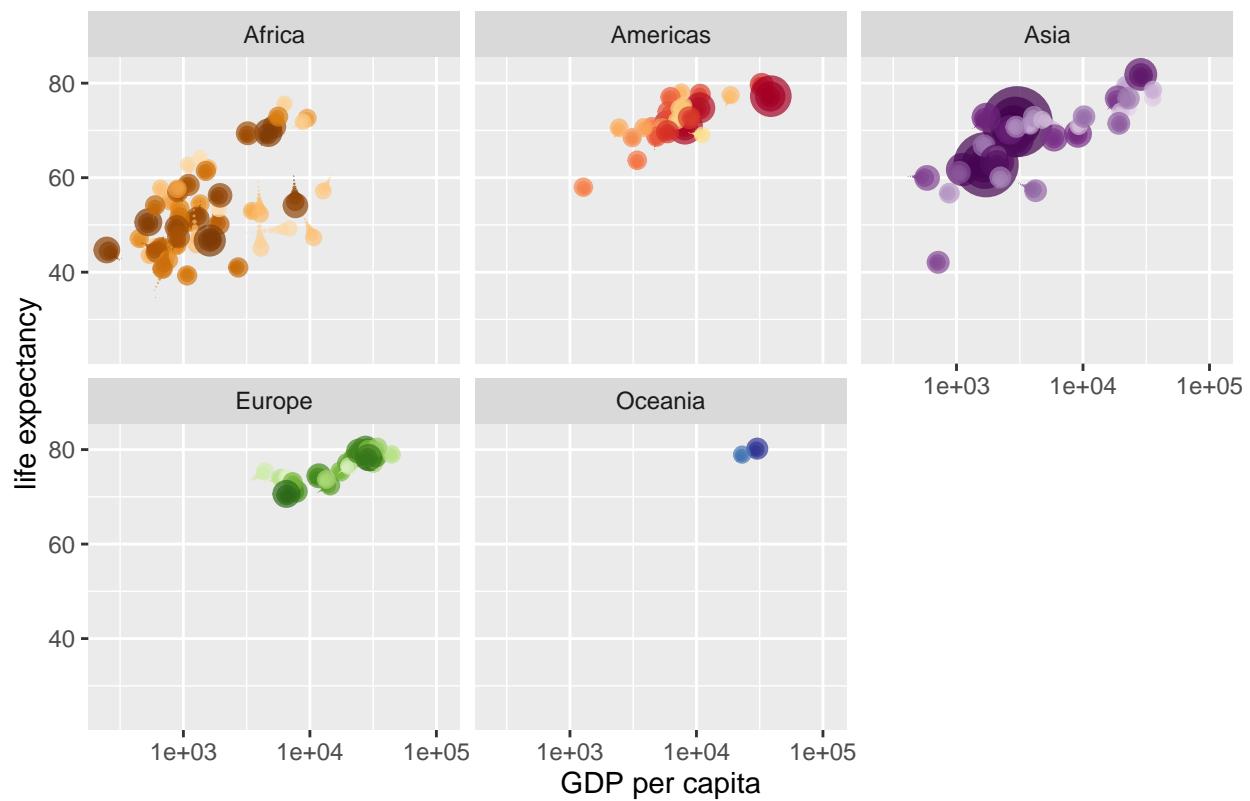
Year: 2000



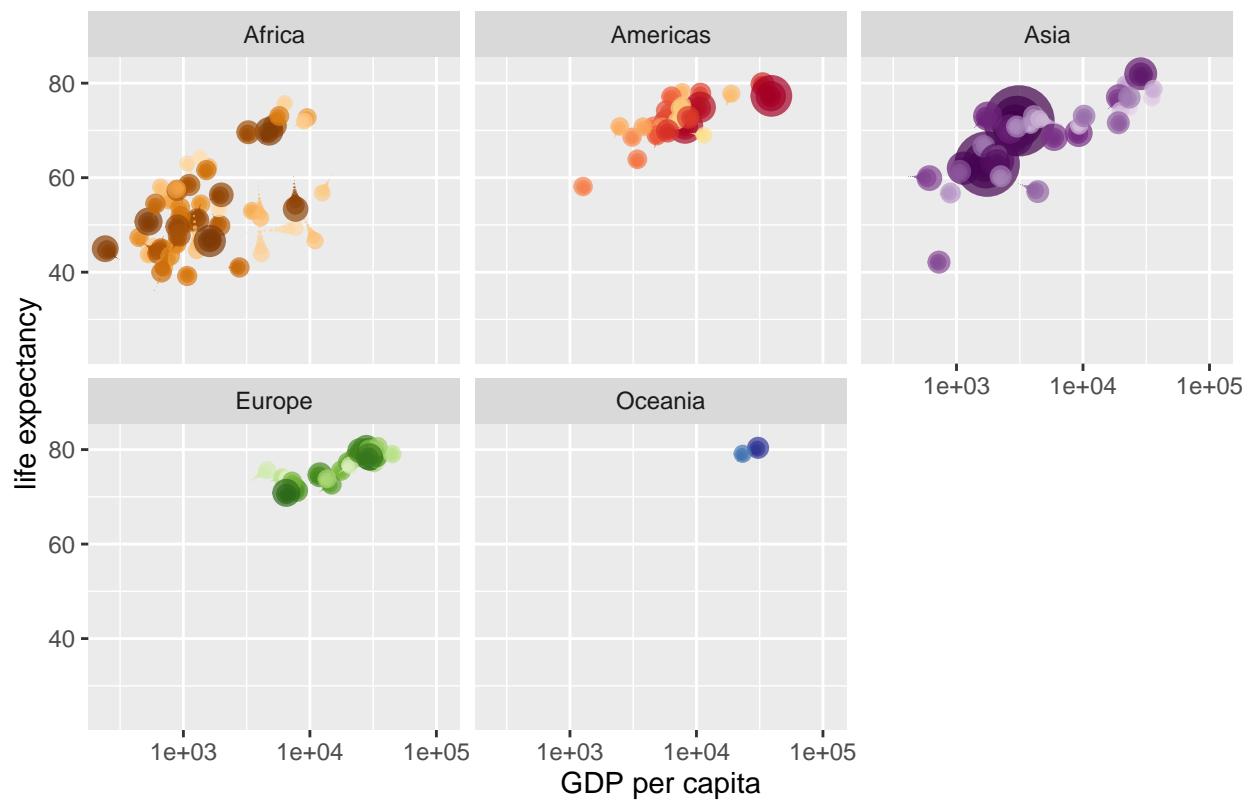
Year: 2001



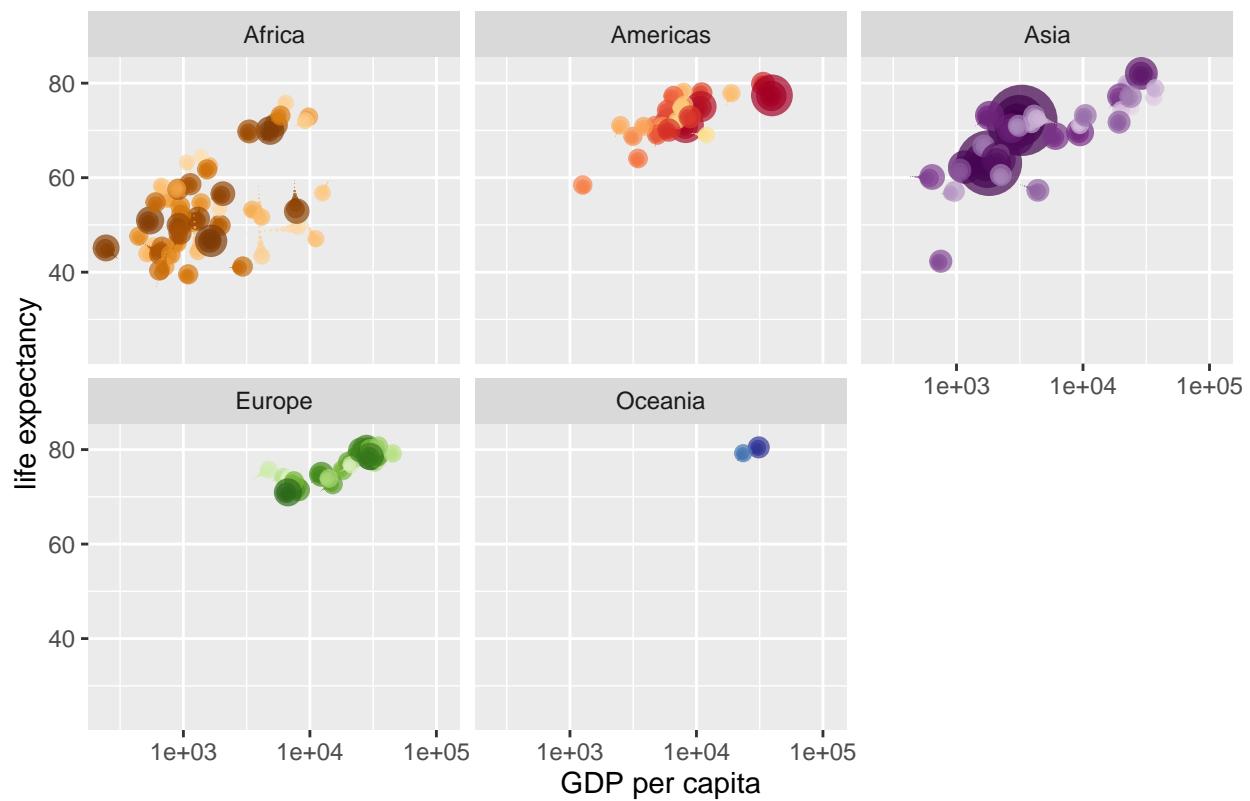
Year: 2001



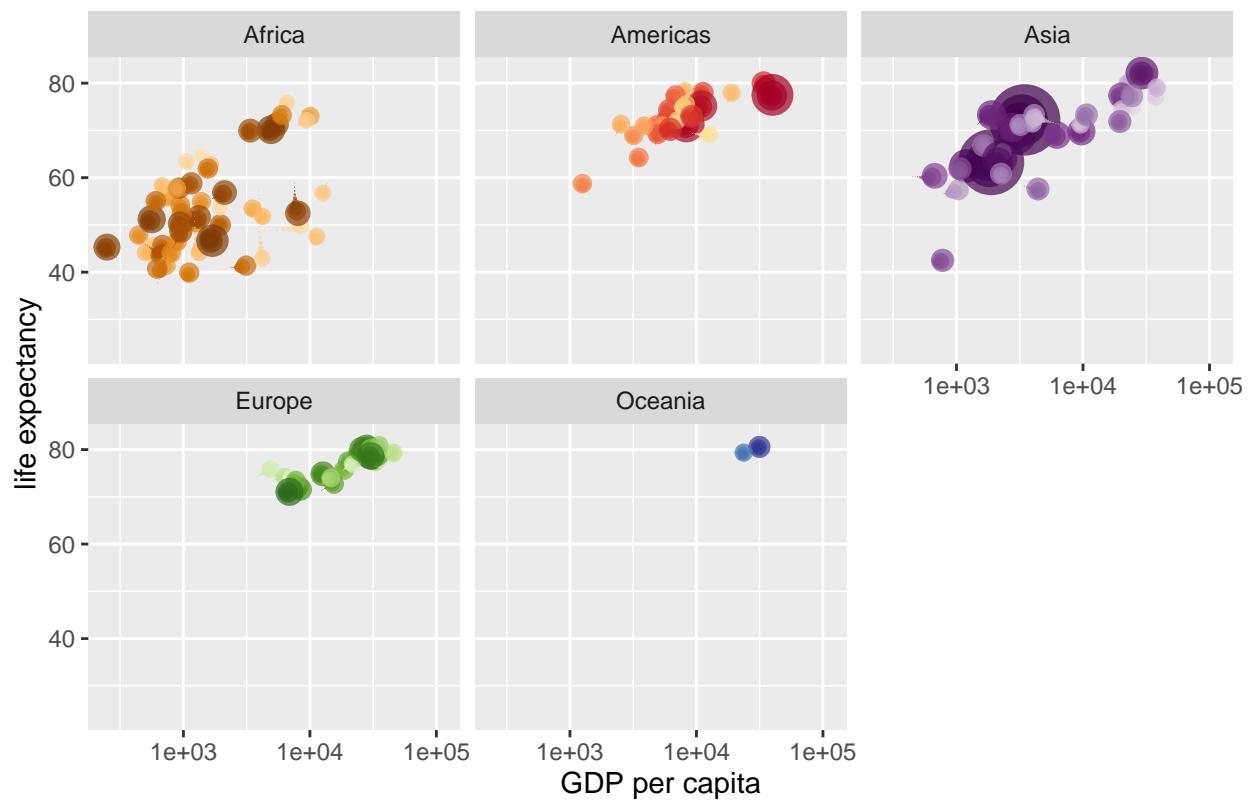
Year: 2002



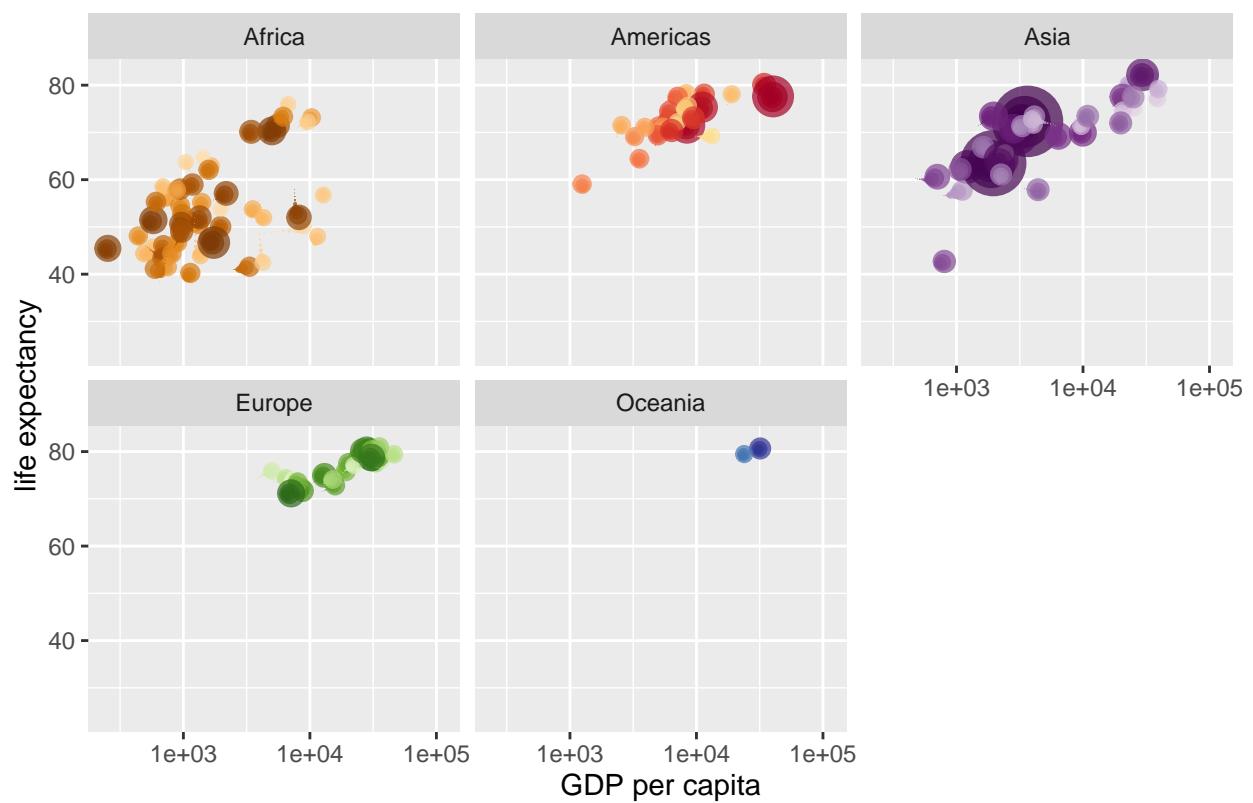
Year: 2003



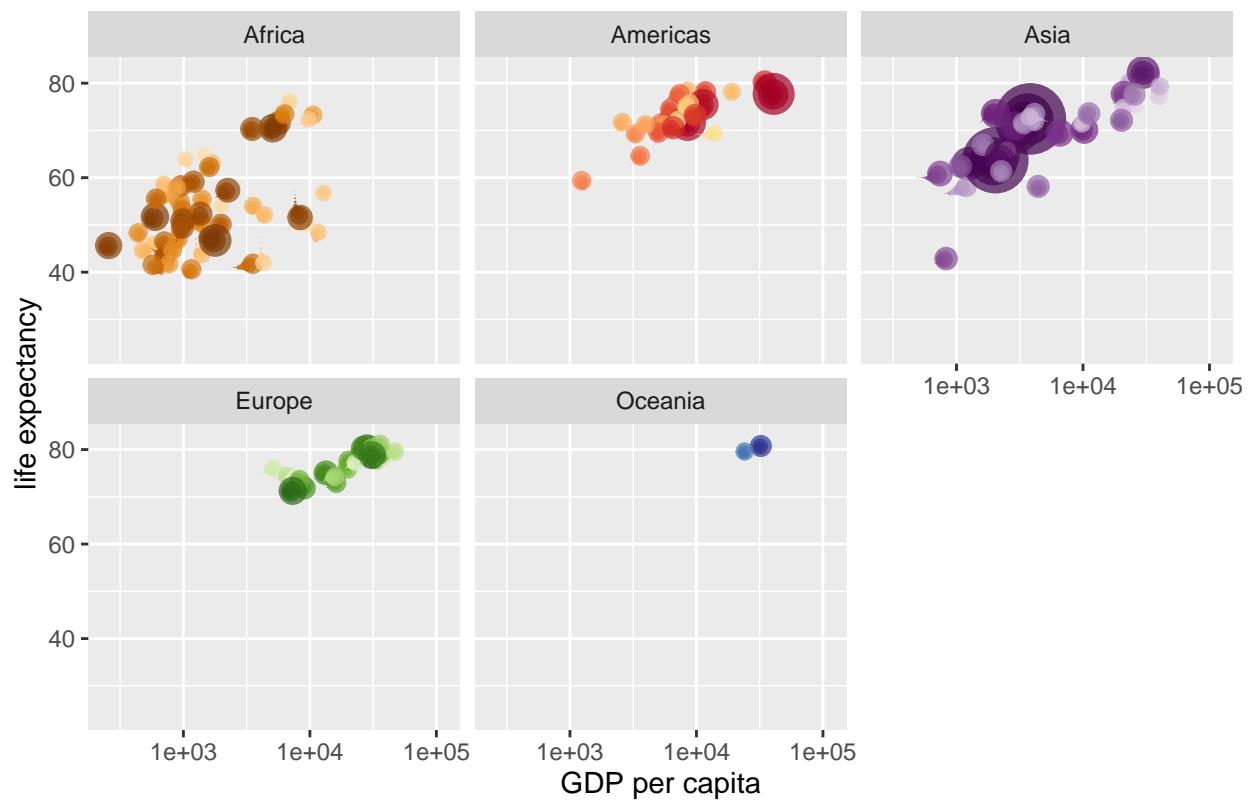
Year: 2003



