

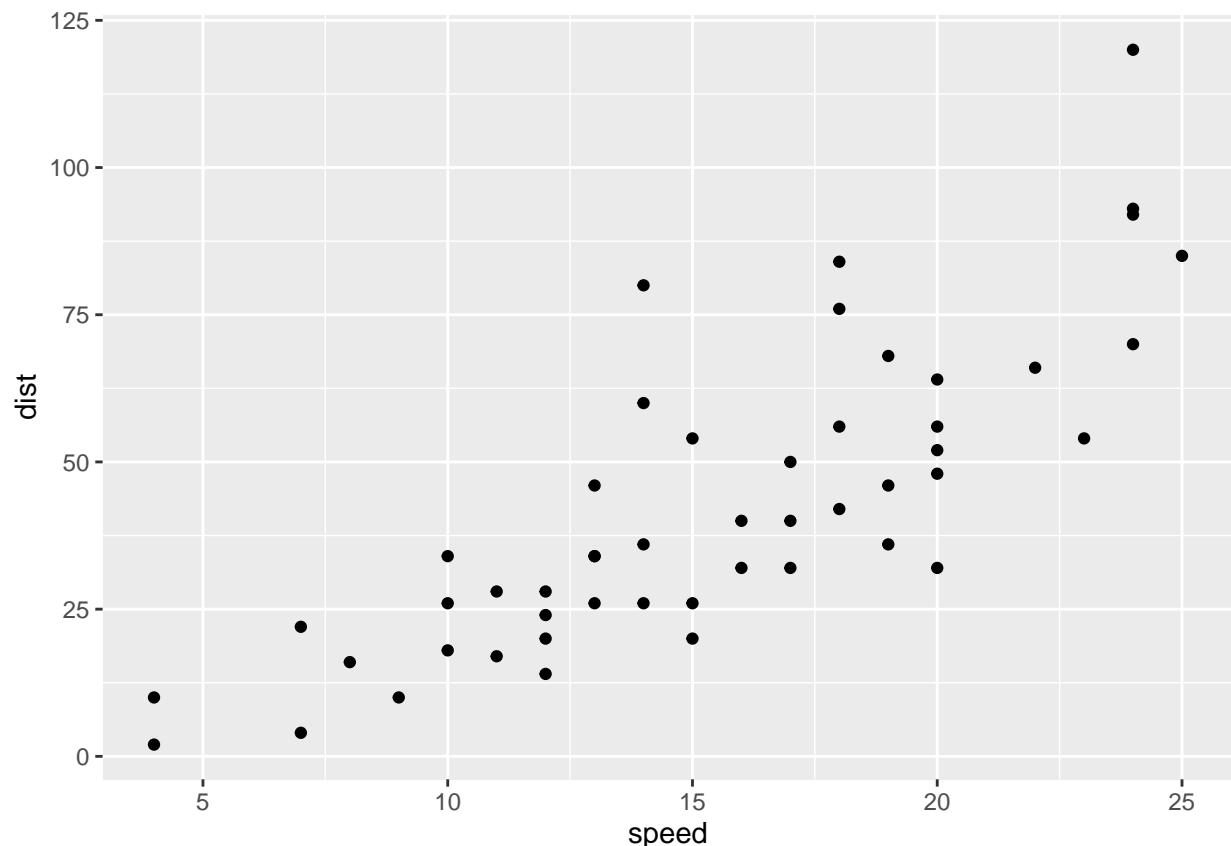
Class 05 Data Visualization

Caroline Mackey (PID: A15522472)

2021-10-16

SCATTERPLOTS

```
# Install, then load ggplot.  
library (ggplot2)  
  
# Every ggplot has data + aes + geom.  
# **First geom_point() plot of cars data:  
ggplot(data=cars) +  
  aes(x=speed, y=dist) +  
  geom_point()
```



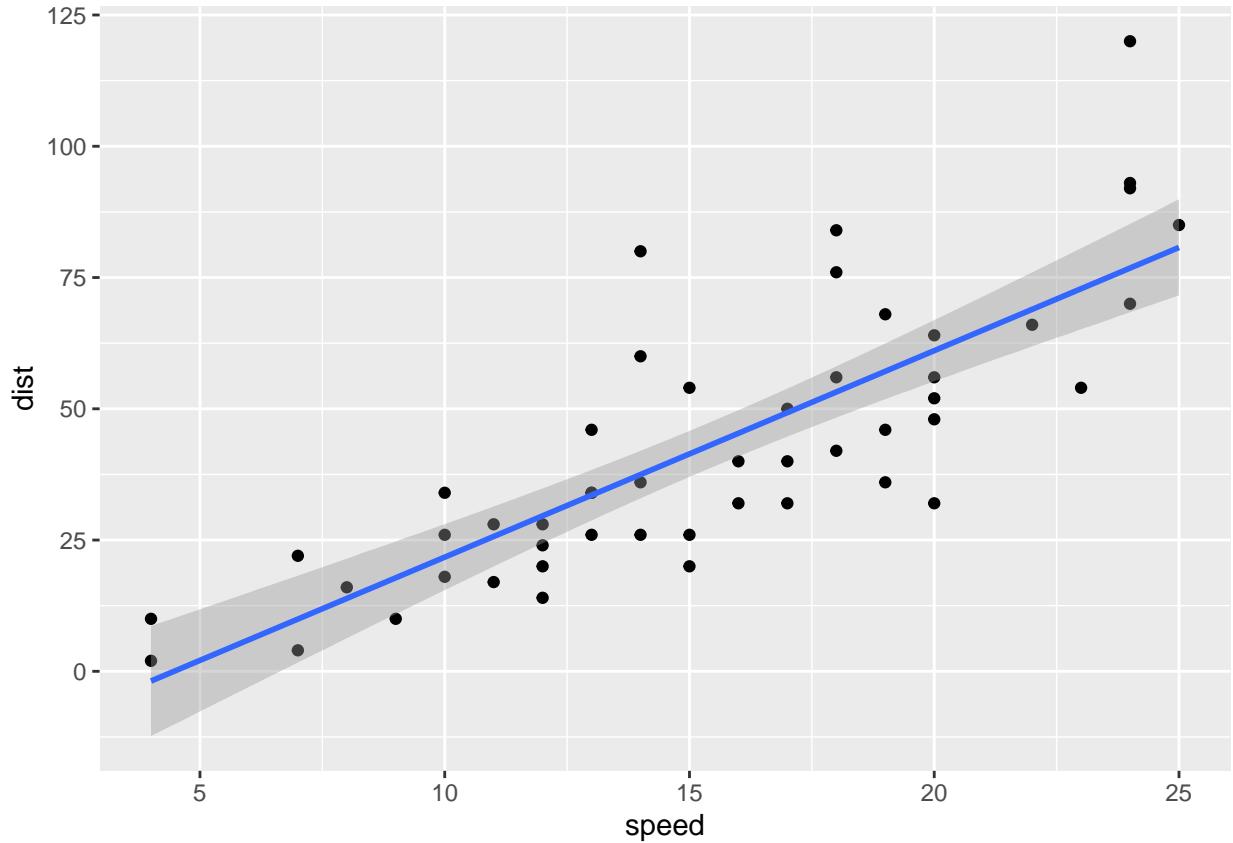
```
# Change to a linear model.  
# **Plot with 2 or more geoms:
```

```

p <- ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth(method="lm")
p

```

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



```

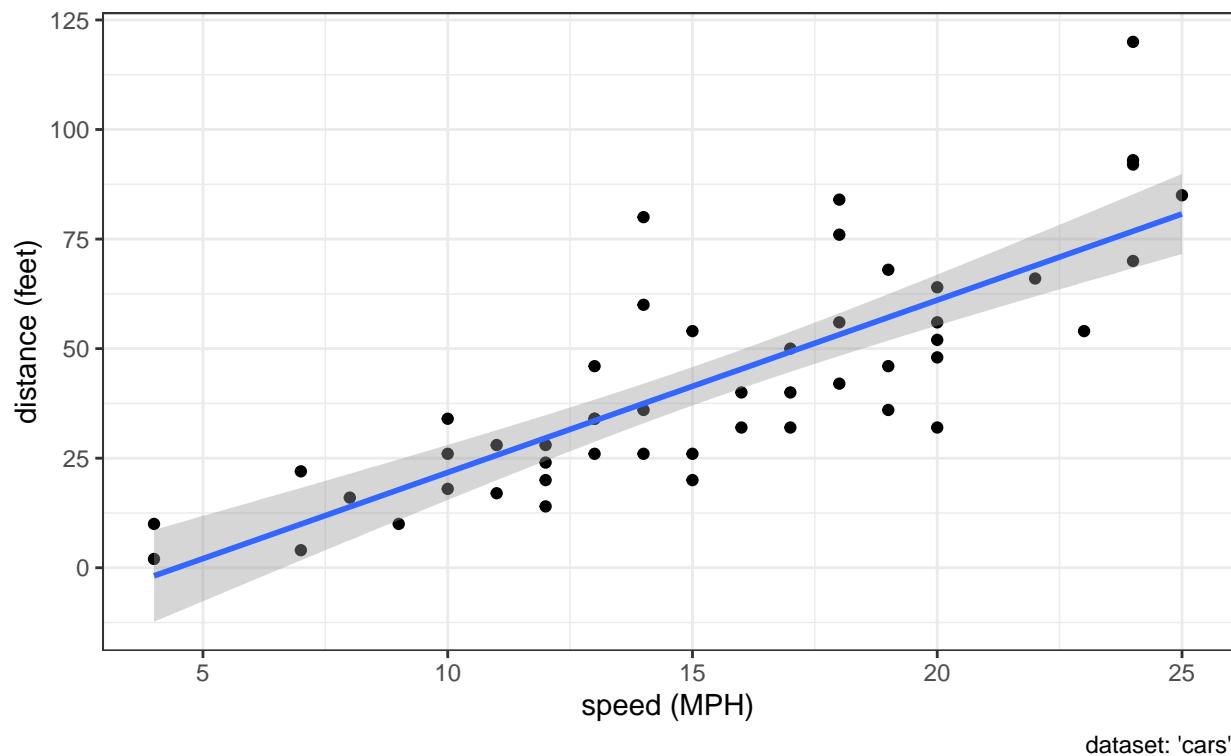
# Adjusting labels & theme.
p + labs(title="Stopping Distance (feet) vs. Speed (MPH) of Cars",
         x=("speed (MPH)"),
         y=("distance (feet)"),
         subtitle=("R Built-in Data Set"),
         caption=("dataset: 'cars'")) +
  theme_bw()

```

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

Stopping Distance (feet) vs. Speed (MPH) of Cars

R Built-in Data Set



```
# New data set...
# **Read an input file

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
```

```
# Q. How many genes are there in the data set?
nrow(genes)
```

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

```
# Q. What are the column names?
colnames(genes)
```

```
## [1] "Gene"        "Condition1" "Condition2" "State"
```

```

# Q. How many columns are there?
ncol(genes)

## [1] 4

# Q. How many 'up' regulated genes are there?
table(genes$State)

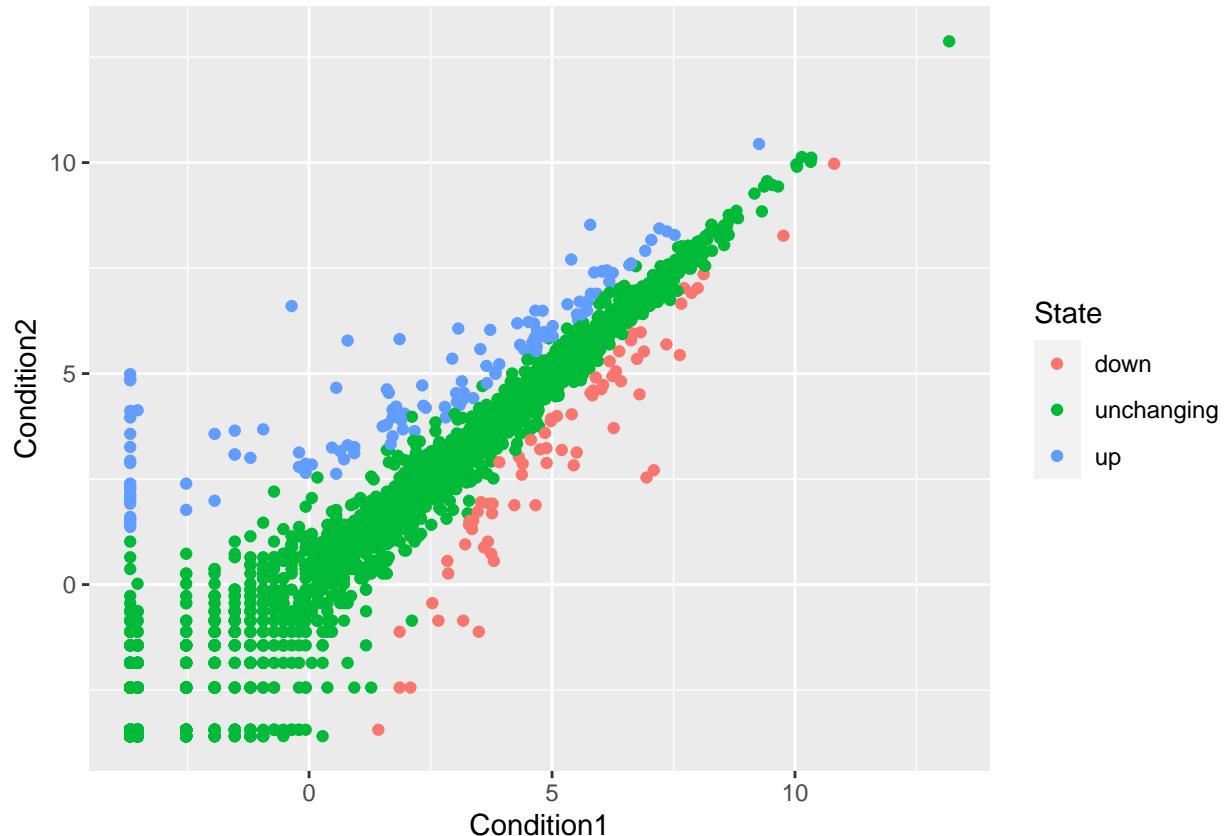
##
##      down unchanged      up
##      72        4997     127

#Q. What fraction of total genes is up-regulated in this dataset? (2 sig figs)
prec <- table(genes$State) / nrow(genes) * 100
round (prec, 2)

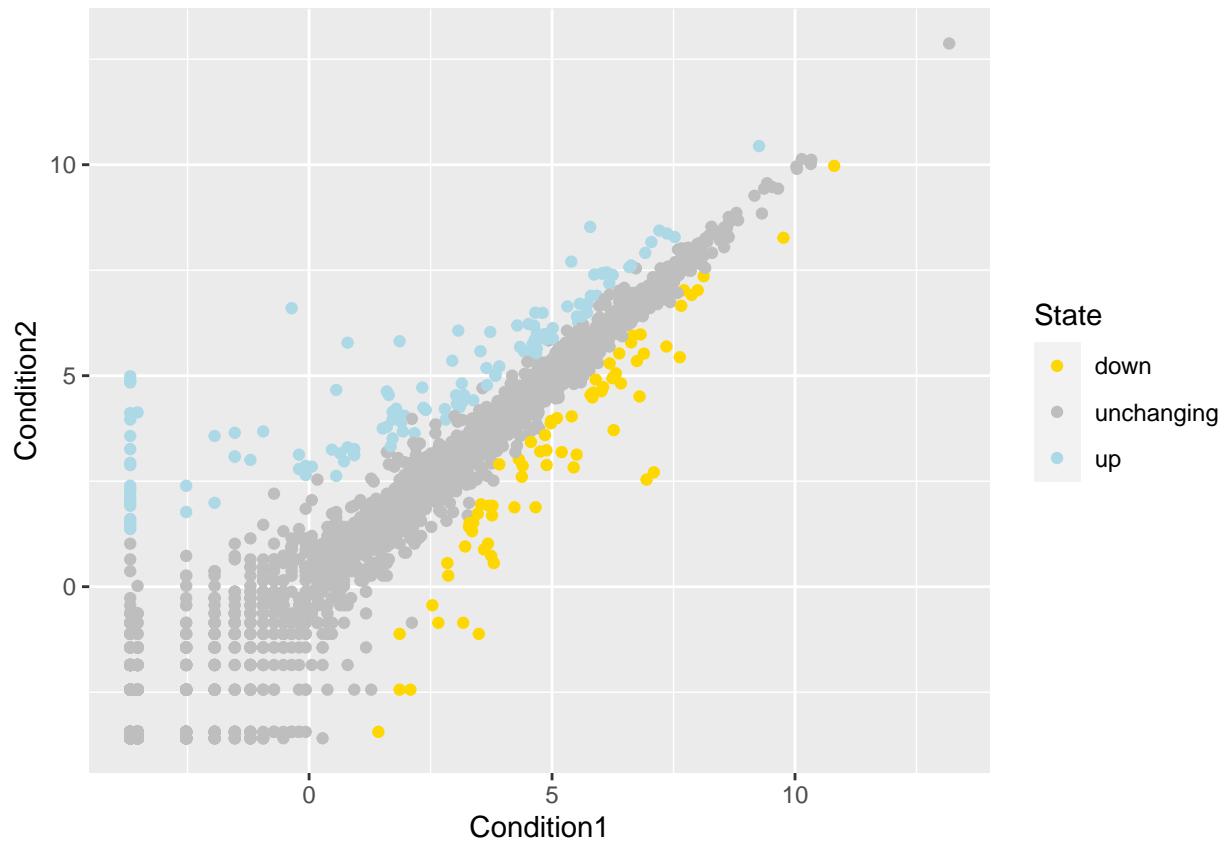
##
##      down unchanged      up
##      1.39        96.17     2.44

# Q. Make plot
q <- ggplot(data=genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point()
q

```

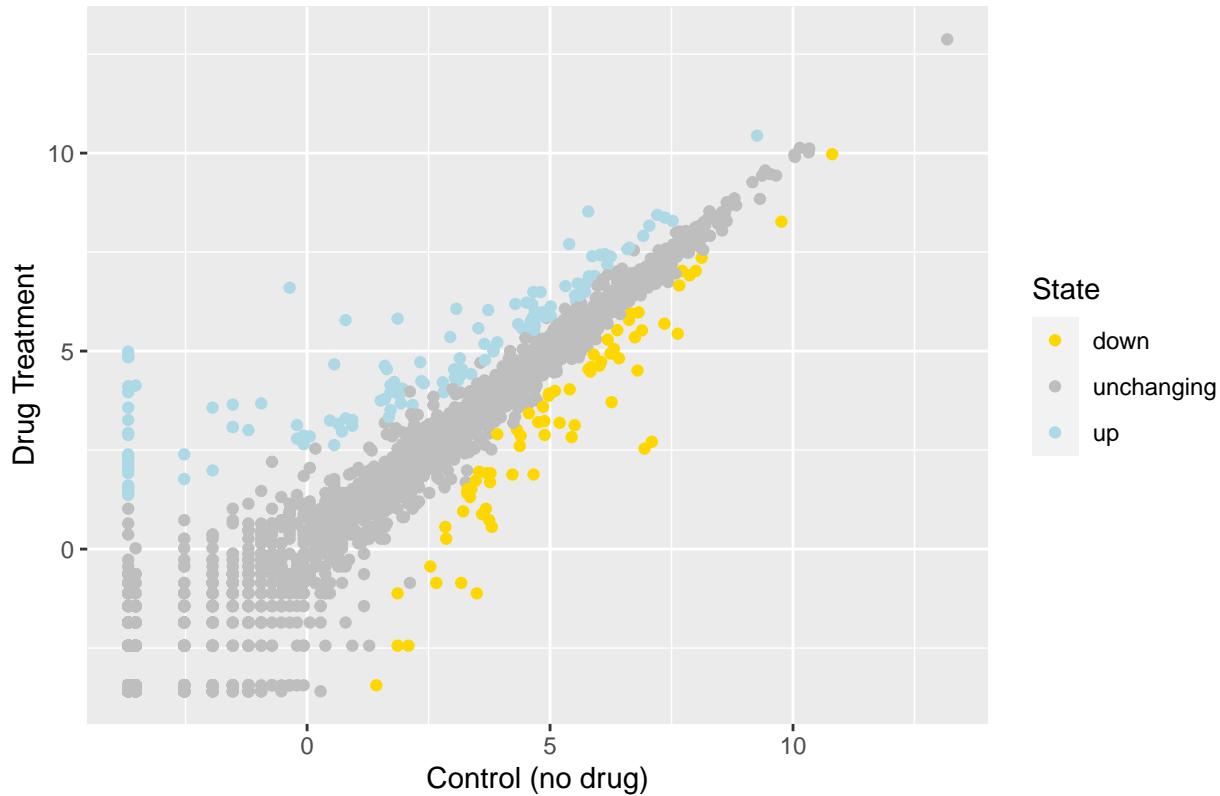


```
# **Plot with custom settings.
q + scale_color_manual(values=c("gold", "gray", "lightblue"))
```



```
# **Plot with labs settings.
q + scale_color_manual(values=c("gold", "gray", "lightblue")) +
  labs (title="Gene Expression Changes Upon Drug Treatment",
        x=("Control (no drug)"),
        y=("Drug Treatment"))
```

Gene Expression Changes Upon Drug Treatment



```
# New data set...

# **Install, then load gapminder.
library (gapminder)

# Install, then load dplyr.
library(dplyr)

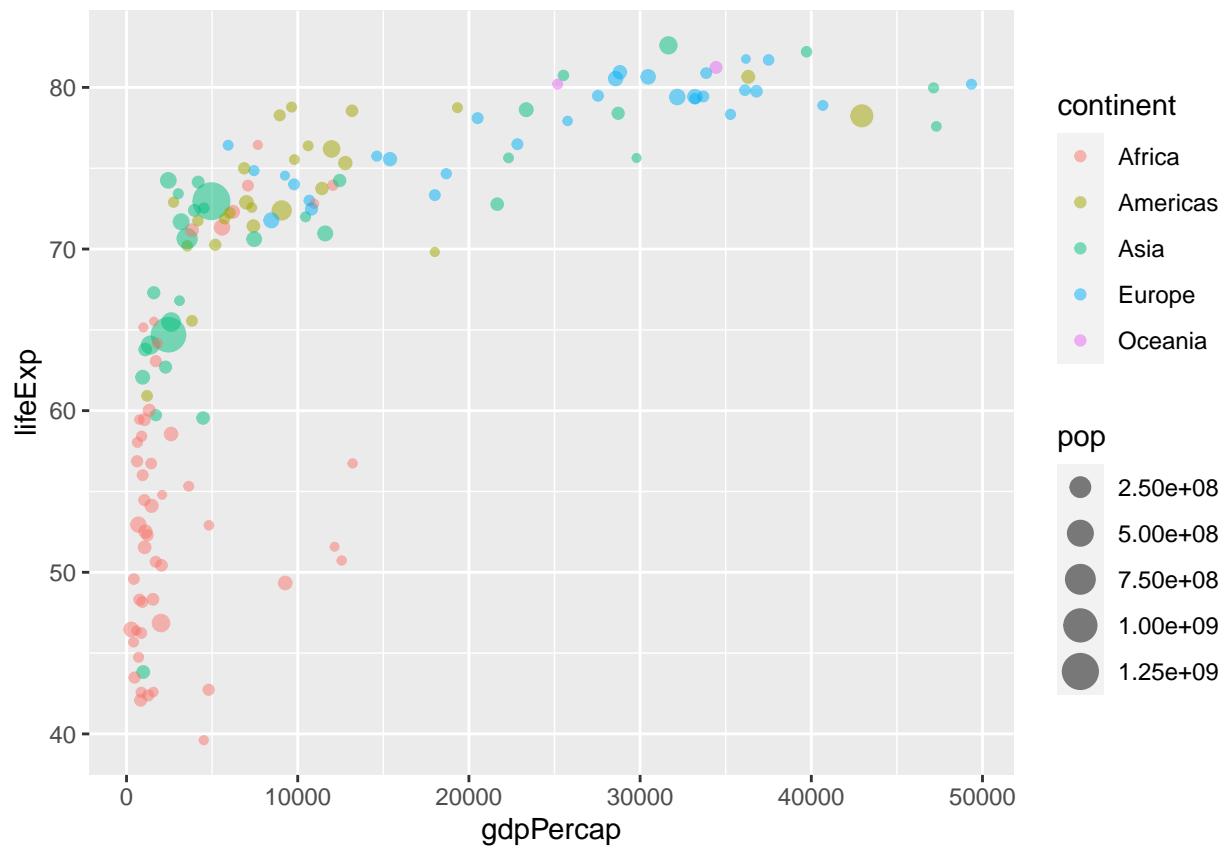
##
## Attaching package: 'dplyr'

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

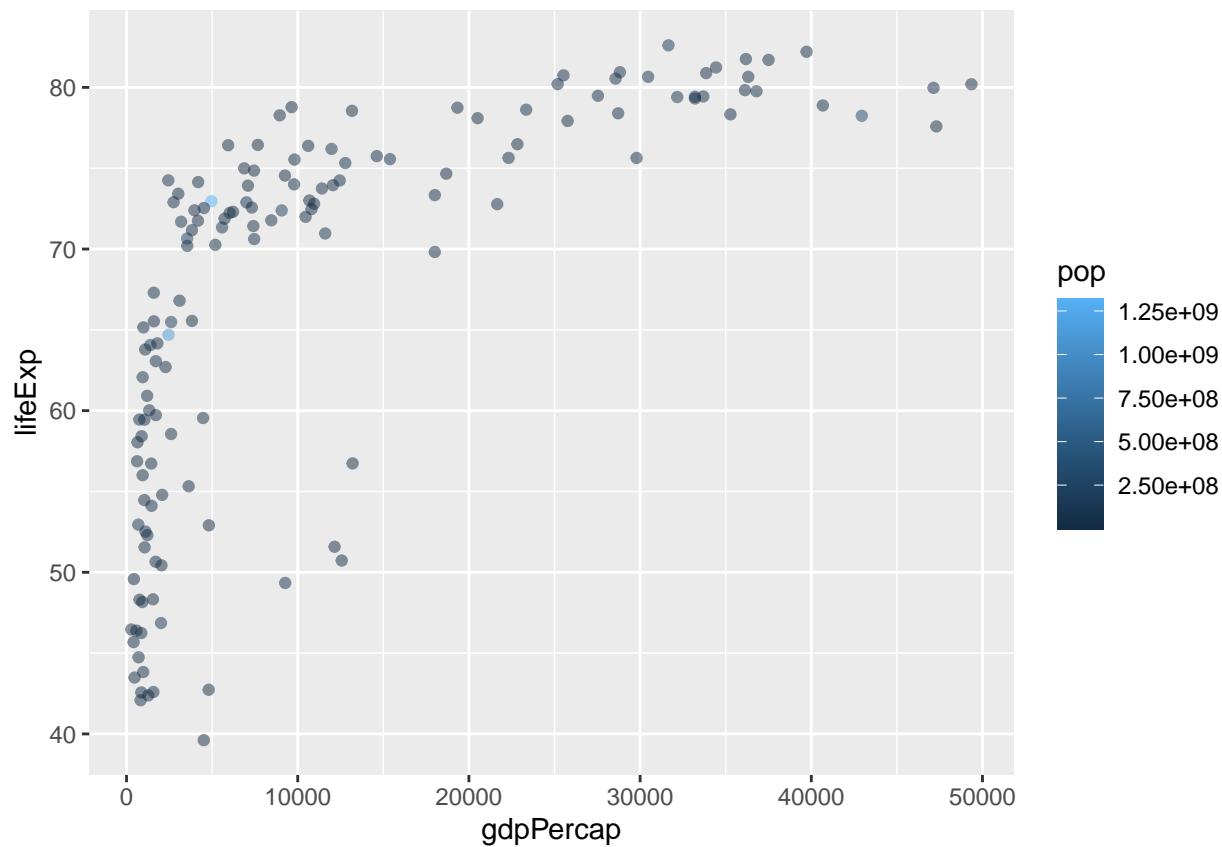
## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

# Filter to year 2007.
gapminder_2007 <- gapminder %>% filter(year==2007)

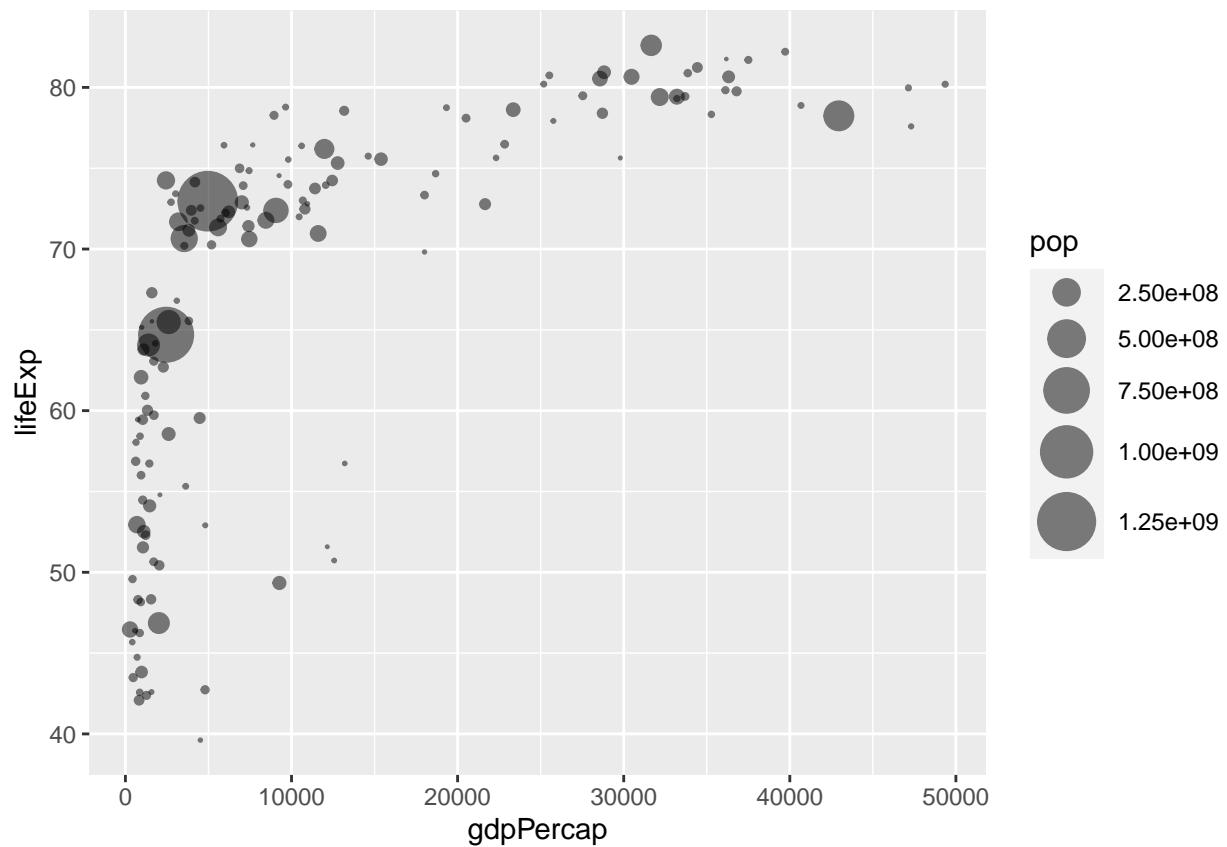
# Q. Scatter plot of gapminder_2007.
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  geom_point(alpha=0.5)
```



```
# Another version: points colored by population size.
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp, color=pop) +
  geom_point(alpha=0.5)
```



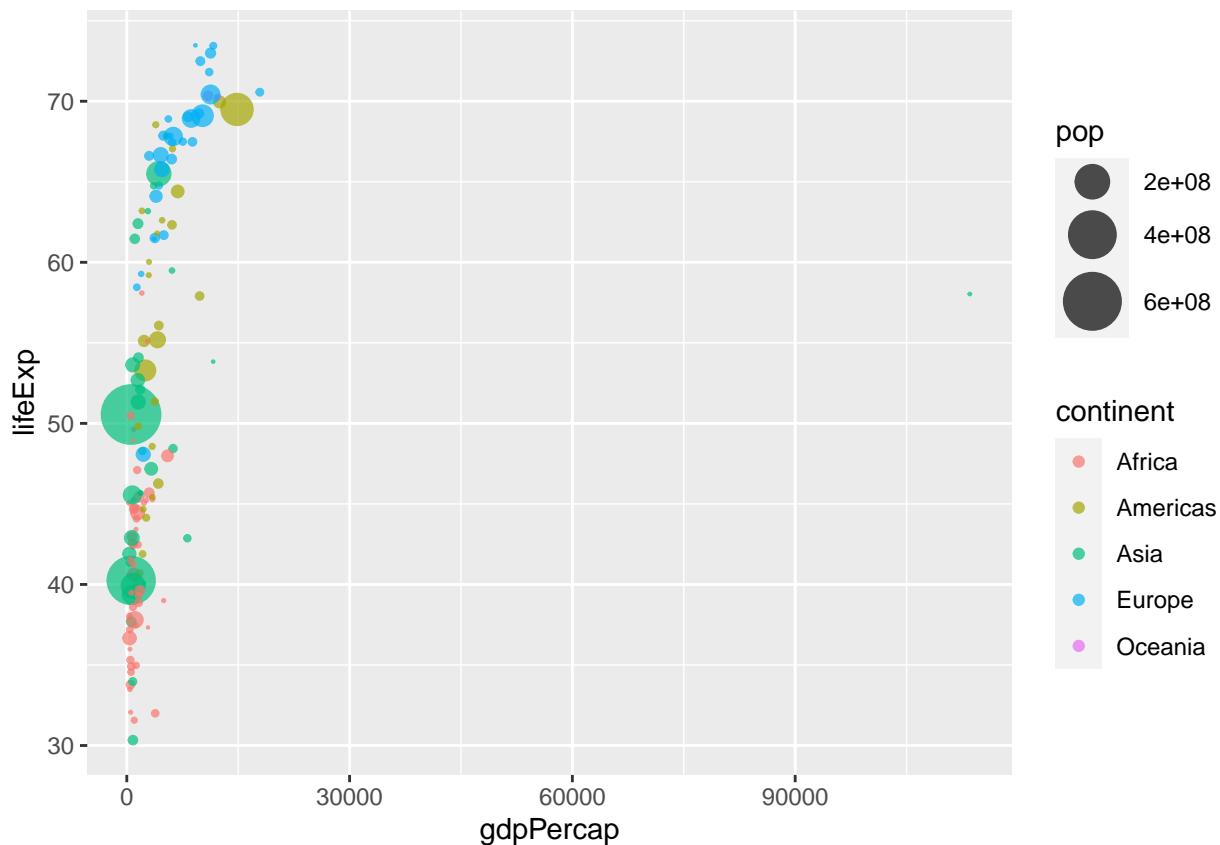
```
# Another version: adjusting point size based on population size.  
ggplot(gapminder_2007) +  
  geom_point(aes(x = gdpPercap, y = lifeExp,  
                 size = pop), alpha=0.5) +  
  scale_size_area(max_size = 10)
```



```
# Q. For the year 1957:

gapminder_1957 <- gapminder %>% filter(year==1957)

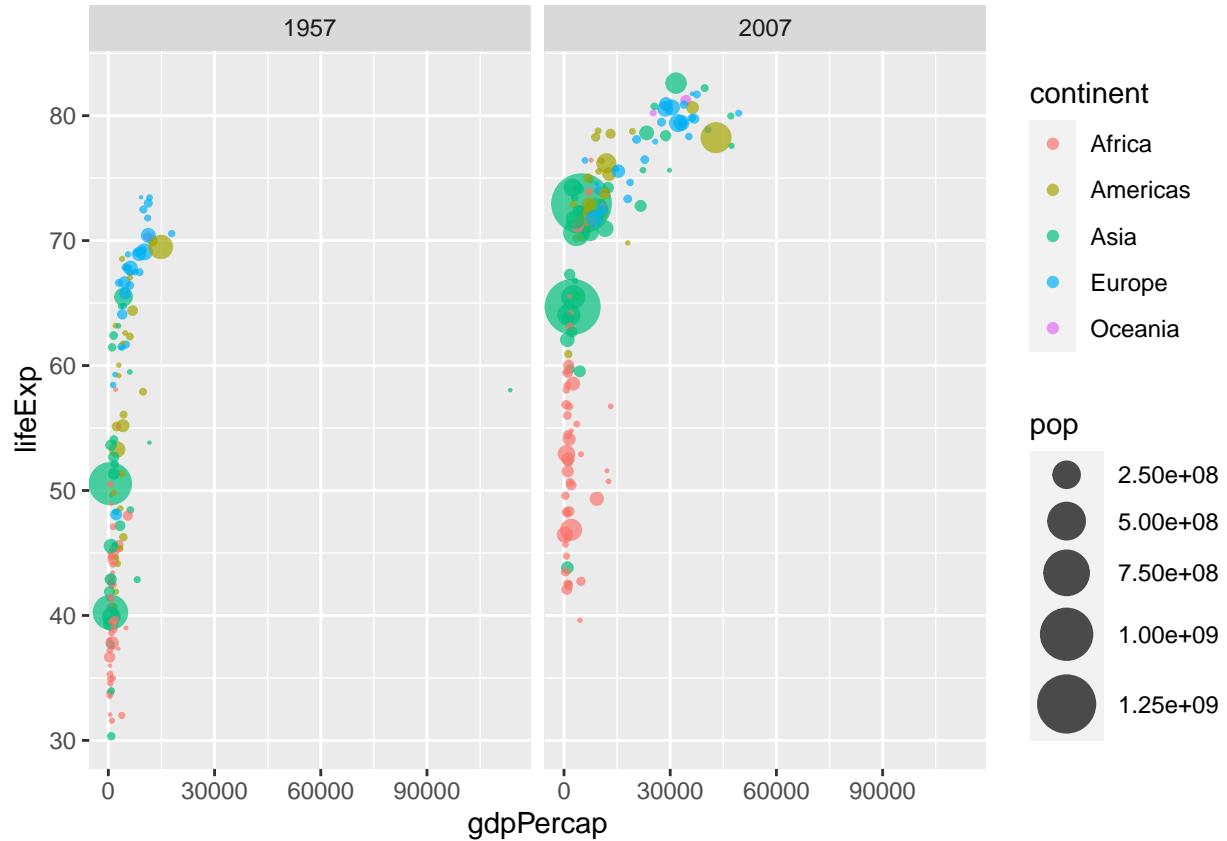
ggplot(gapminder_1957) +
  aes(x=gdpPercap, y=lifeExp,
      color=continent,
      size=pop)+
  geom_point(alpha=0.7)+
  scale_size_area(max_size=10)
```



```
# Q. For the years 1957 AND 2007"
```

```
gapminder_1957.2007 <- gapminder %>% filter(year==1957 | year==2007)

ggplot(gapminder_1957.2007) +
  aes(x=gdpPercap, y=lifeExp,
      color=continent,
      size=pop)+
  geom_point(alpha=0.7)+
  scale_size_area(max_size=10) +
  facet_wrap(~year)
```



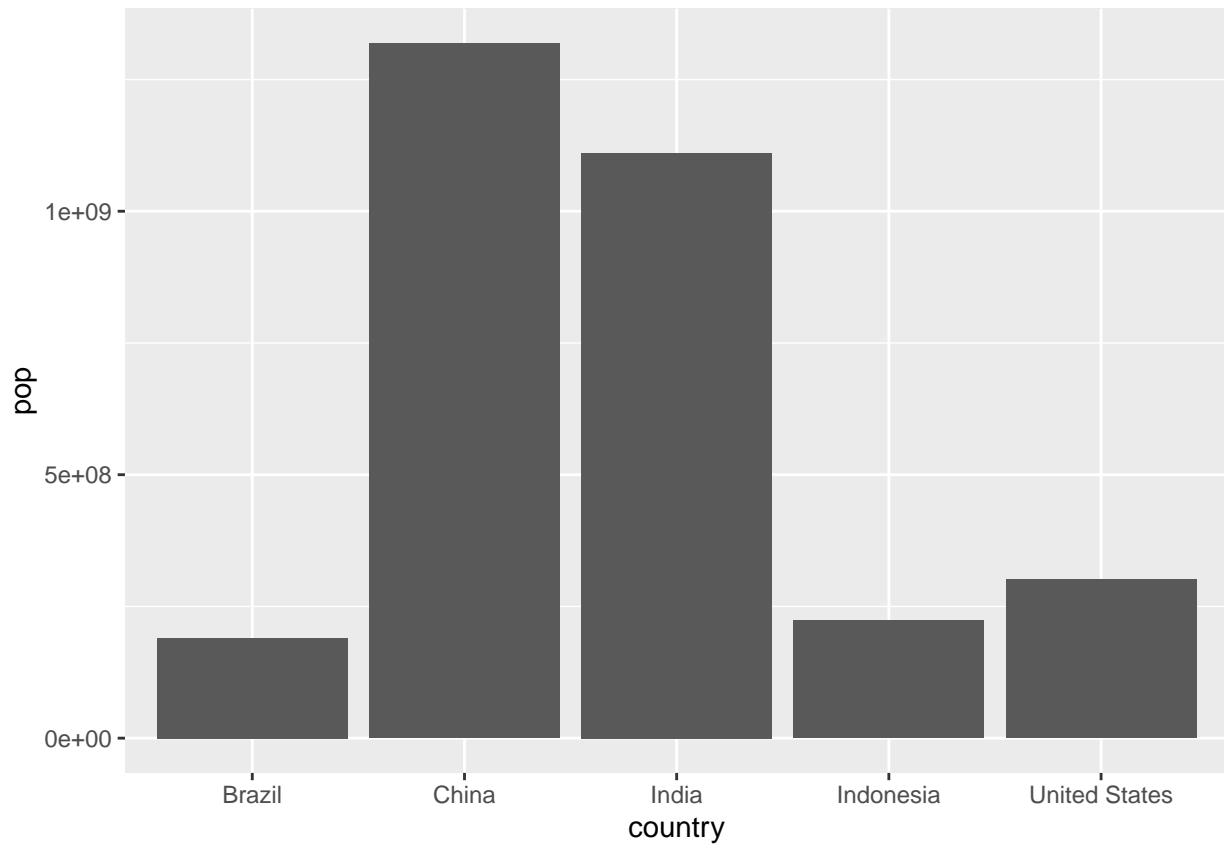
BAR CHARTS

```
# Data for 5 biggest countries:
gapminder_top5 <- gapminder %>%
  filter(year==2007) %>%
  arrange(desc(pop)) %>%
  top_n(5, pop)

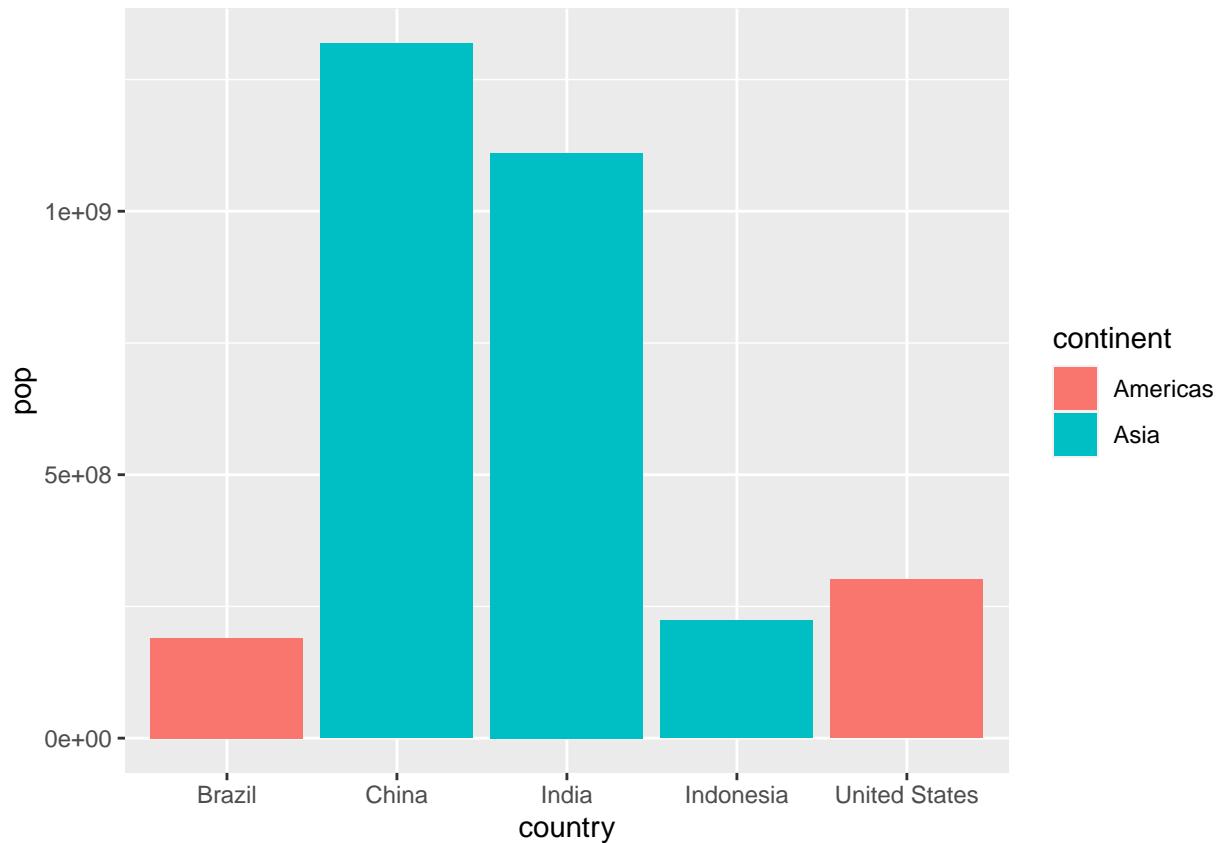
gapminder_top5
```

```
## # A tibble: 5 x 6
##   country     continent year lifeExp      pop gdpPercap
##   <fct>       <fct>    <int>   <dbl>     <int>     <dbl>
## 1 China        Asia     2007    73.0 1318683096     4959.
## 2 India        Asia     2007    64.7 1110396331     2452.
## 3 United States Americas 2007    78.2 301139947     42952.
## 4 Indonesia    Asia     2007    70.6 223547000     3541.
## 5 Brazil       Americas 2007    72.4 190010647     9066.
```

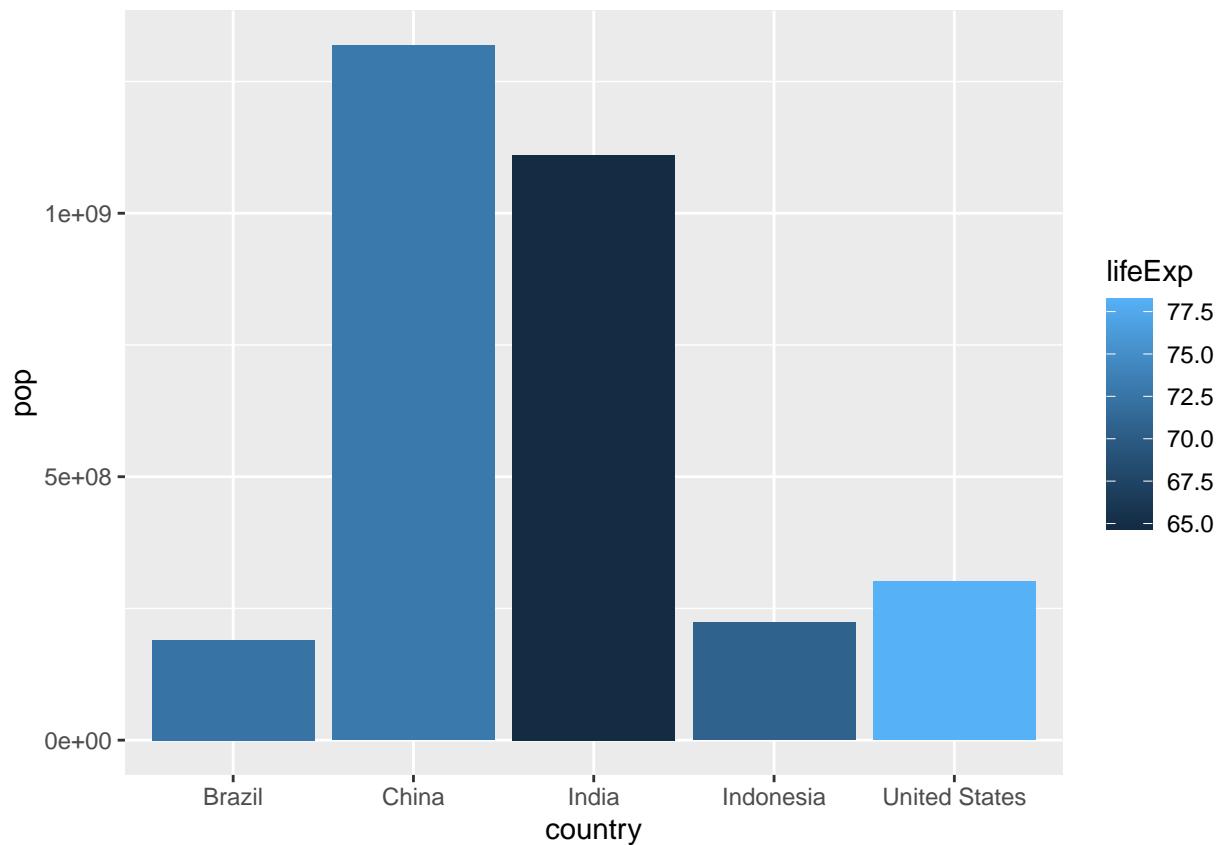
```
# A simple bar chart:
ggplot(gapminder_top5) +
  geom_col(aes(x=country,y=pop))
```



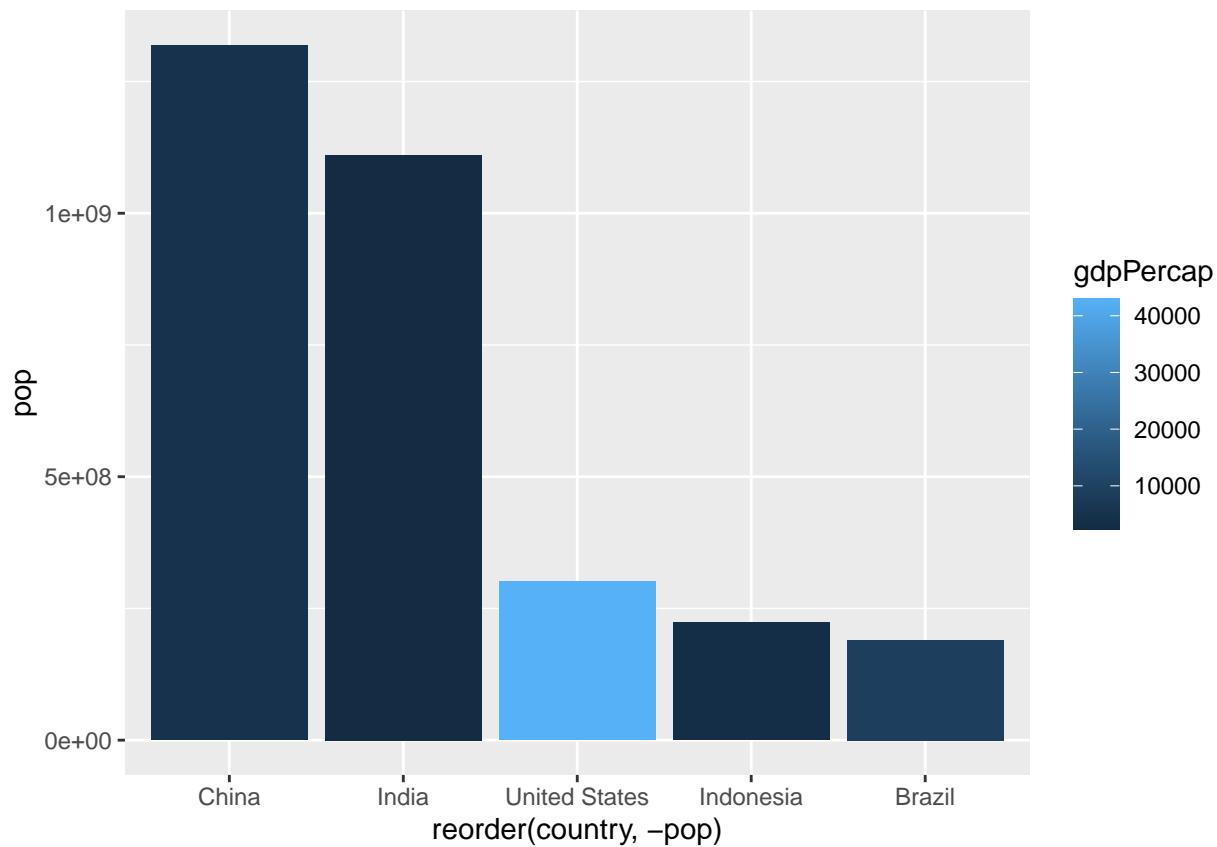
```
# Fill by continent:  
ggplot(gapminder_top5) +  
  geom_col(aes(x=country,y=pop,  
               fill=continent))
```



```
# Fill by life expectancy:  
ggplot(gapminder_top5) +  
  geom_col(aes(x=country,y=pop,  
               fill=lifeExp) )
```

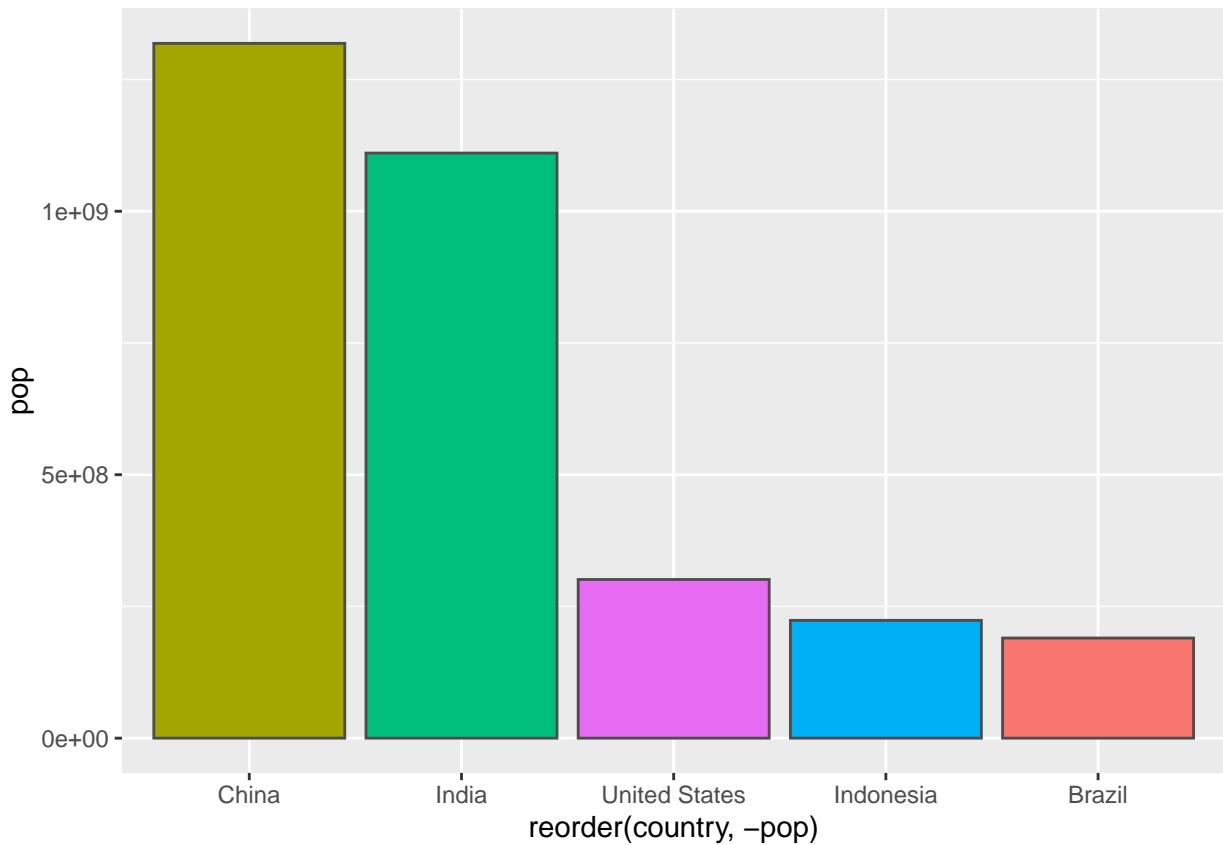


```
# Fill by GDP per capita, change the order of bars:  
ggplot(gapminder_top5) +  
  aes(x=reorder(country,-pop),y=pop, fill=gdpPercap)+  
  geom_col()
```



```
# Just fill by country
ggplot(gapminder_top5) +
  aes(x=reorder(country, -pop), y=pop, fill=country) +
  geom_col(col="gray30") +
  guides(fill=FALSE)

## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```



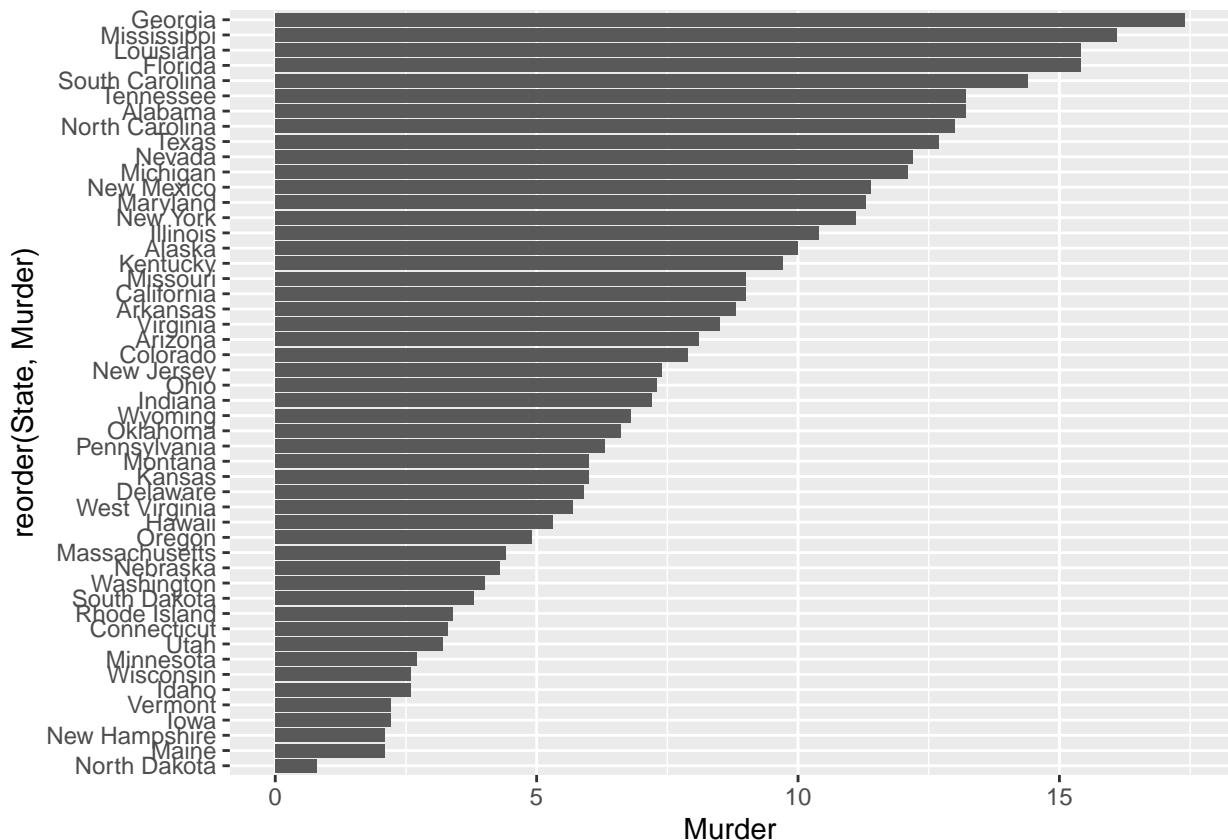
```
# New data set...
```

```
head(USArrests)
```

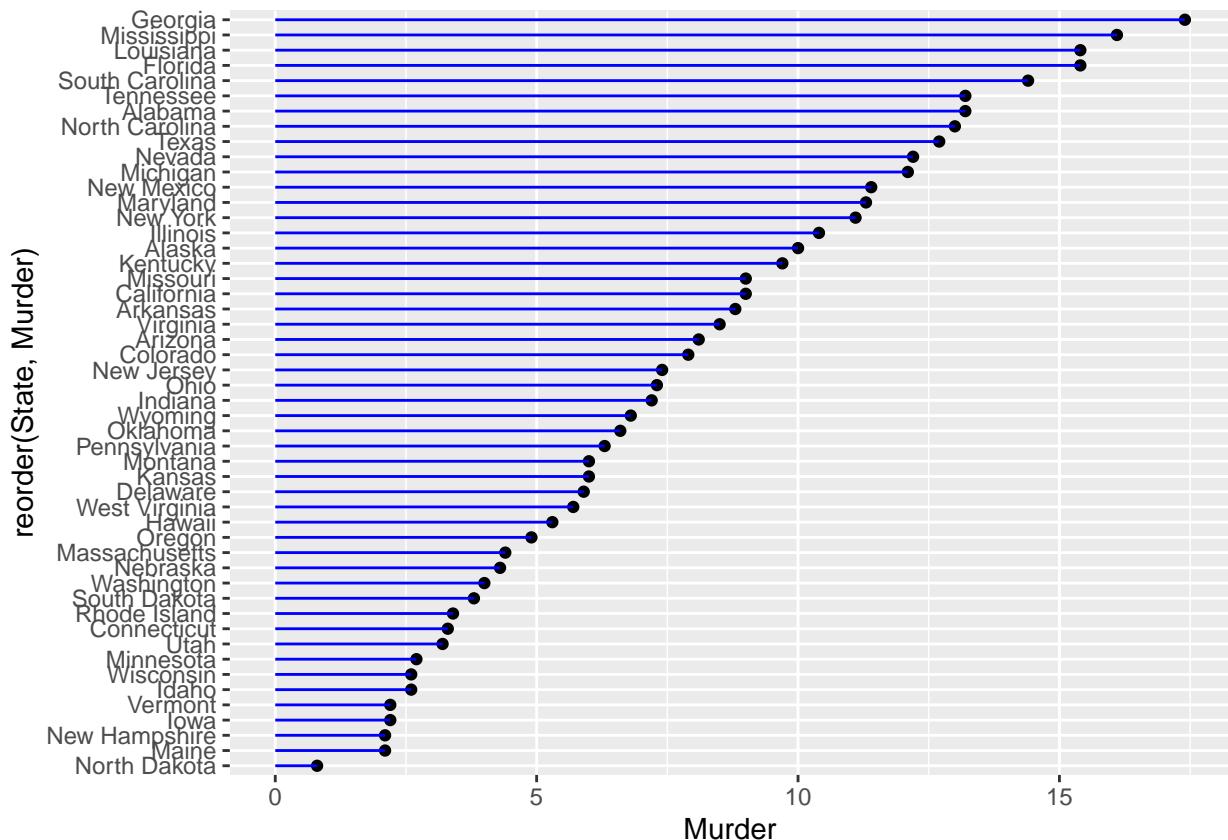
```
##          Murder Assault UrbanPop Rape
## Alabama    13.2     236      58 21.2
## Alaska     10.0     263      48 44.5
## Arizona     8.1     294      80 31.0
## Arkansas    8.8     190      50 19.5
## California   9.0     276      91 40.6
## Colorado    7.9     204      78 38.7
```

```
# Flipped bar chart:
```

```
USArrests$State <- rownames(USArrests)
ggplot(USArrests) +
  aes(x=reorder(State,Murder),y=Murder) +
  geom_col() +
  coord_flip()
```



```
# New format:
USArrests$State <- rownames(USArrests)
ggplot(USArrests) +
  aes(x=reorder(State,Murder) , y=Murder) +
  geom_point() +
  geom_segment(aes(x=State,
                    xend=State,
                    y=0,
                    yend=Murder) , color="blue") +
  coord_flip()
```

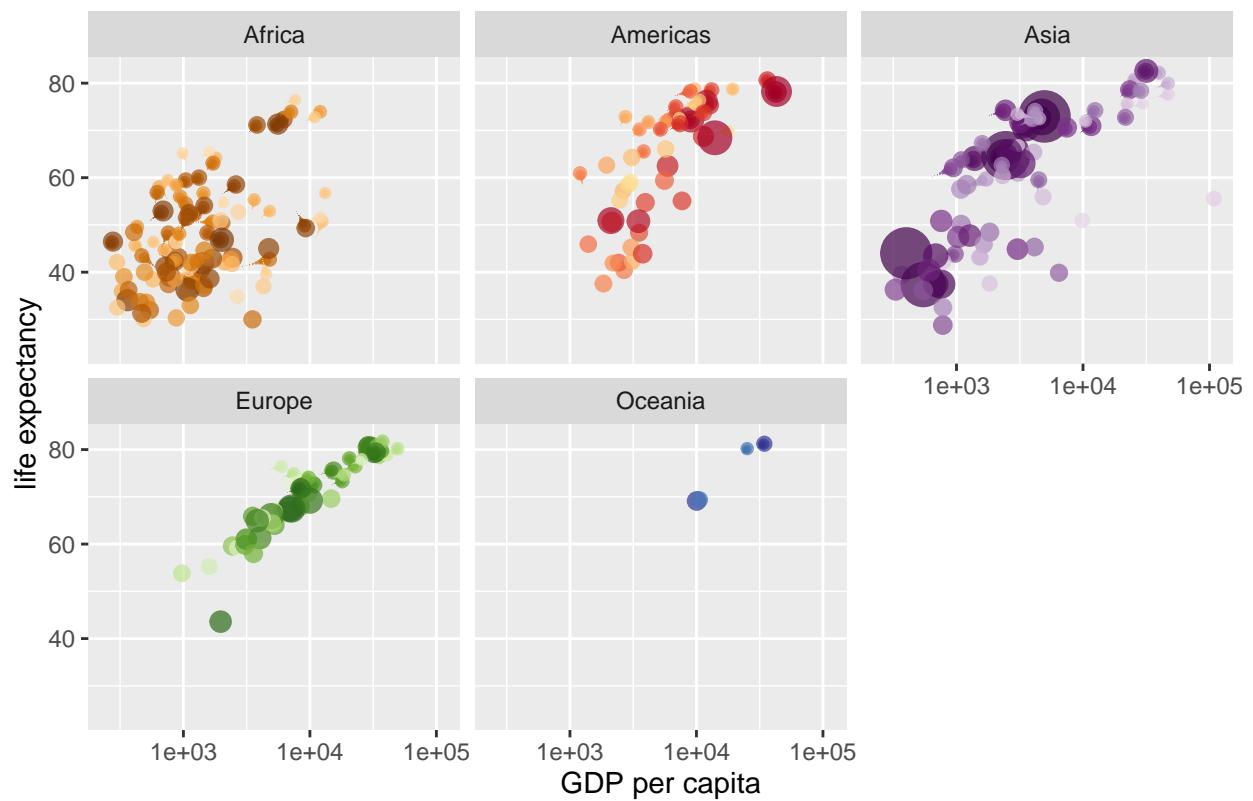


```
# PLOT ANIMATION
```

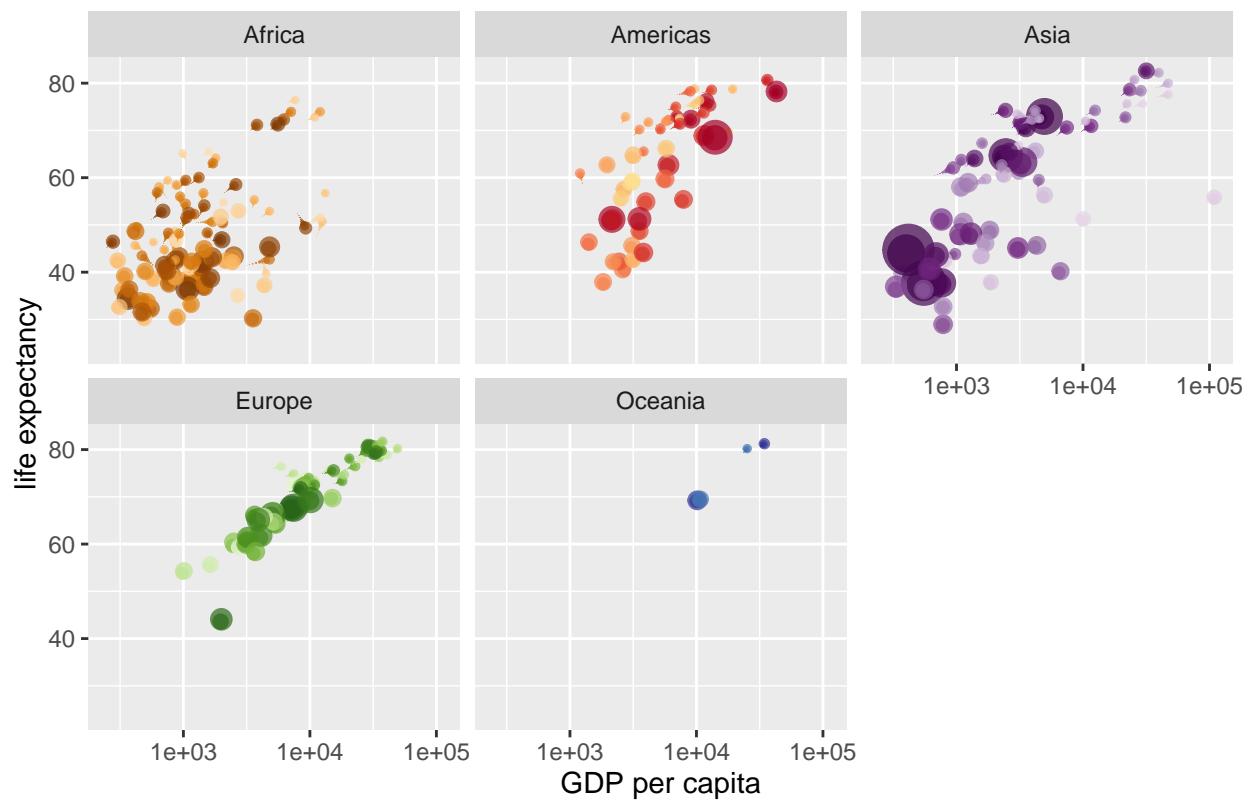
```
# Install, then load gifski & gganimate.
library (gganimate)
library (gifski)