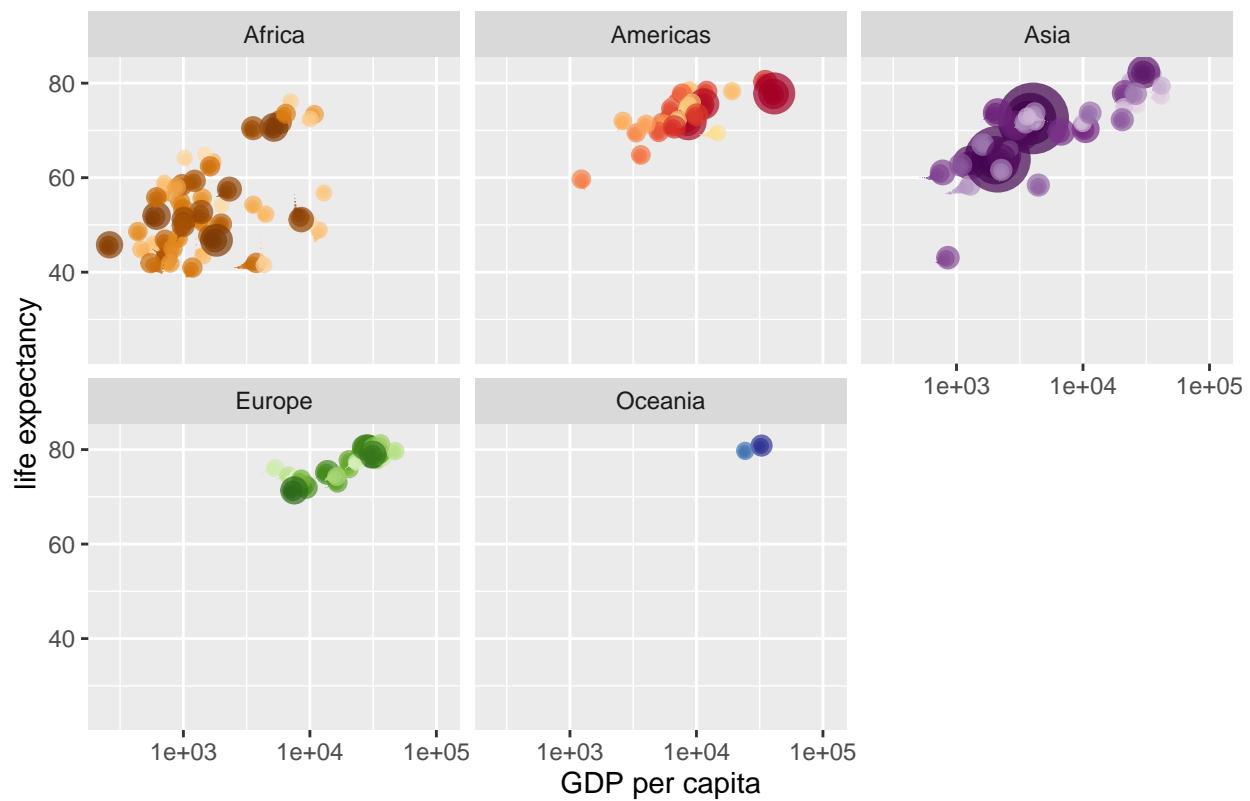
Year: 2004



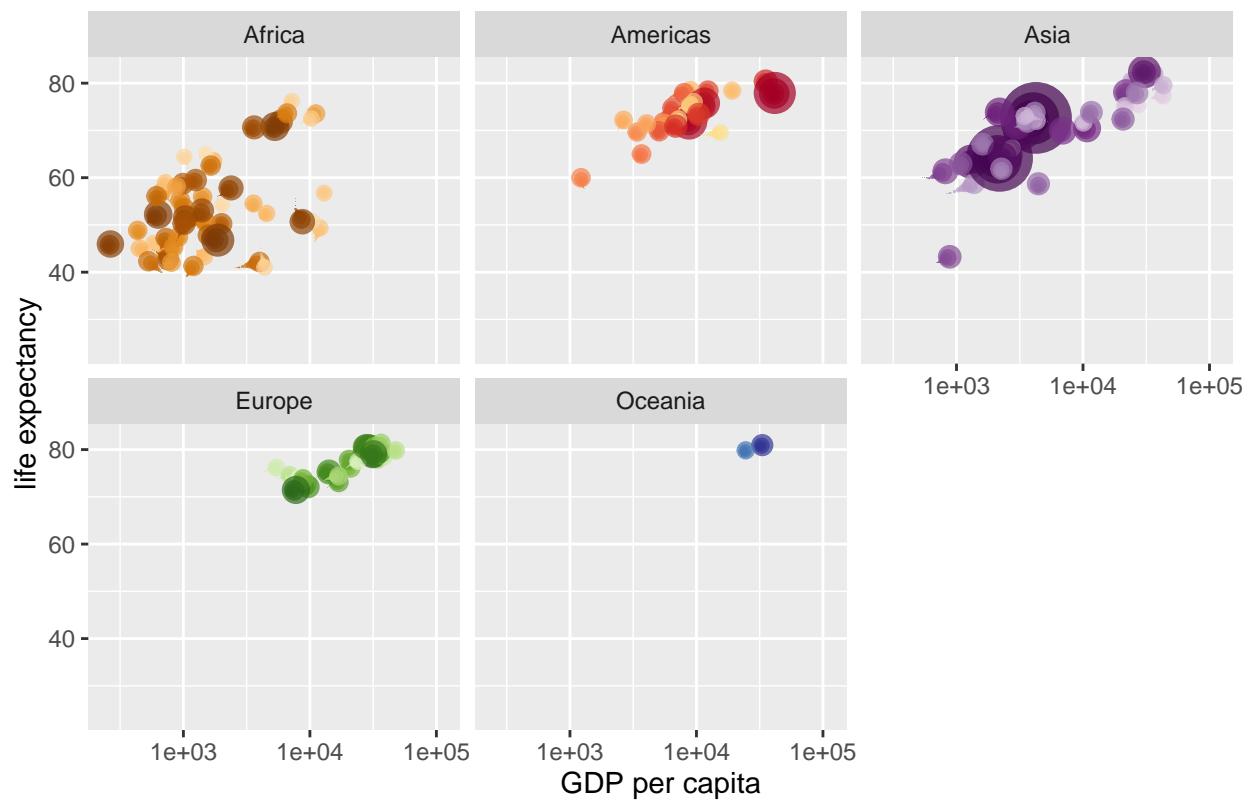
Year: 2004



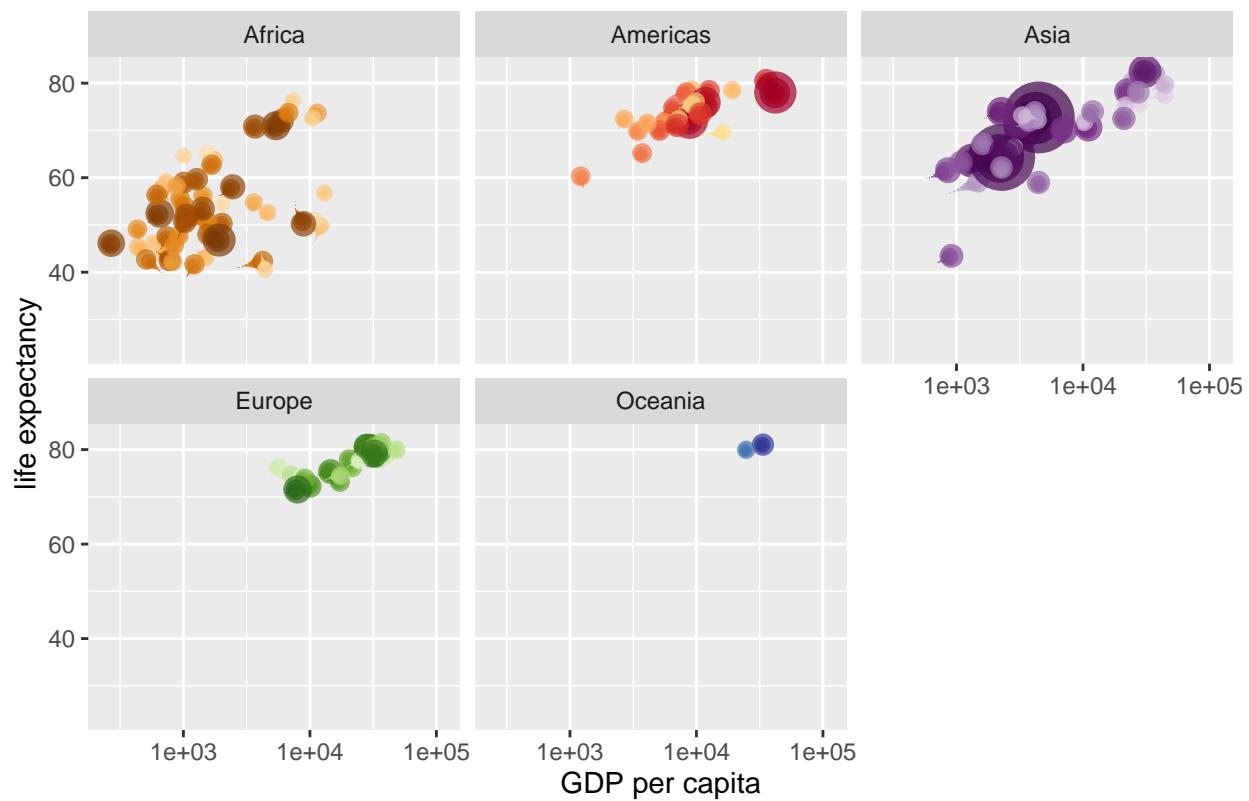
Year: 2005



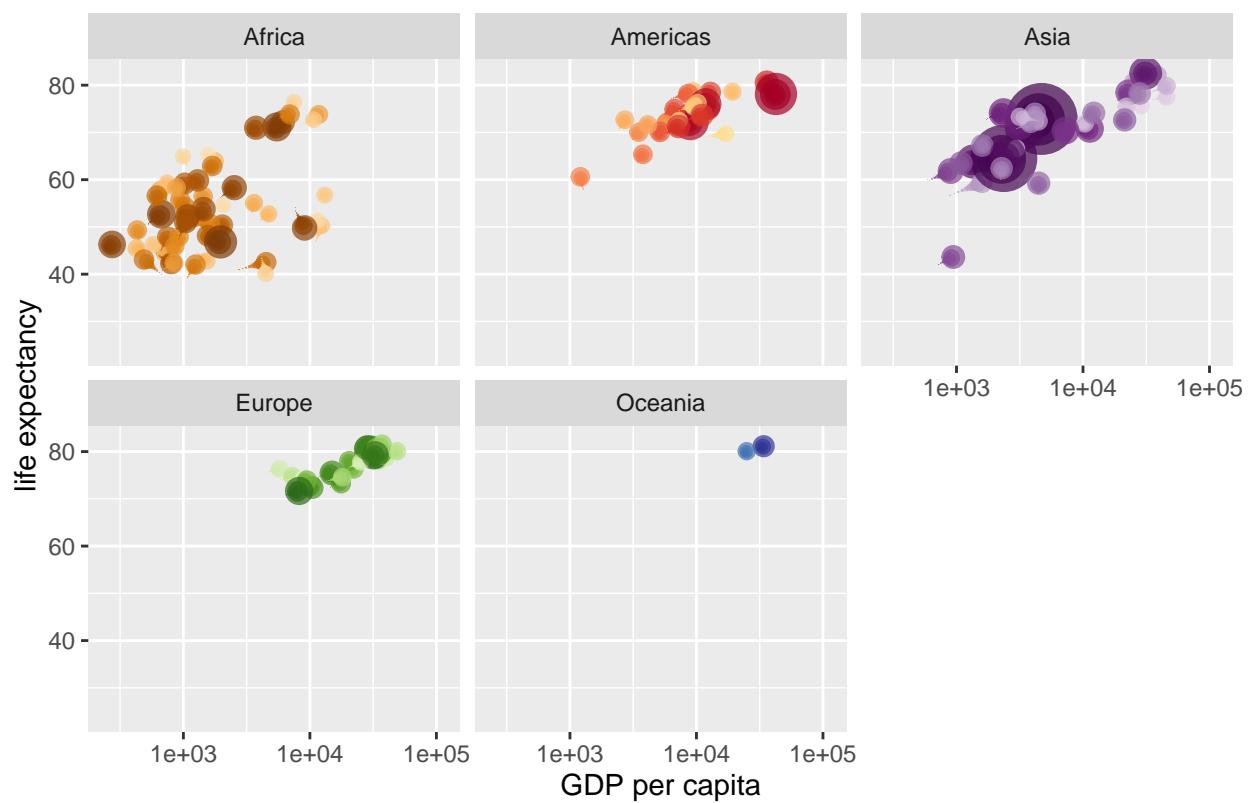
Year: 2005



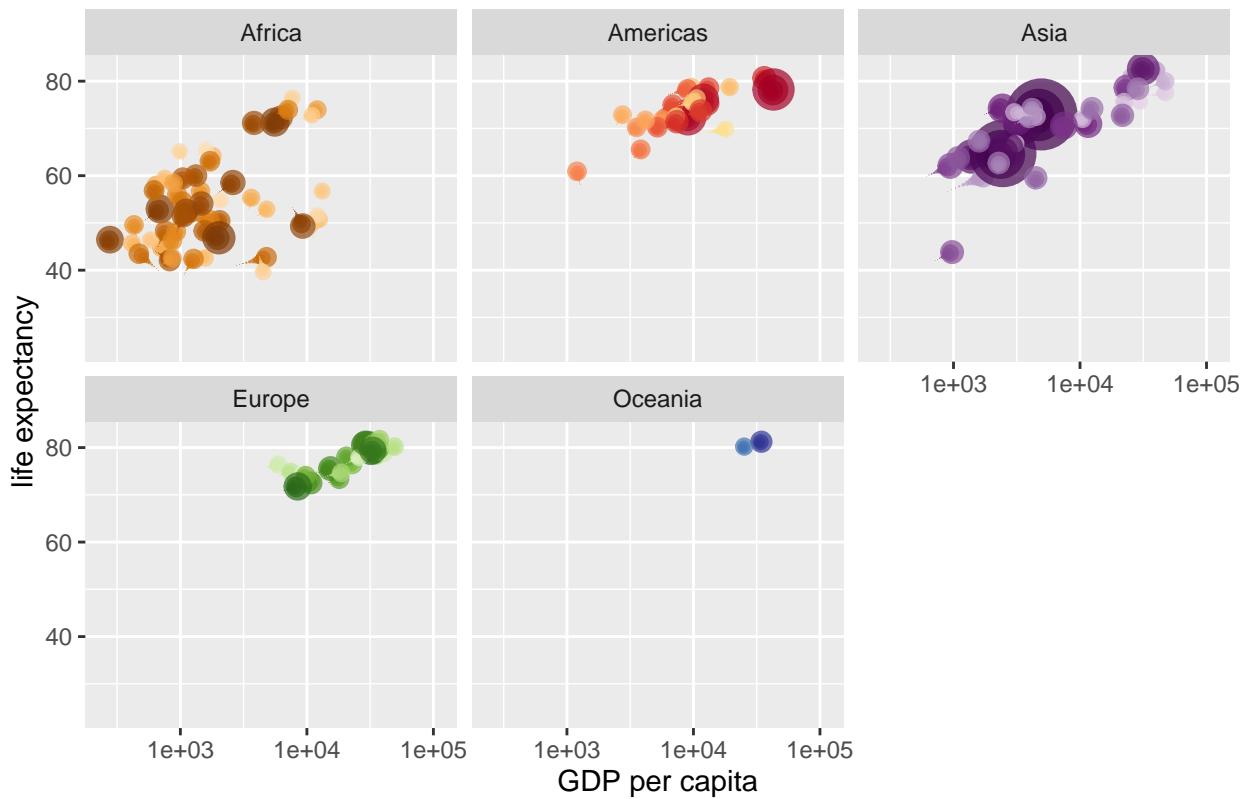
Year: 2006



Year: 2006



Year: 2007

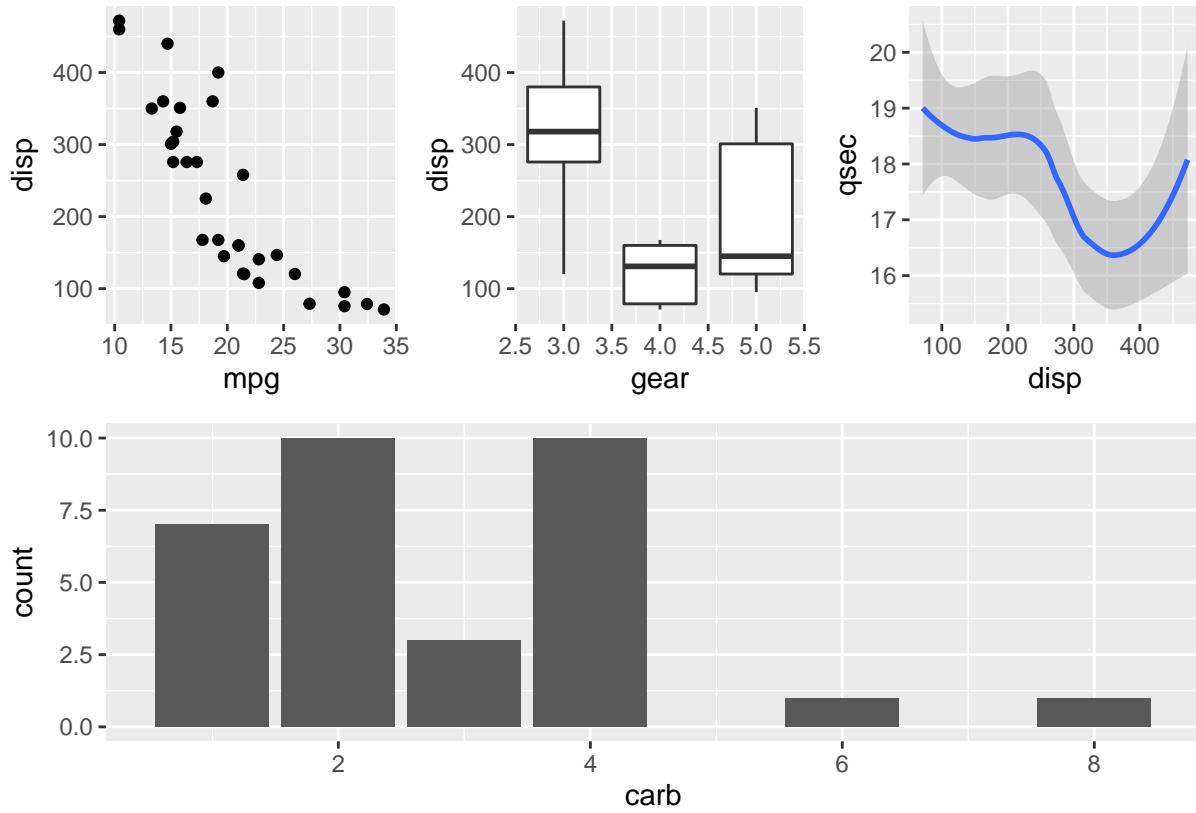