# Normal ggplot of gapminder data:
ggplot (gapminder, aes(gdpPercap, lifeExp, size=pop, color=country))+ 
  geom_point(alpha=0.7, show.legend=FALSE) + 
  scale_color_manual(values=country_colors) + 
  scale_size(range= c(2,12)) + 
  scale_x_log10()+
  # Facet by continent
  facet_wrap (~continent) +
  # Animations :-)
  labs(title = 'Year: {frame_time}', x = 'GDP per capita', y = 'life expectancy') + 
  transition_time(year) + 
  shadow_wake(wake_length = 0.1, alpha = FALSE)
```

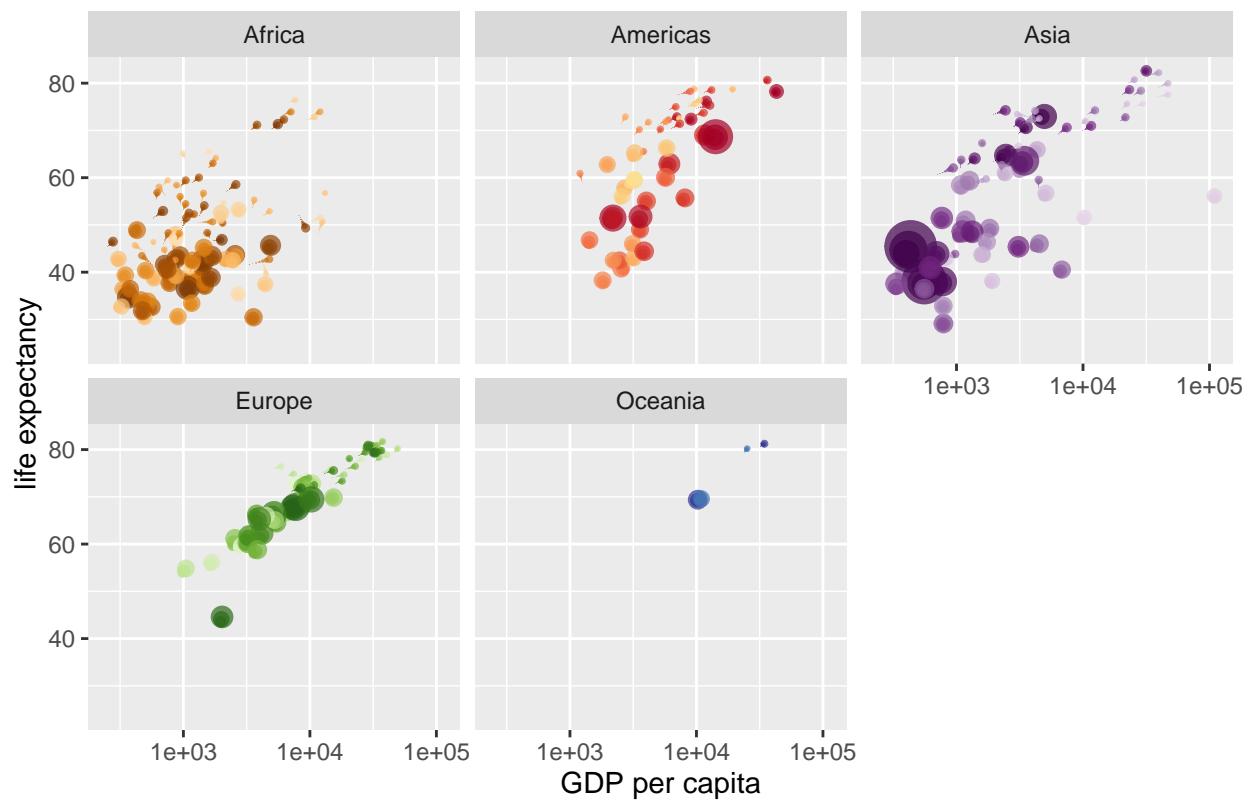
Year: 1952



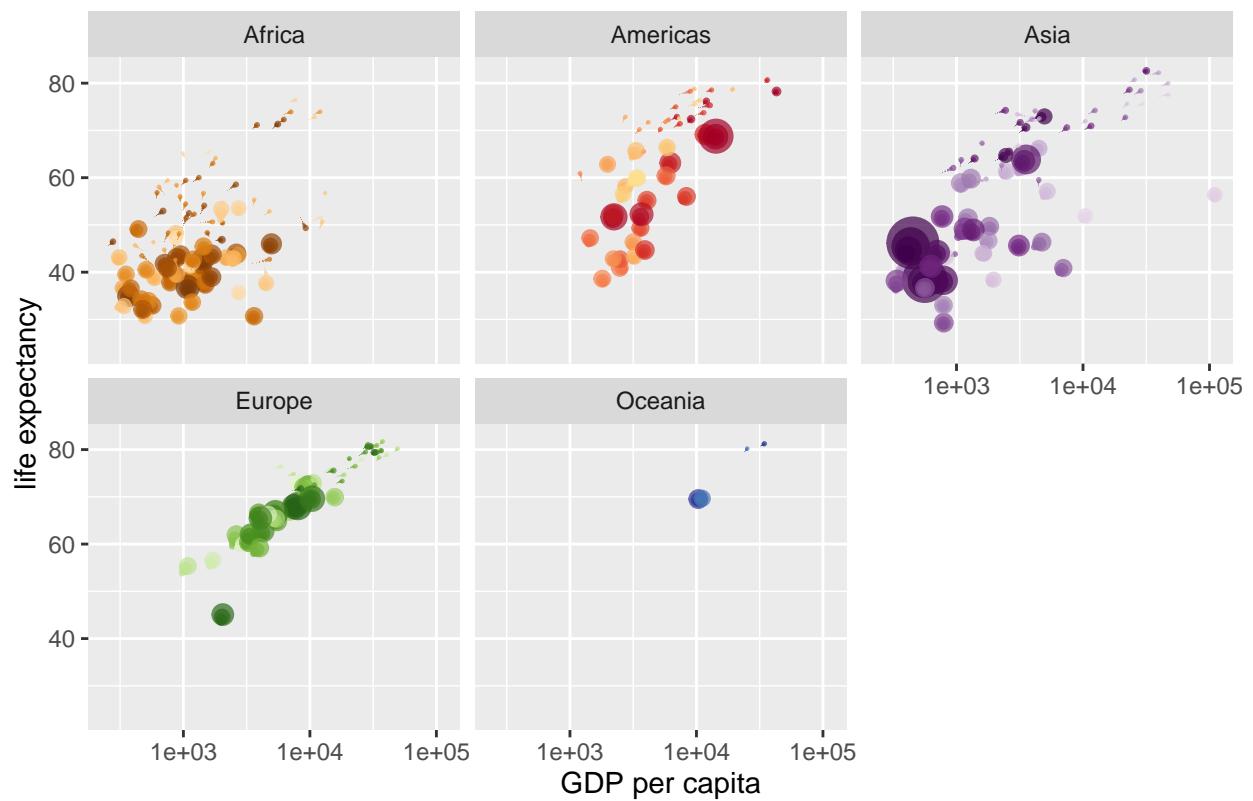
Year: 1953



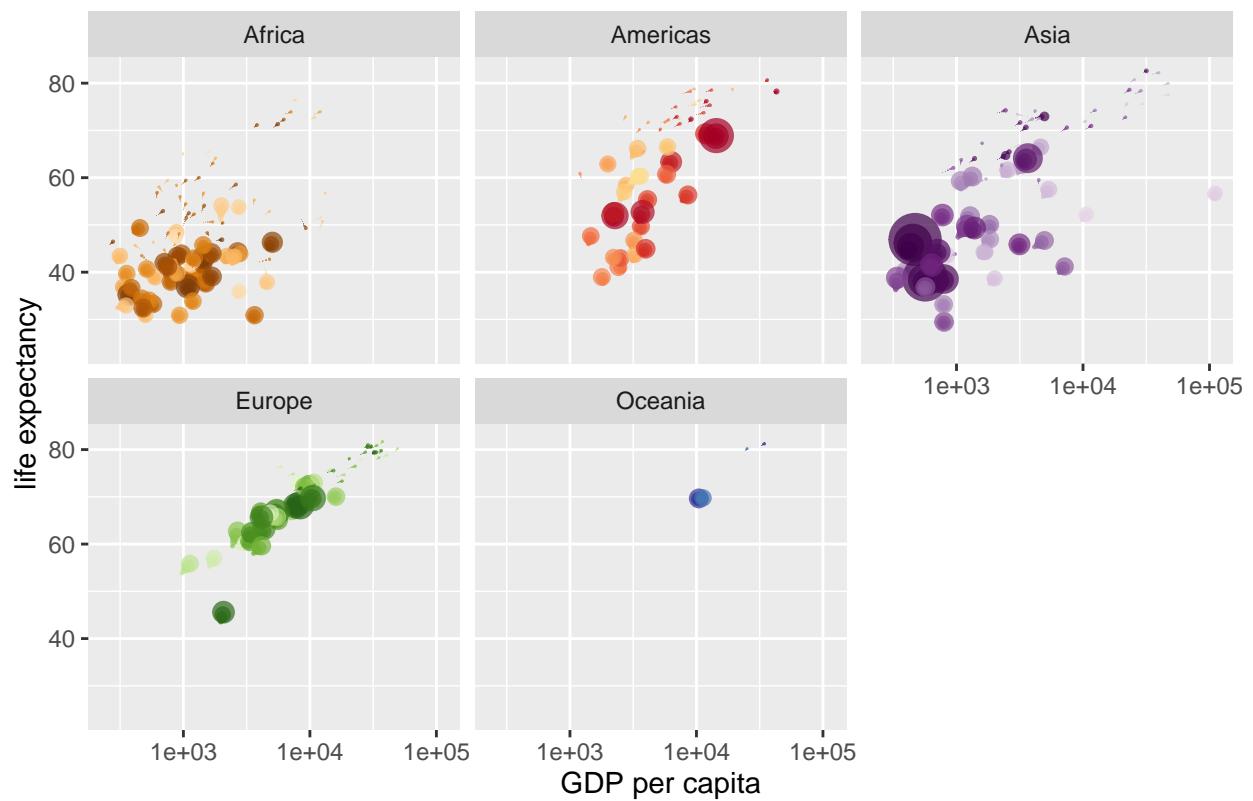
Year: 1953



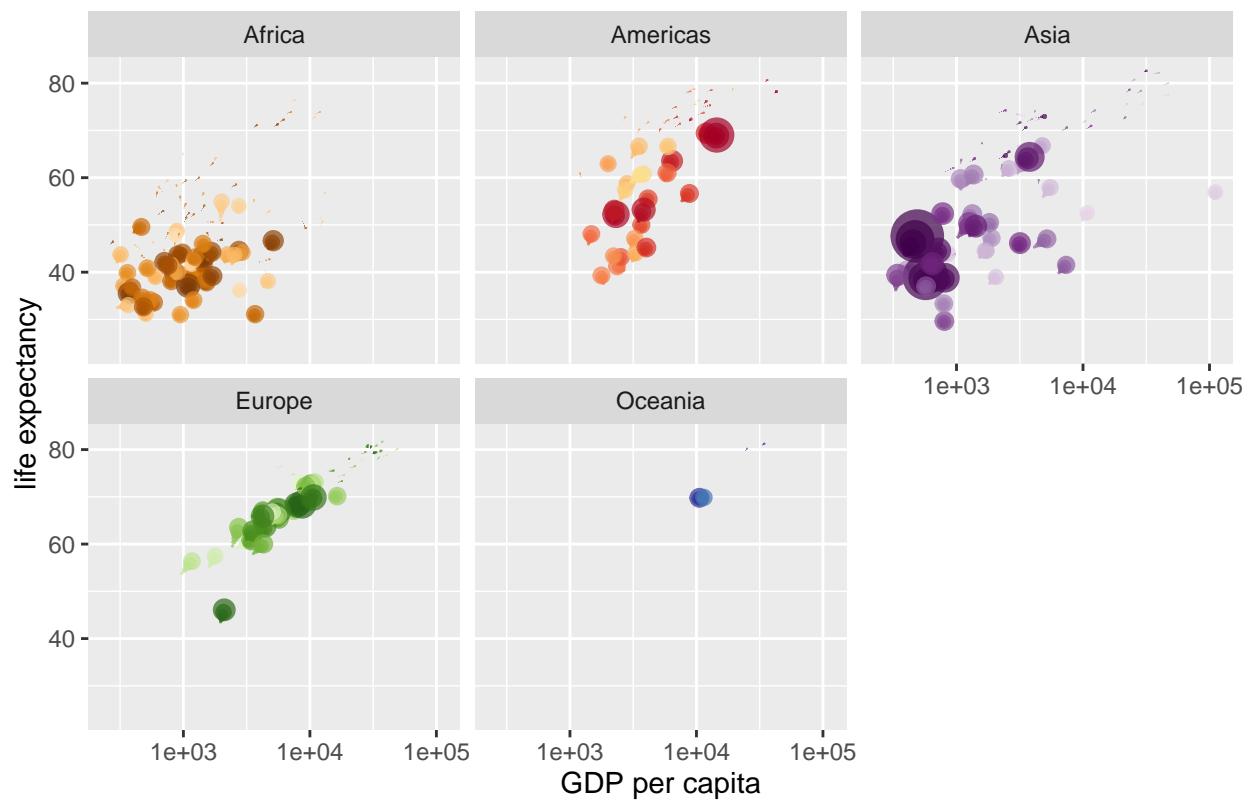
Year: 1954



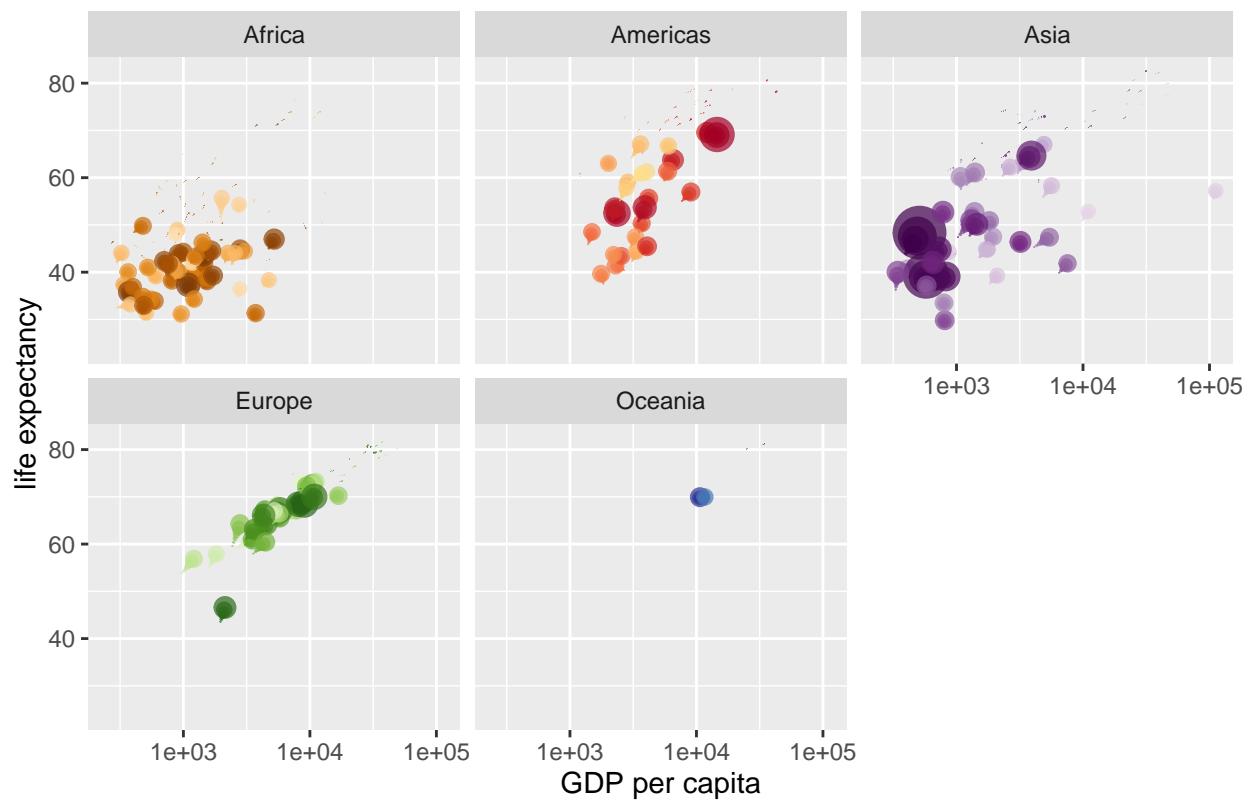
Year: 1954



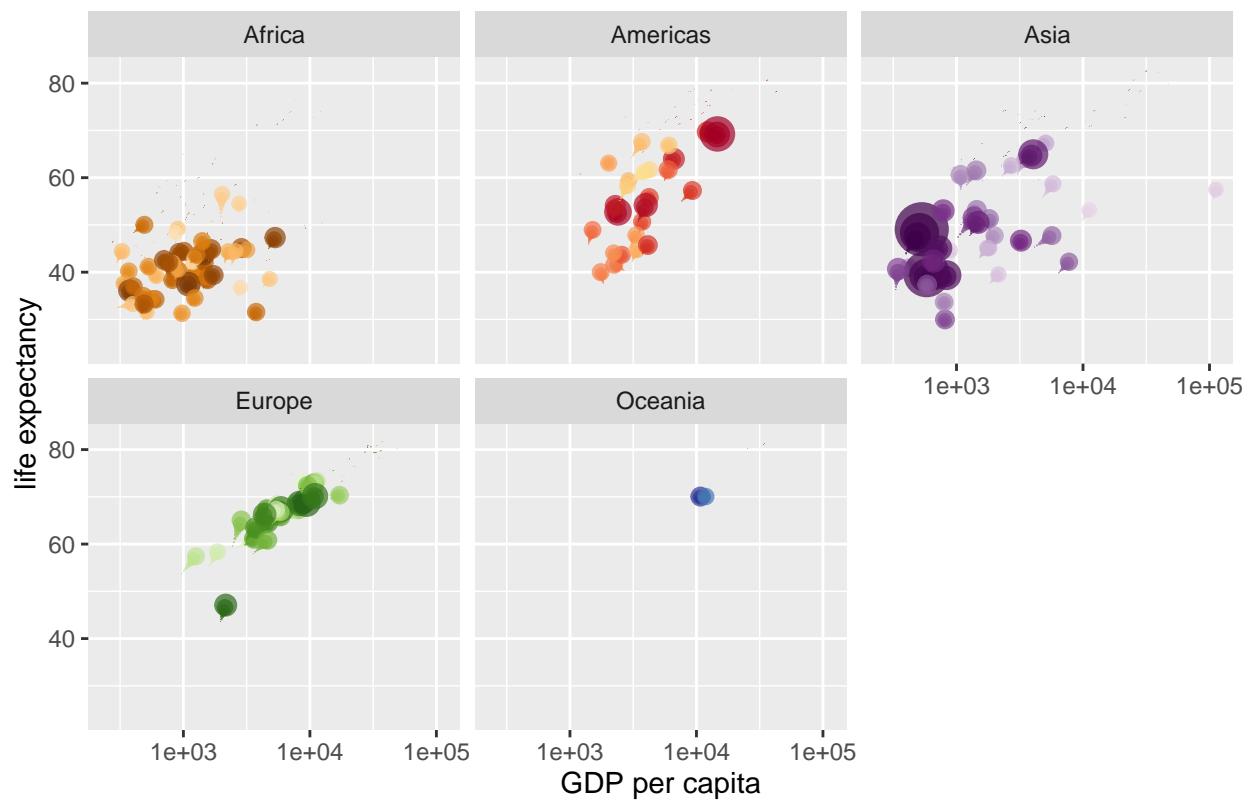
Year: 1955



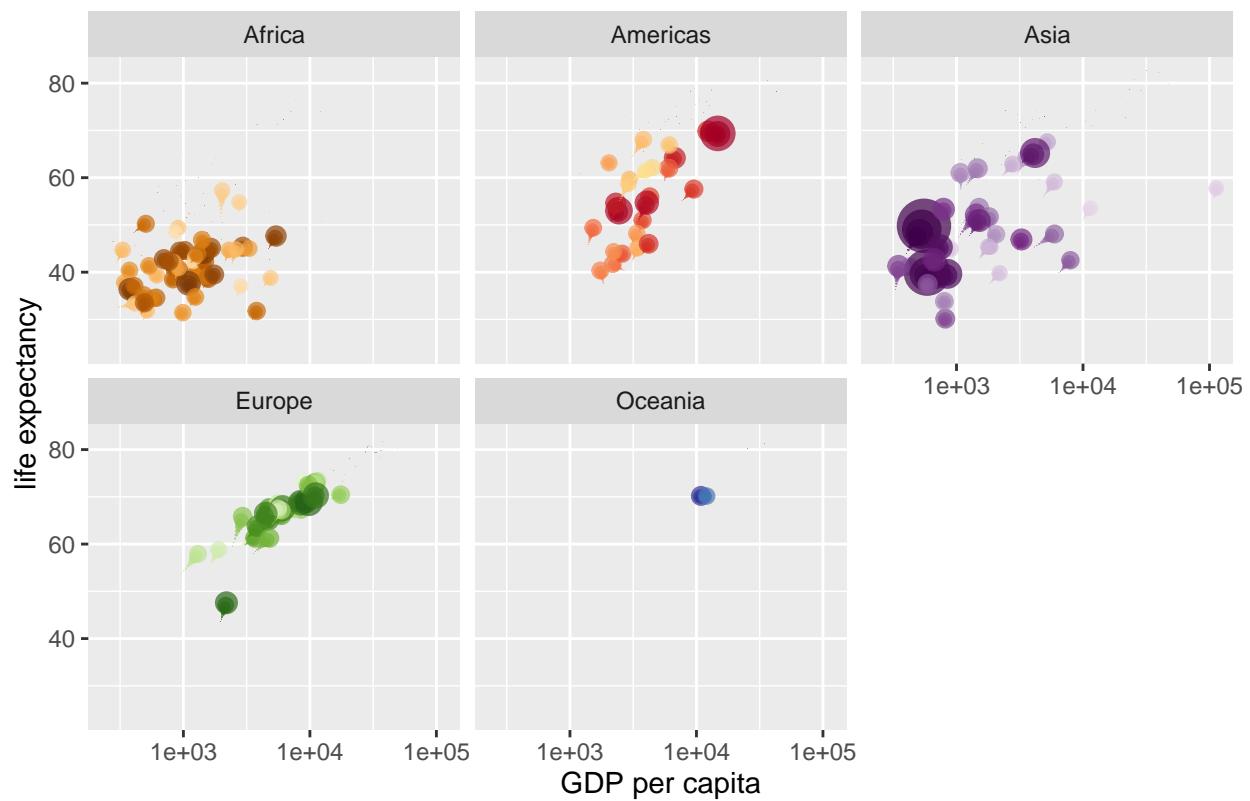
Year: 1955



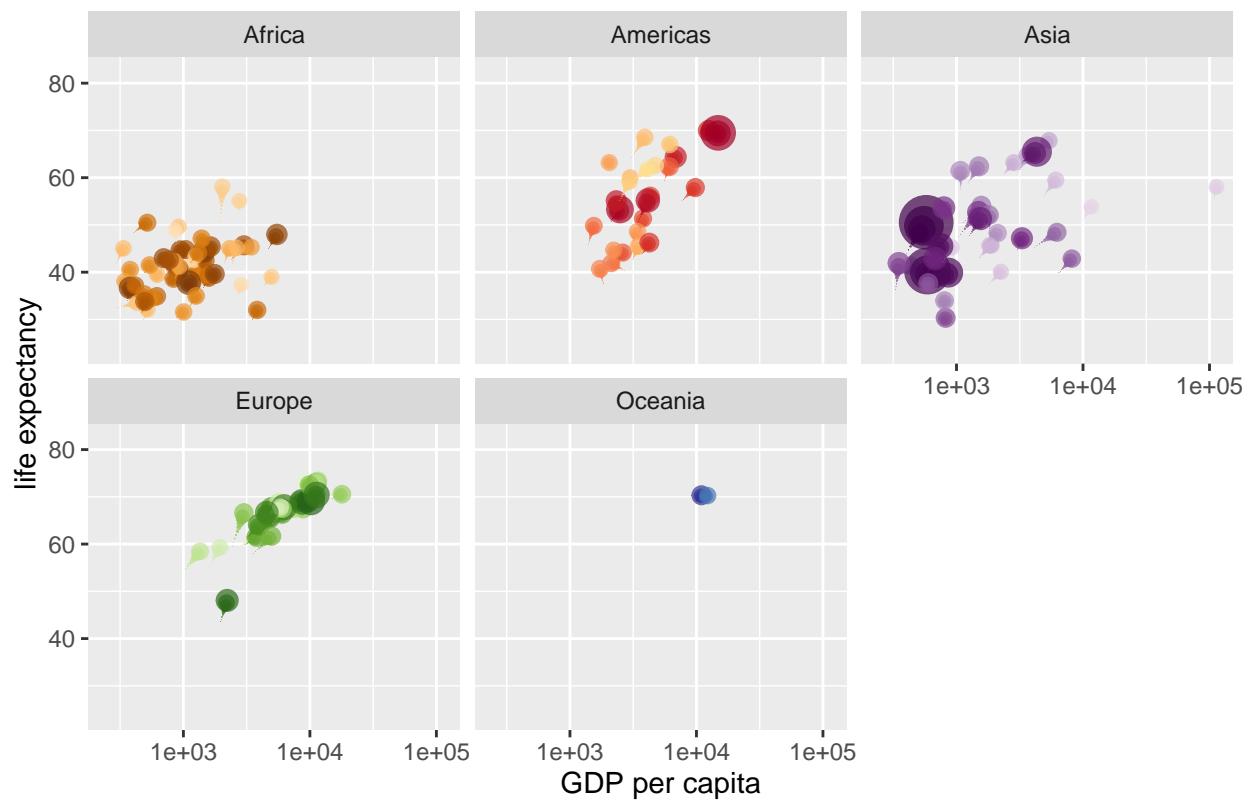
Year: 1956



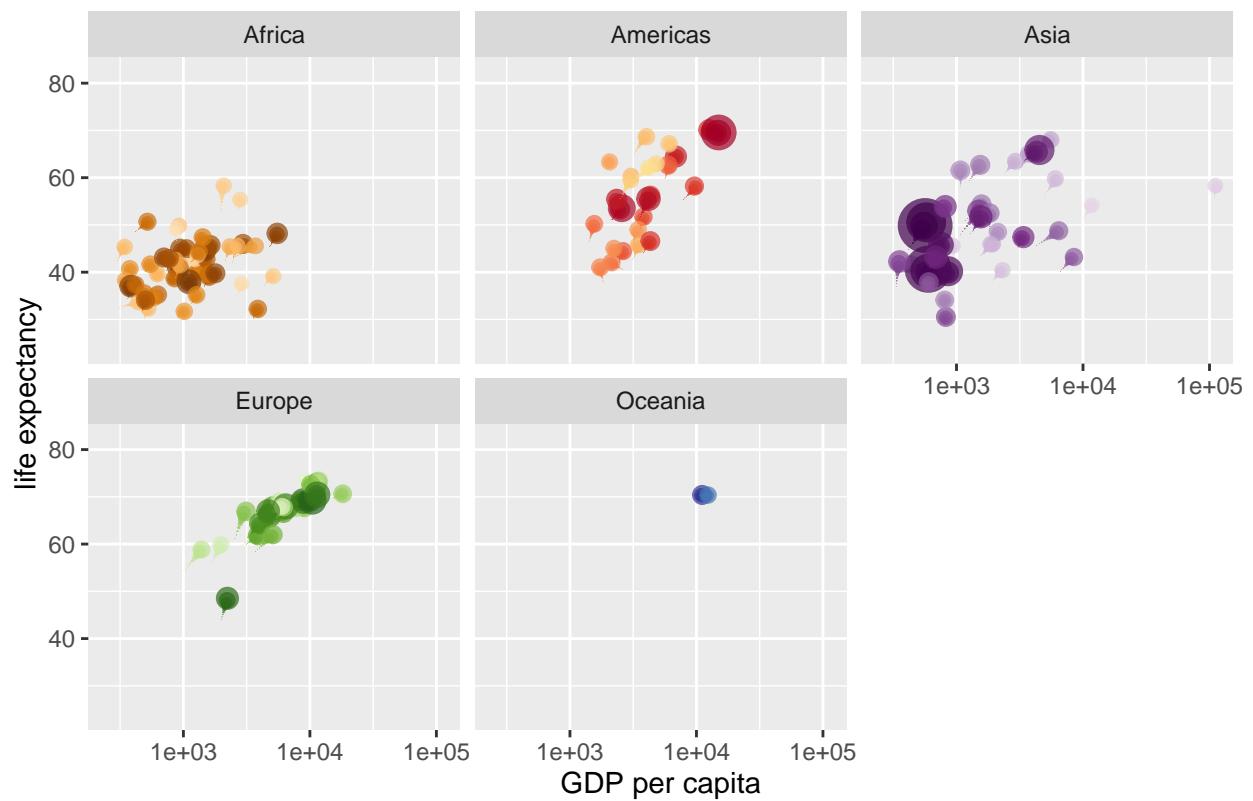
Year: 1956



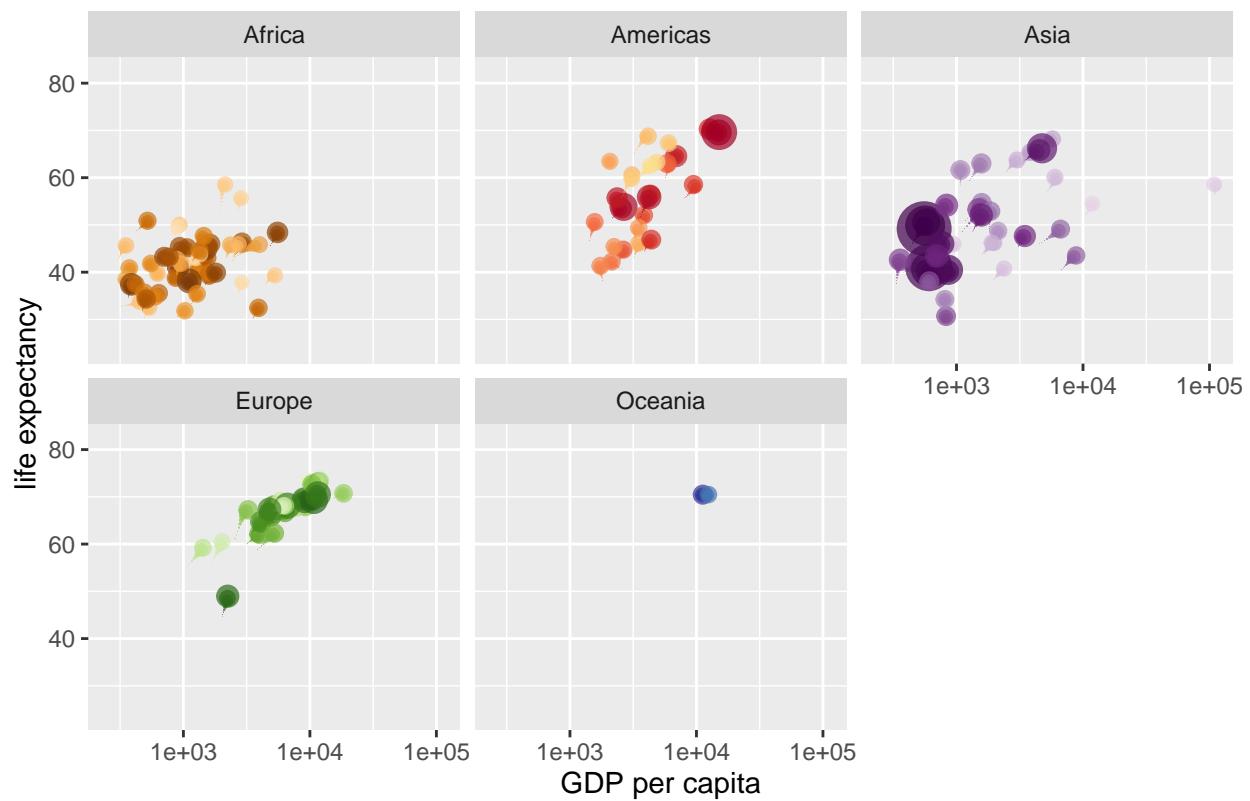
Year: 1957



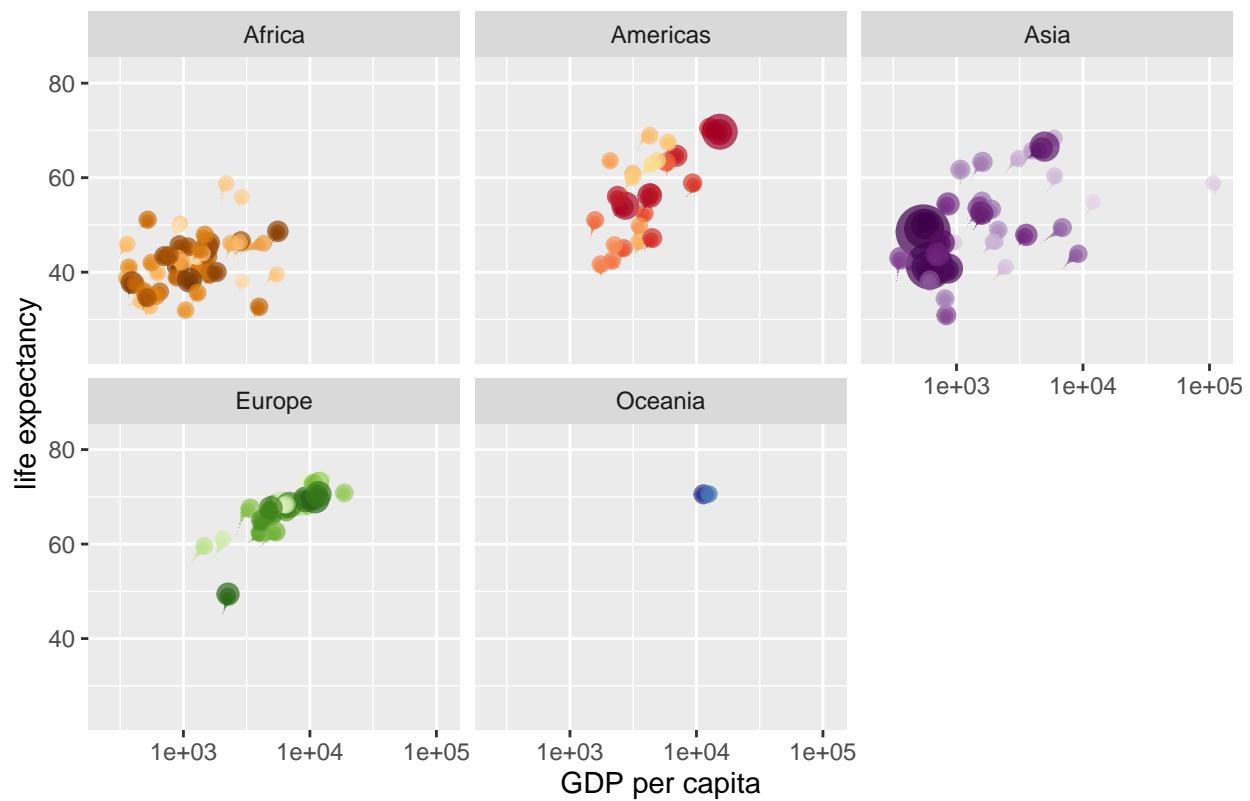
Year: 1958



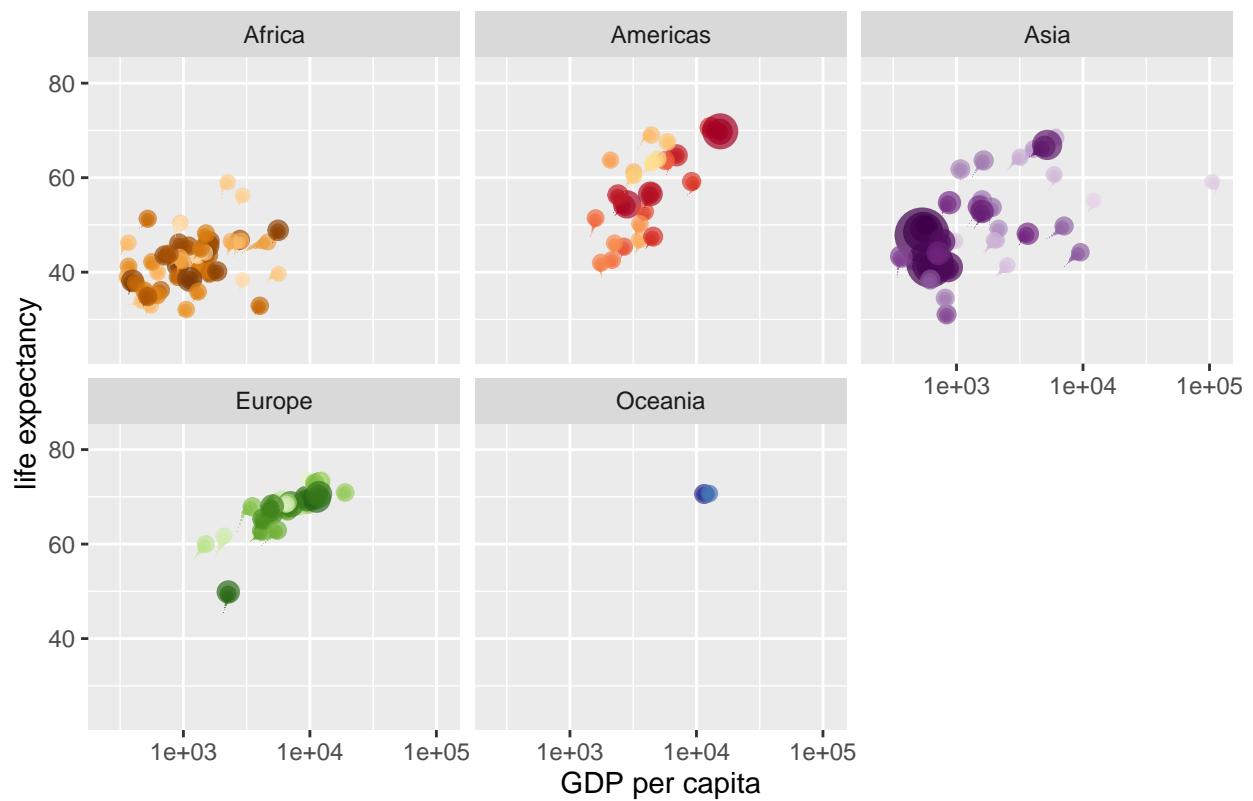
Year: 1958



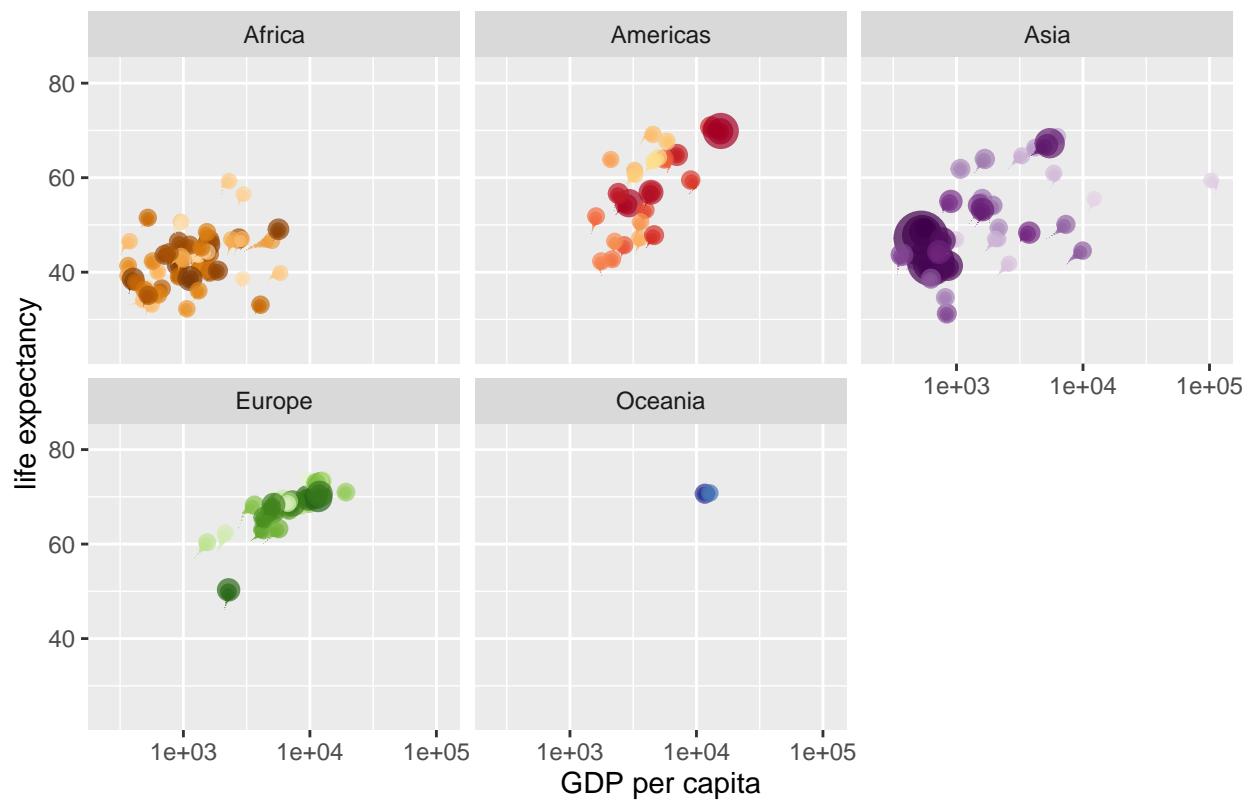
Year: 1959



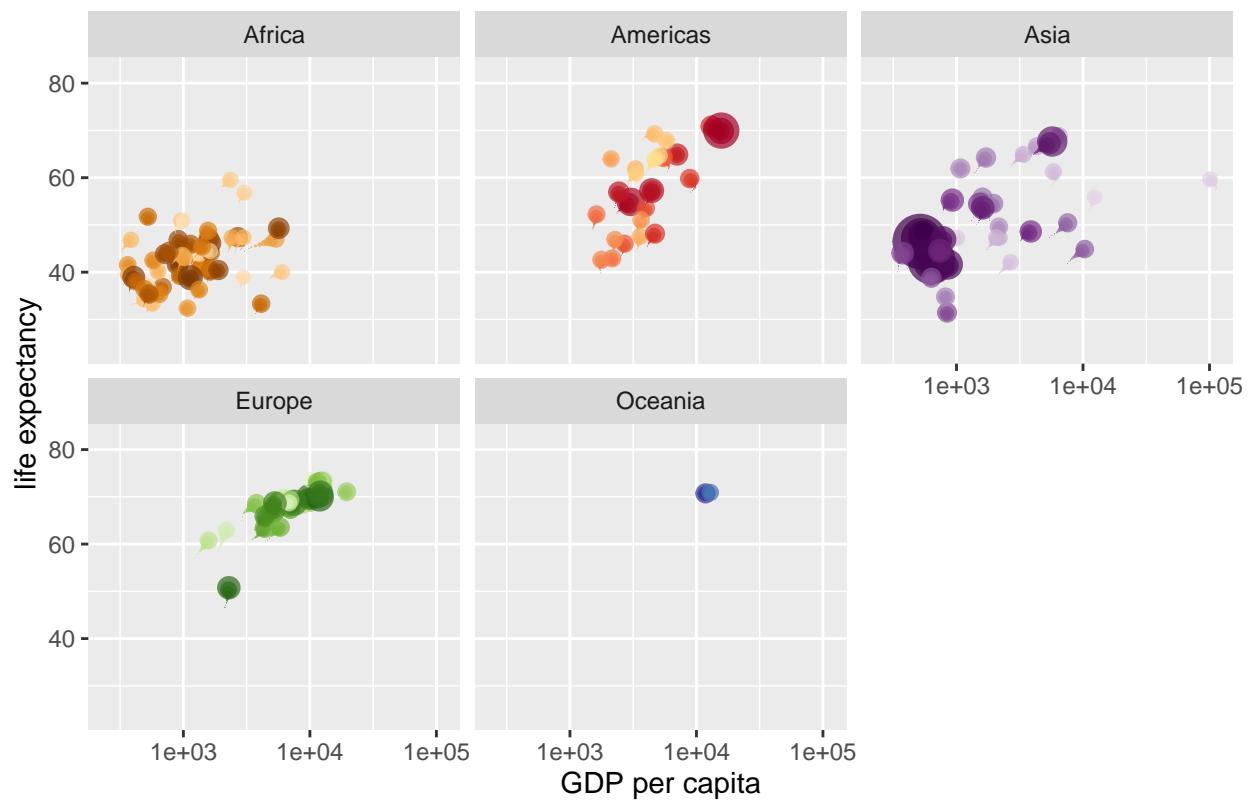
Year: 1959



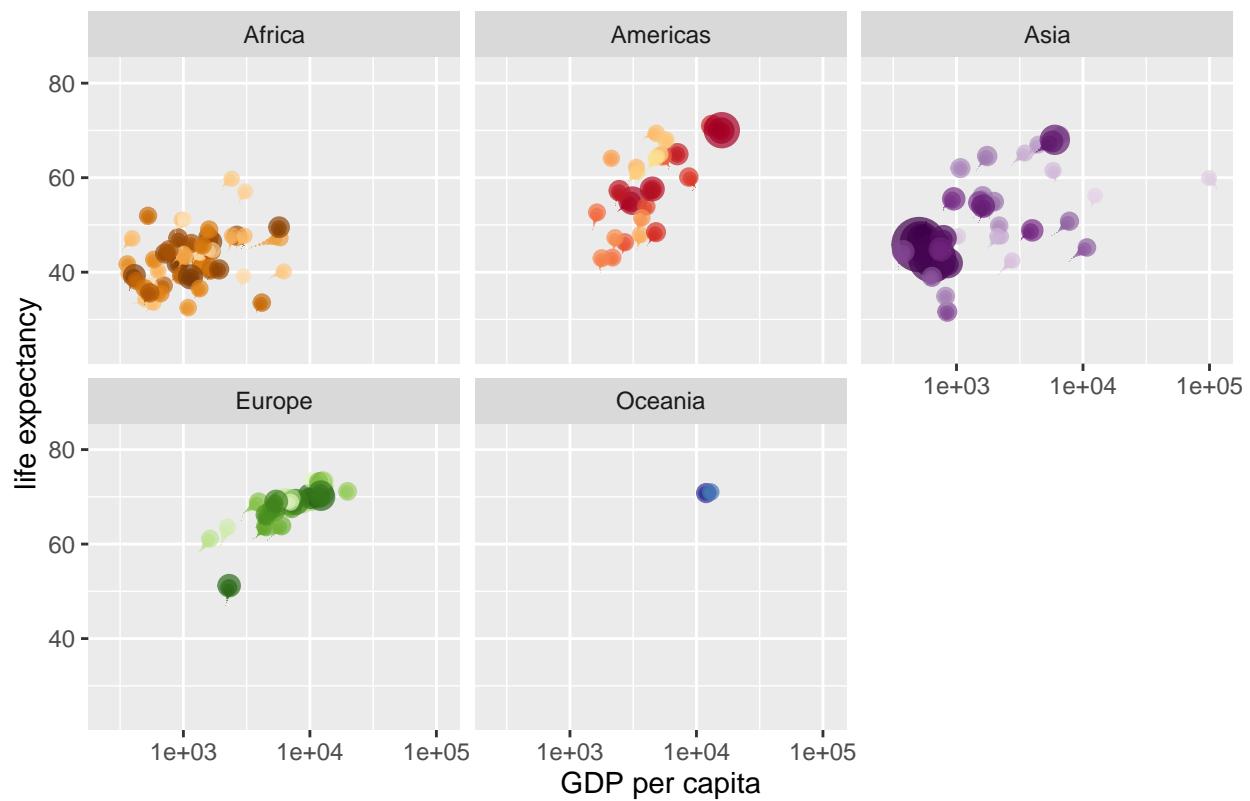
Year: 1960



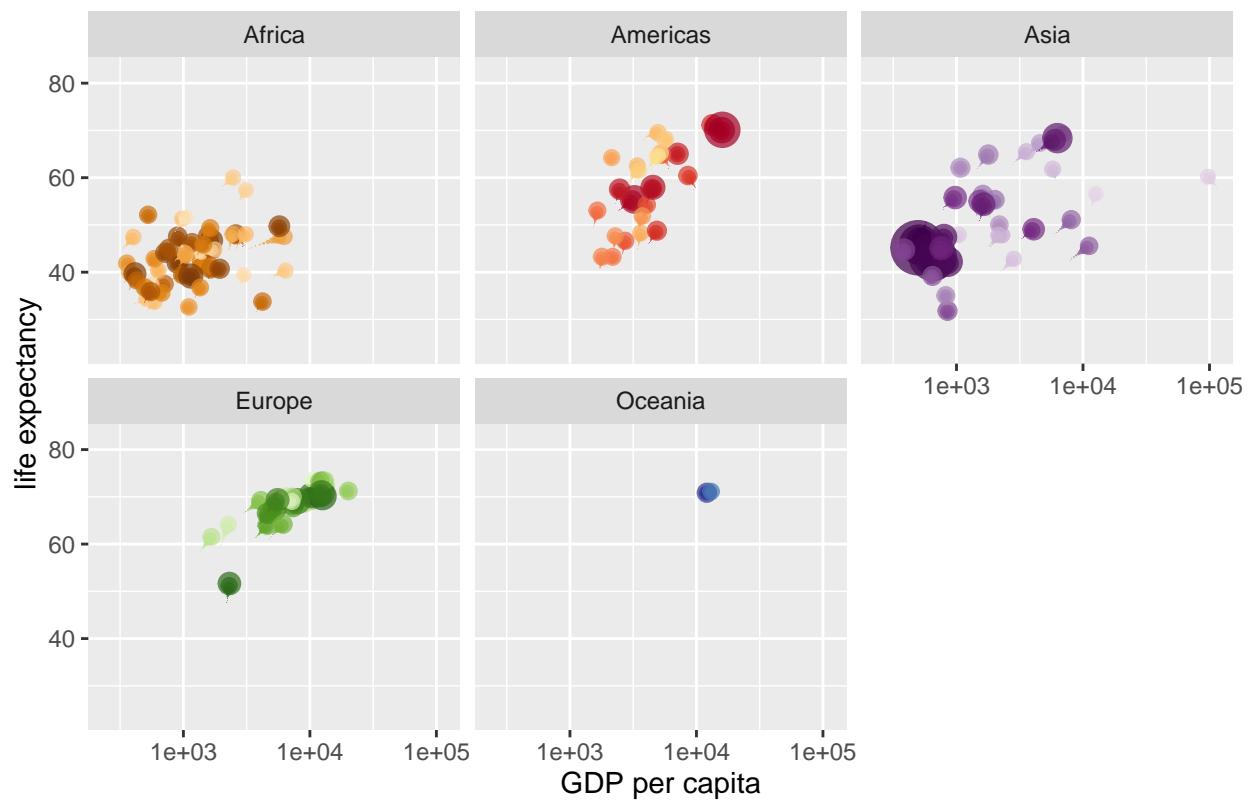
Year: 1960



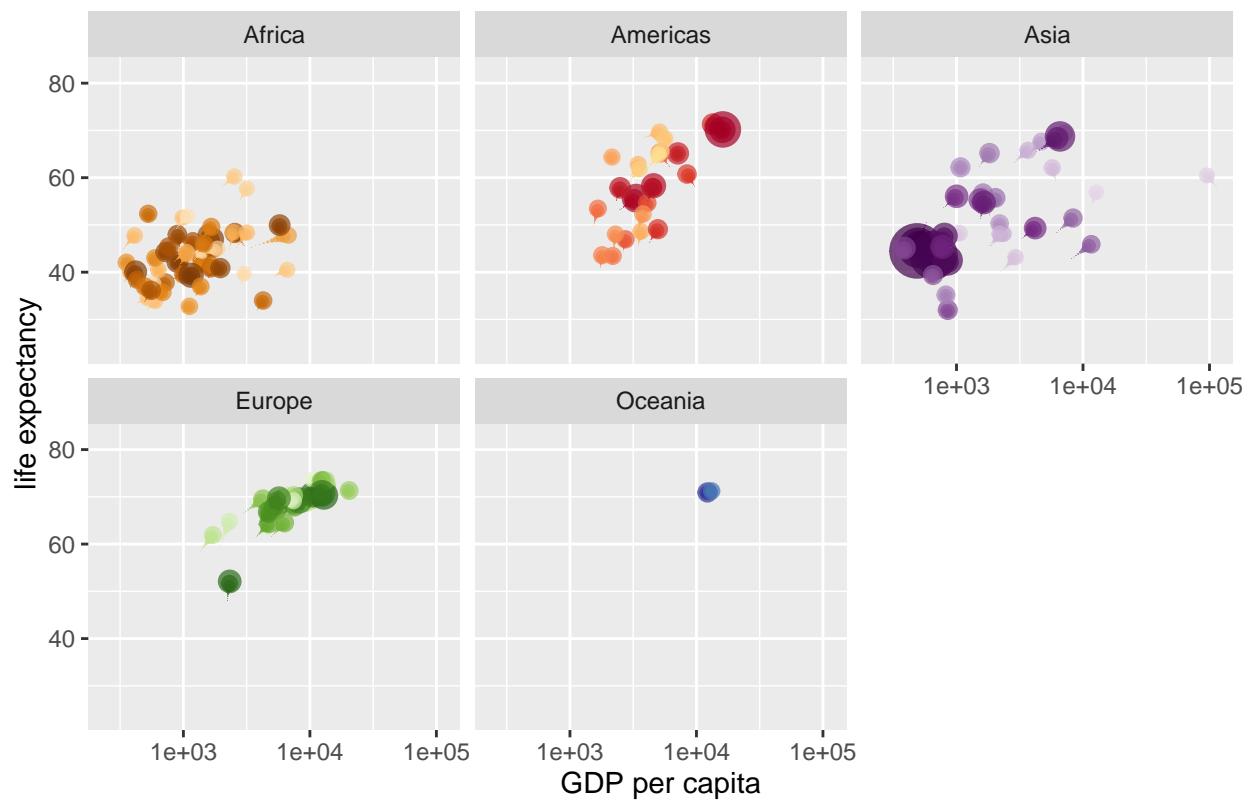
Year: 1961



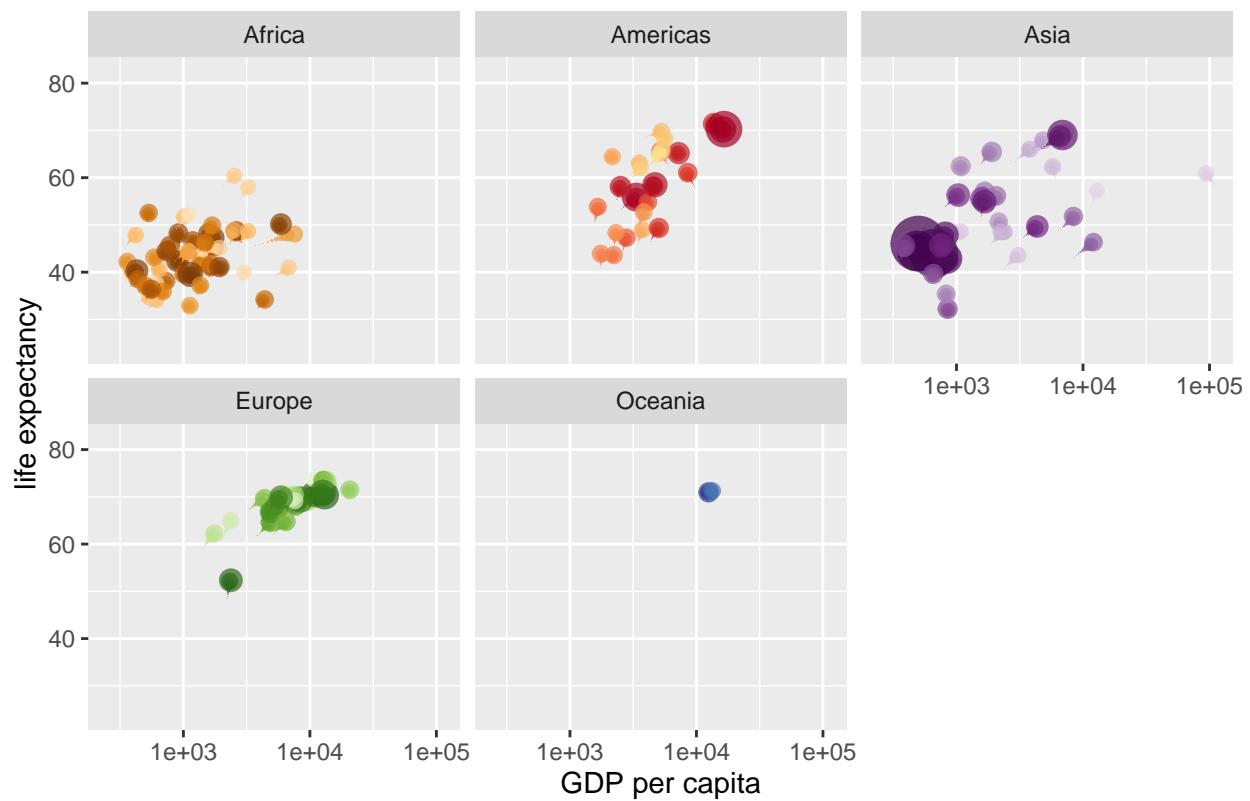
Year: 1961



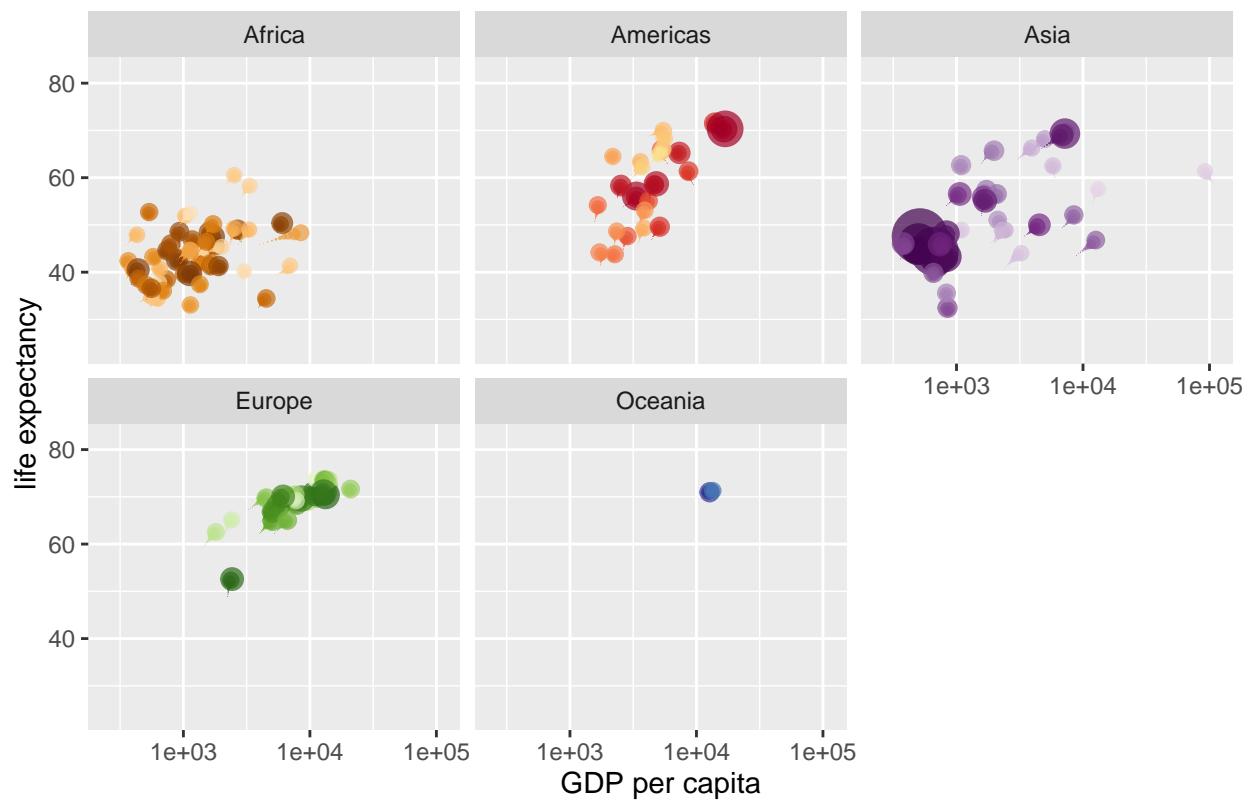
Year: 1962



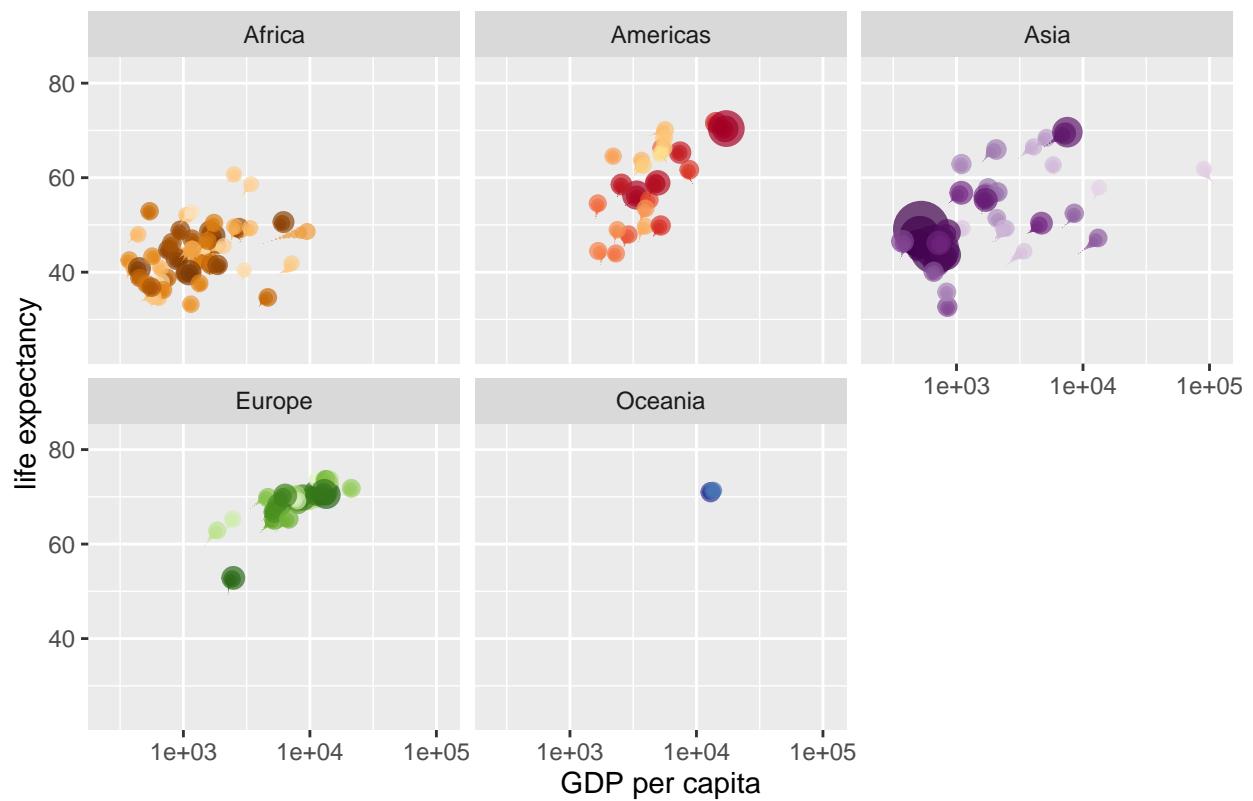
Year: 1963



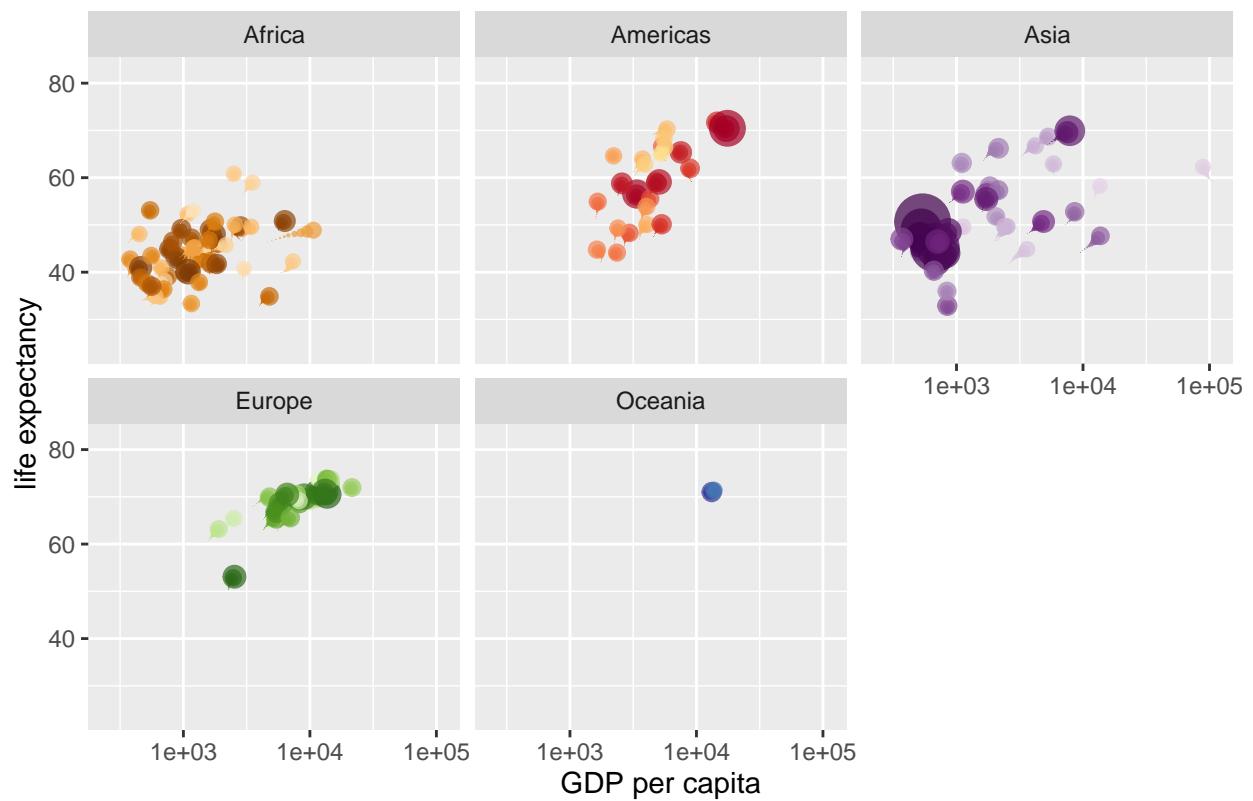
Year: 1963



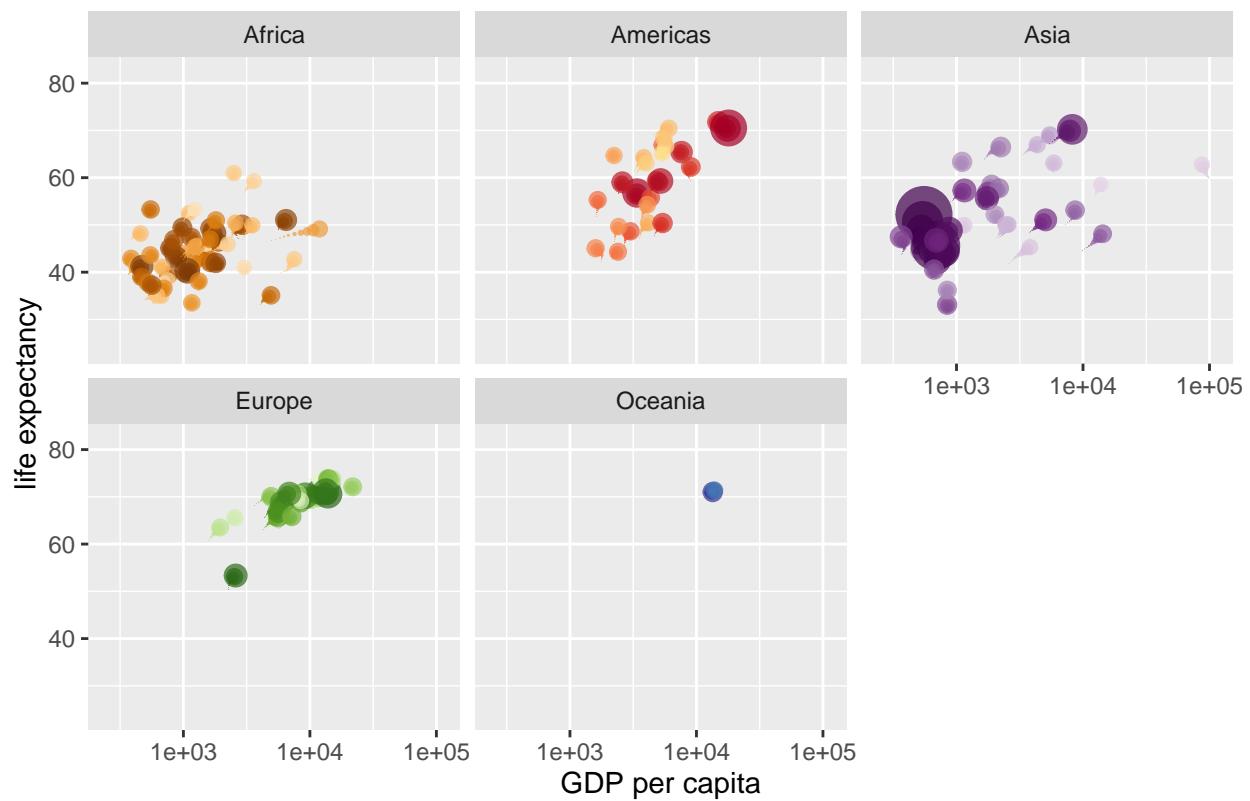
Year: 1964



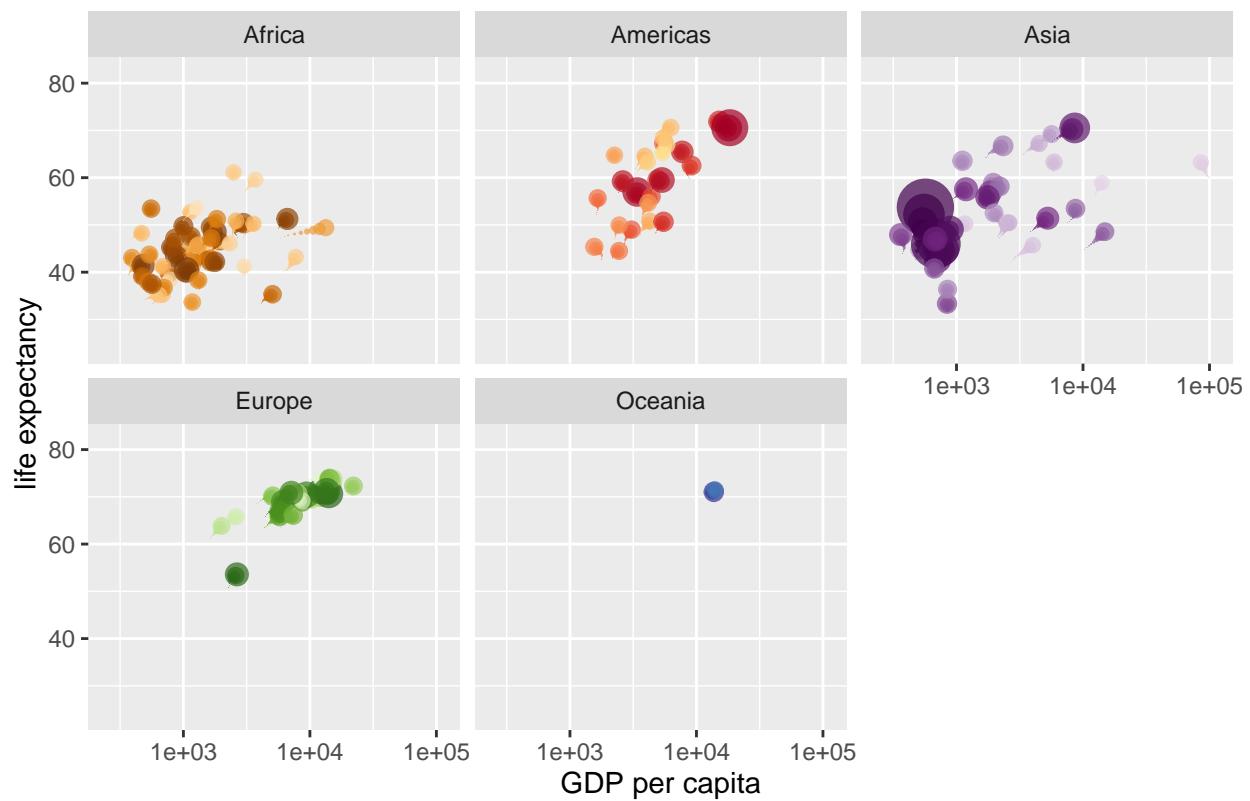
Year: 1964



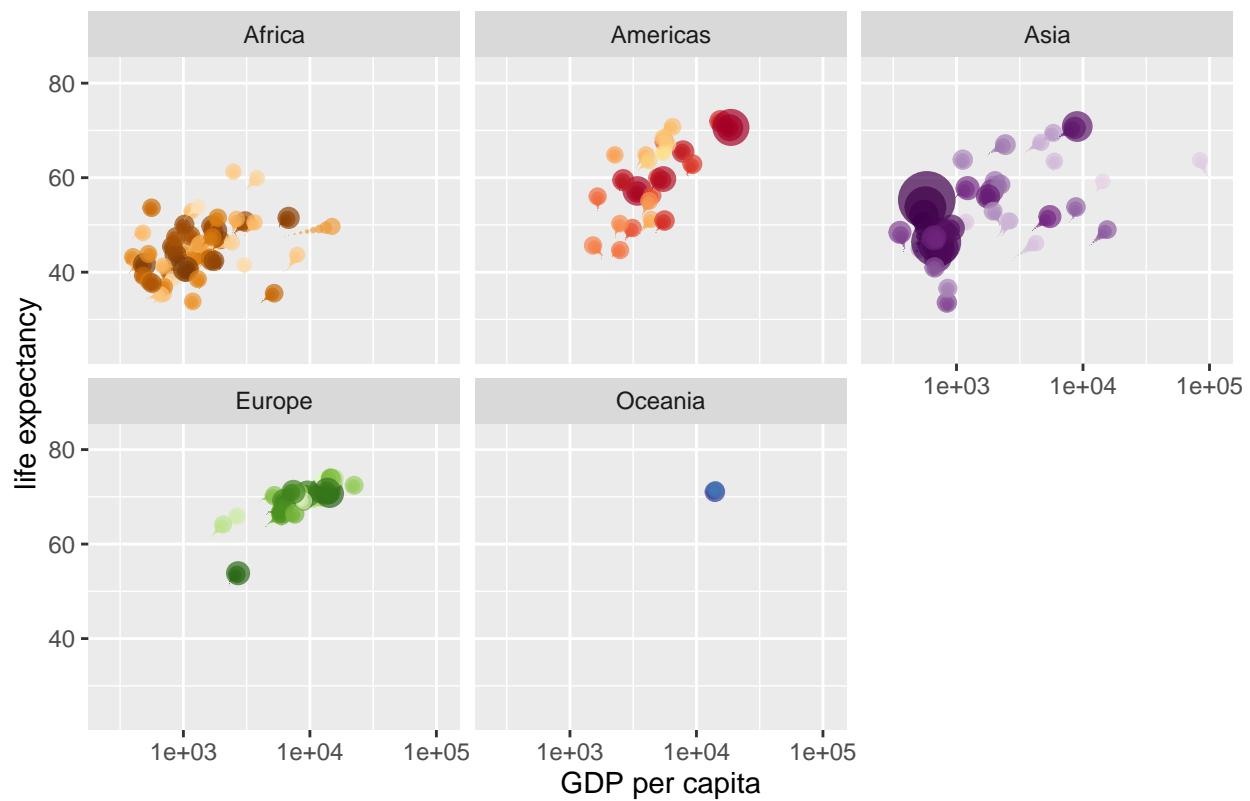
Year: 1965



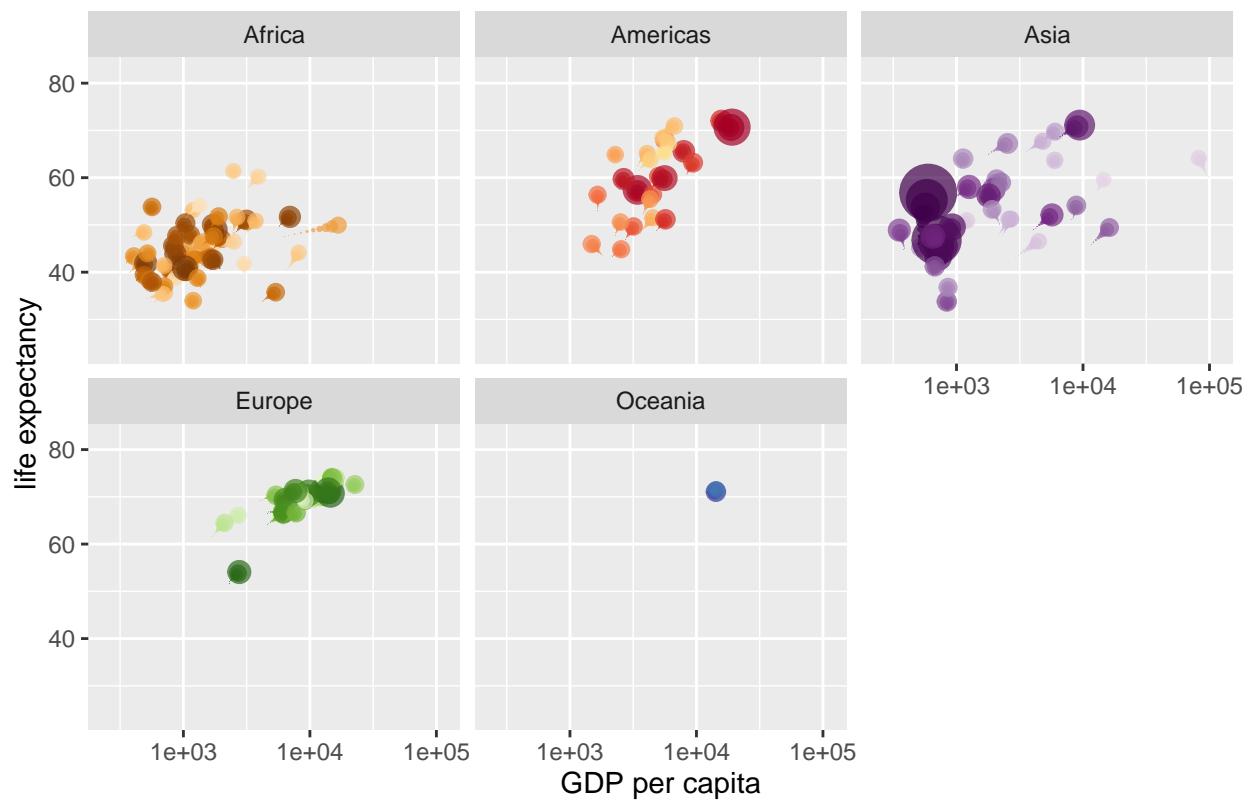
Year: 1965



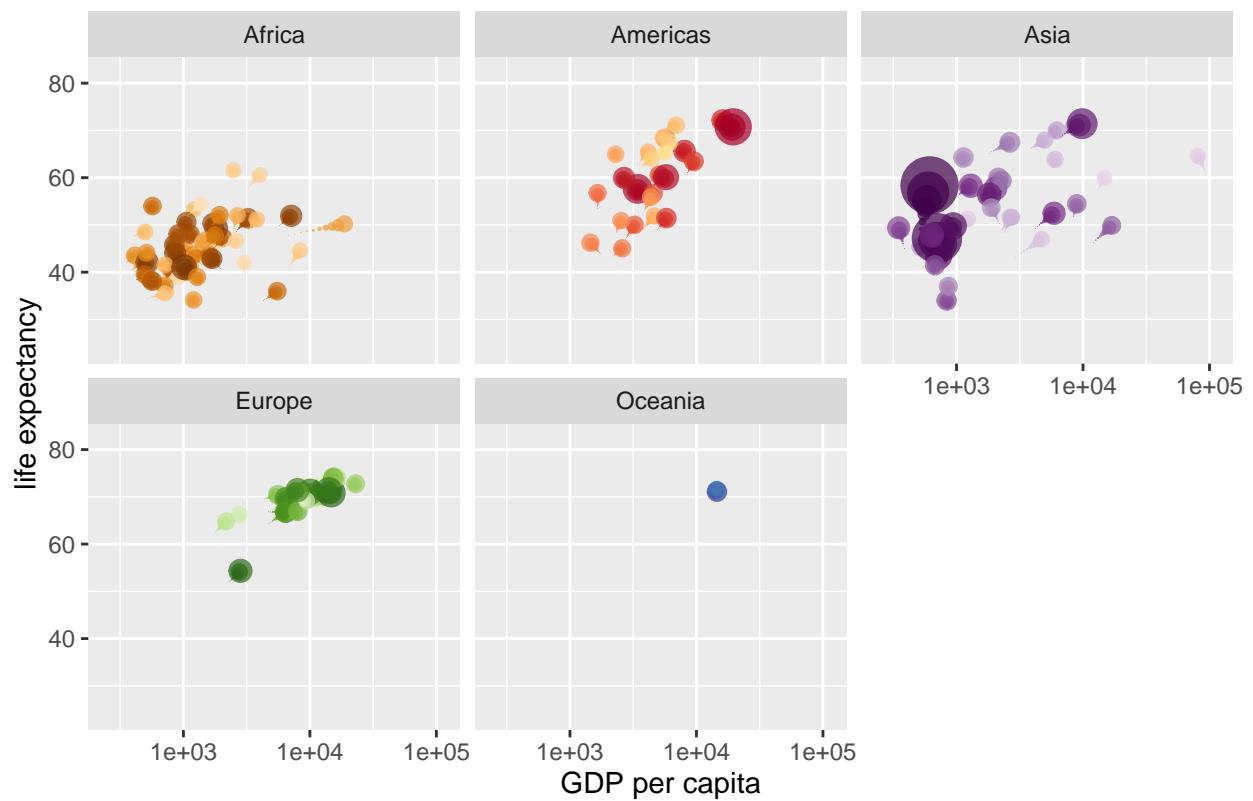
Year: 1966



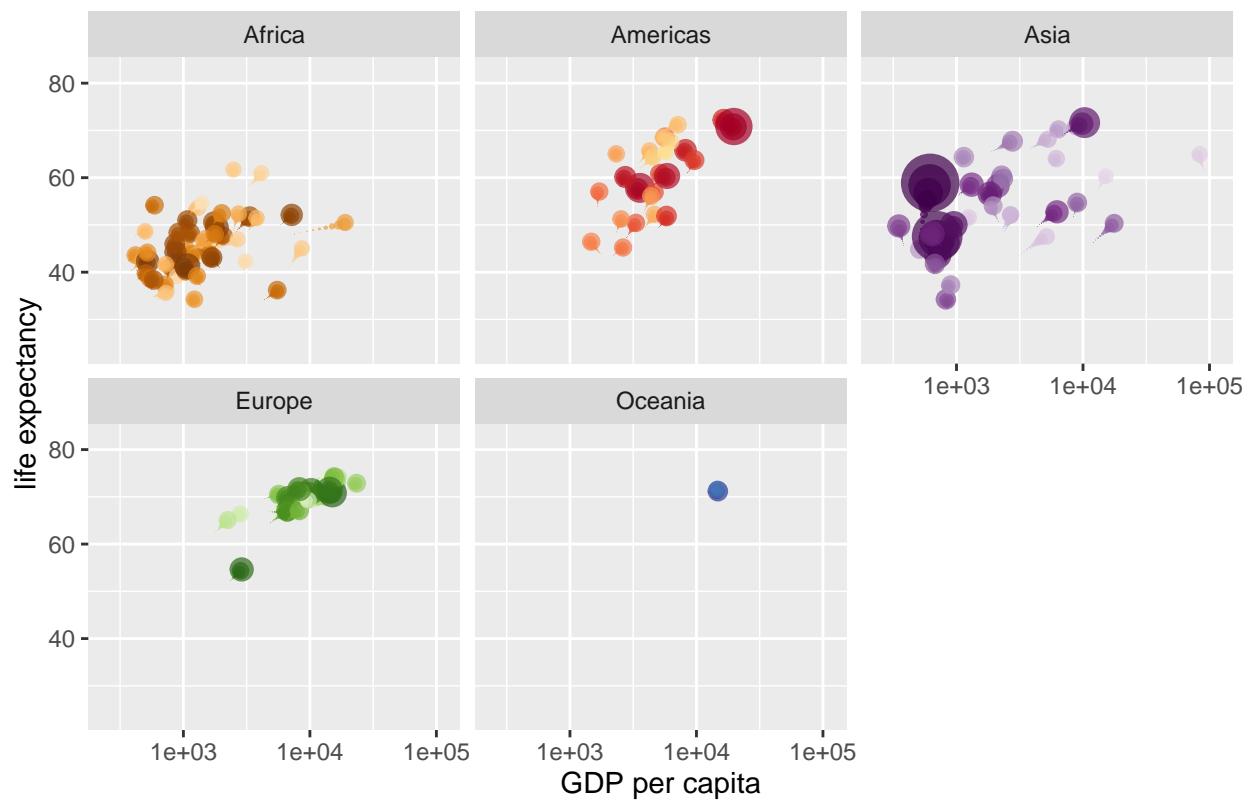
Year: 1966



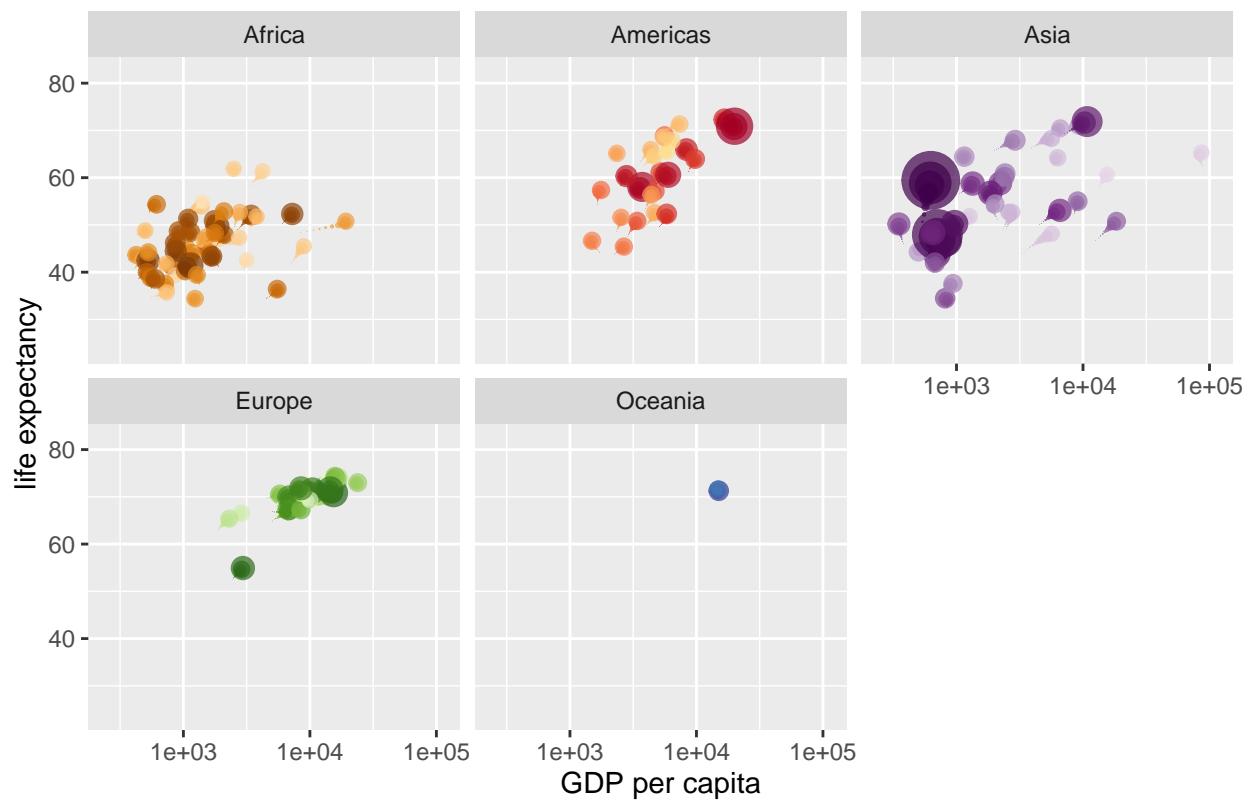