```
#Combining plots
library(ggplot2)
library(patchwork)

# Setup some example plots
p1 <- ggplot(mtcars) + geom_point(aes(mpg, disp))
p2 <- ggplot(mtcars) + geom_boxplot(aes(gear, disp, group = gear))
p3 <- ggplot(mtcars) + geom_smooth(aes(disp, qsec))
p4 <- ggplot(mtcars) + geom_bar(aes(carb))

# Use patchwork to combine them here:
(p1 | p2 | p3) /
  p4

## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



```
sessionInfo()
```

```
## R version 4.1.2 (2021-11-01)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Mojave 10.14.6
##
## Matrix products: default
## BLAS:    /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRblas.0.dylib
## LAPACK:  /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats      graphics   grDevices  utils       datasets   methods    base
##
## other attached packages:
## [1] patchwork_1.1.1 gganimate_1.0.7 dplyr_1.0.7      gapminder_0.3.0
## [5] ggplot2_3.3.5
##
## loaded via a namespace (and not attached):
## [1] Rcpp_1.0.8        plyr_1.8.6        pillar_1.7.0      compiler_4.1.2
## [5] highr_0.9         prettyunits_1.1.1 progress_1.2.2     tools_4.1.2
## [9] digest_0.6.29     lattice_0.20-45   nlme_3.1-153     evaluate_0.14
## [13] lifecycle_1.0.1    tibble_3.1.6       gtable_0.3.0      mgcv_1.8-38
## [17] pkgconfig_2.0.3    rlang_1.0.0       Matrix_1.3-4      cli_3.1.1
```

```
## [21] yaml_2.2.2          xfun_0.29           fastmap_1.1.0      withr_2.4.3
## [25] stringr_1.4.0        knitr_1.37          hms_1.1.1          generics_0.1.2
## [29] vctrs_0.3.8          grid_4.1.2          tidyselect_1.1.1   glue_1.6.1
## [33] R6_2.5.1              gifski_1.4.3-1    fansi_1.0.2        rmarkdown_2.11
## [37] tweenr_1.0.2          purrrr_0.3.4       farver_2.1.0       magrittr_2.0.2
## [41] splines_4.1.2         scales_1.1.1       ellipsis_0.3.2    htmltools_0.5.2
## [45] colorspace_2.0-2     labeling_0.4.2     utf8_1.2.2         stringi_1.7.6
## [49] munsell_0.5.0         crayon_1.4.2
```