Year: 1967



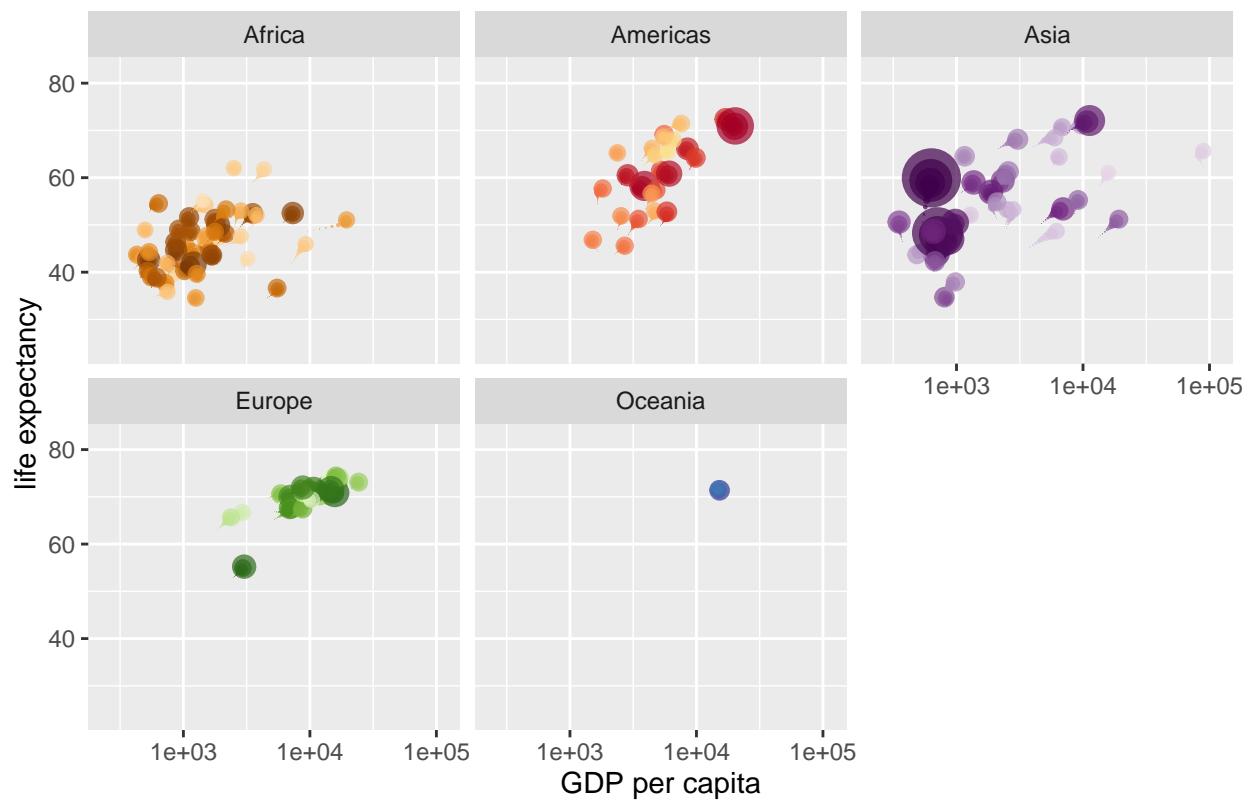
Year: 1968



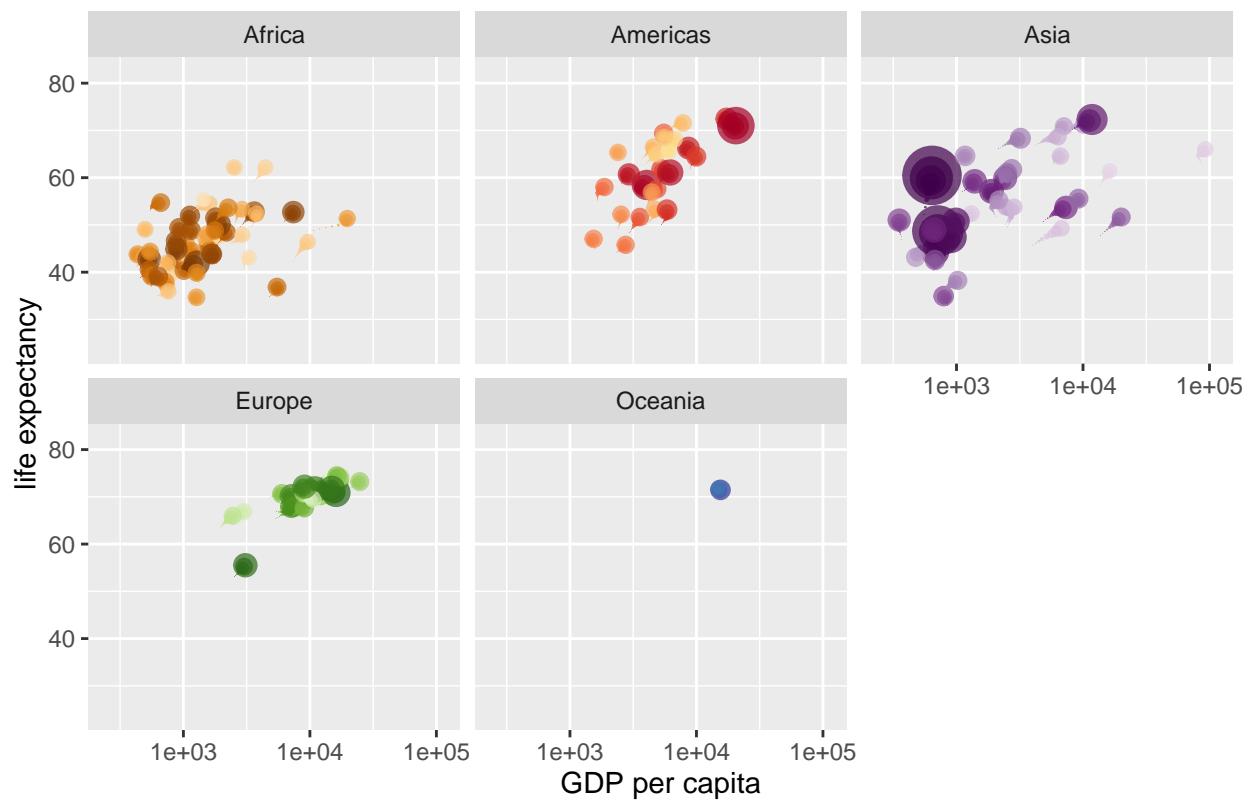
Year: 1968



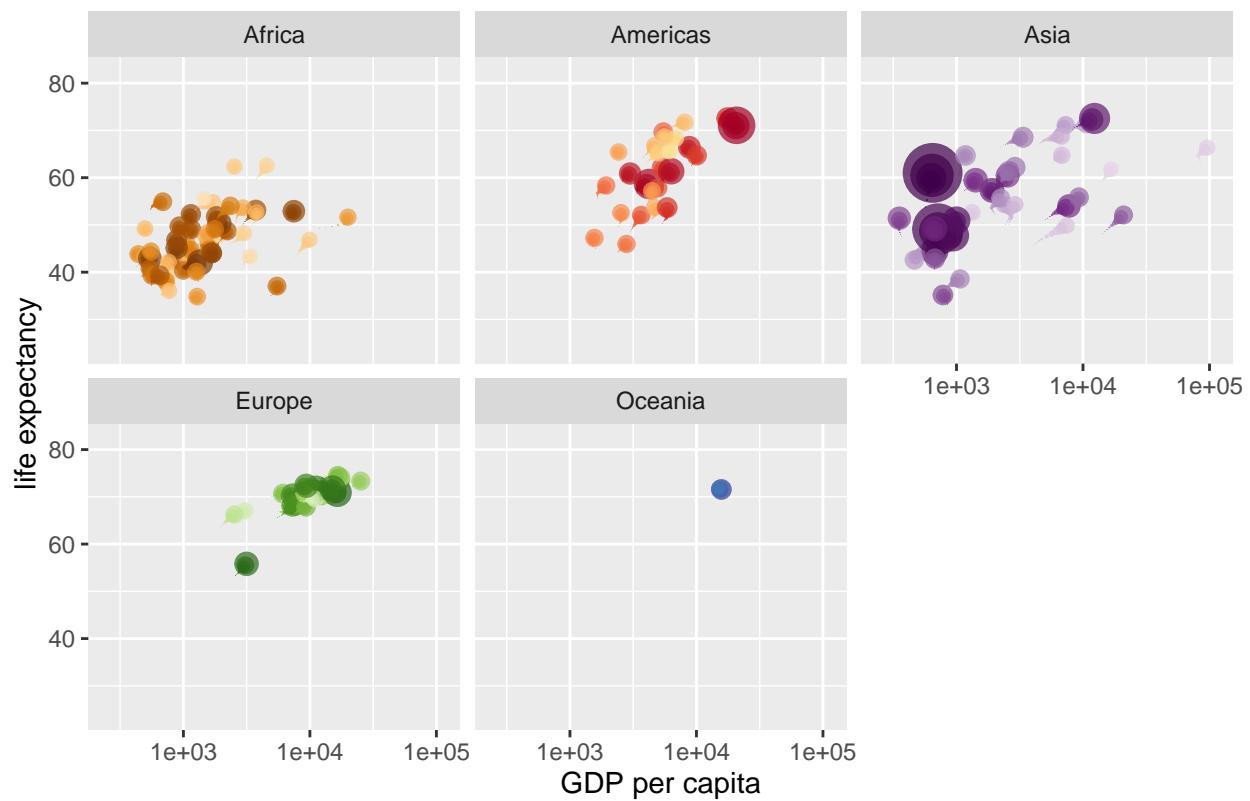
Year: 1969



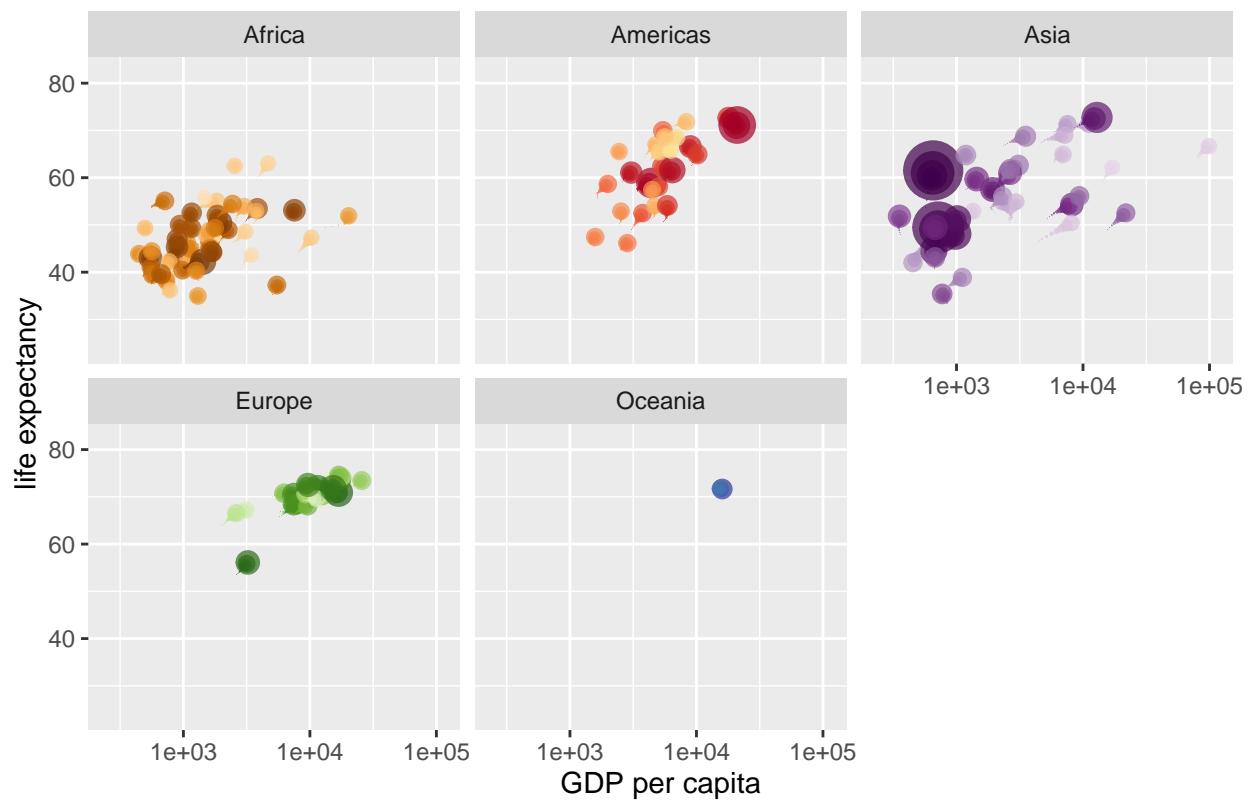
Year: 1969



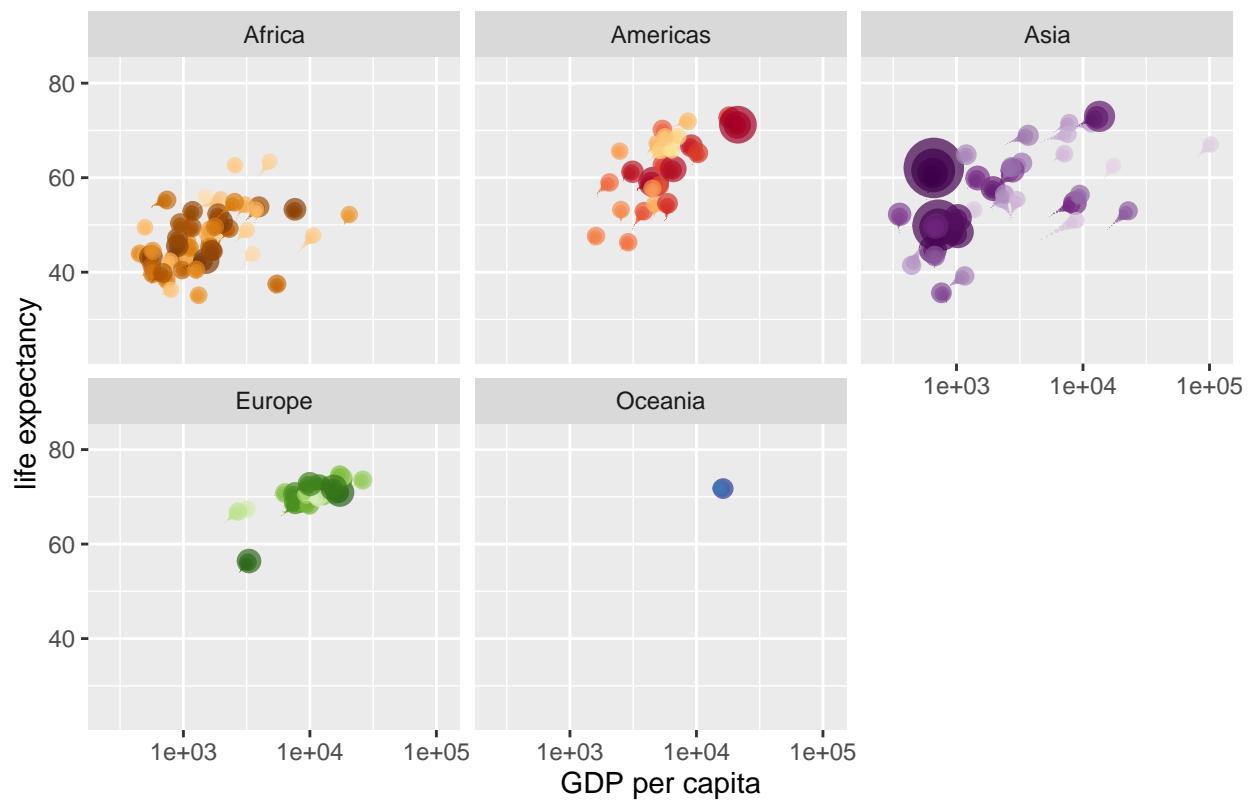
Year: 1970



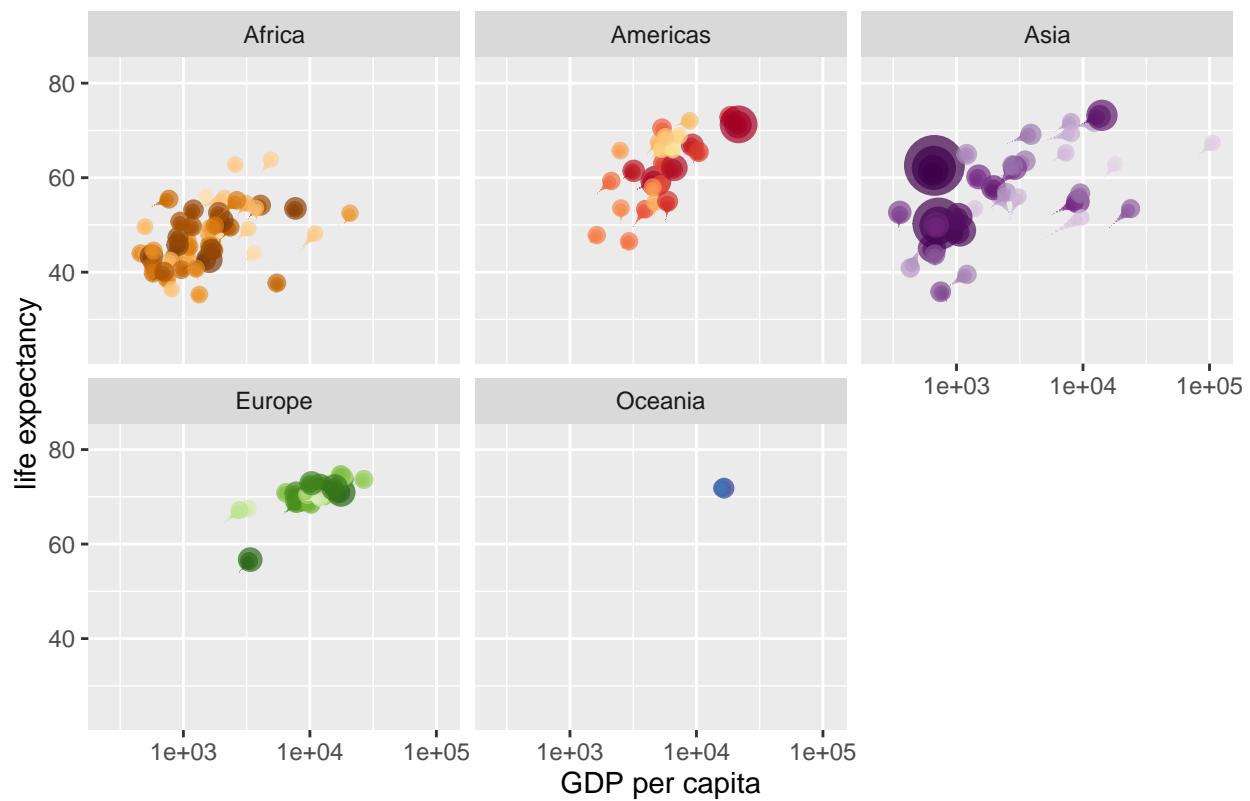
Year: 1970



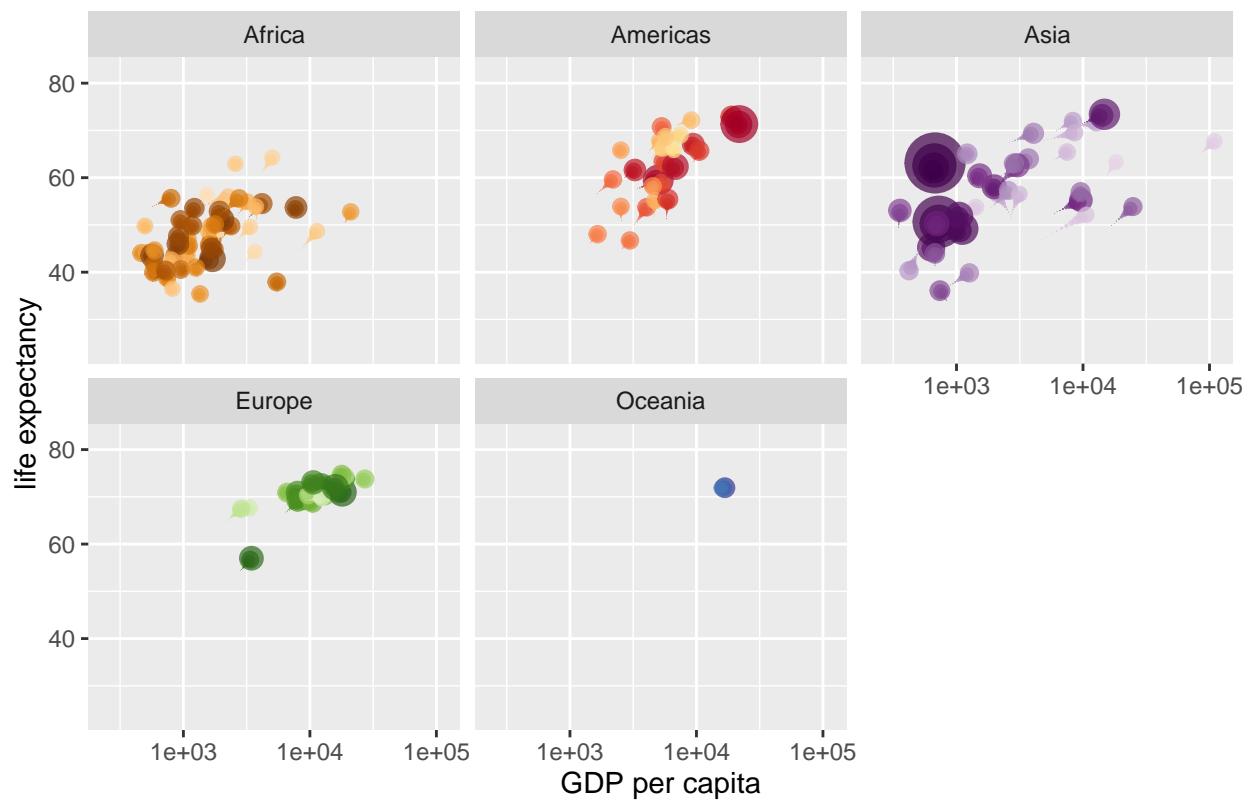
Year: 1971



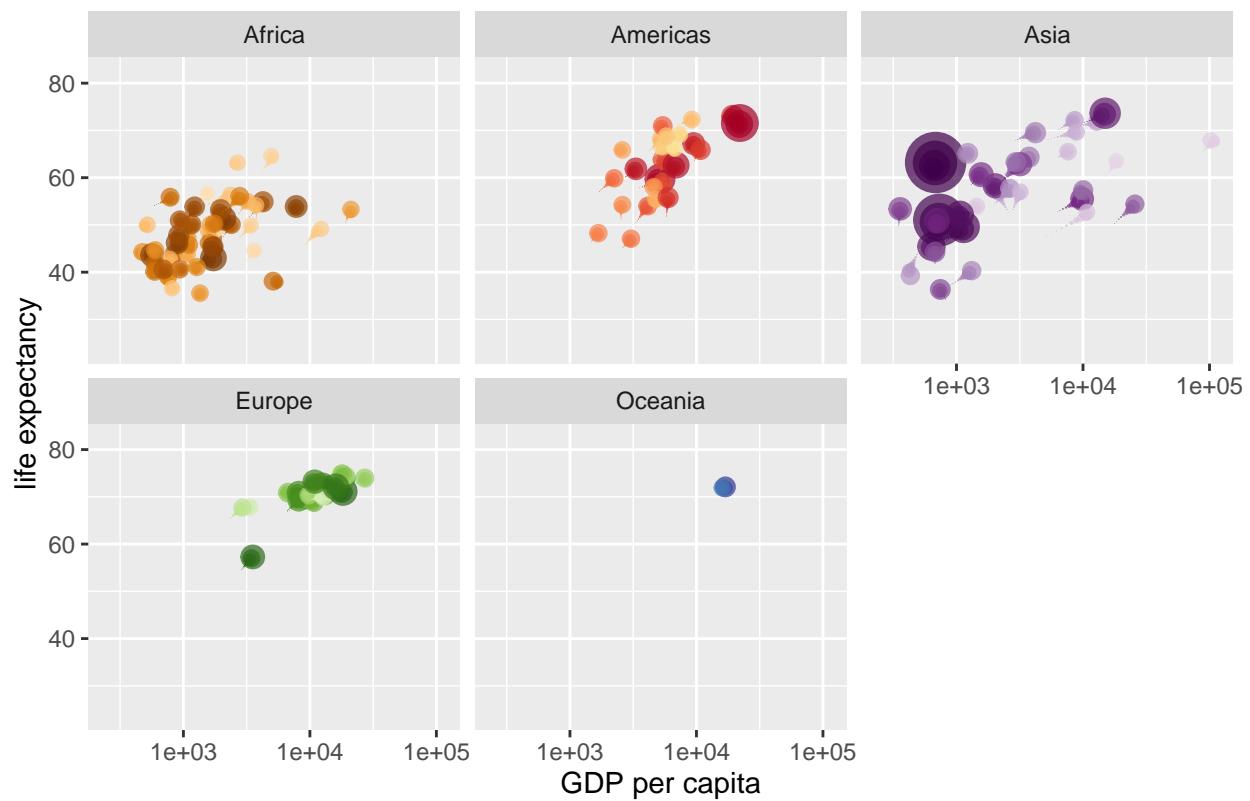
Year: 1971



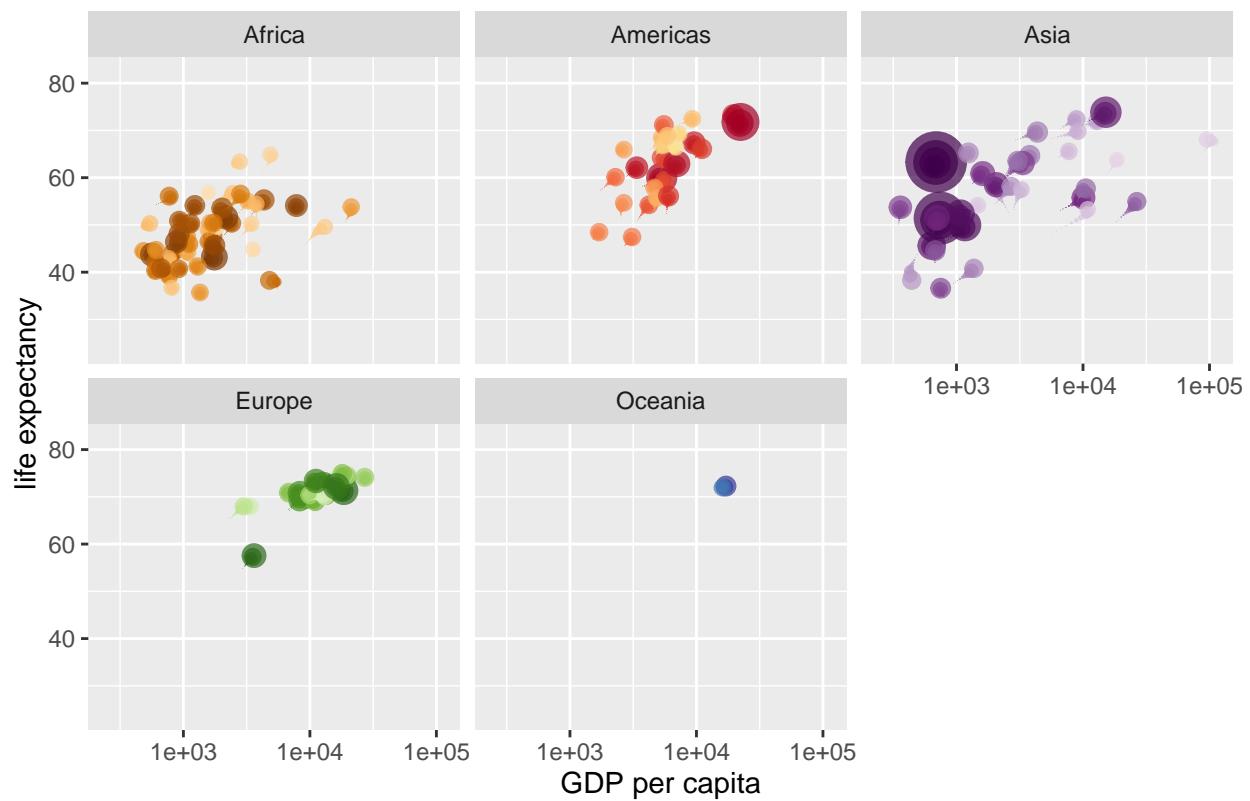
Year: 1972



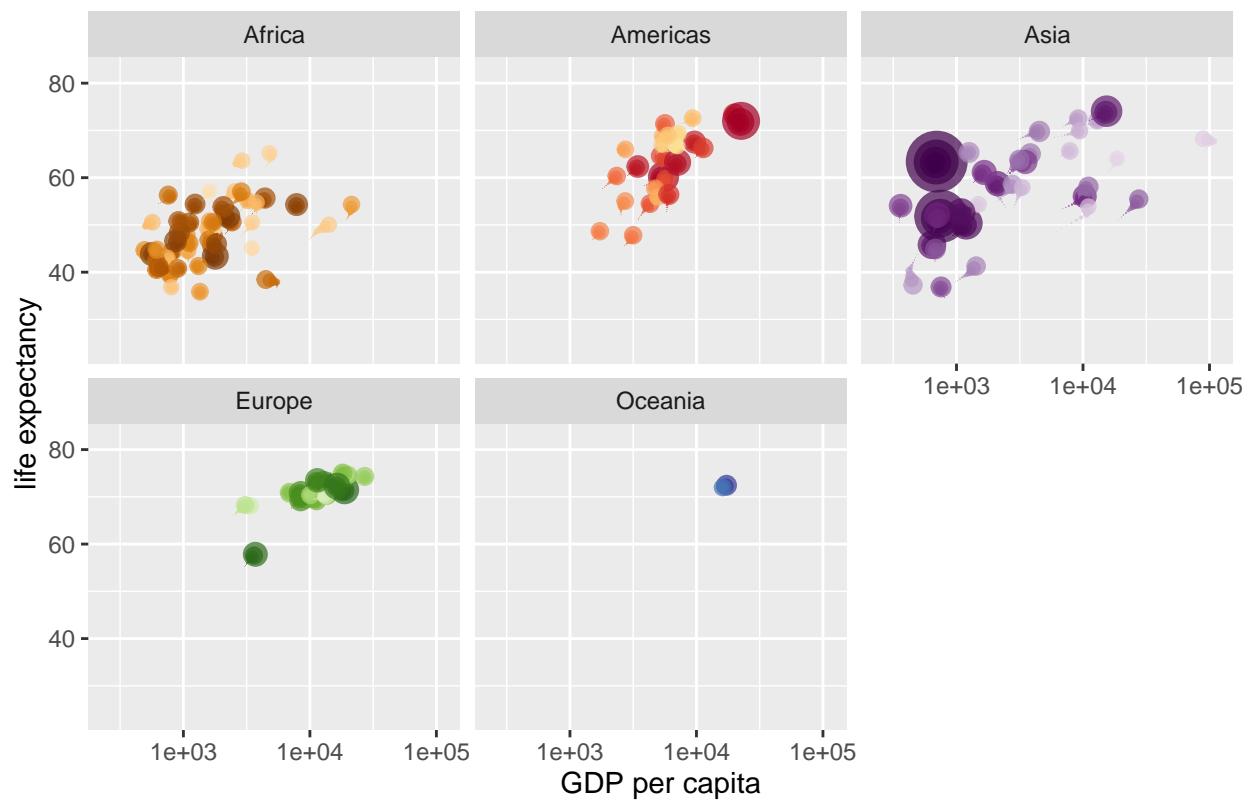
Year: 1973



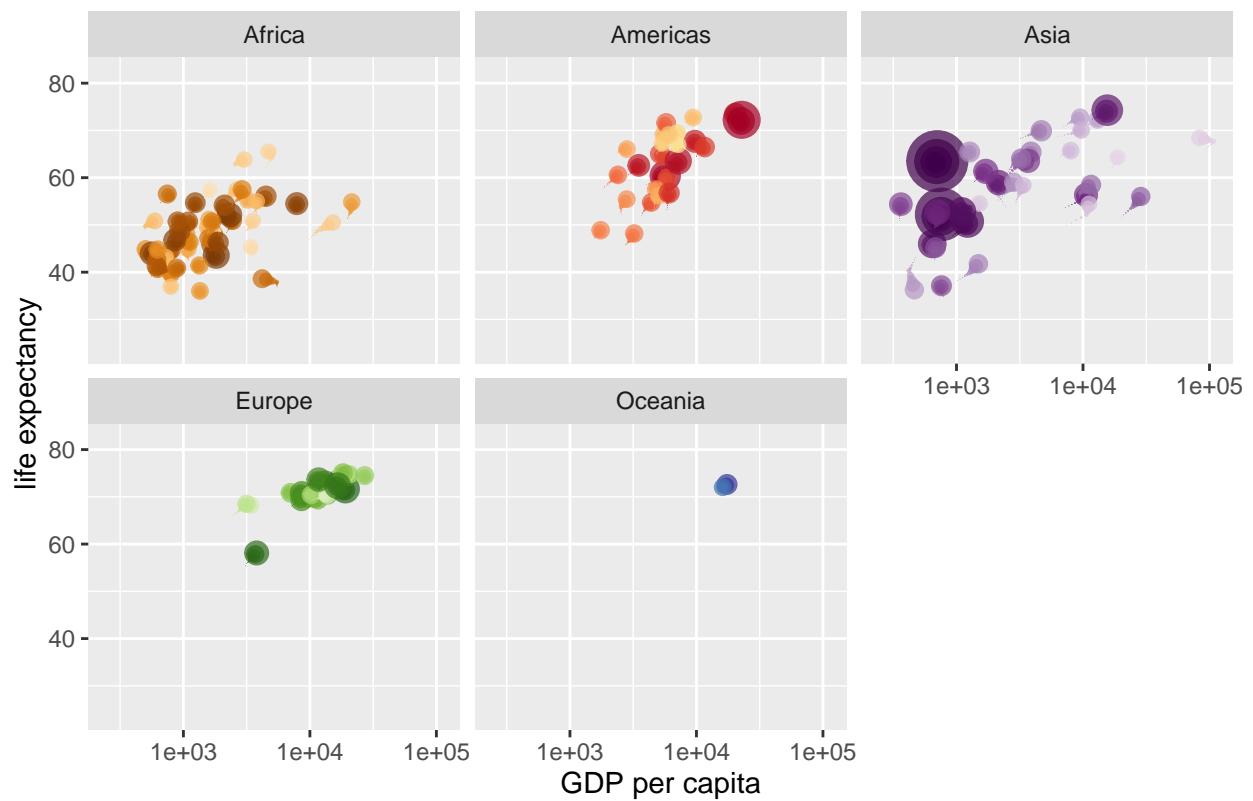
Year: 1973



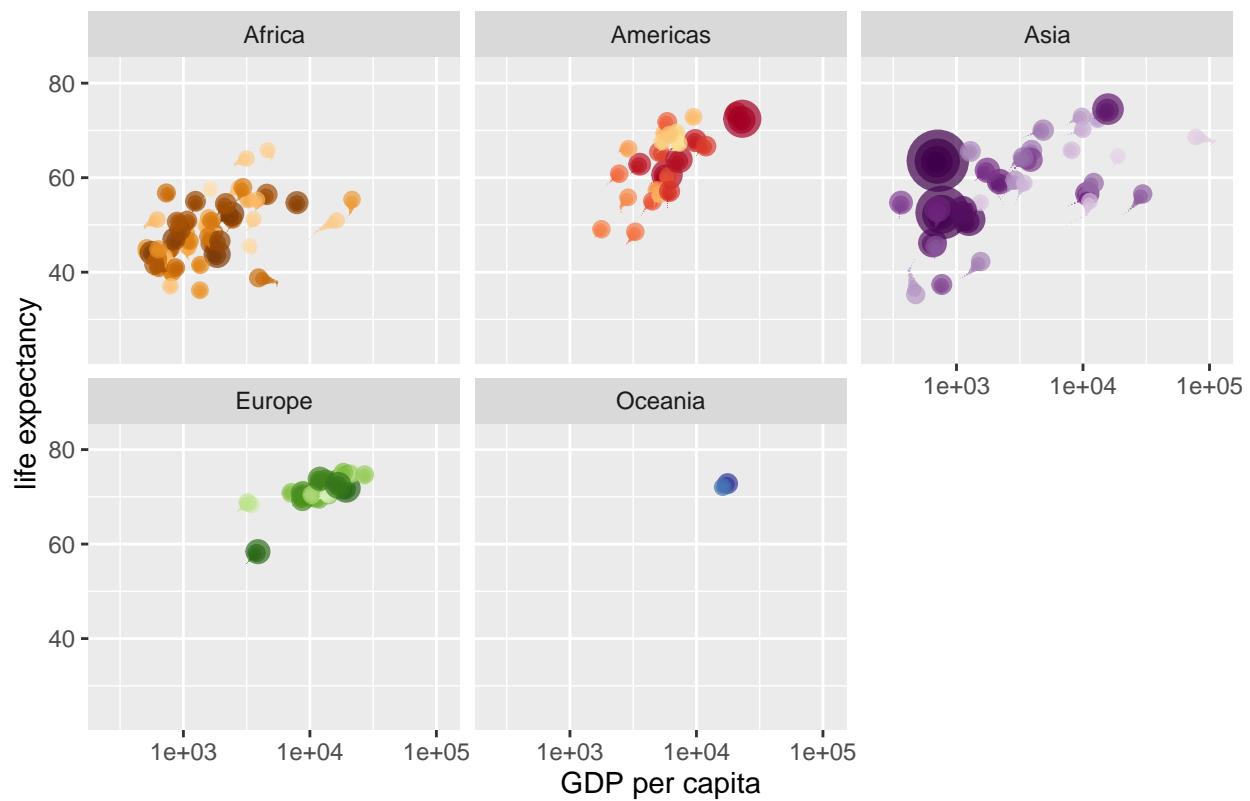
Year: 1974



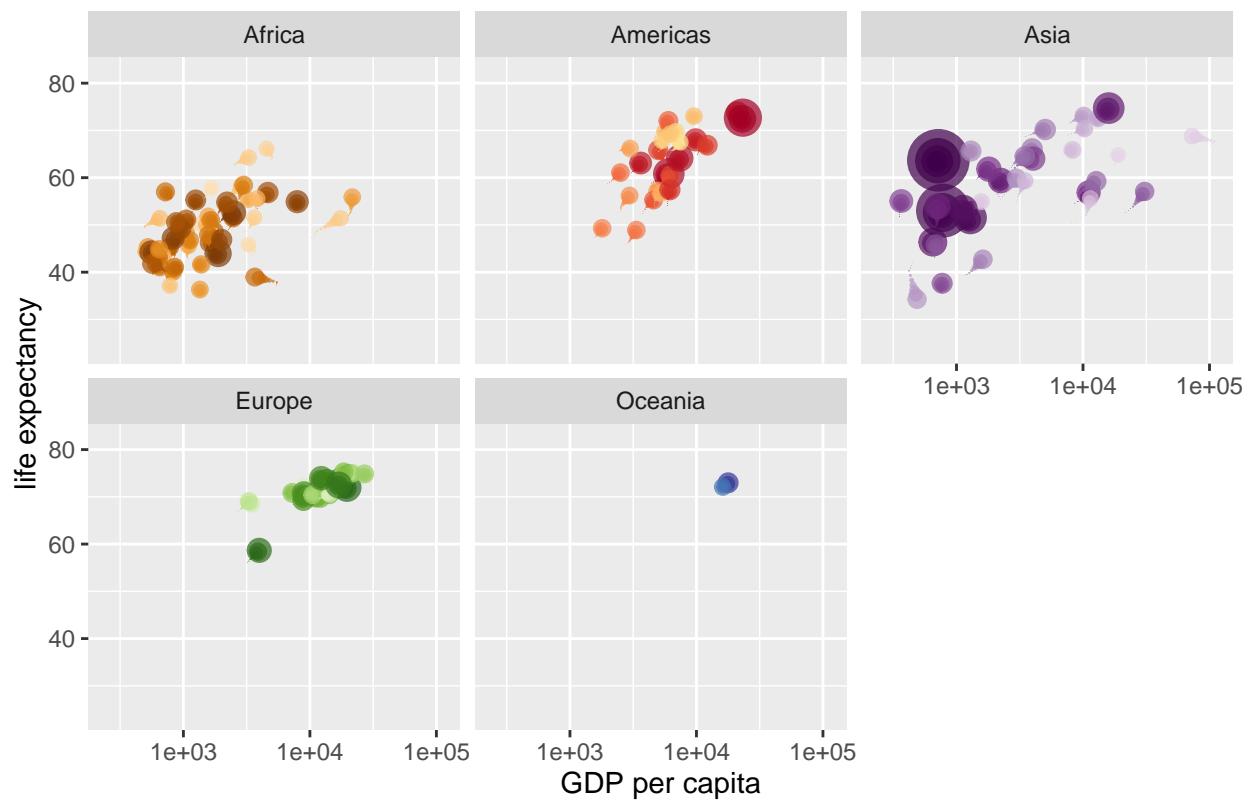
Year: 1974



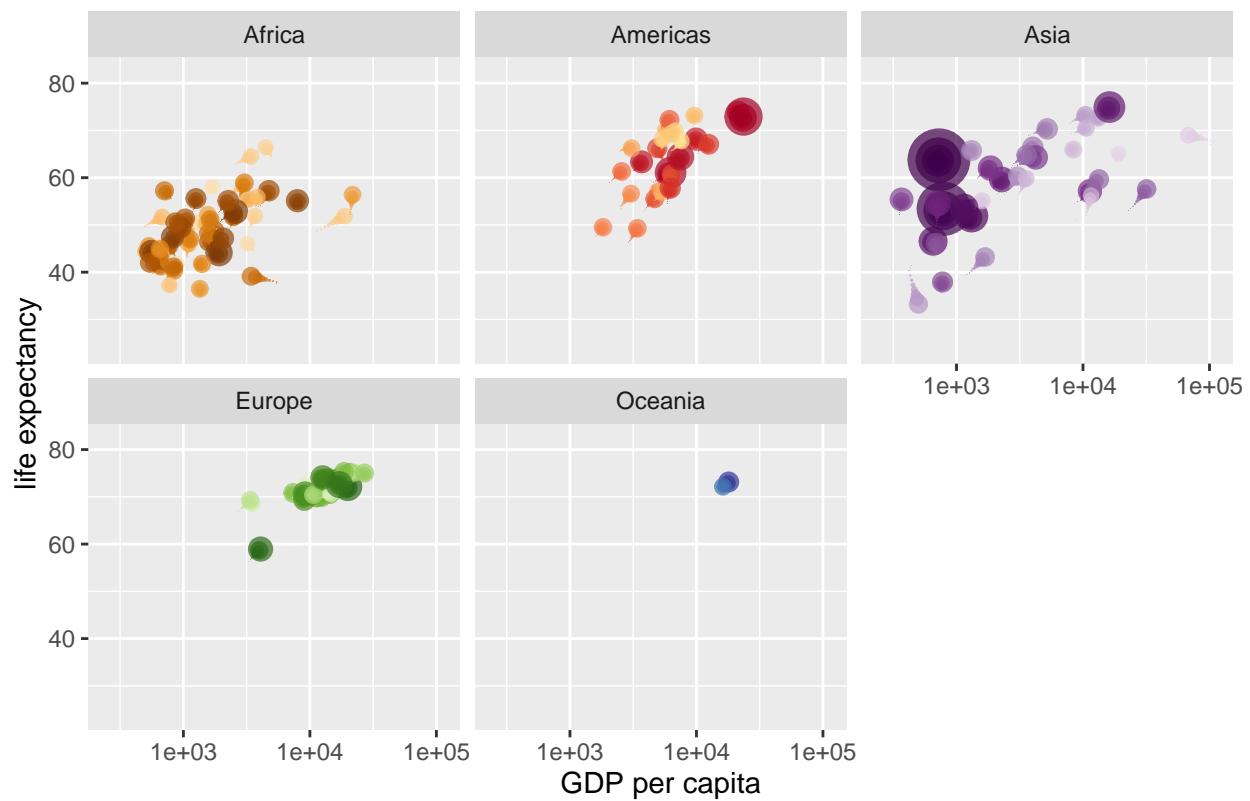
Year: 1975



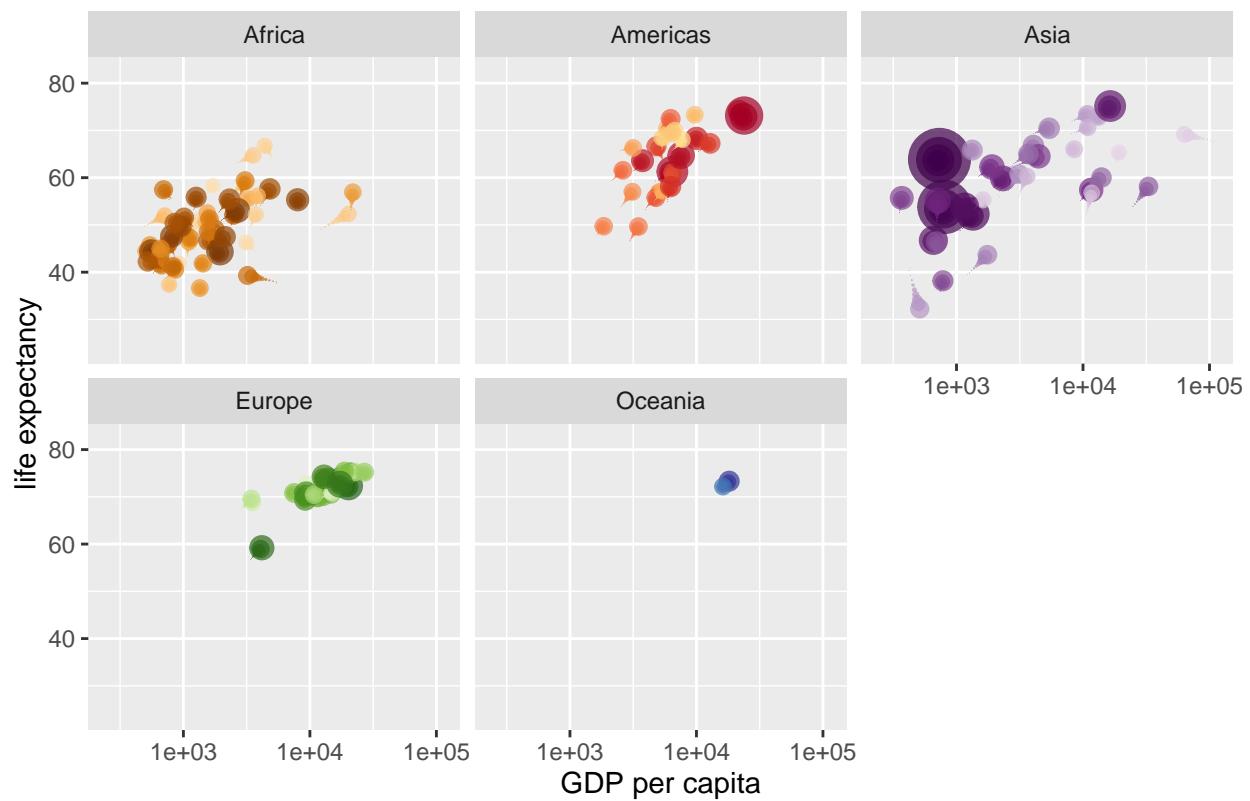
Year: 1975



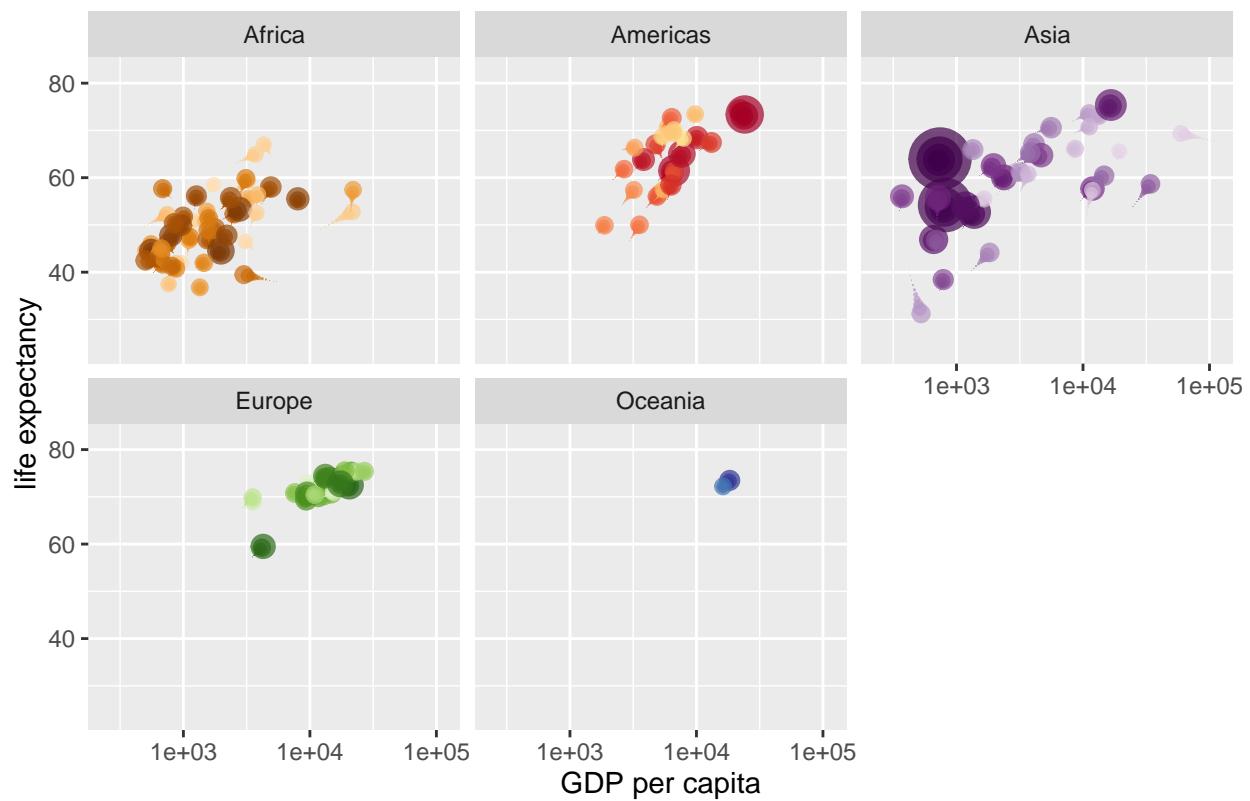
Year: 1976



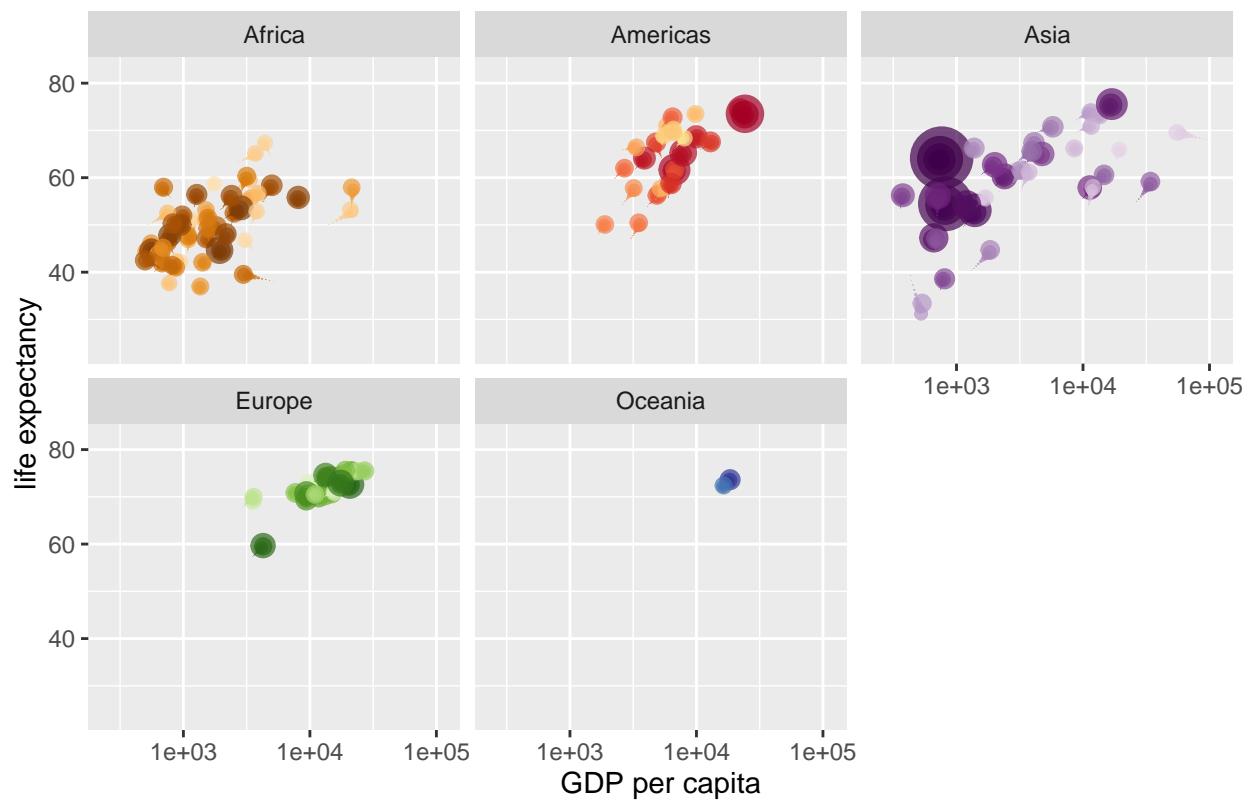
Year: 1976



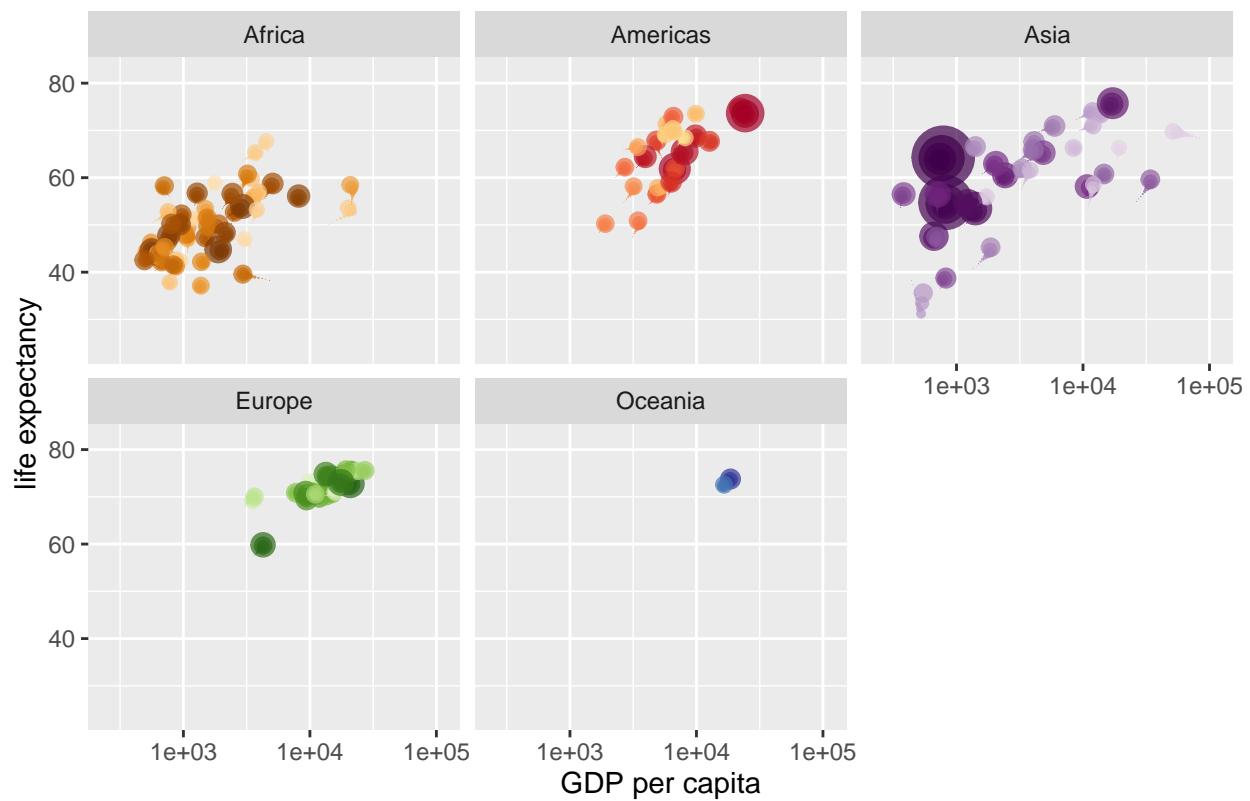
Year: 1977



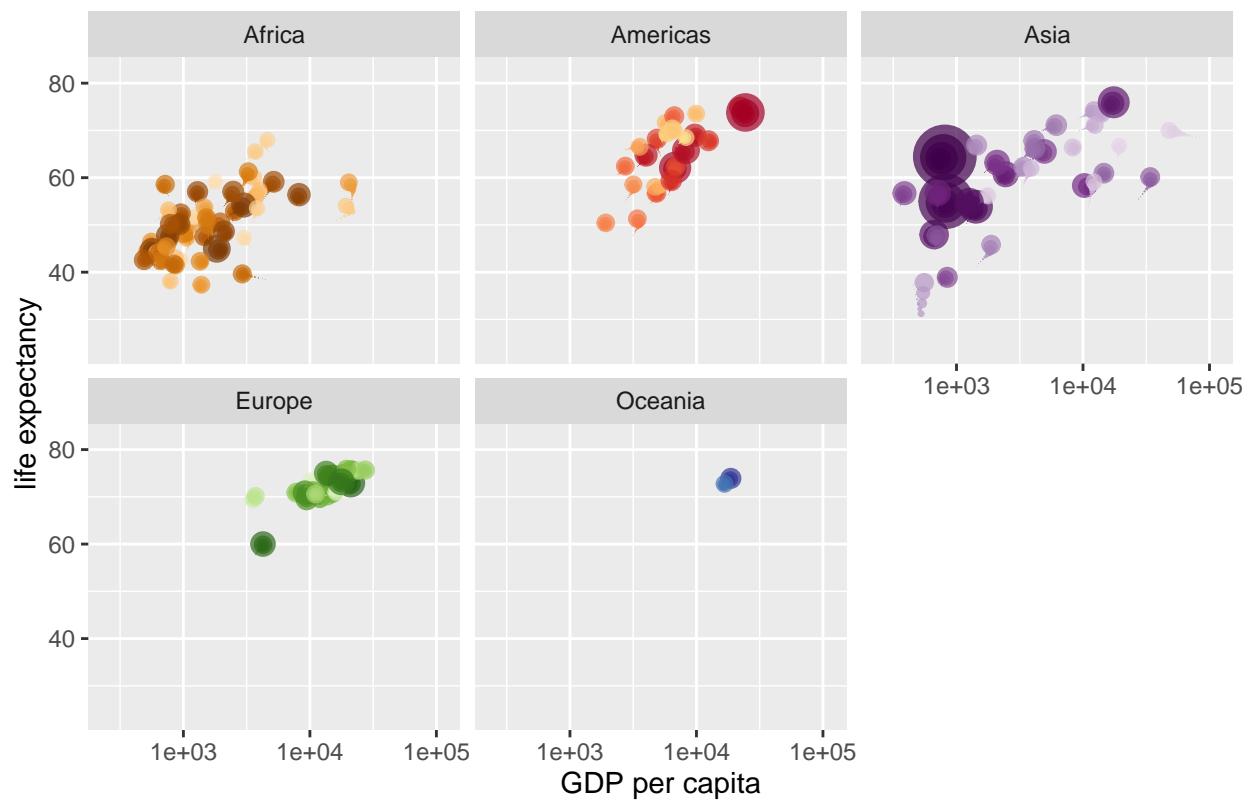
Year: 1978



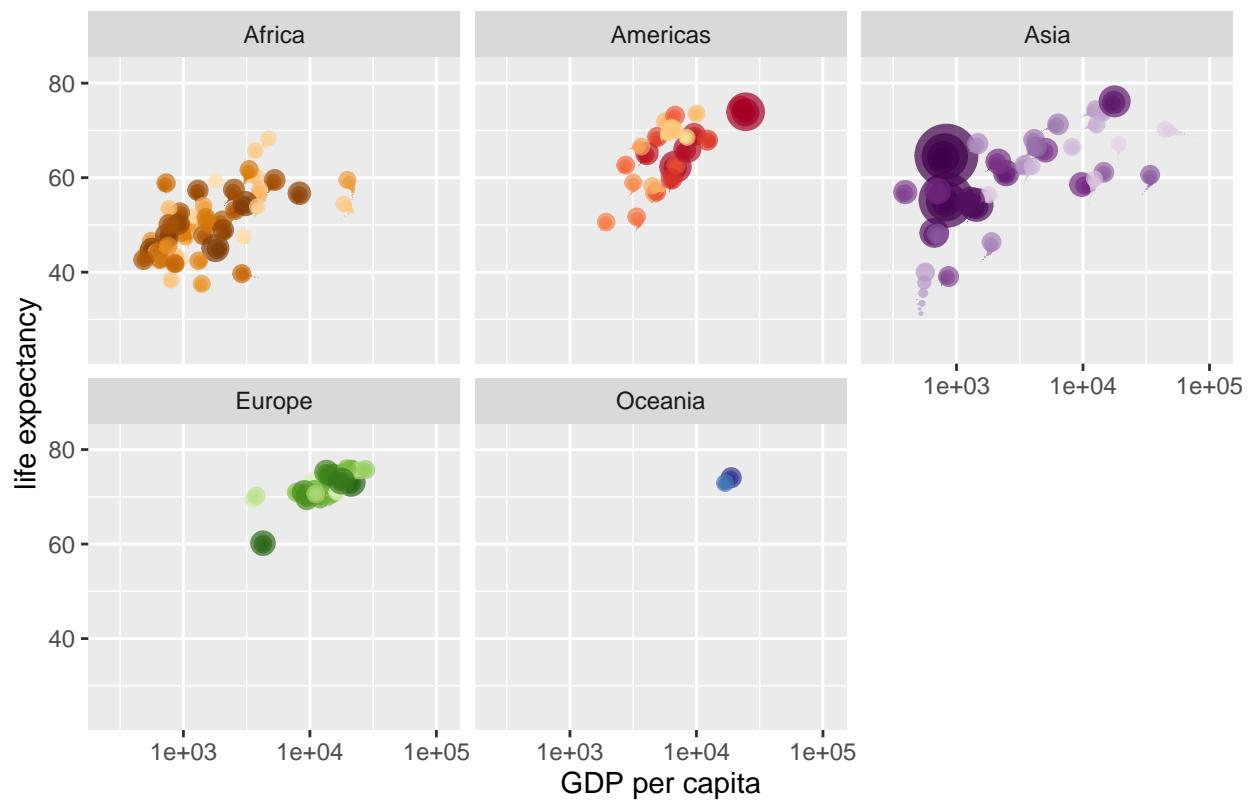
Year: 1978



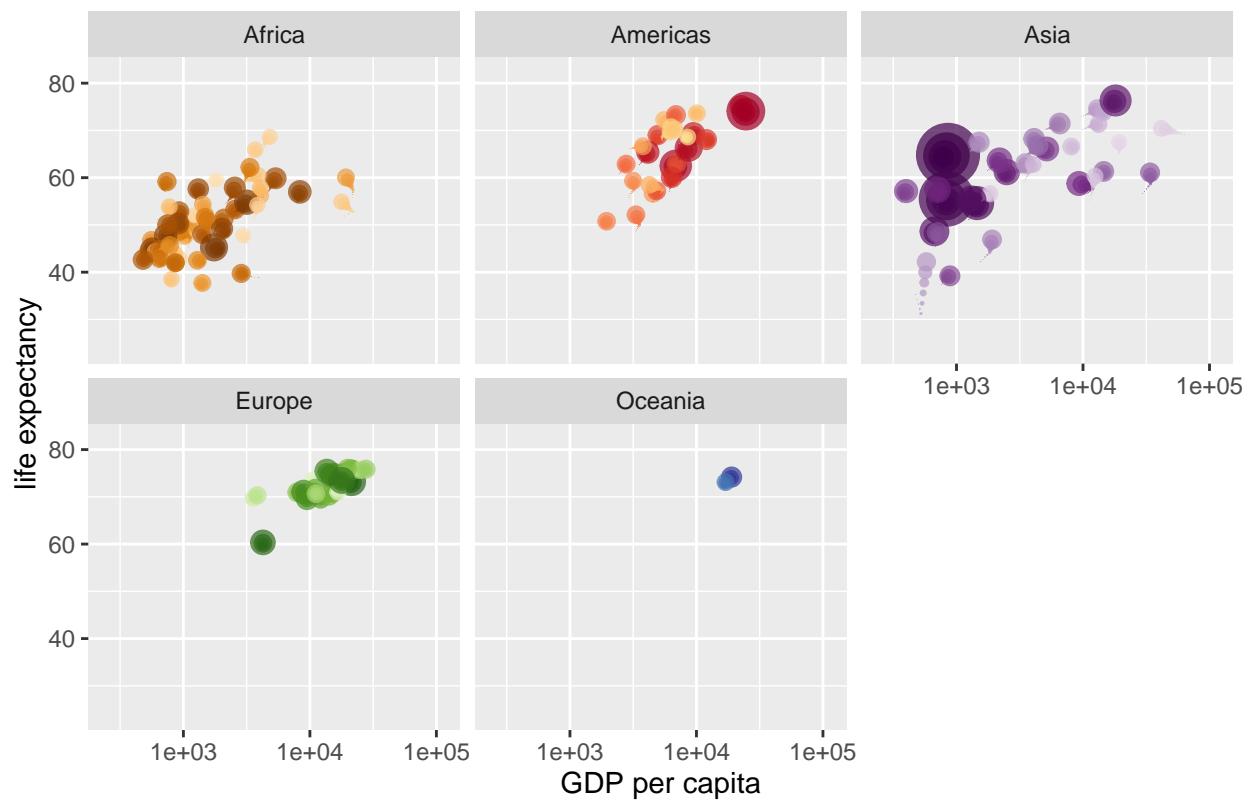
Year: 1979



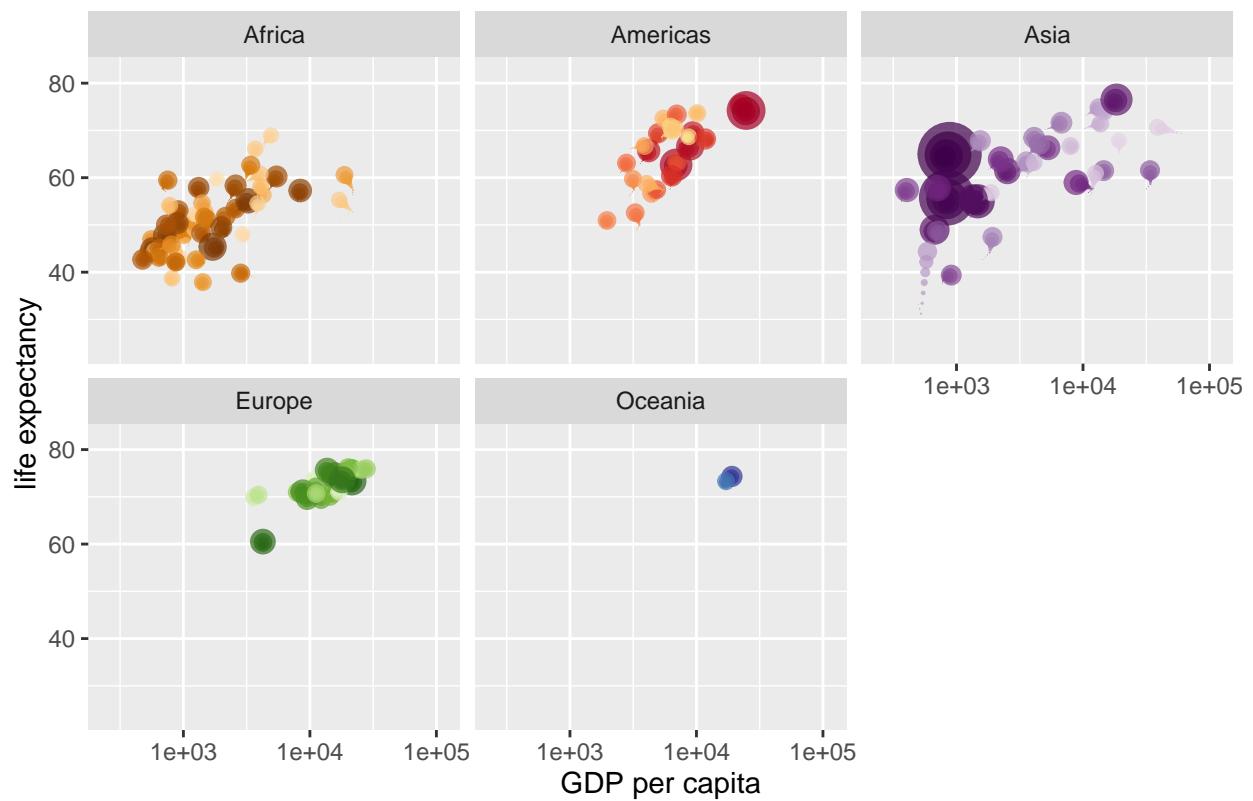
Year: 1979



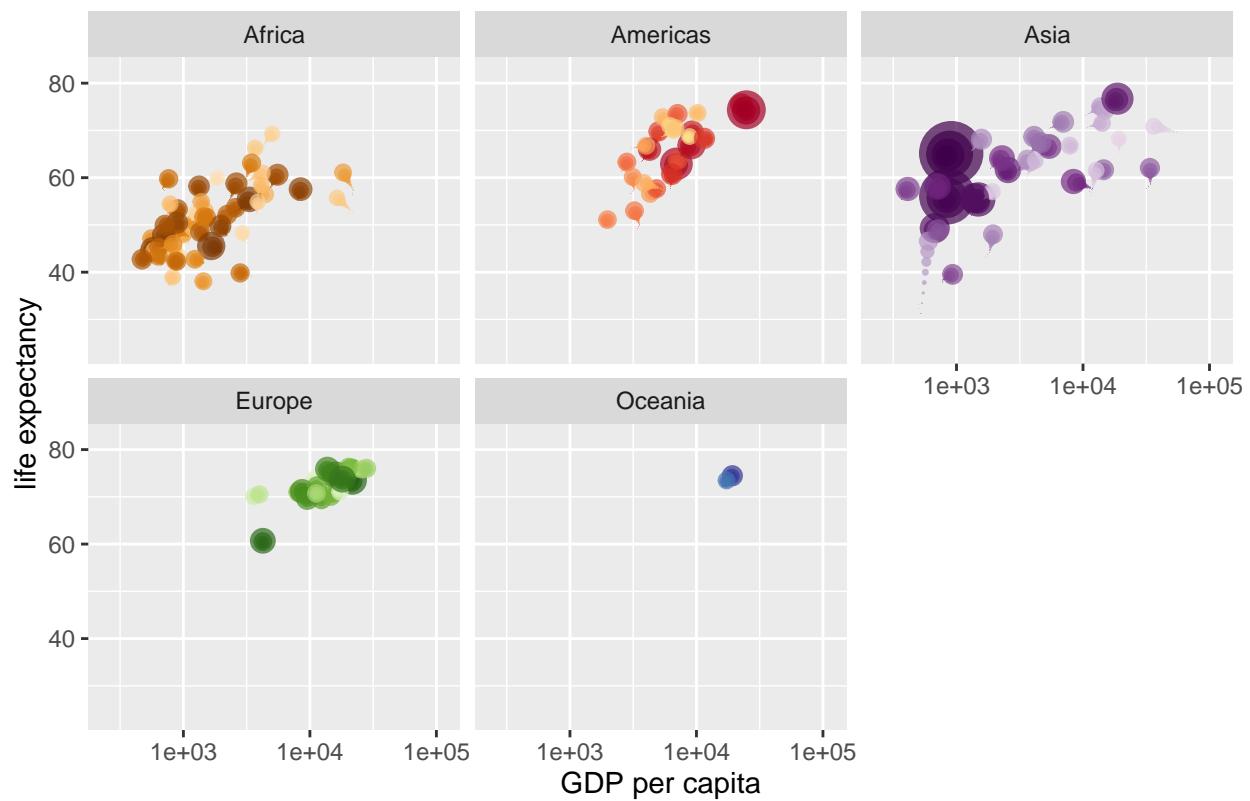
Year: 1980



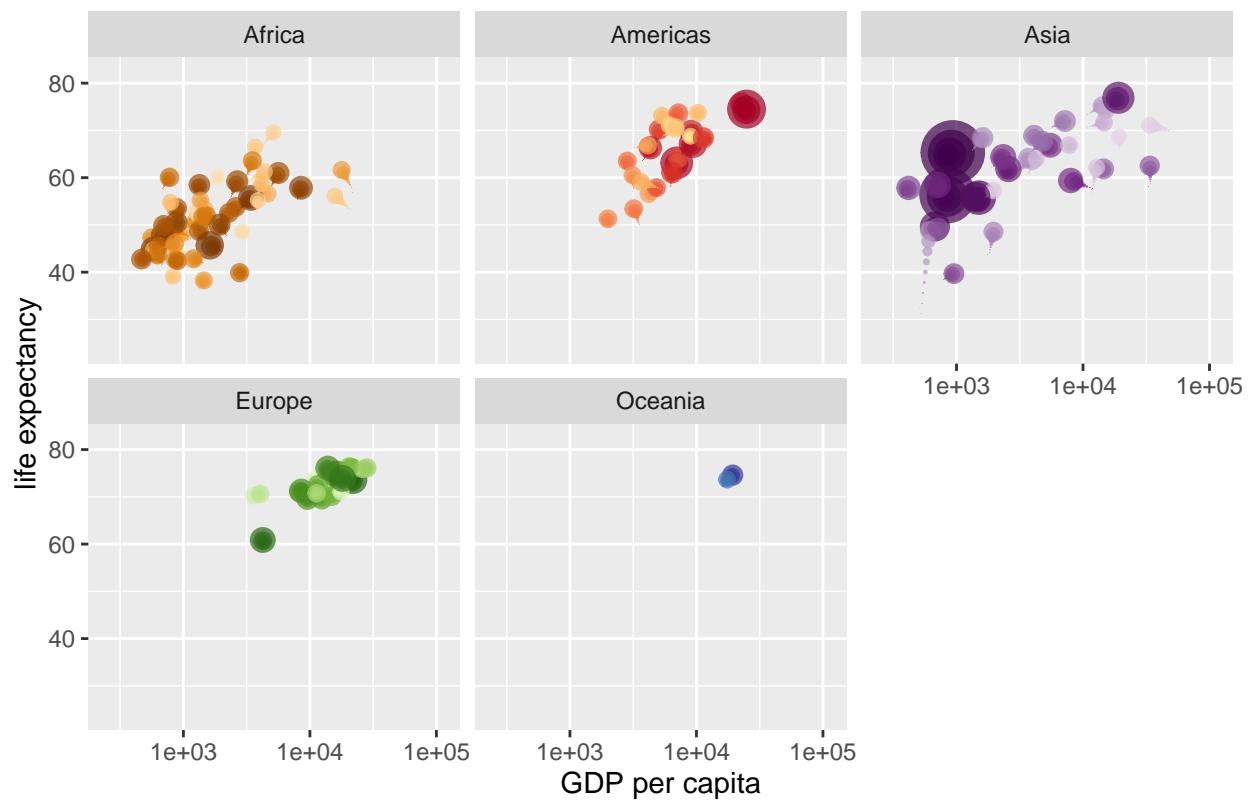
Year: 1980



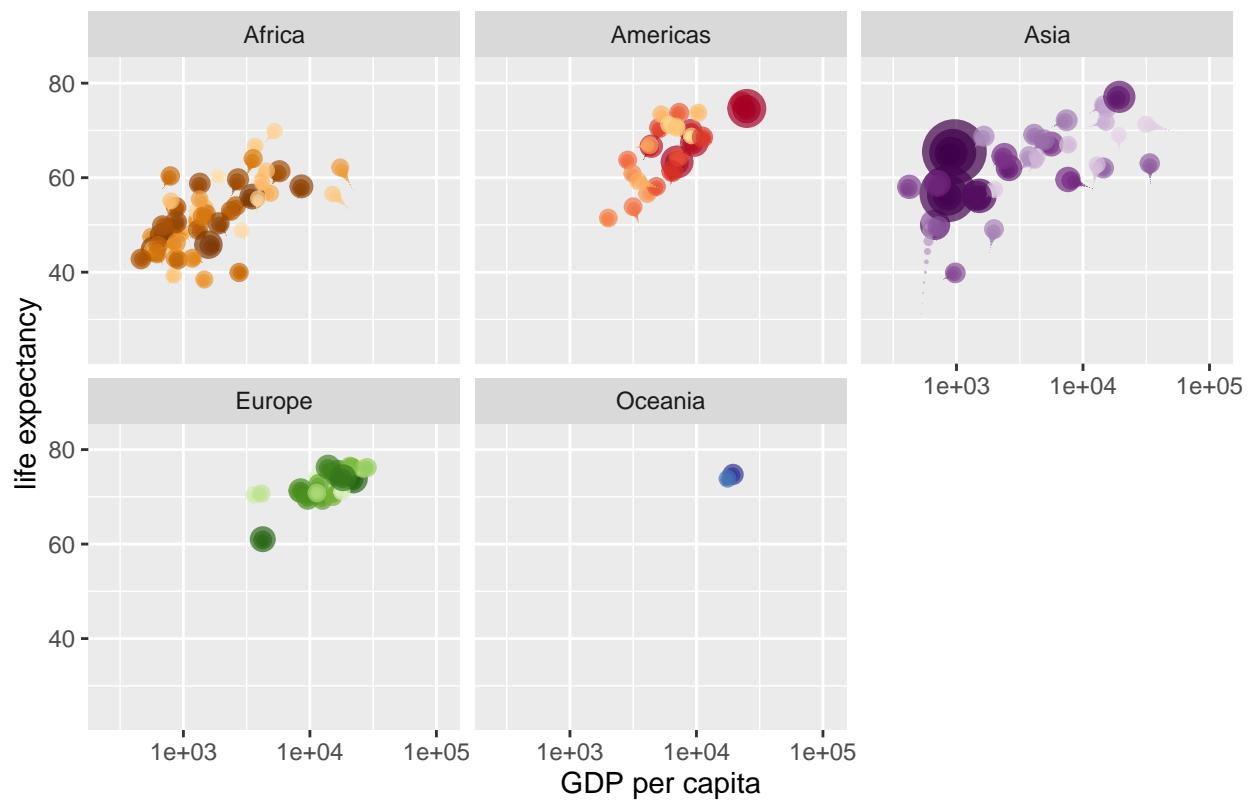
Year: 1981



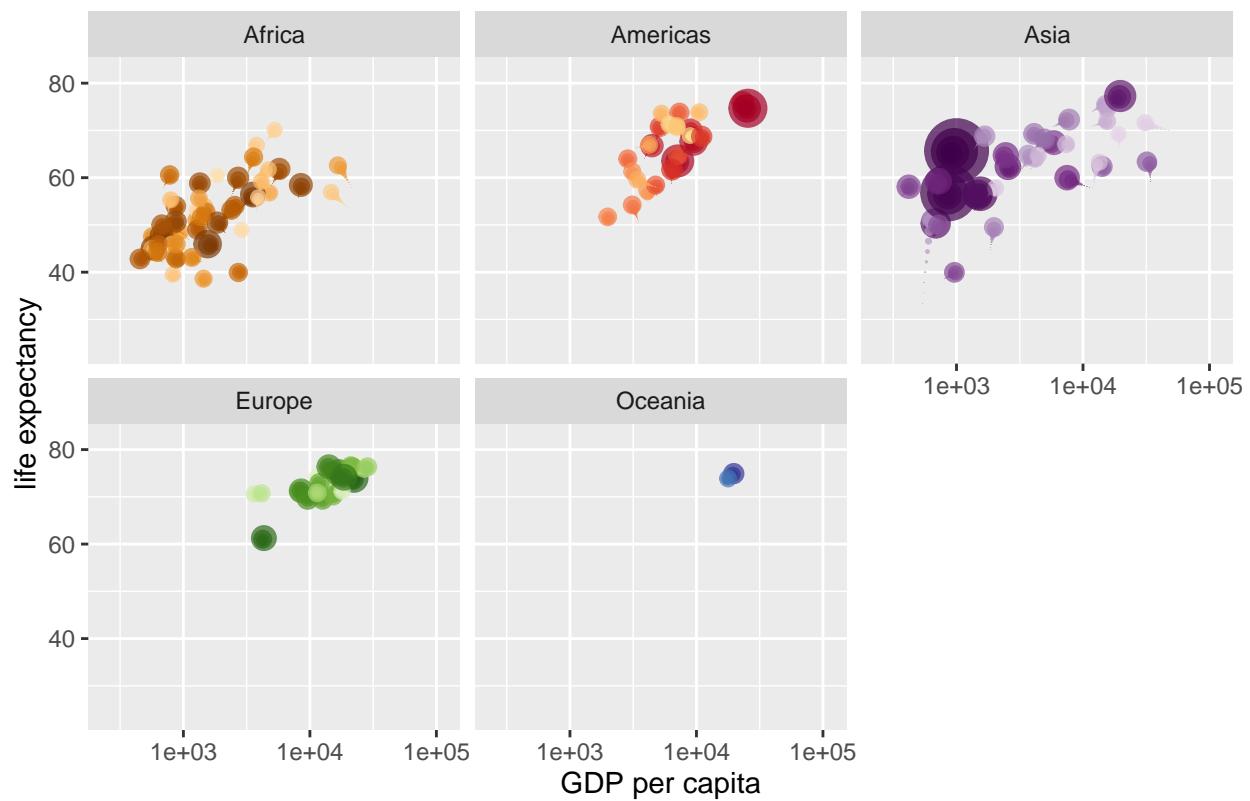
Year: 1981



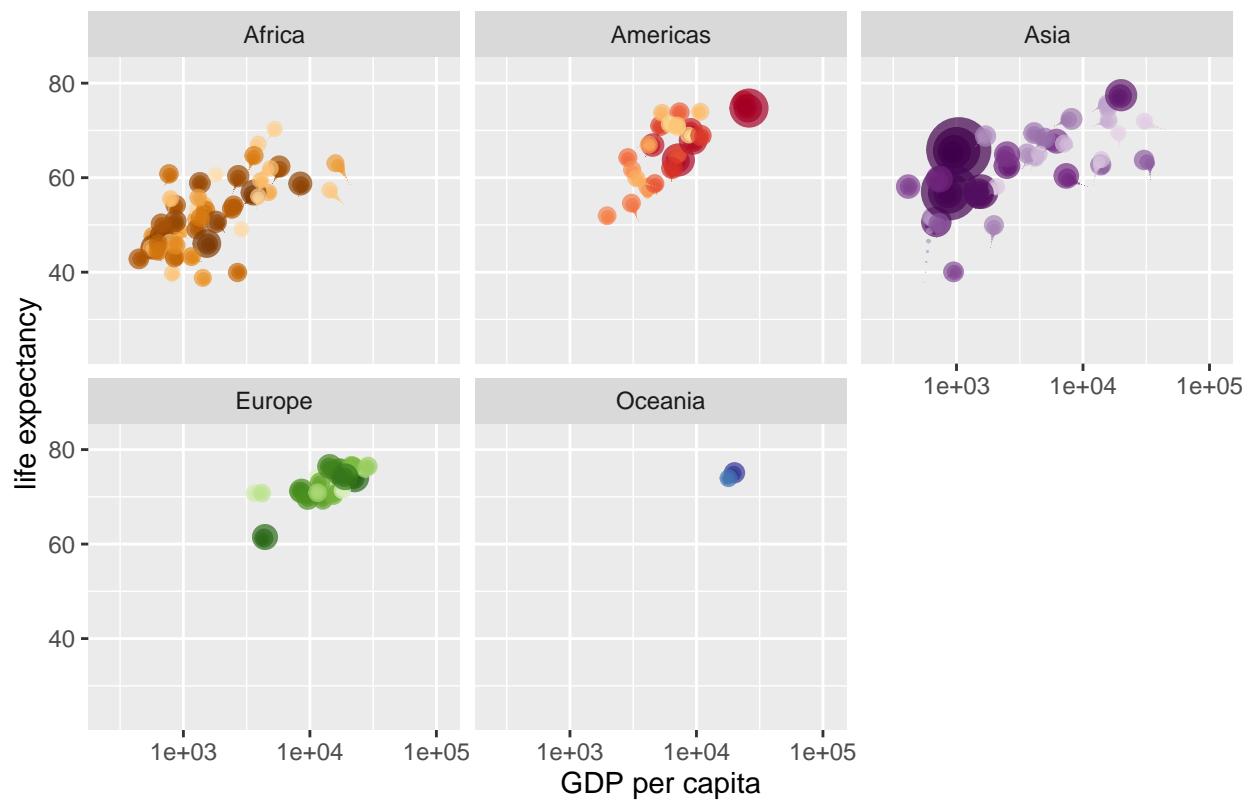
Year: 1982



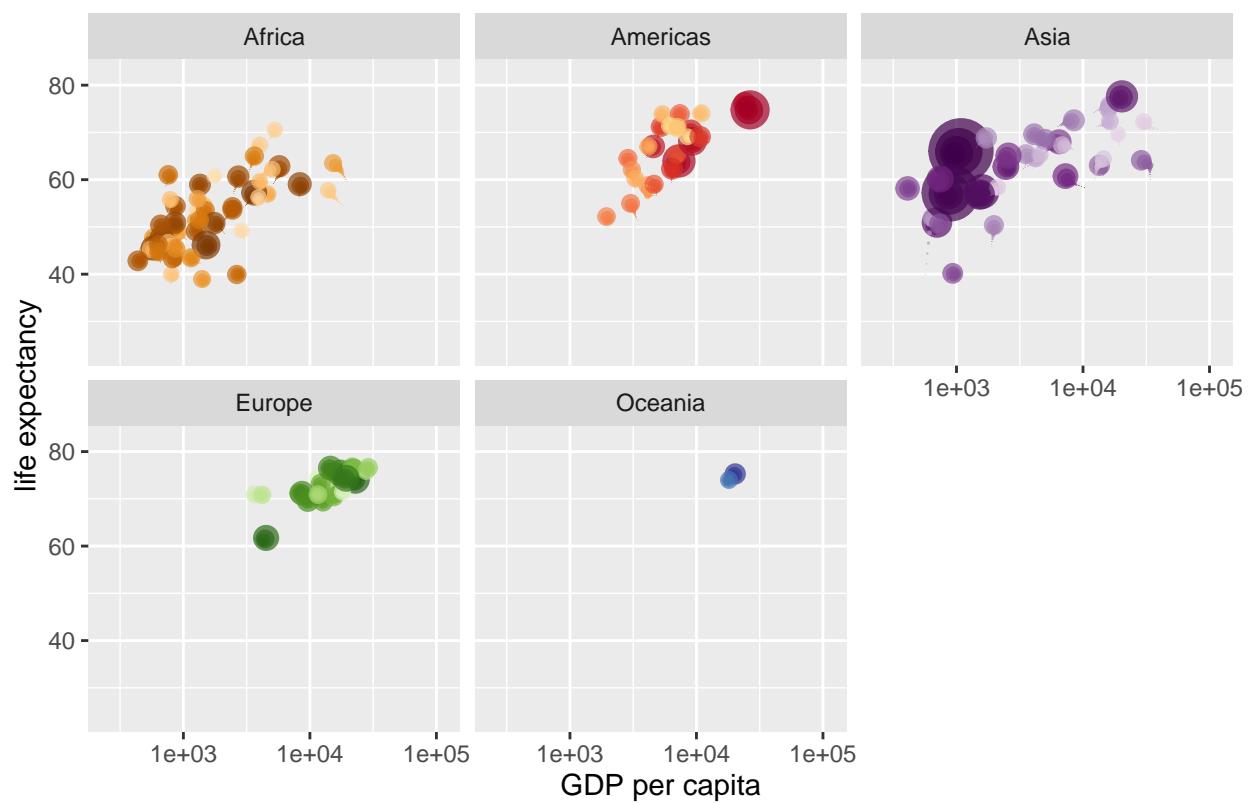
Year: 1983



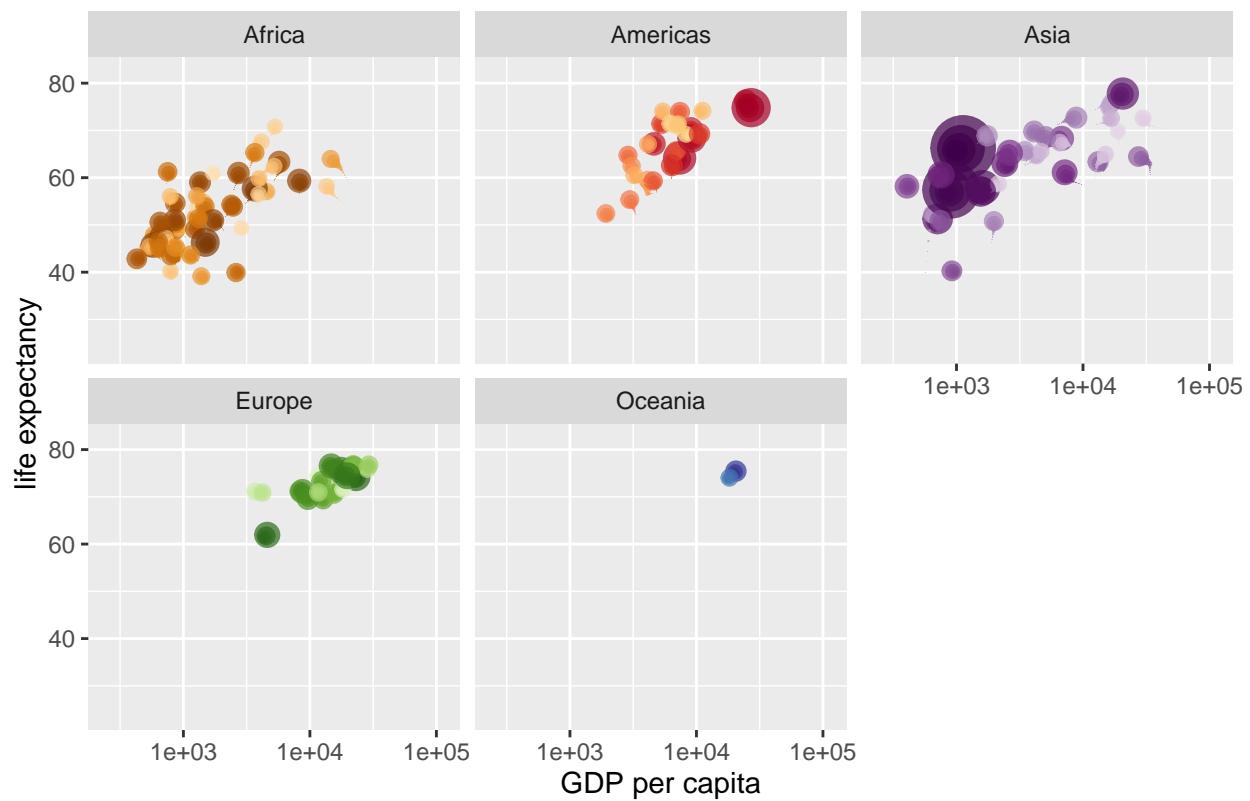
Year: 1983



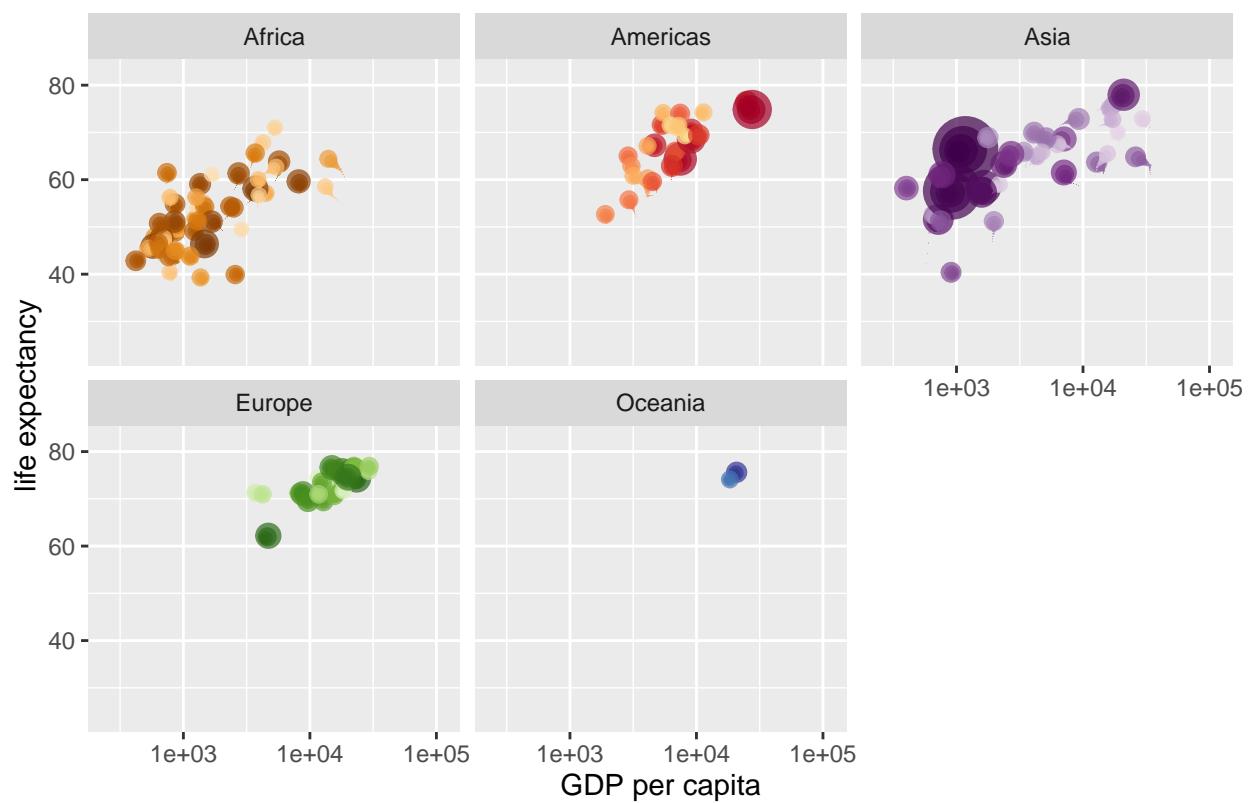
Year: 1984



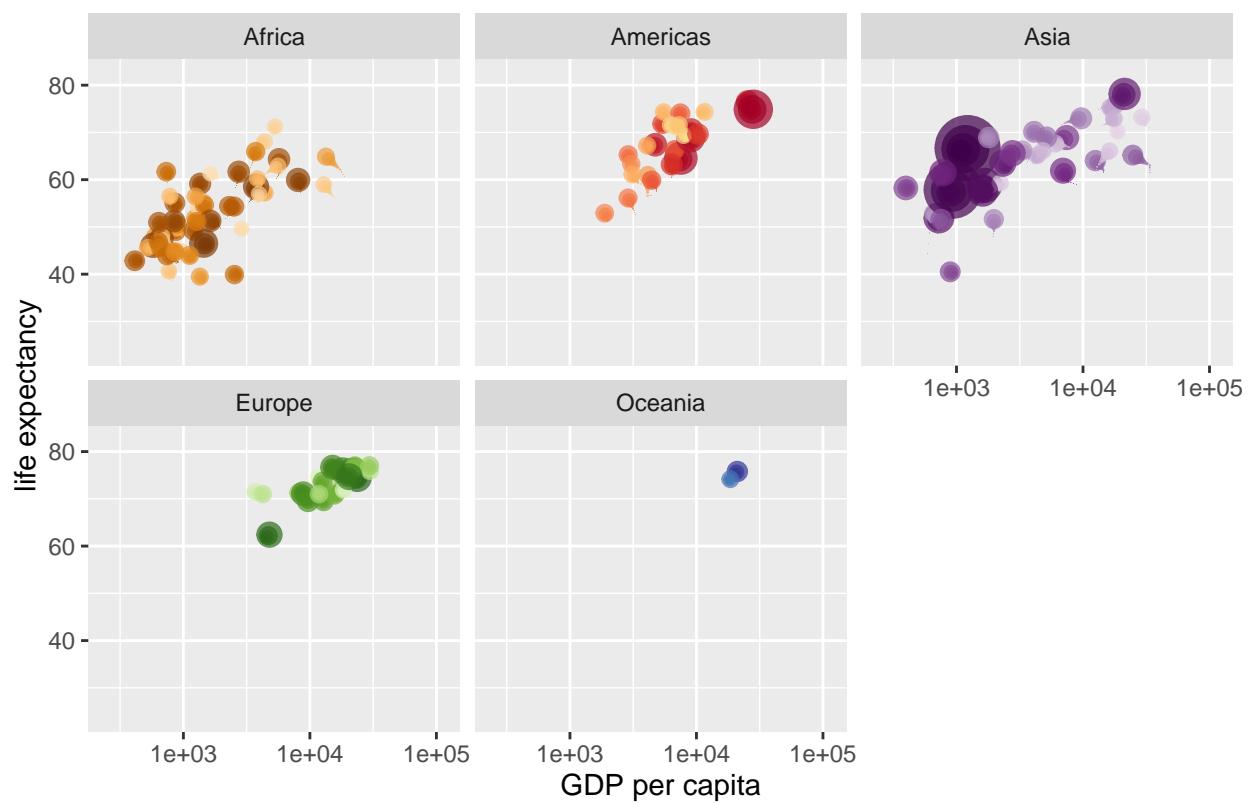
Year: 1984



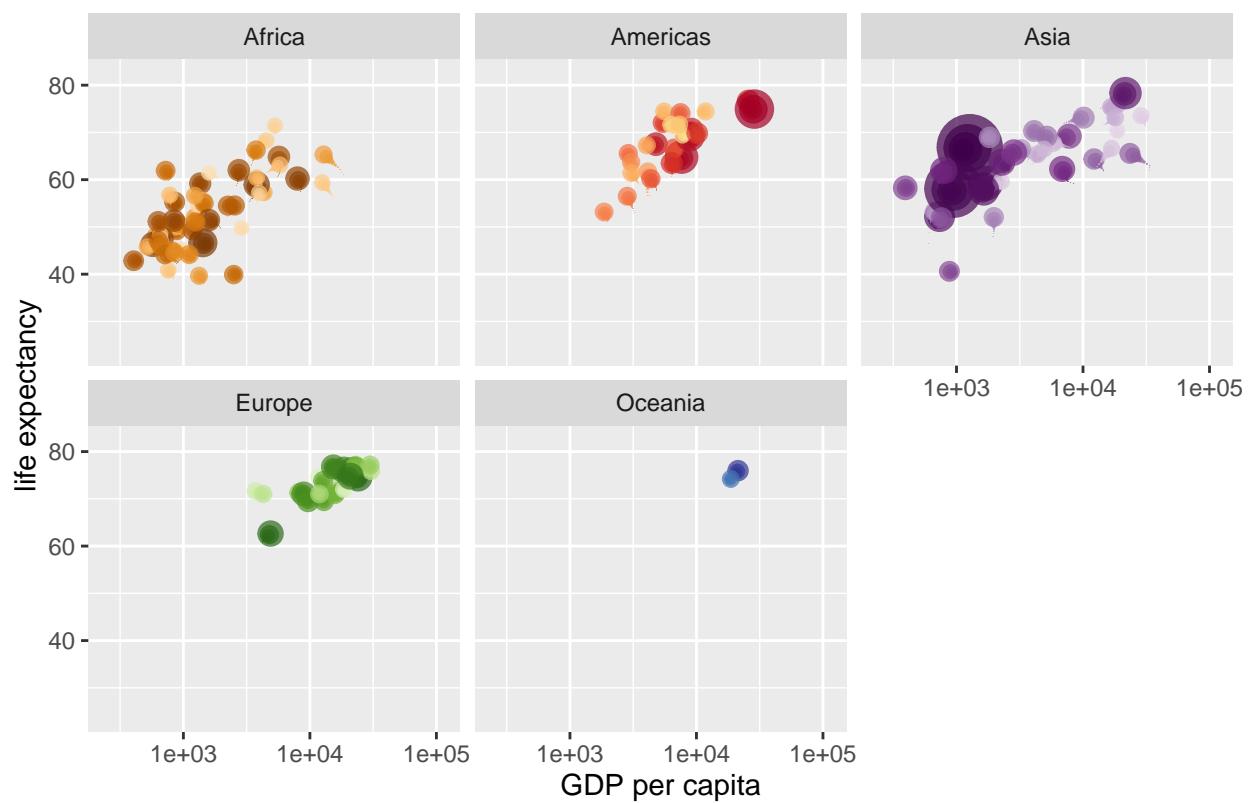
Year: 1985



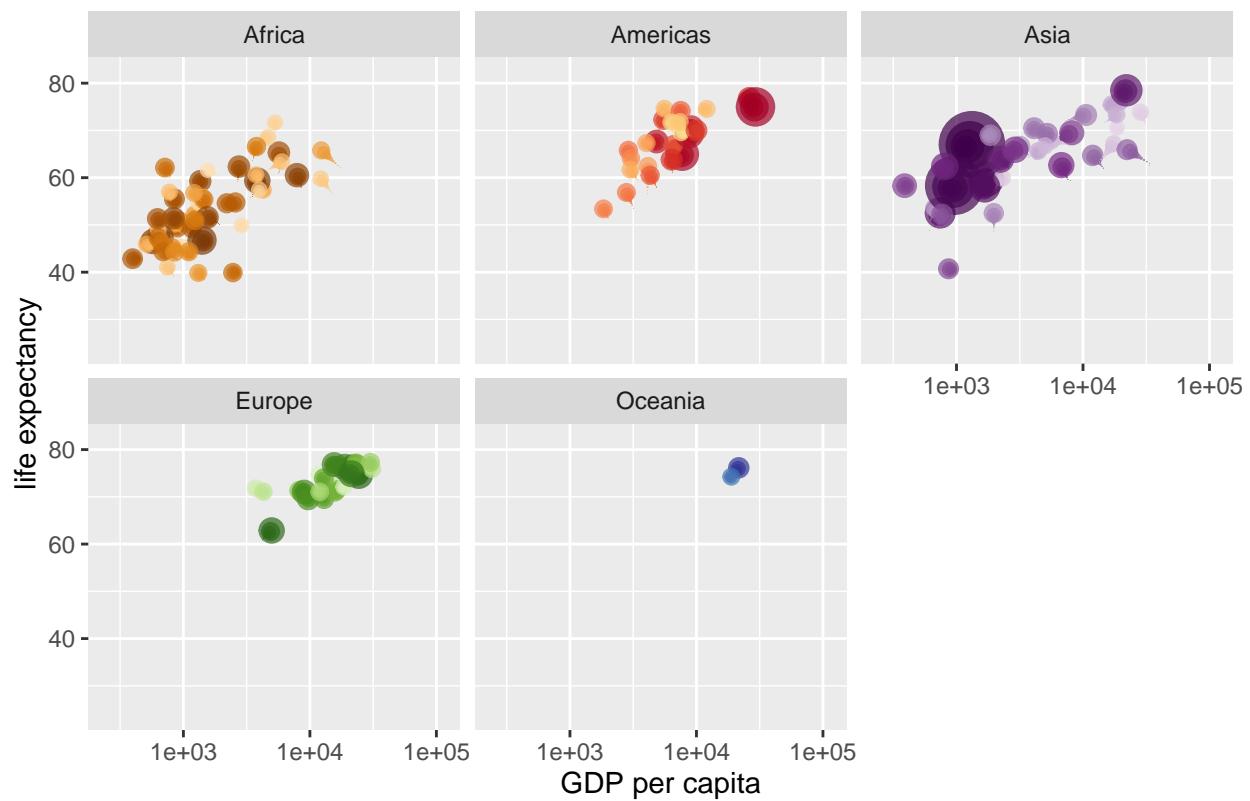
Year: 1985



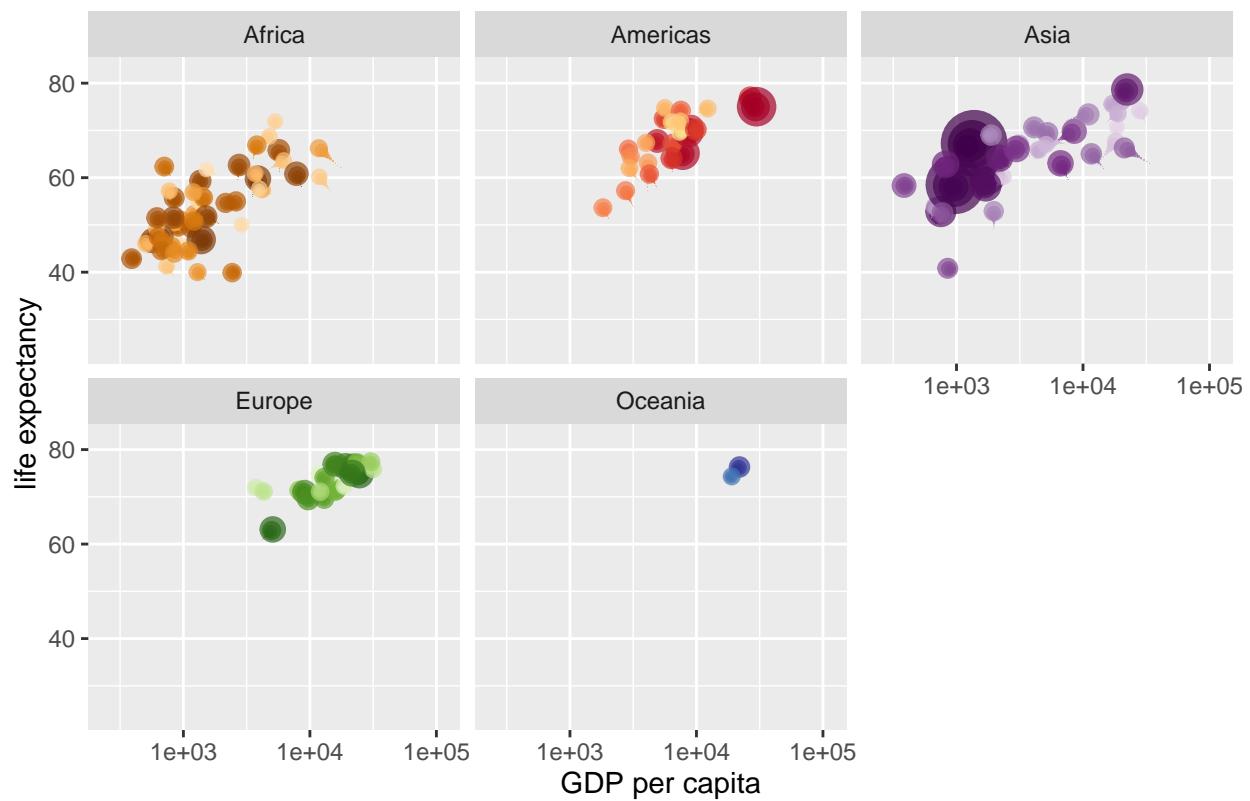
Year: 1986



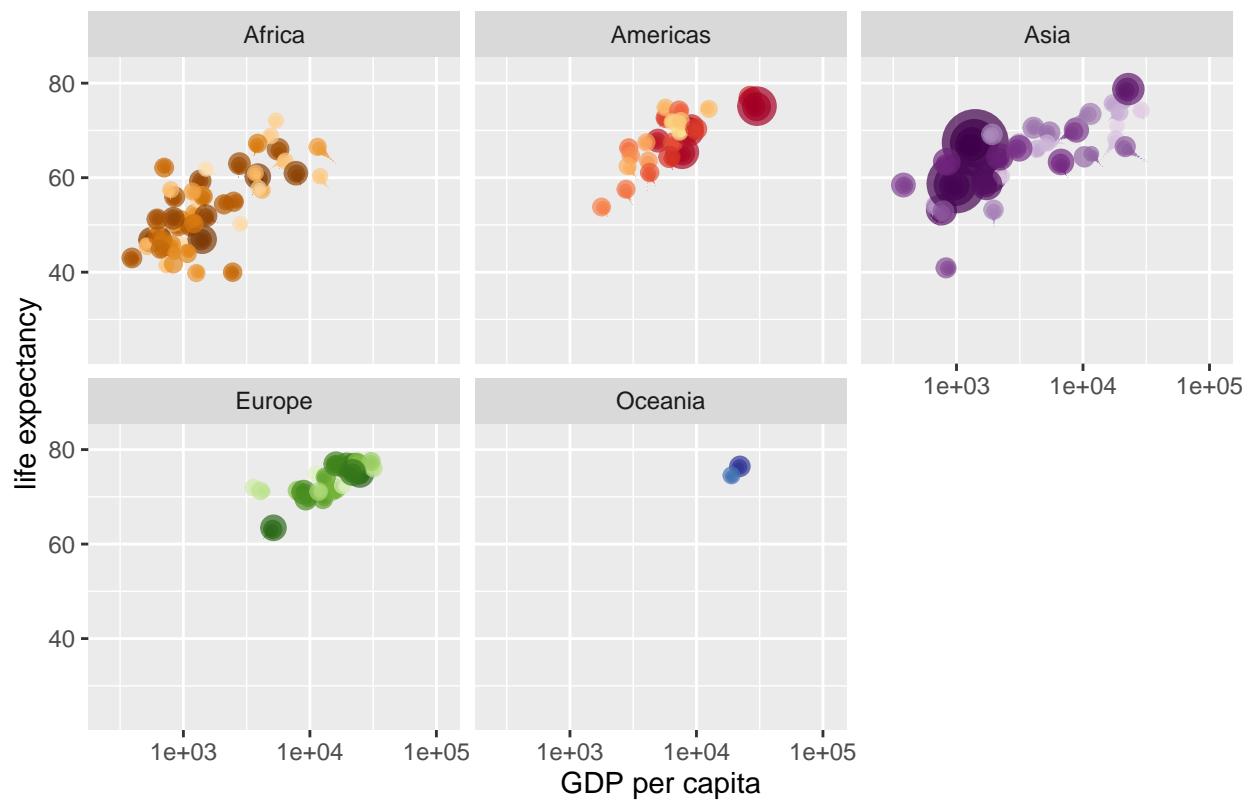
Year: 1986



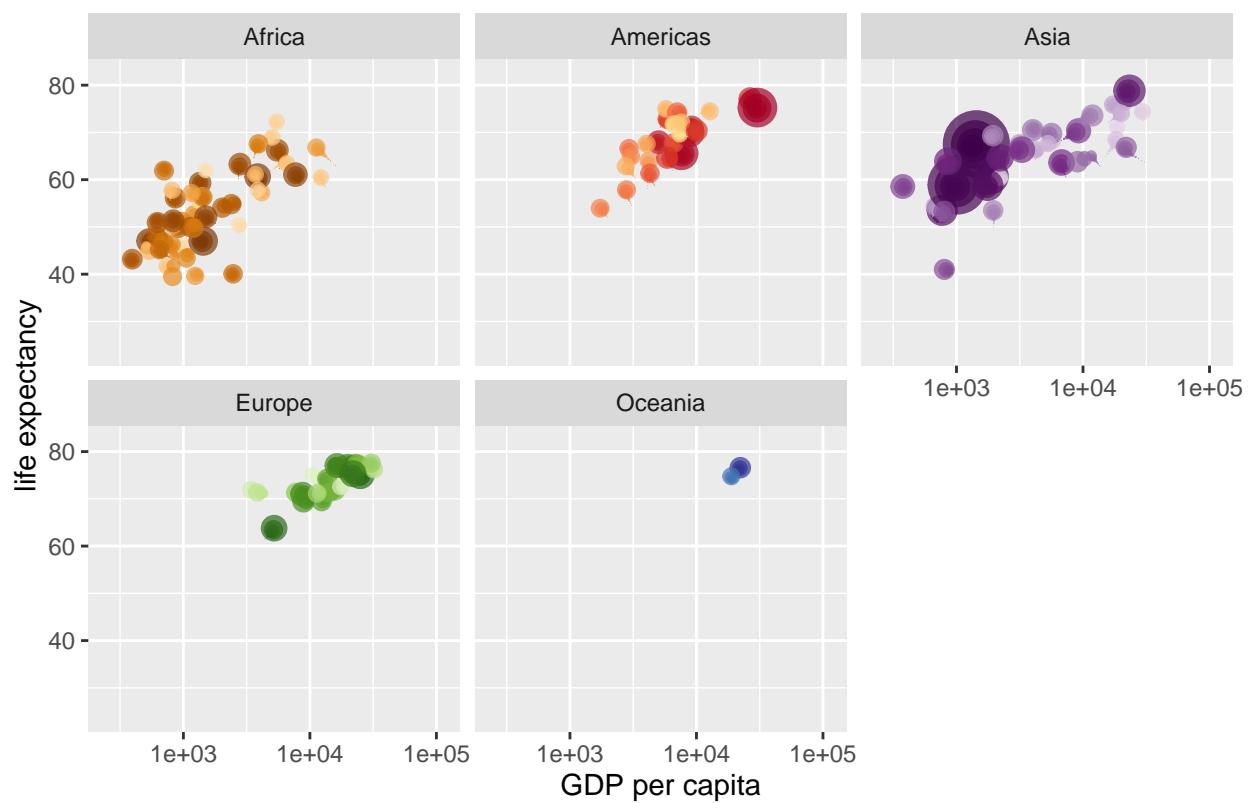
Year: 1987



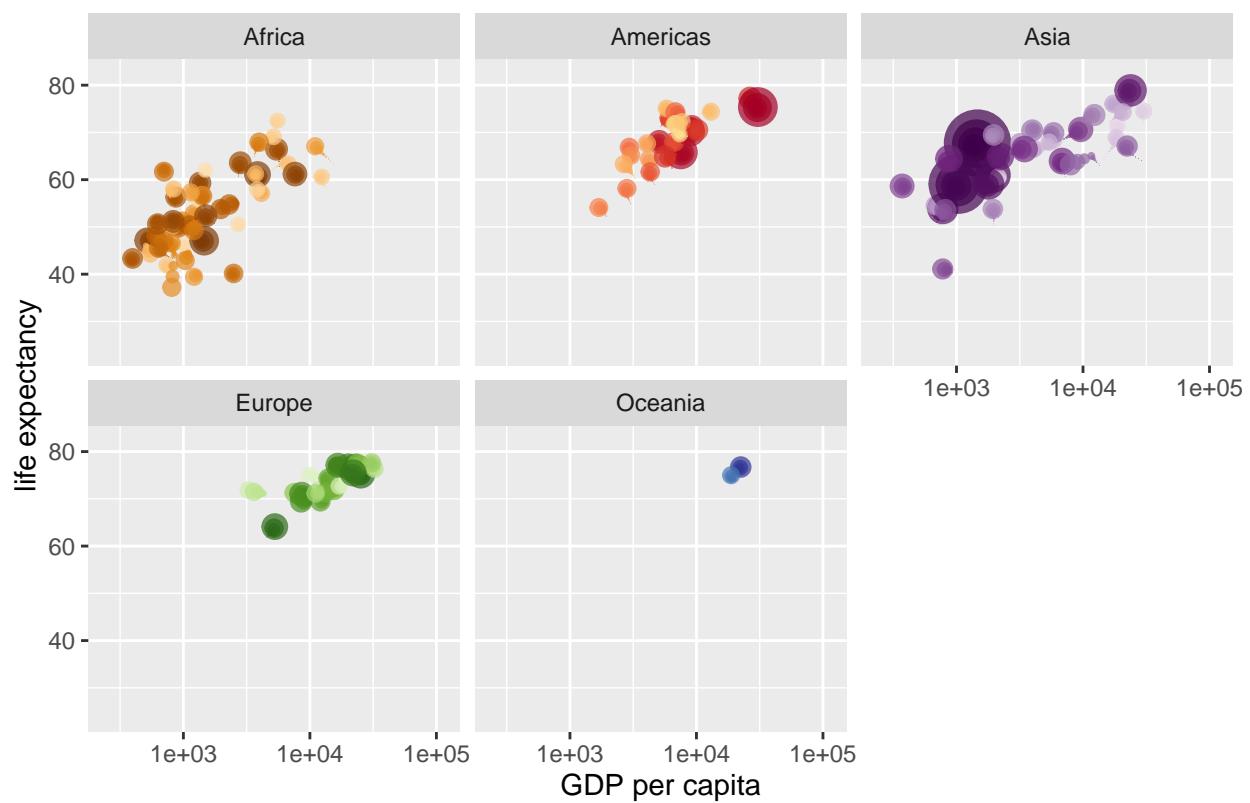
Year: 1988



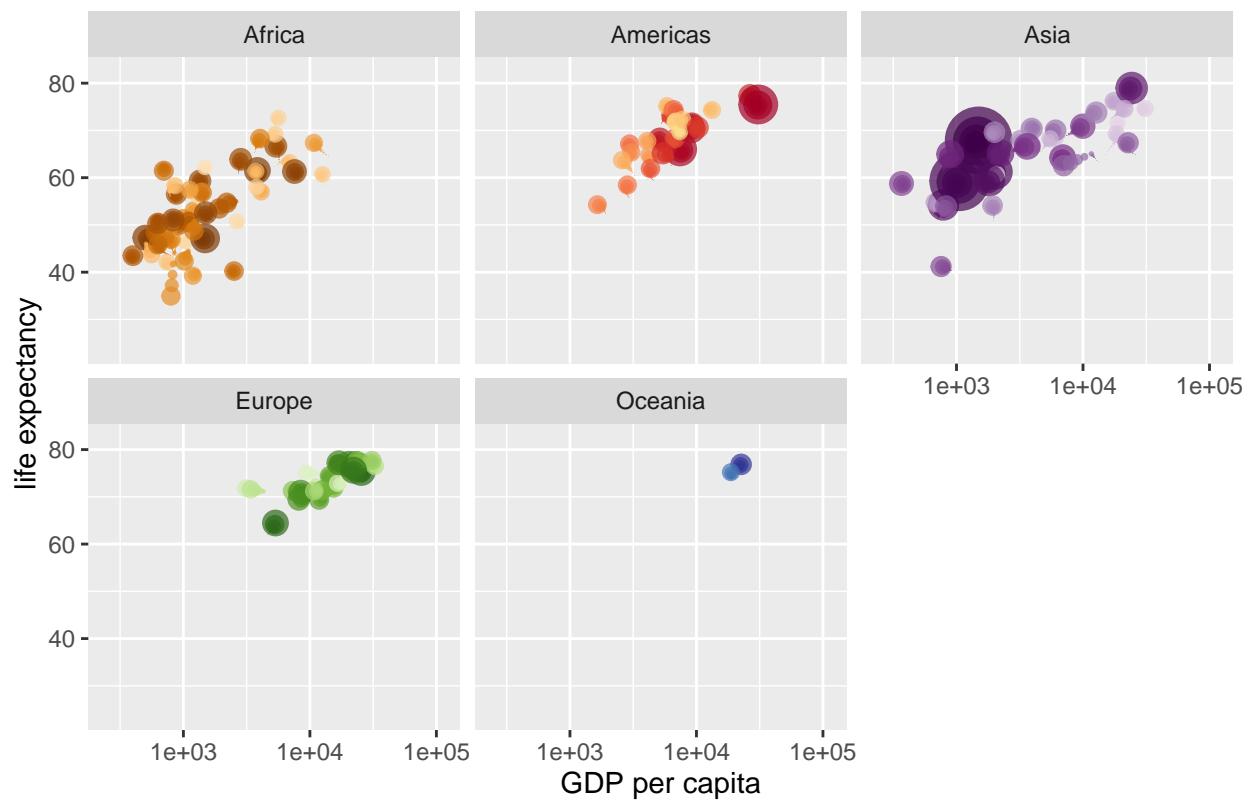
Year: 1988



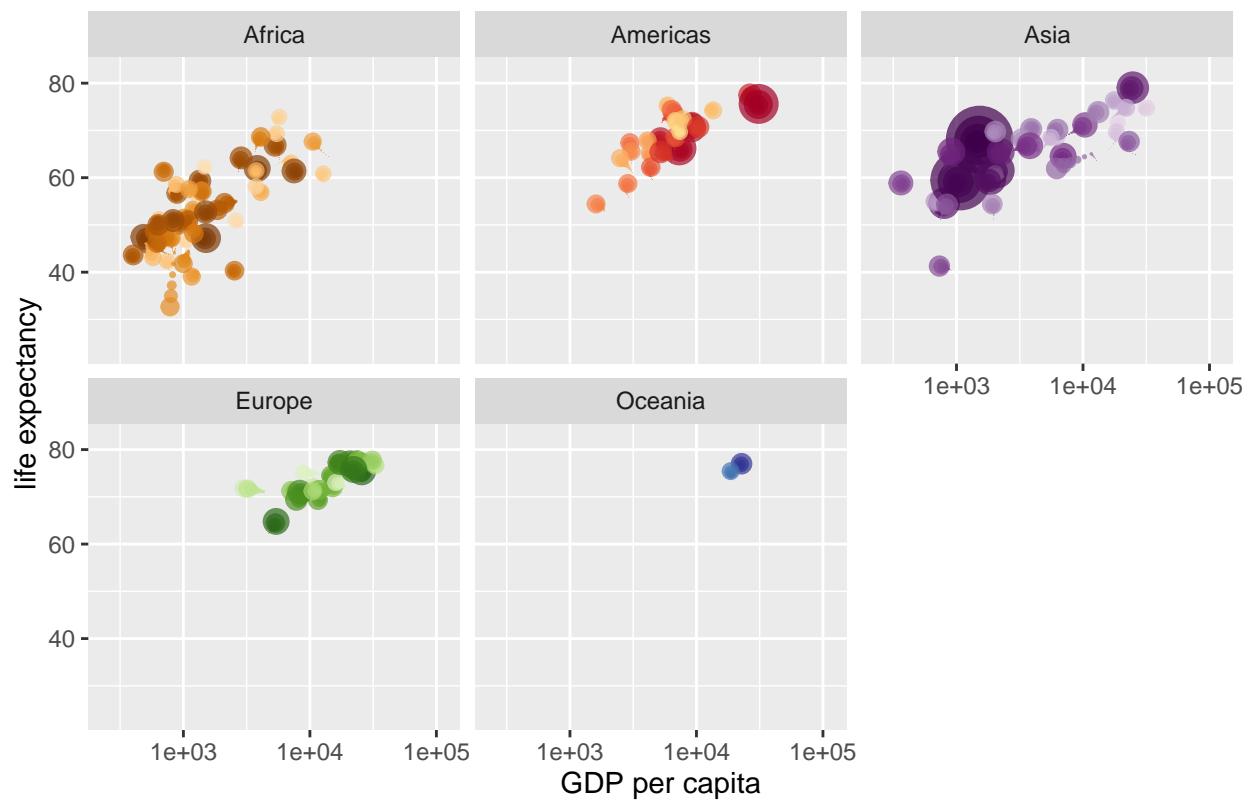
Year: 1989



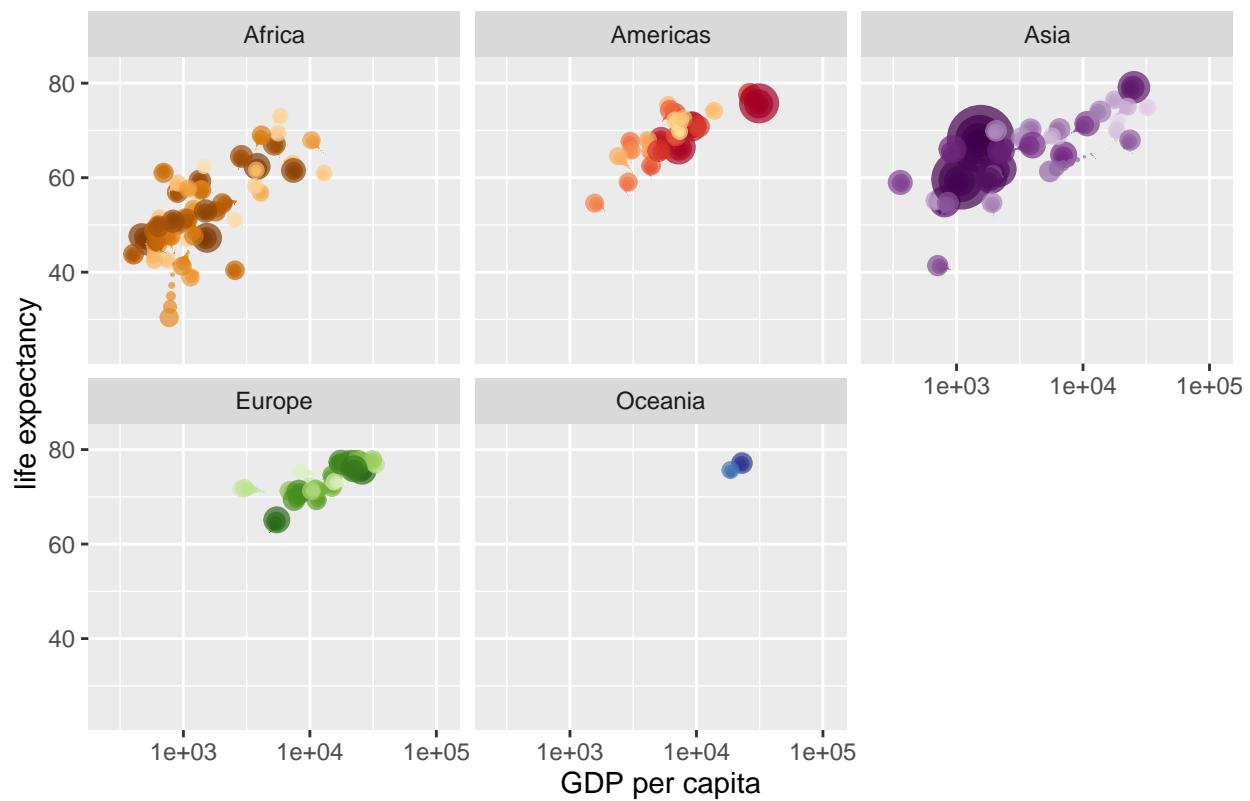
Year: 1989



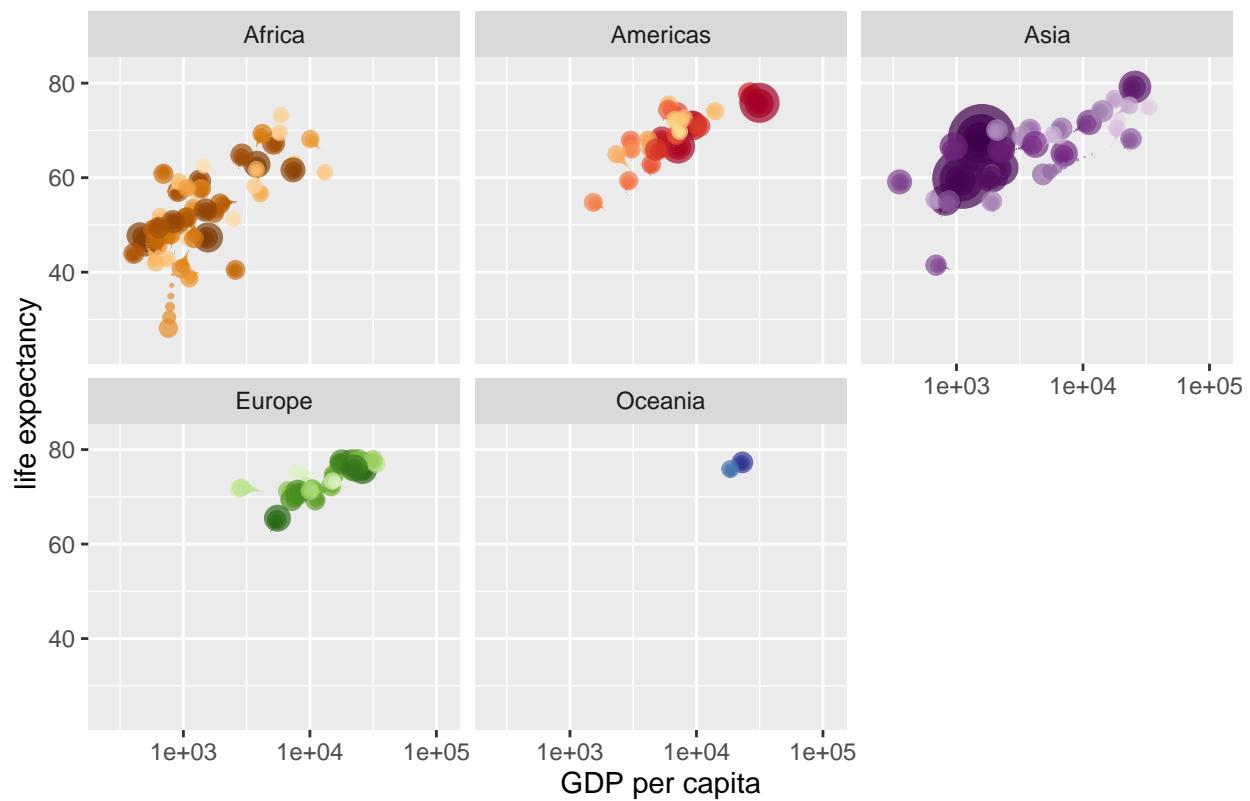
Year: 1990



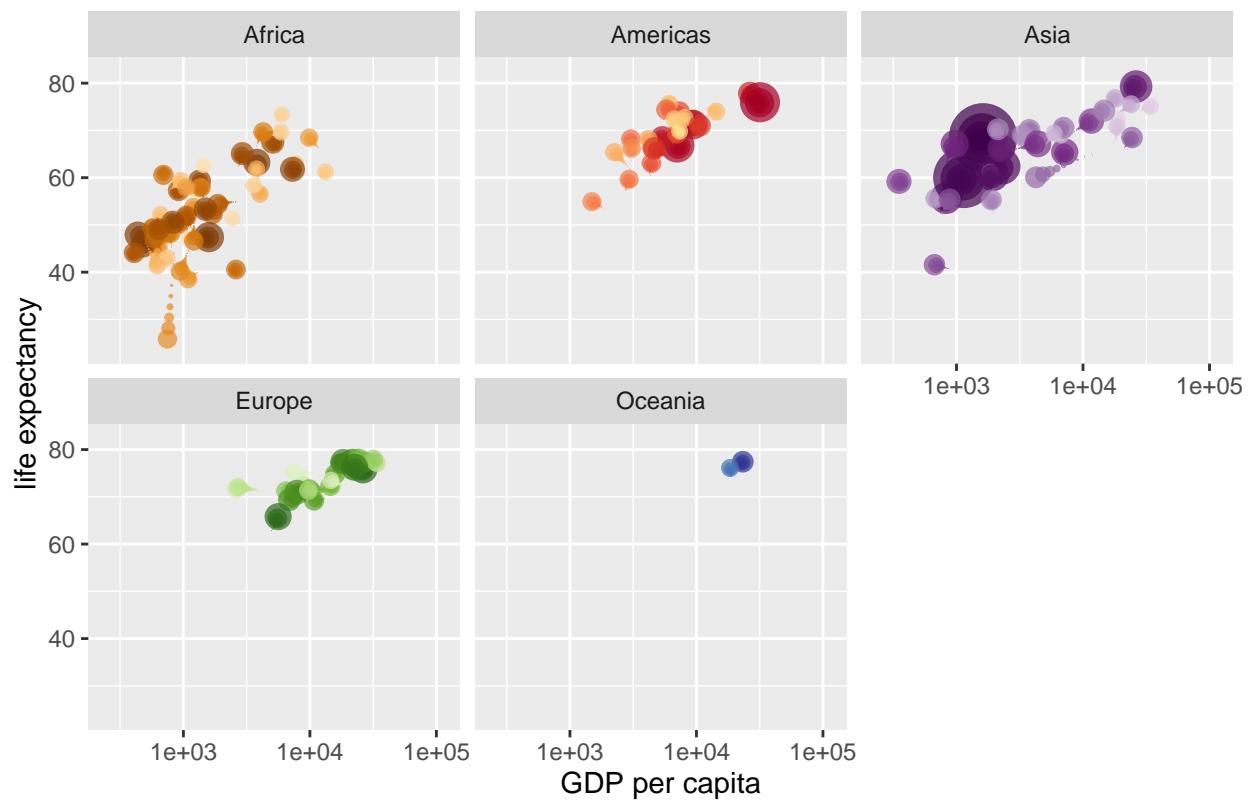
Year: 1990



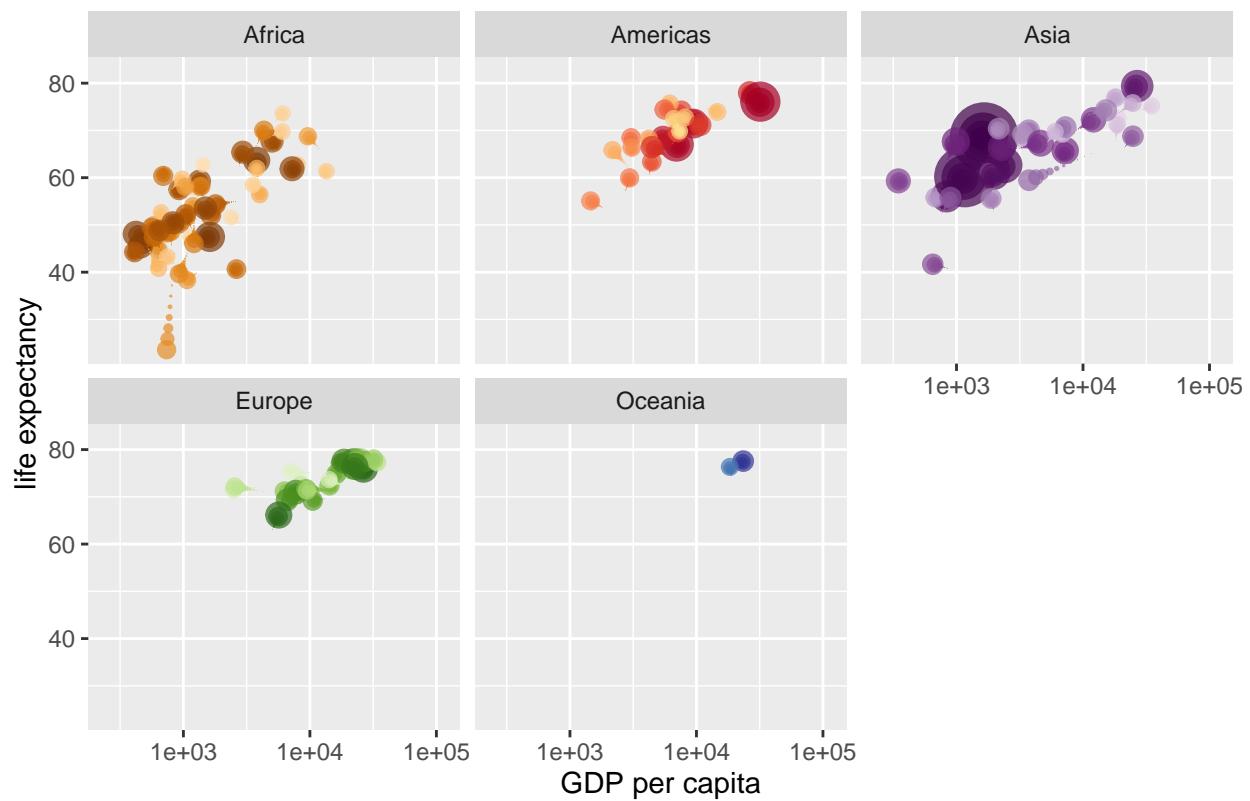
Year: 1991



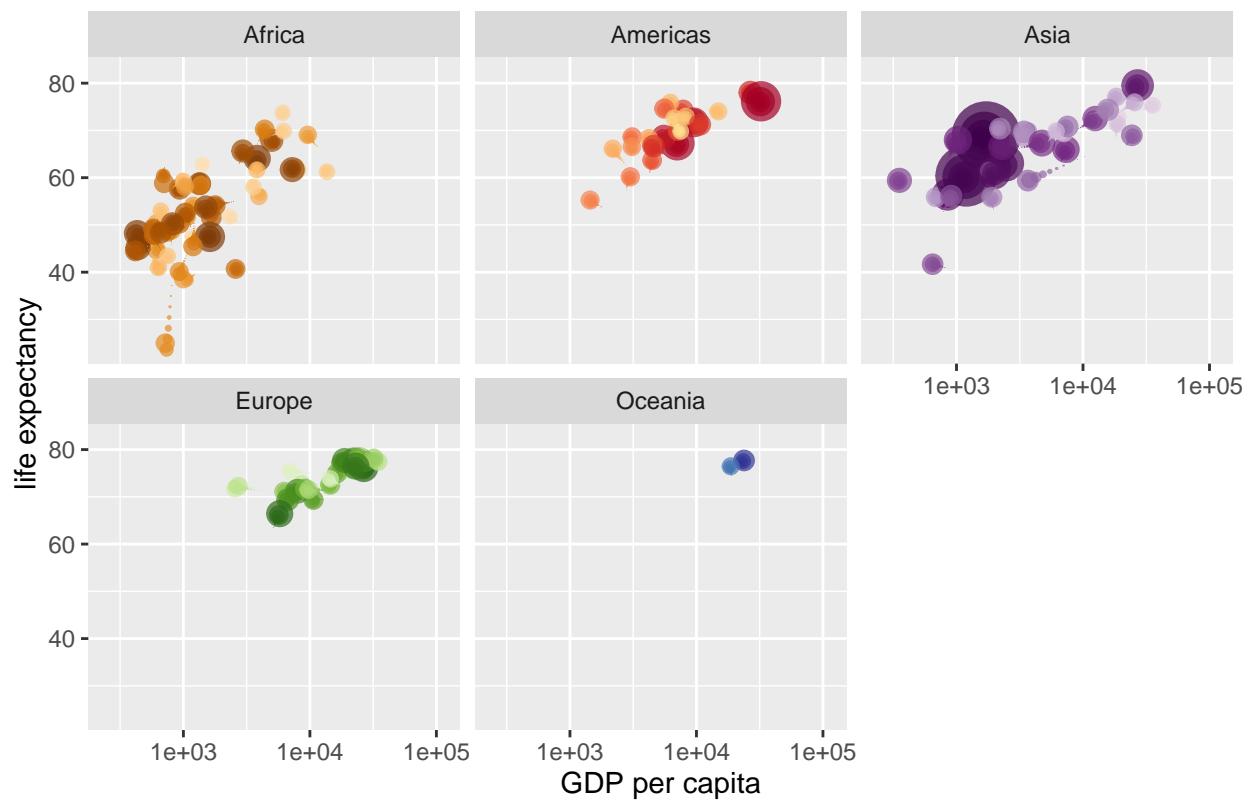
Year: 1991



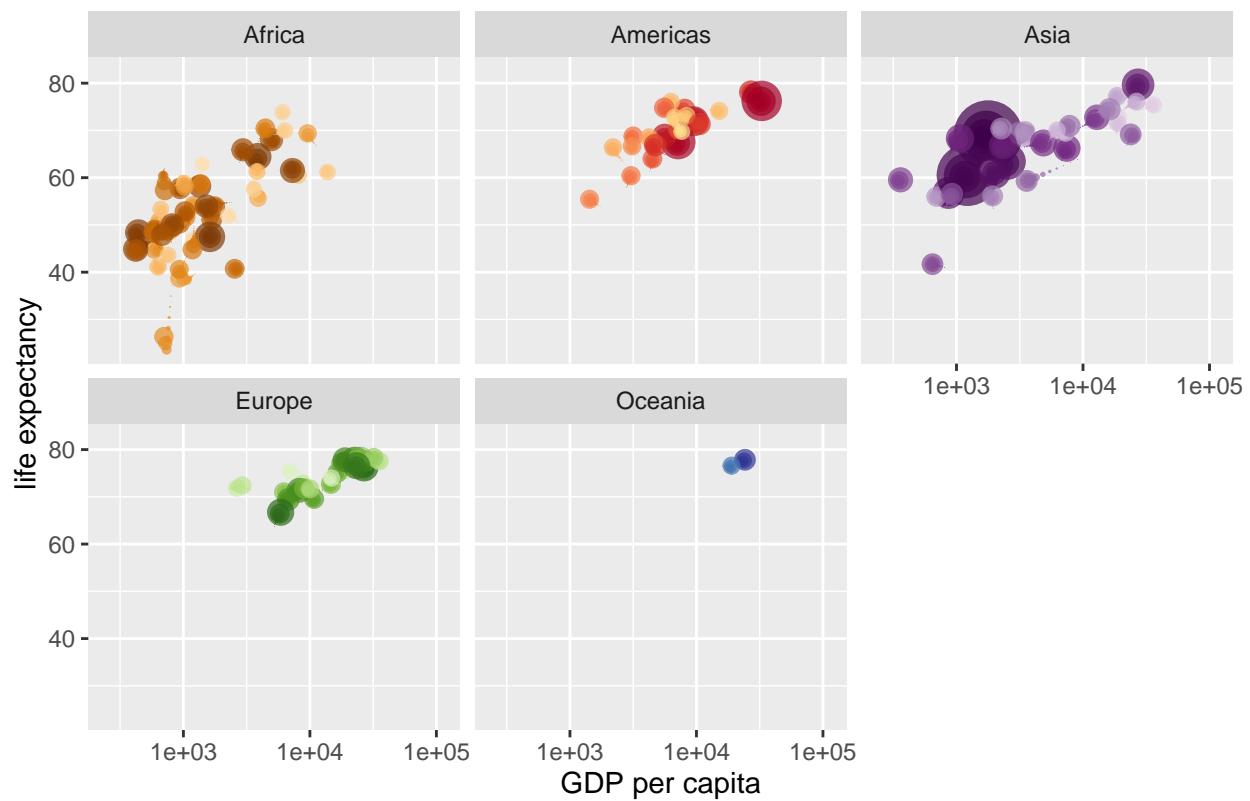
Year: 1992



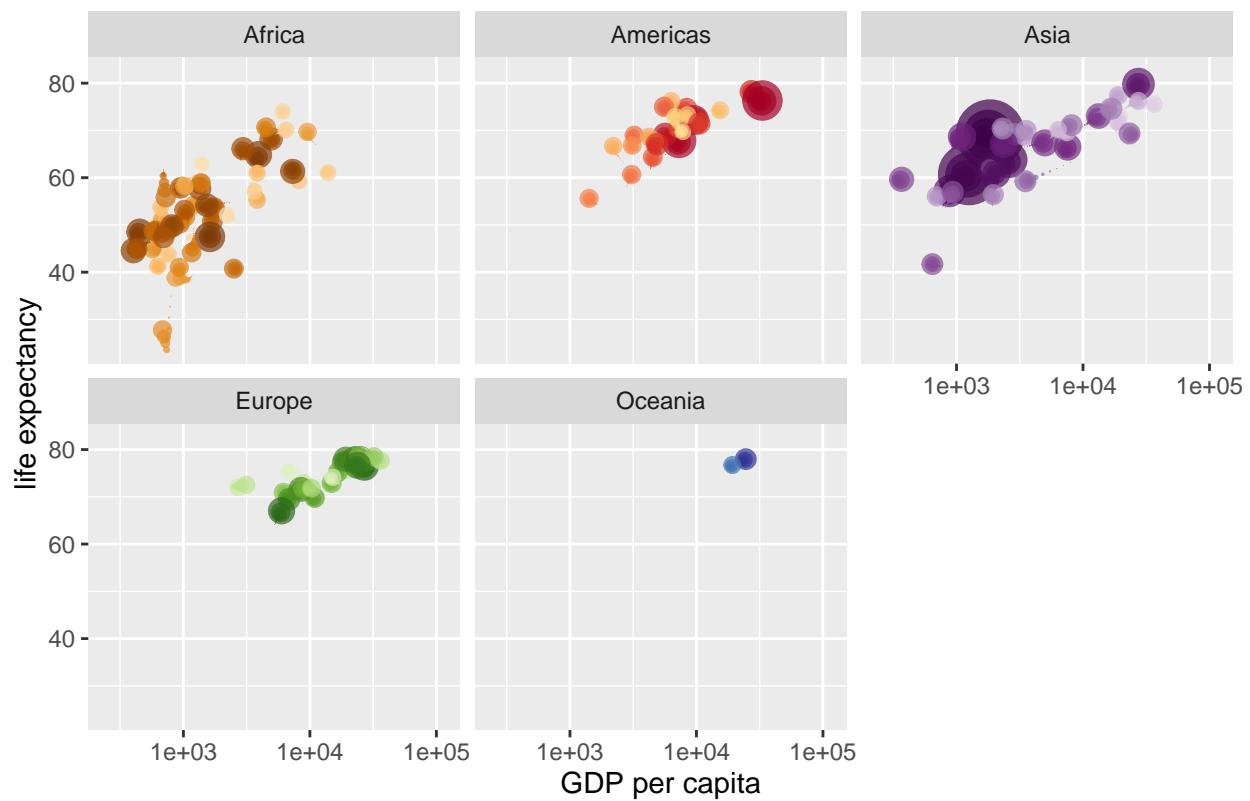
Year: 1993



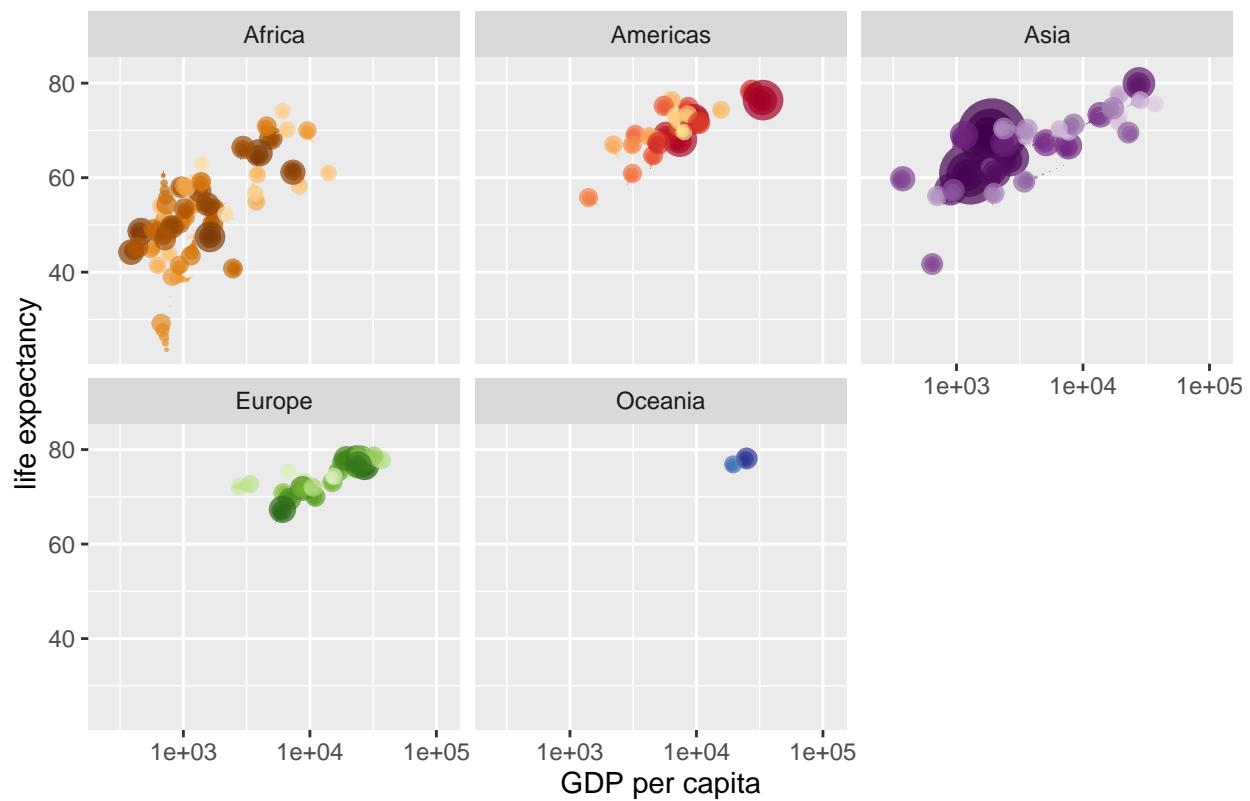
Year: 1993



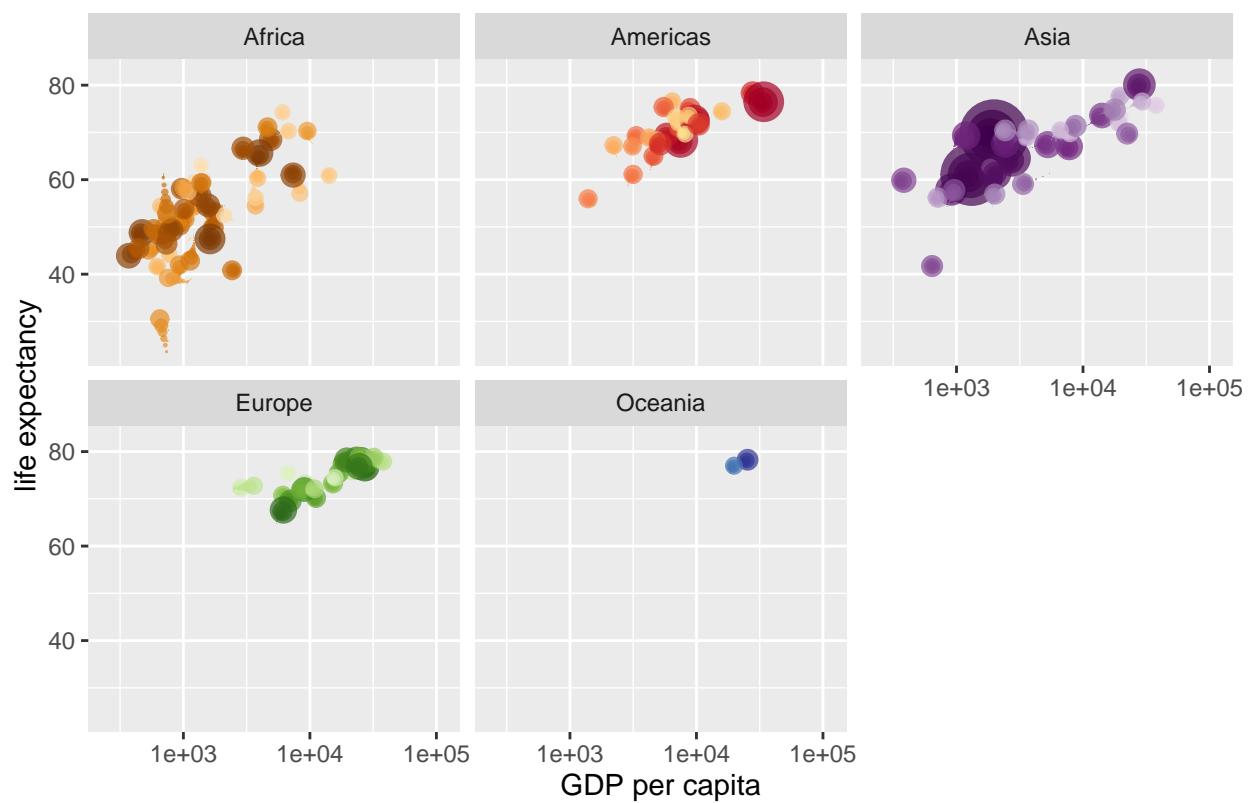
Year: 1994



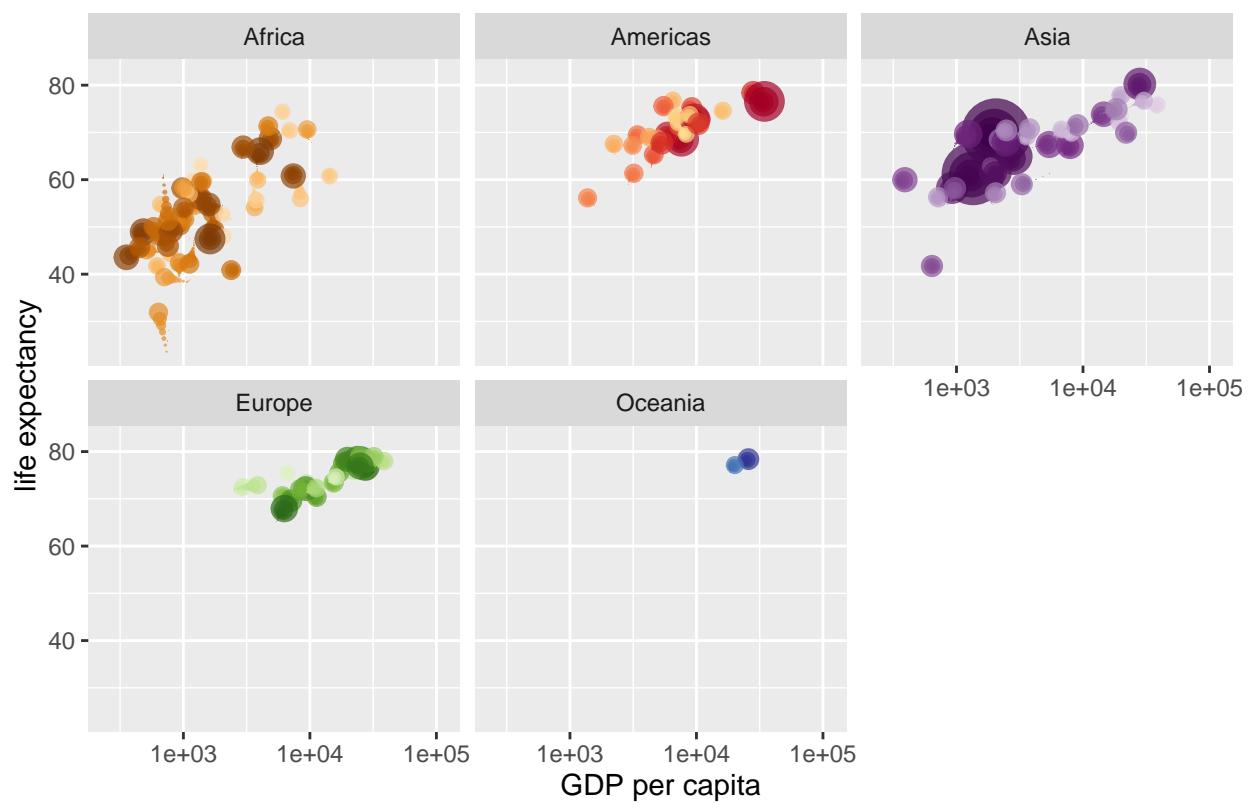
Year: 1994



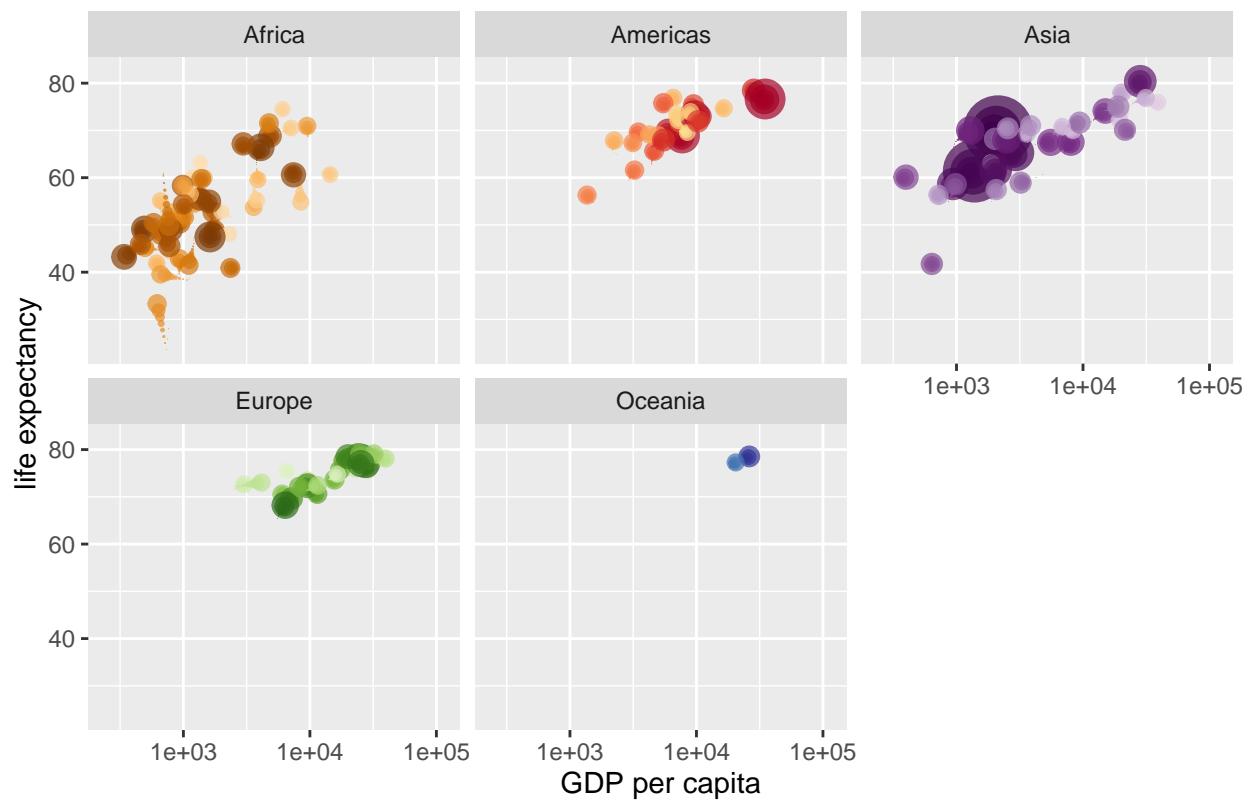
Year: 1995



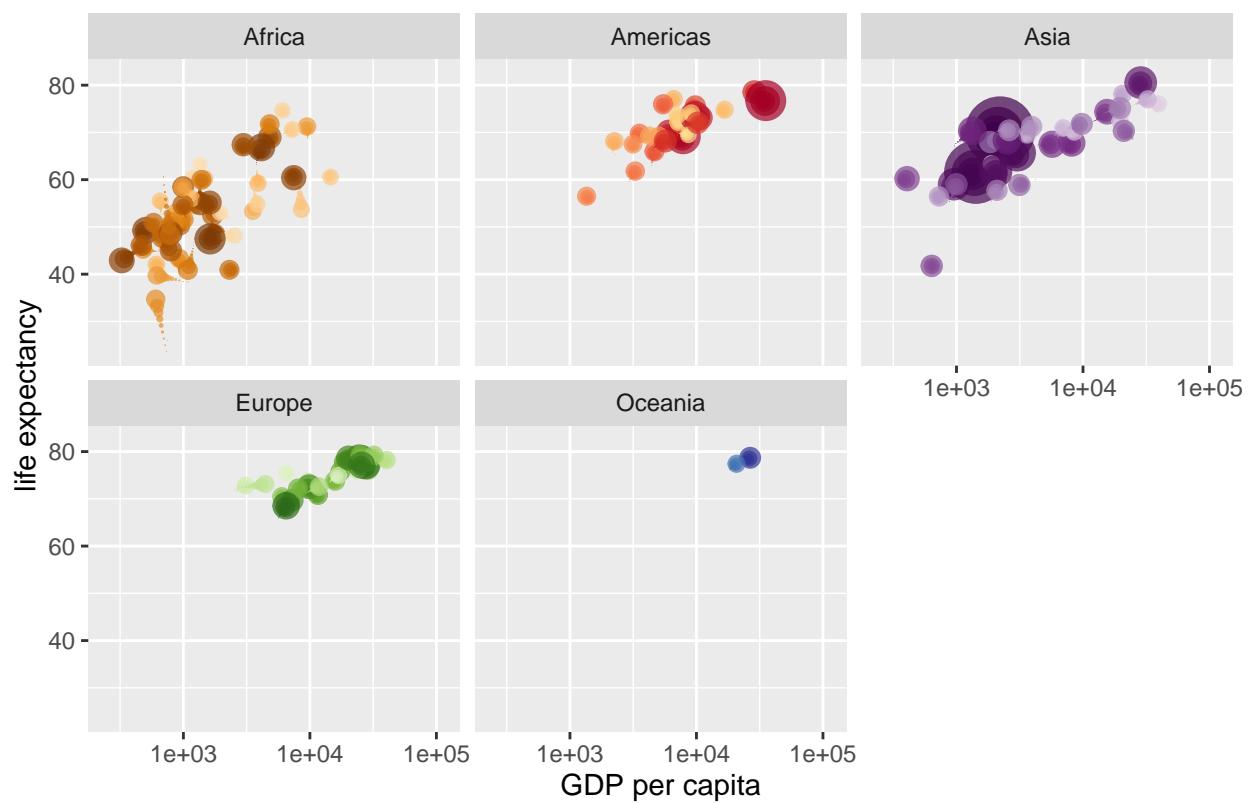
Year: 1995



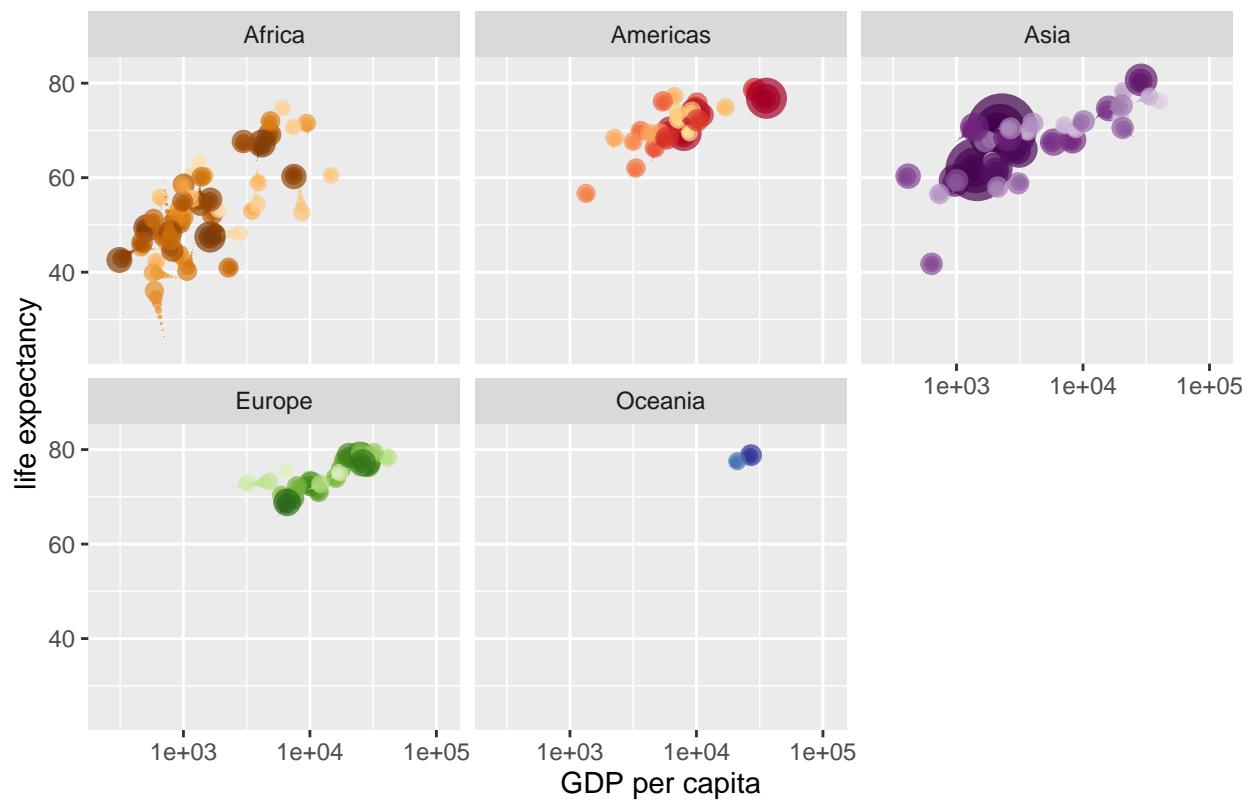
Year: 1996



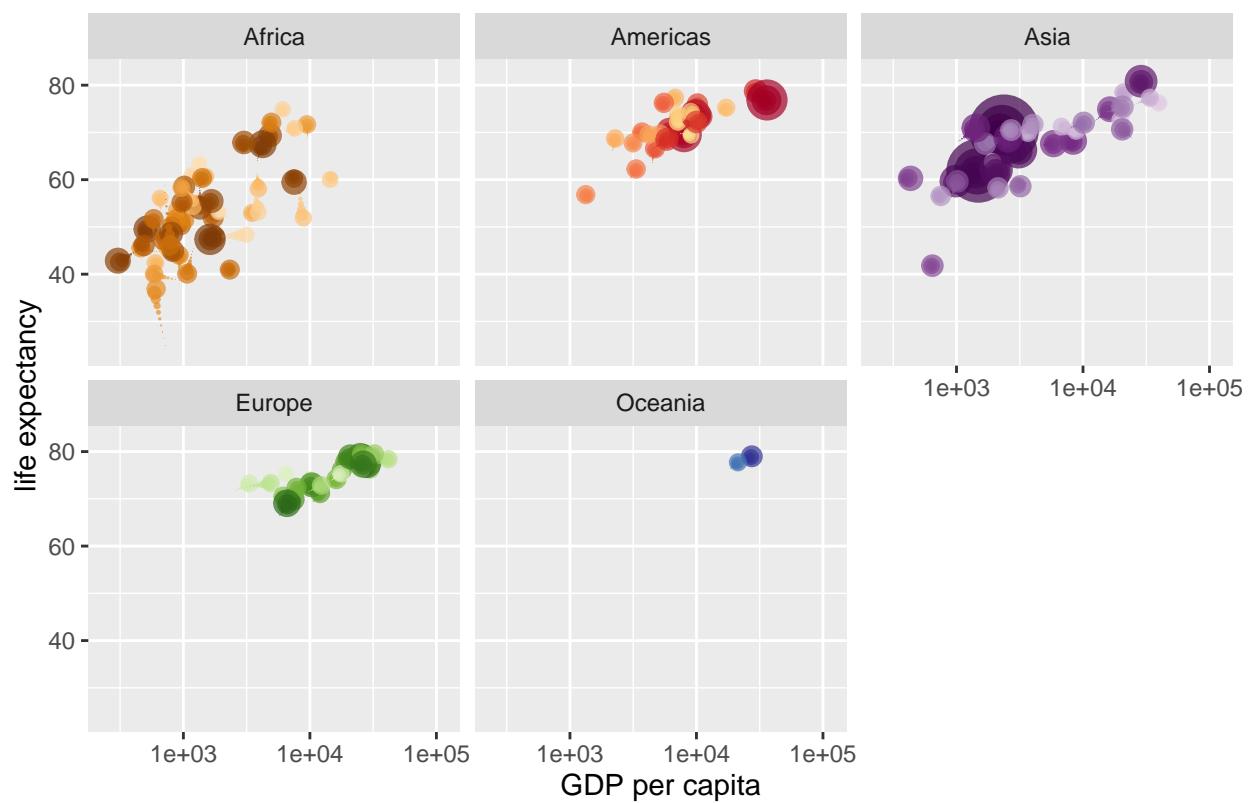
Year: 1996



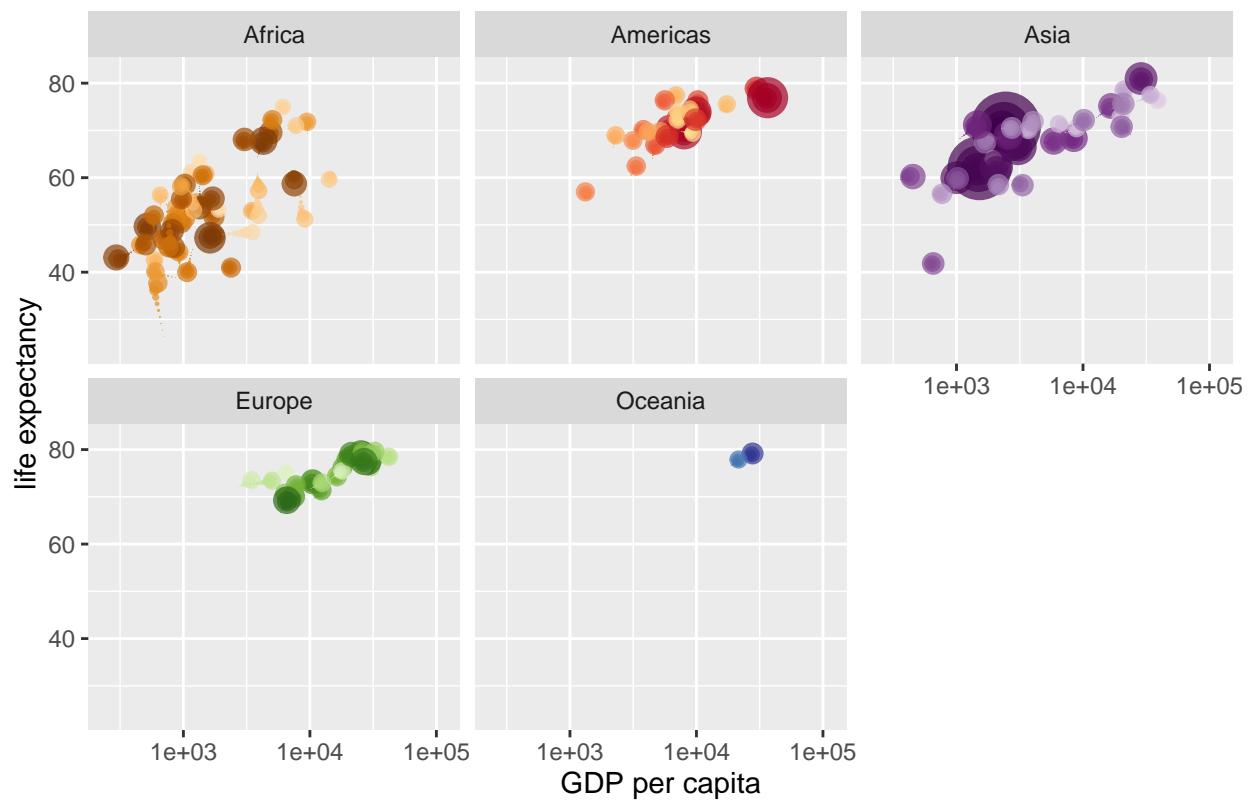
Year: 1997



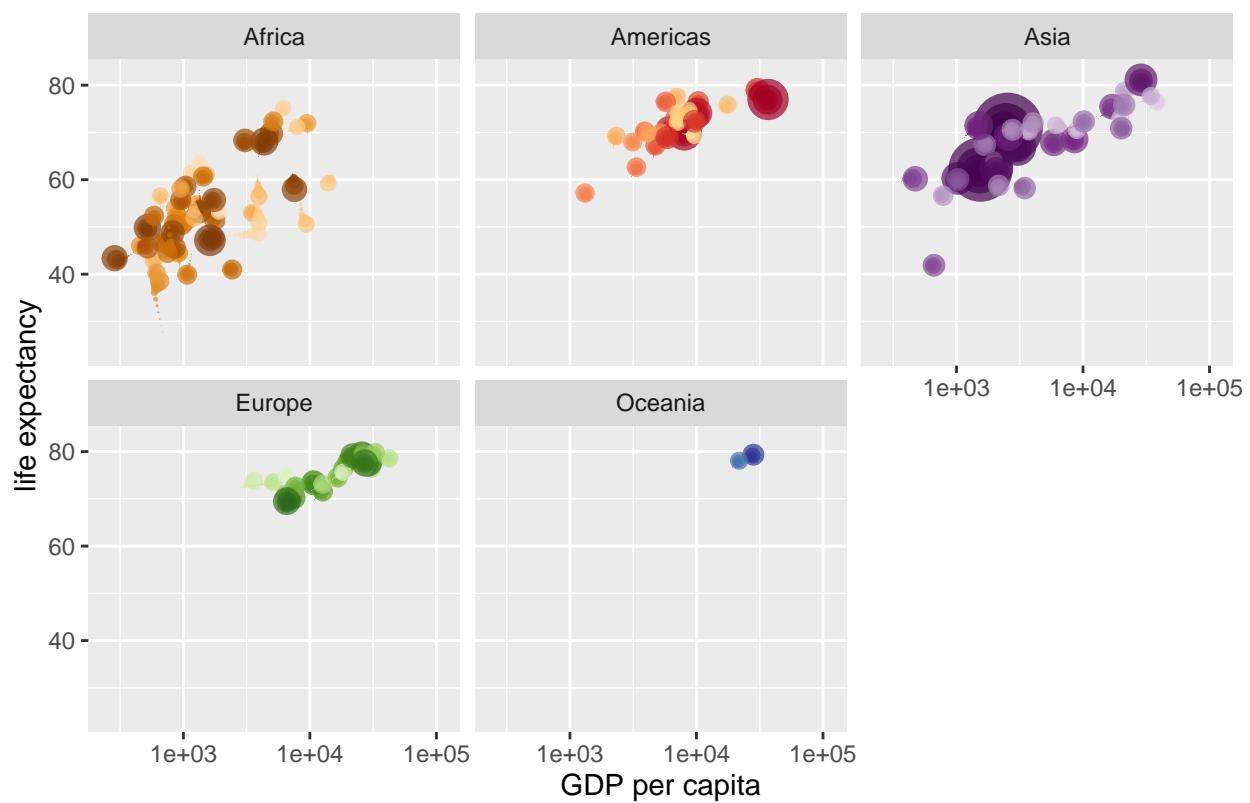
Year: 1998



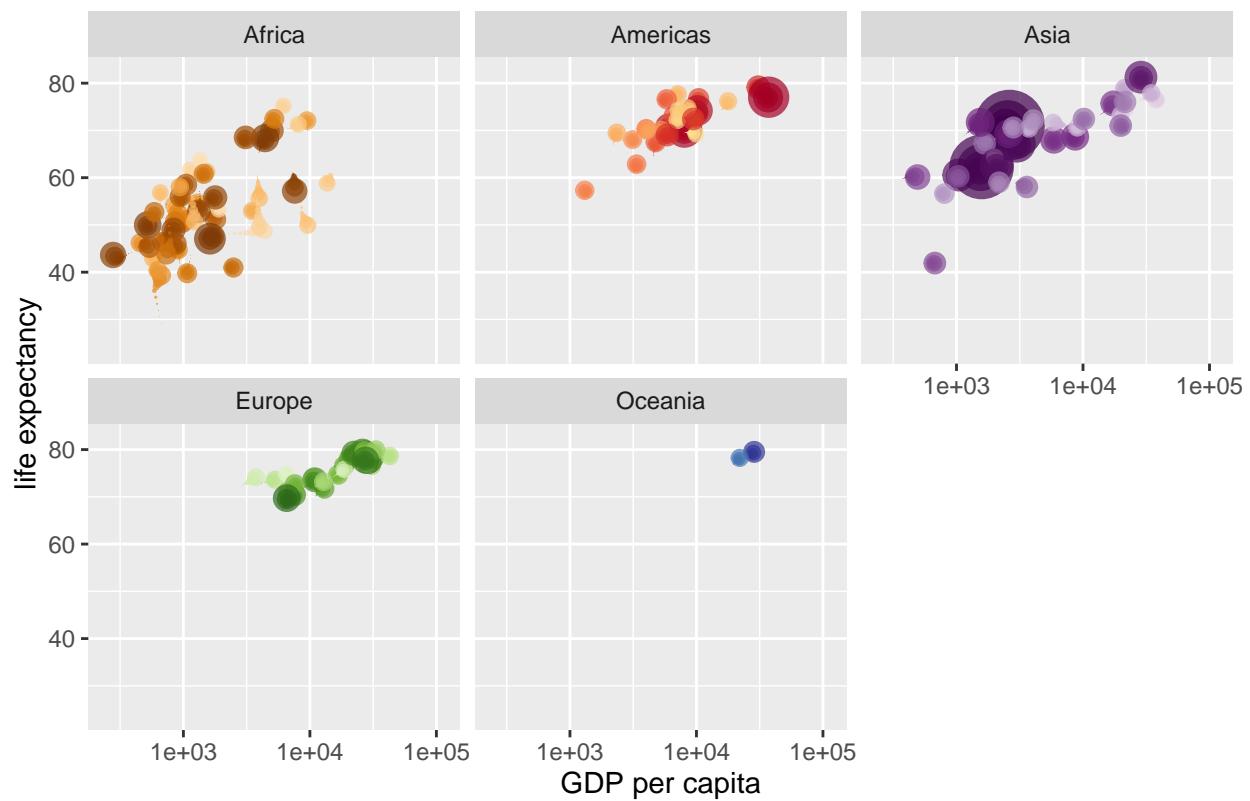
Year: 1998



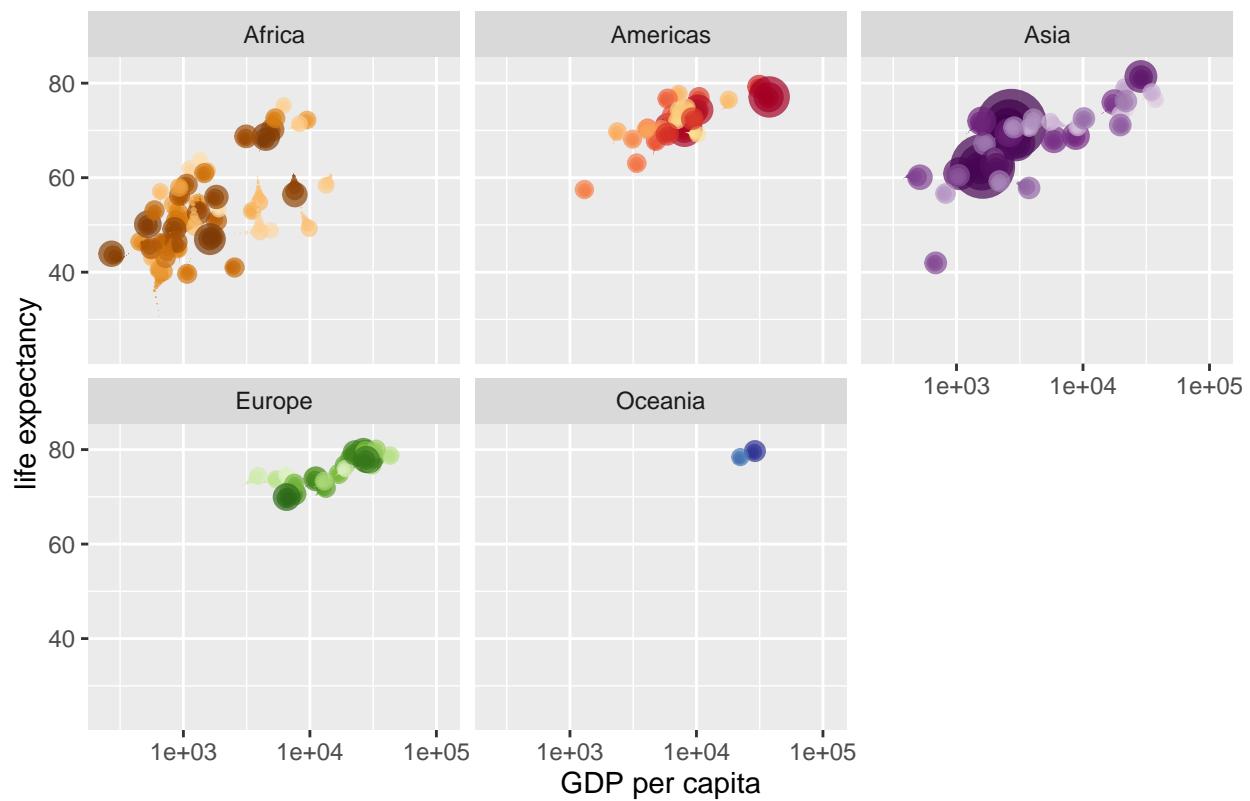
Year: 1999



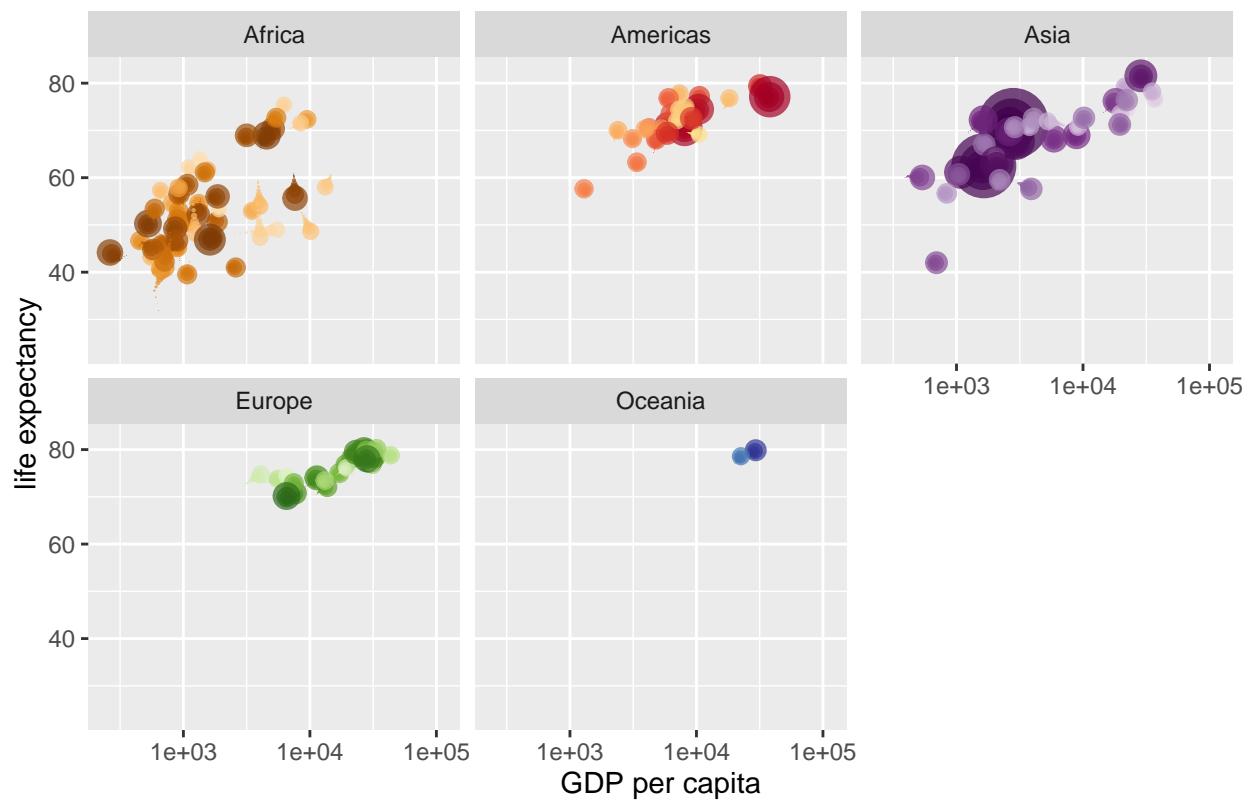
Year: 1999



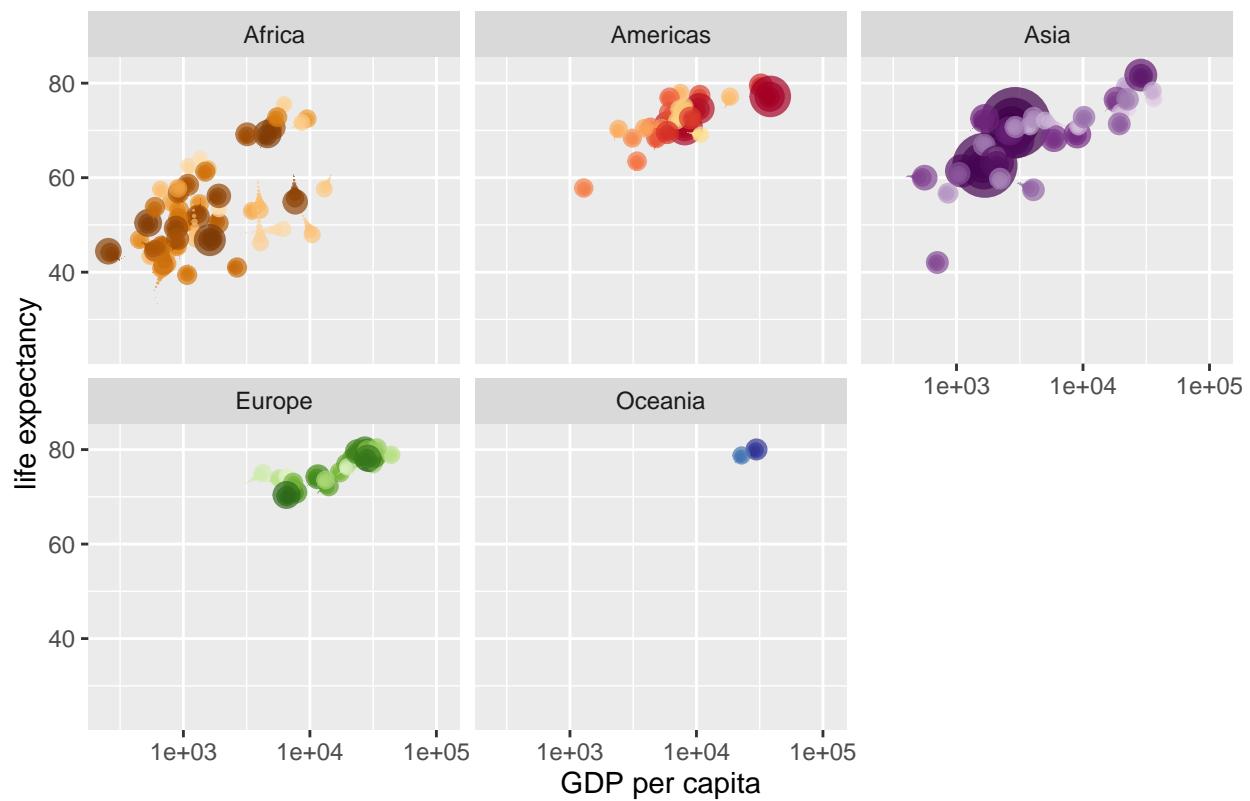
Year: 2000



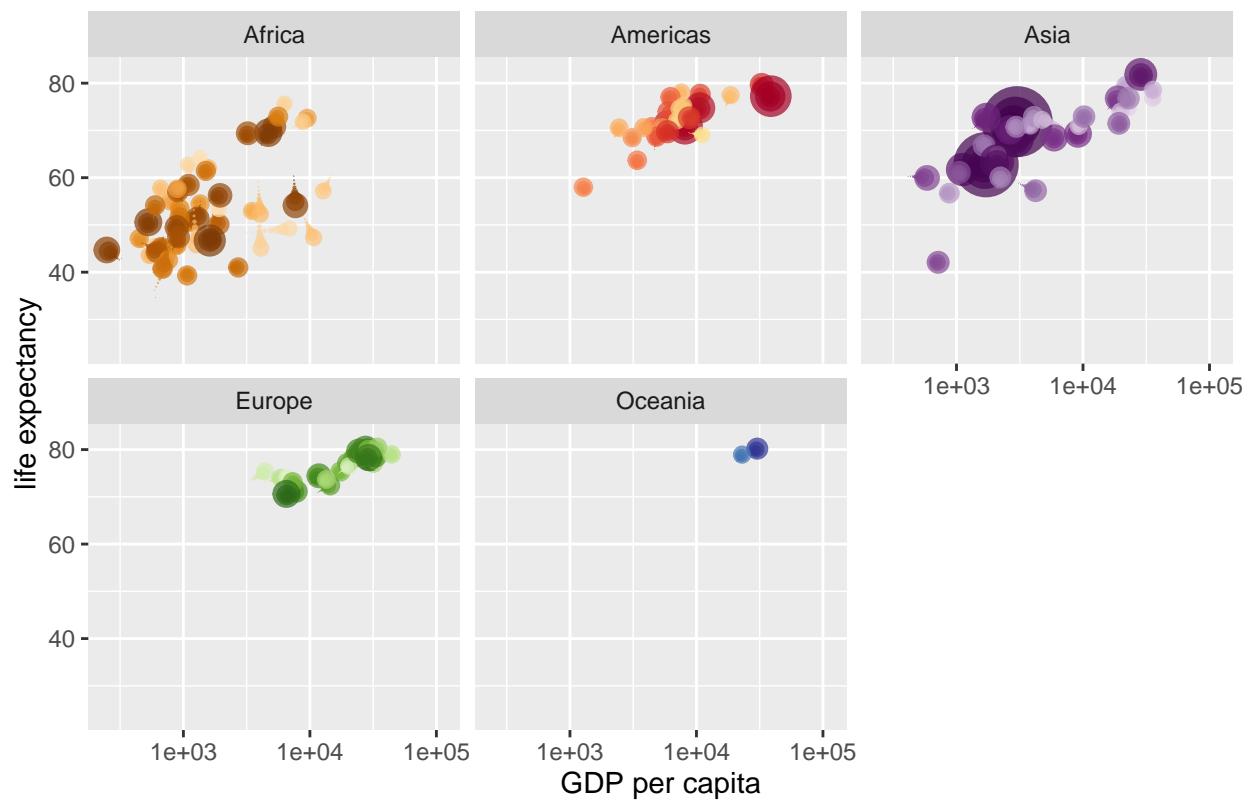
Year: 2000



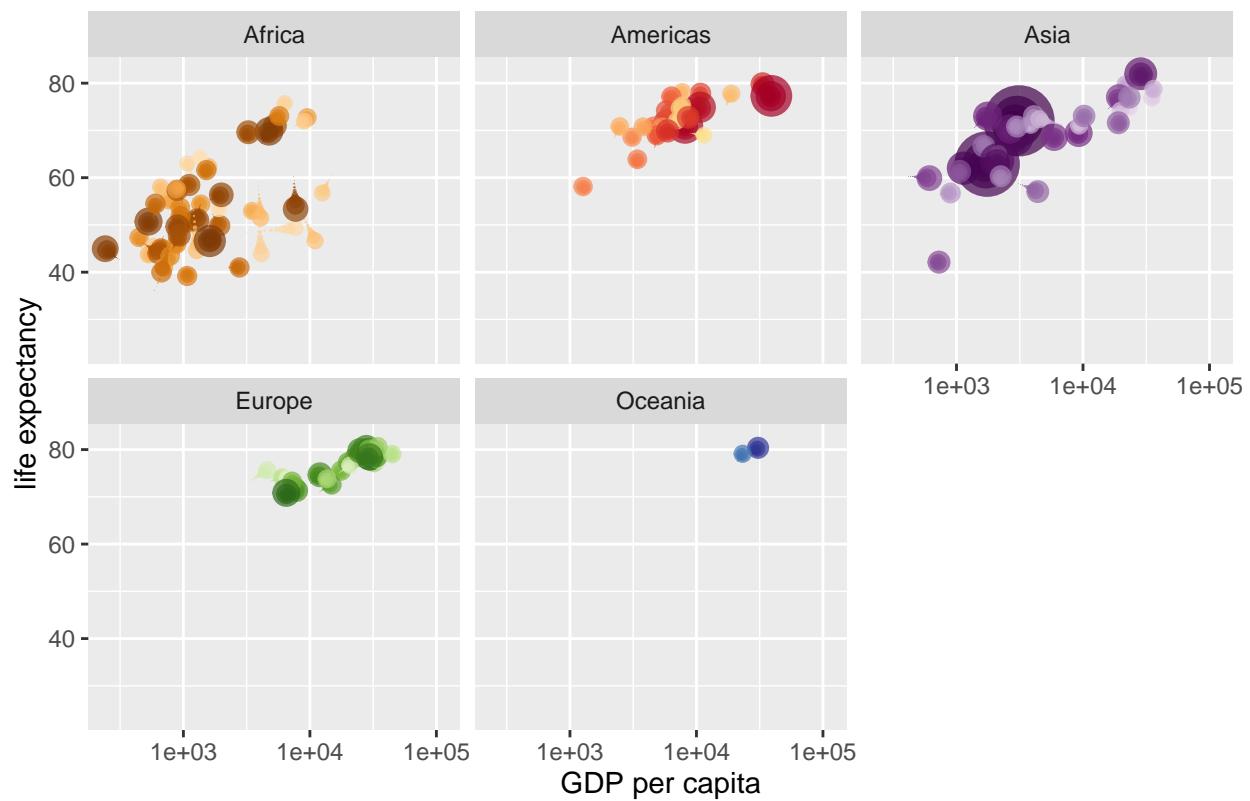
Year: 2001



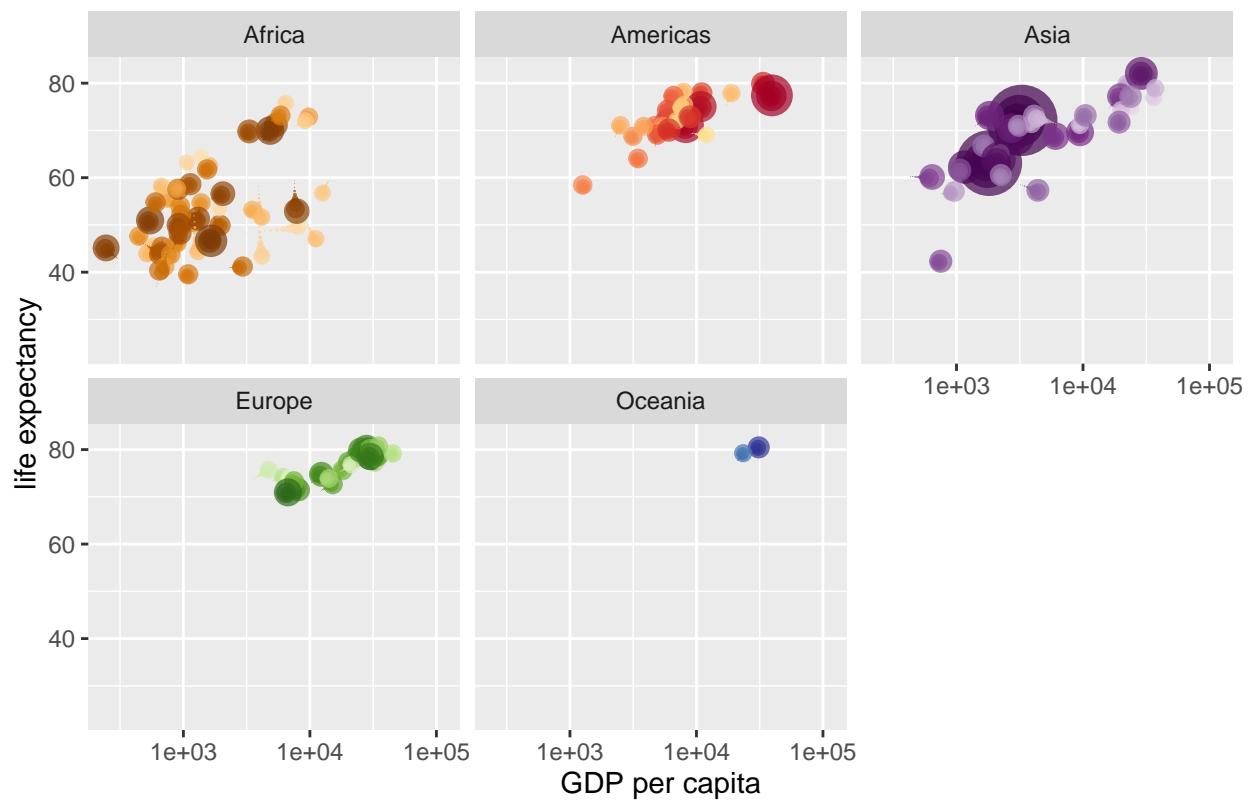
Year: 2001



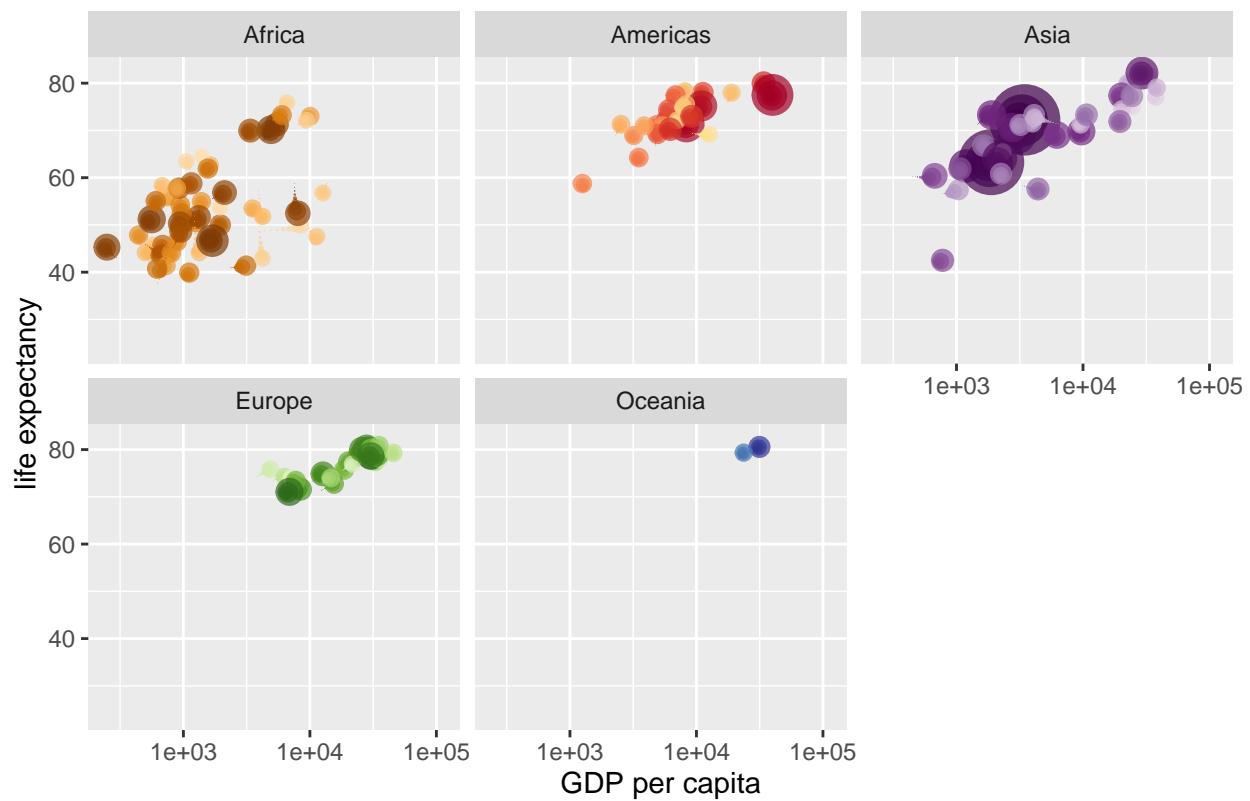
Year: 2002



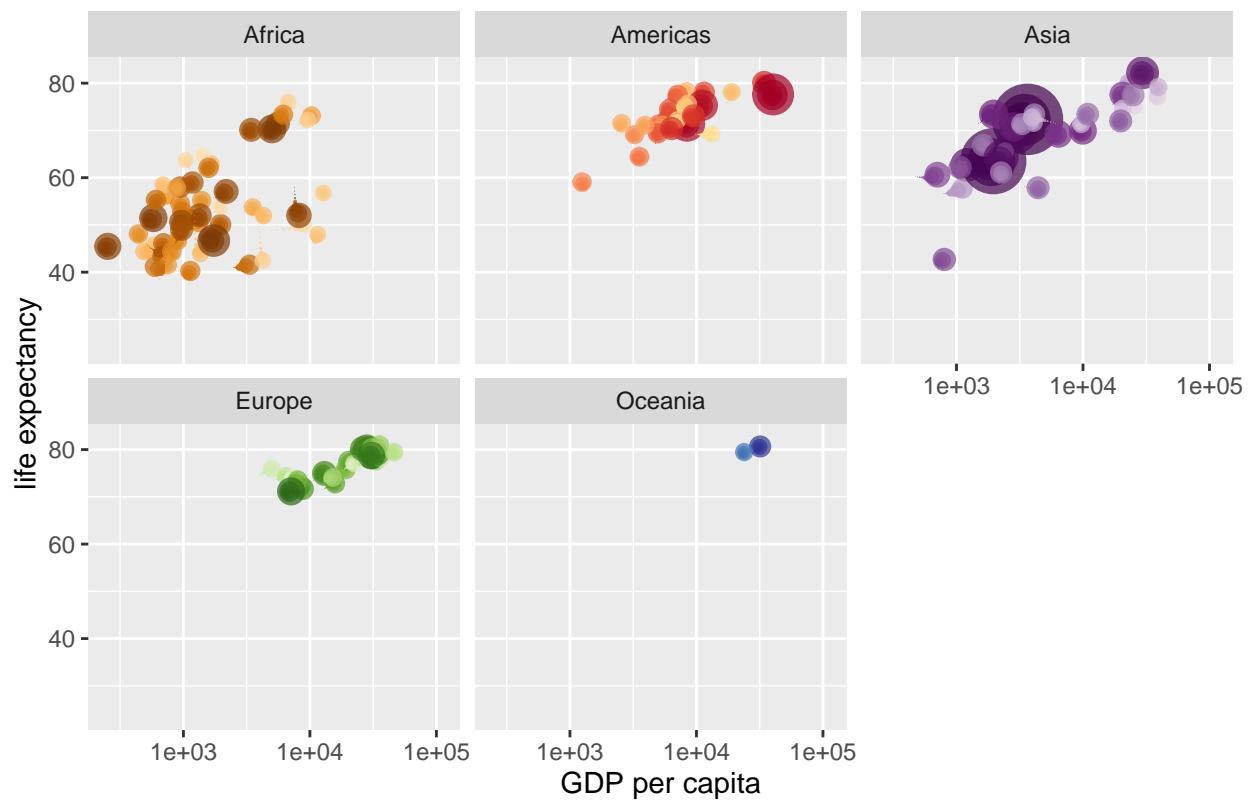
Year: 2003



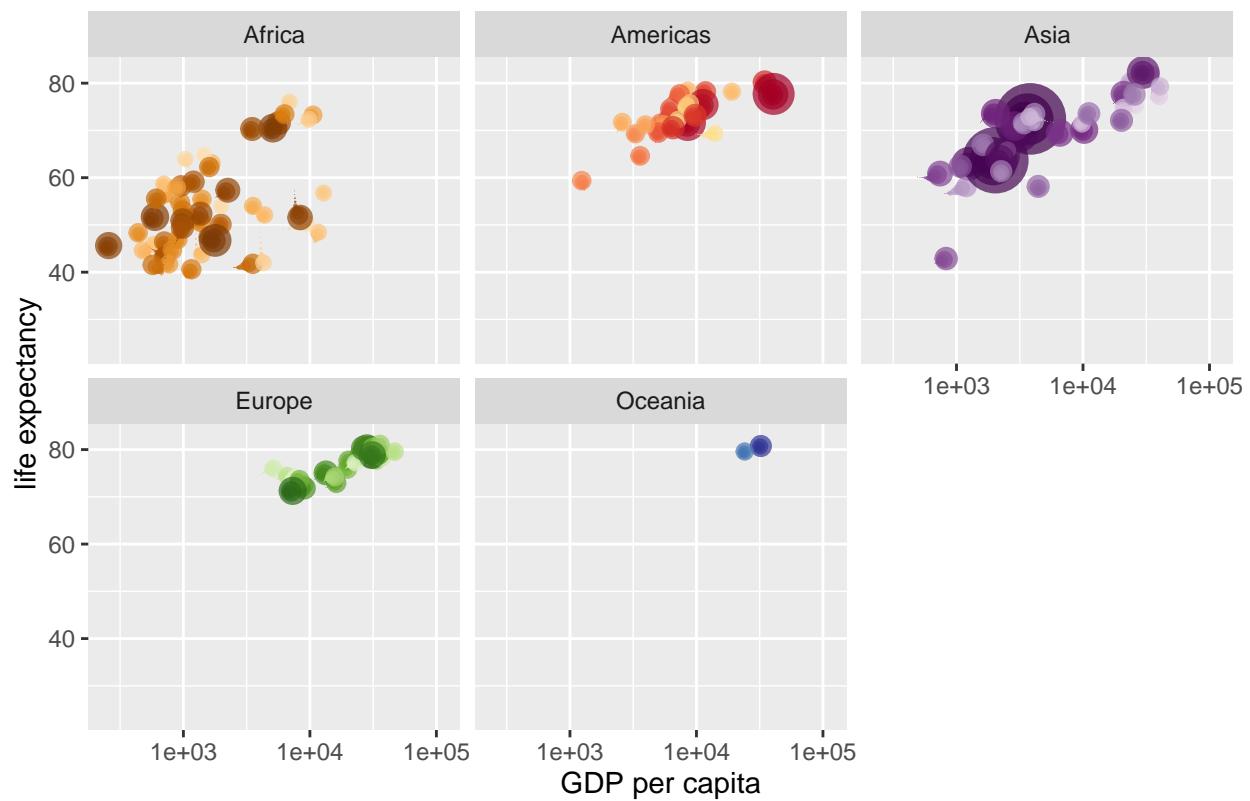
Year: 2003



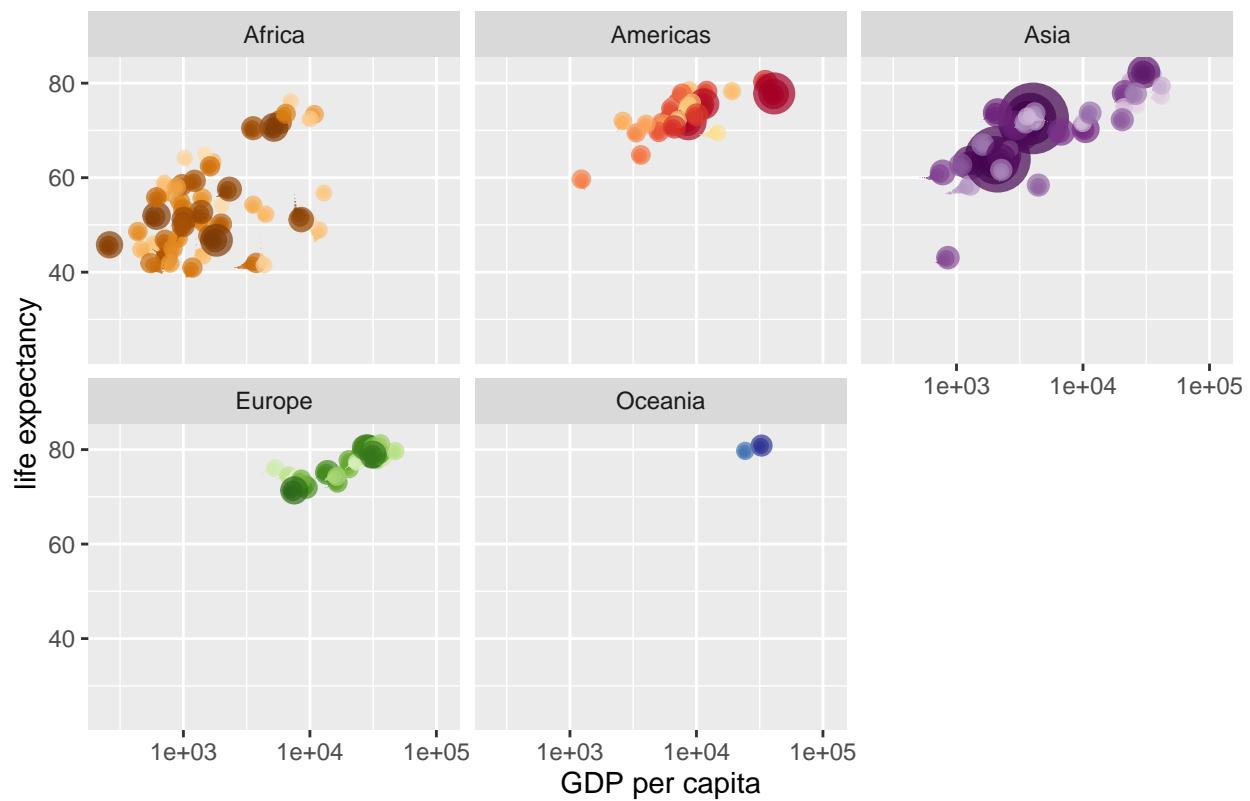
Year: 2004



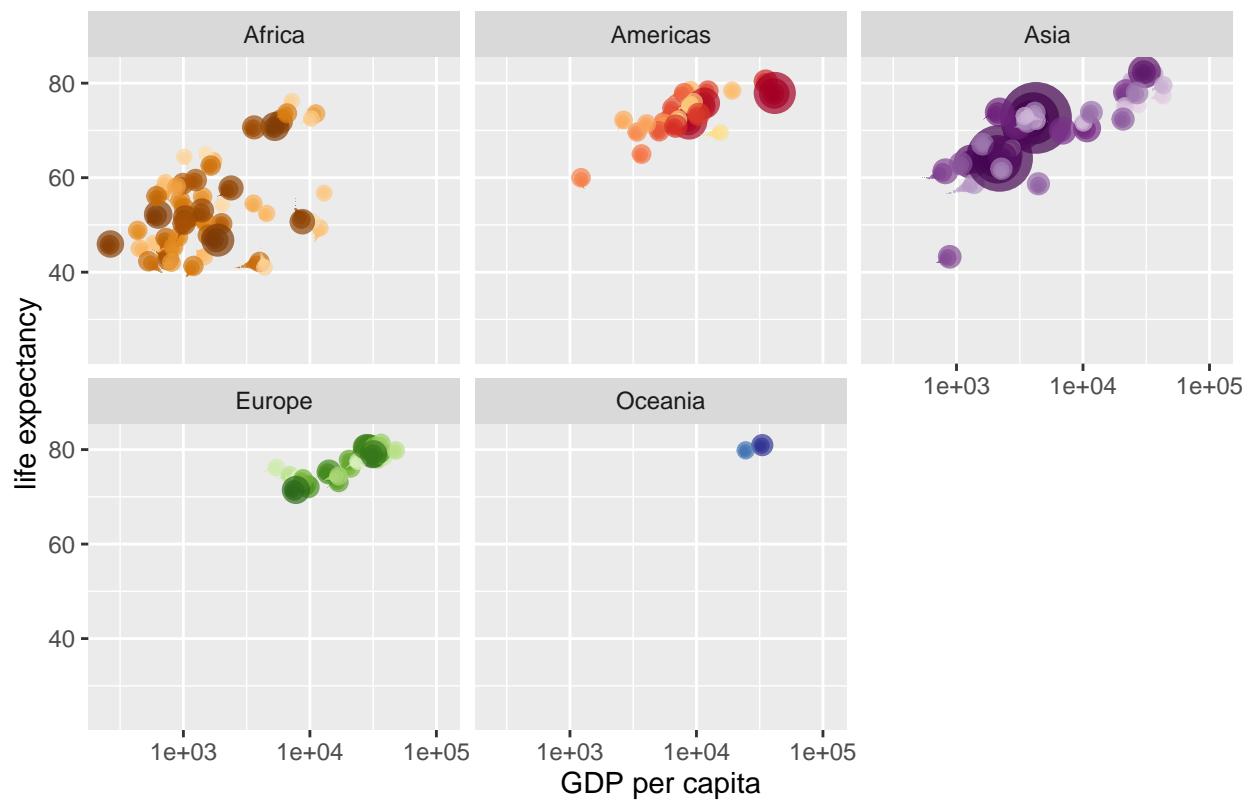
Year: 2004



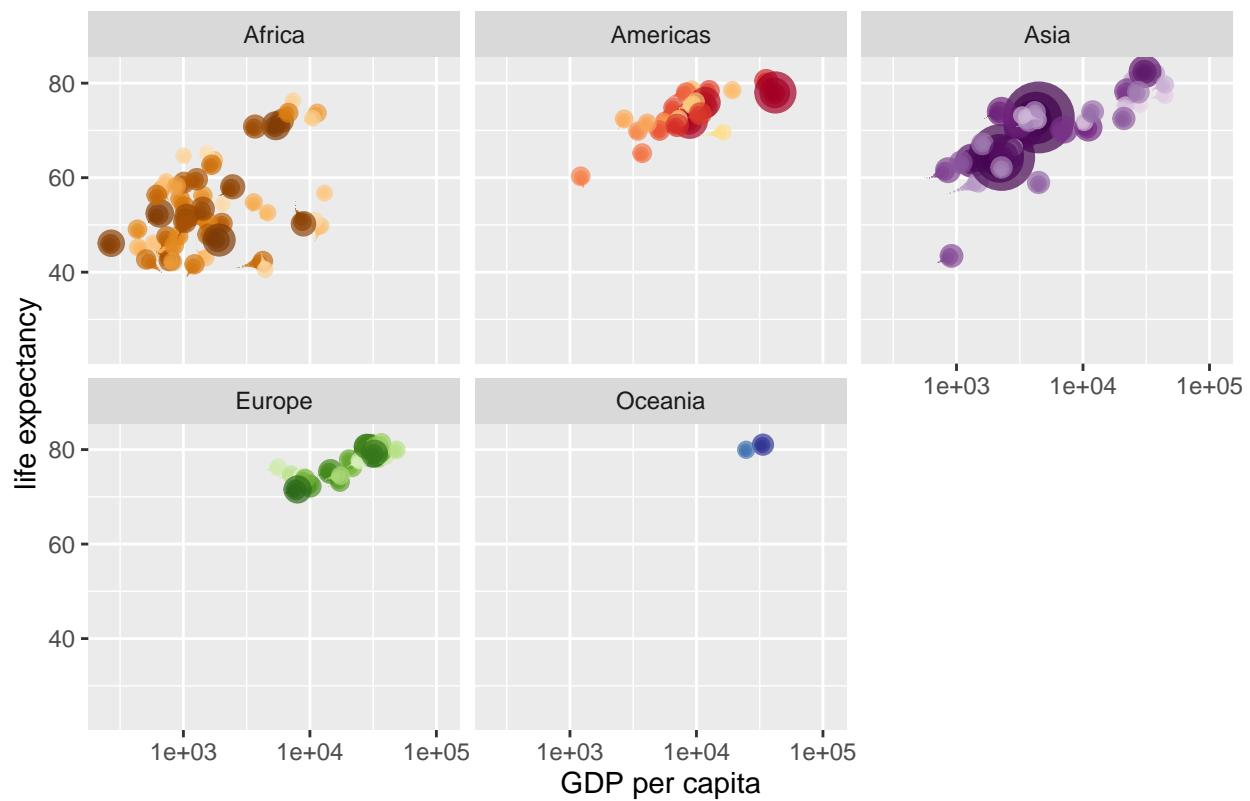
Year: 2005



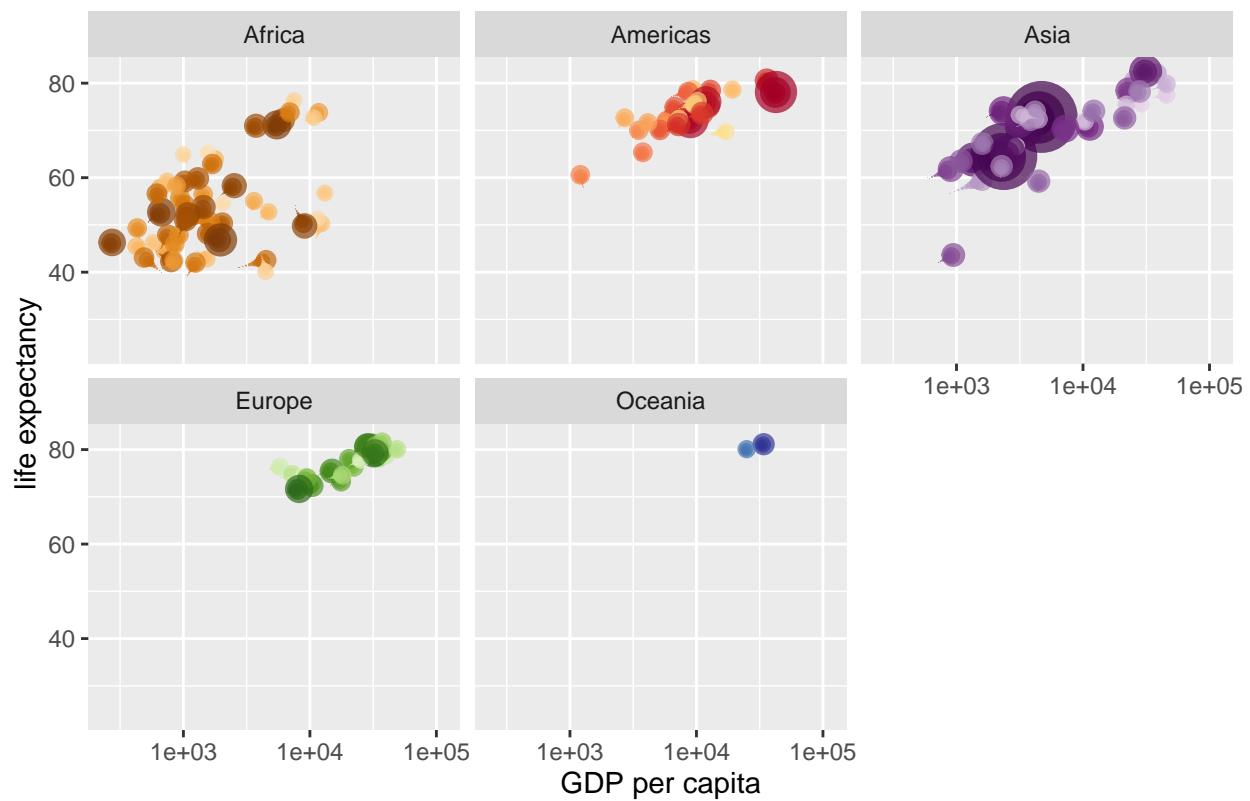
Year: 2005



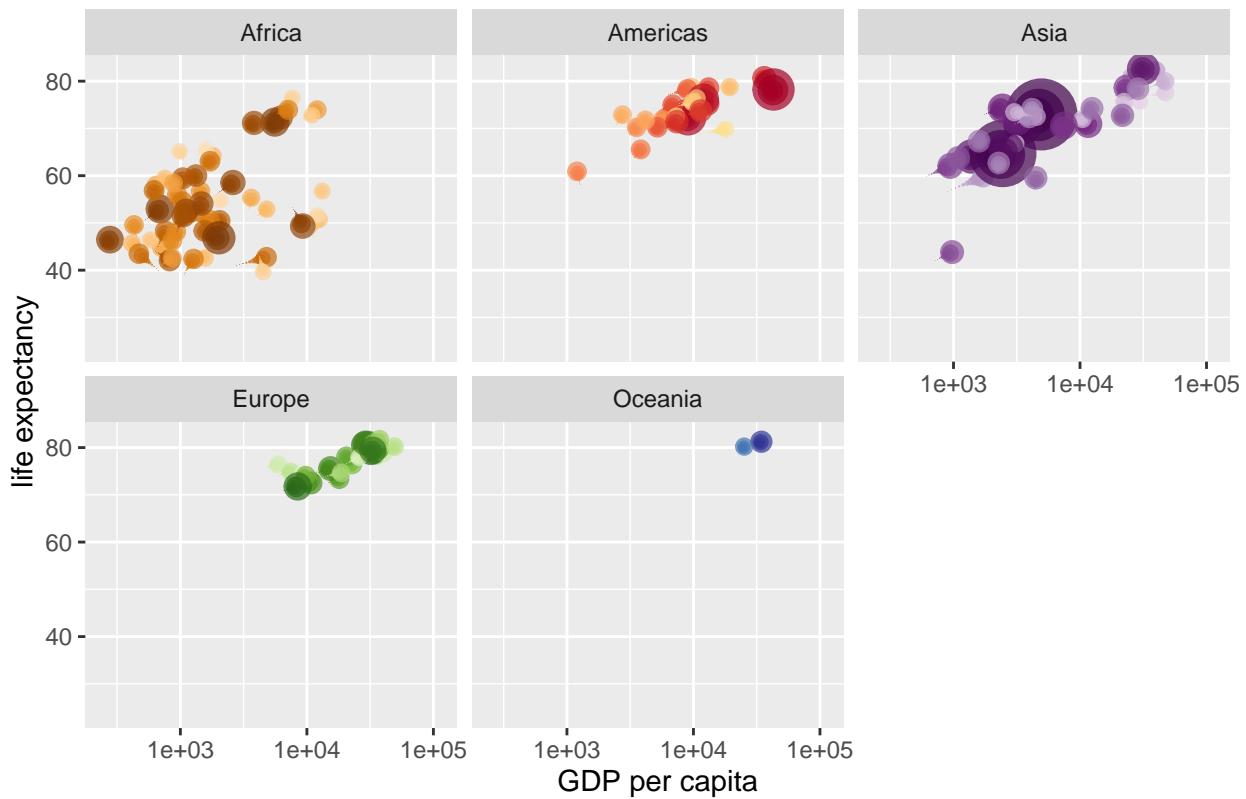
Year: 2006



Year: 2006



Year: 2007



```
# Combining plots
```

```
# Install, then load patchwork:
```

```
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'
```

