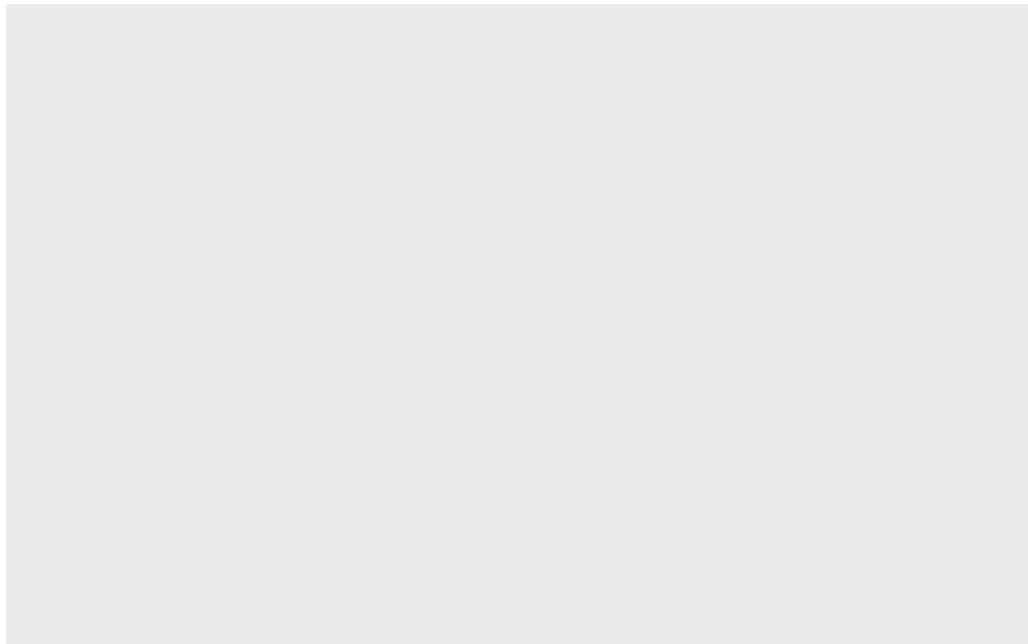


# Class 05: Data Visualization with GGPLOT

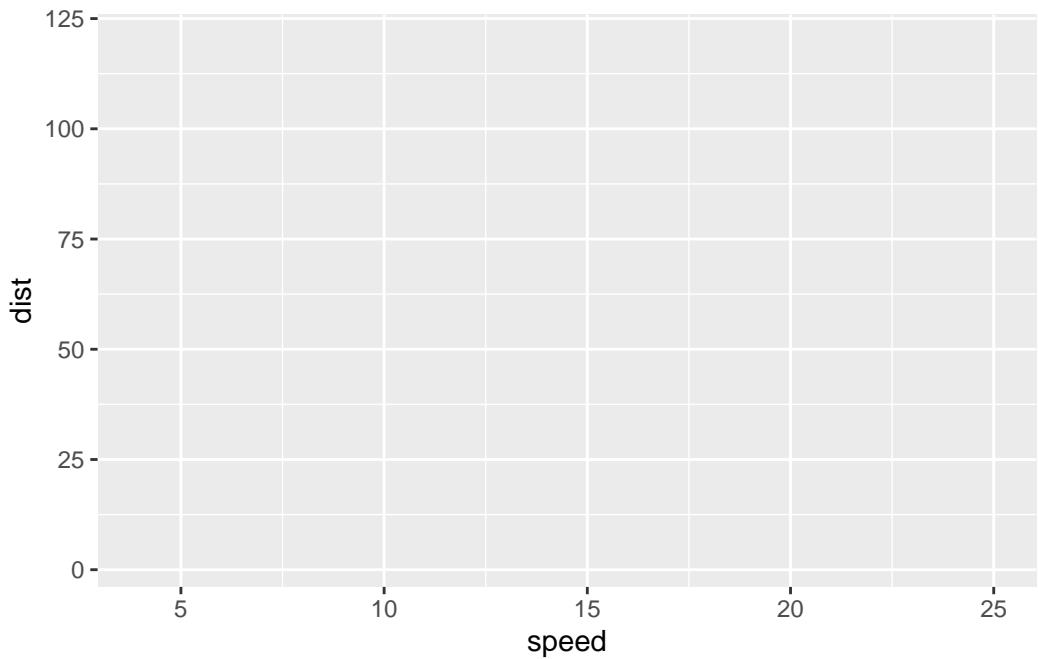
Anh Tran

Q1: All of the above Q2: FALSE Q3: Network graphs Q4: ggplot2 is the only way to create plots in R Q5: geom\_point()

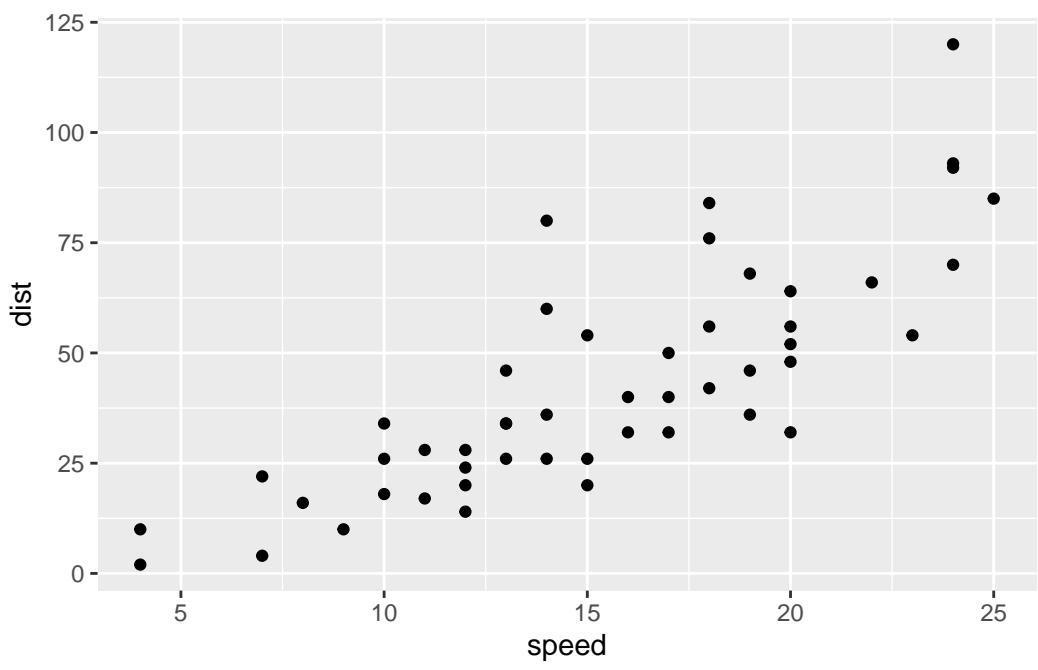
```
library(ggplot2)  
ggplot(cars)
```



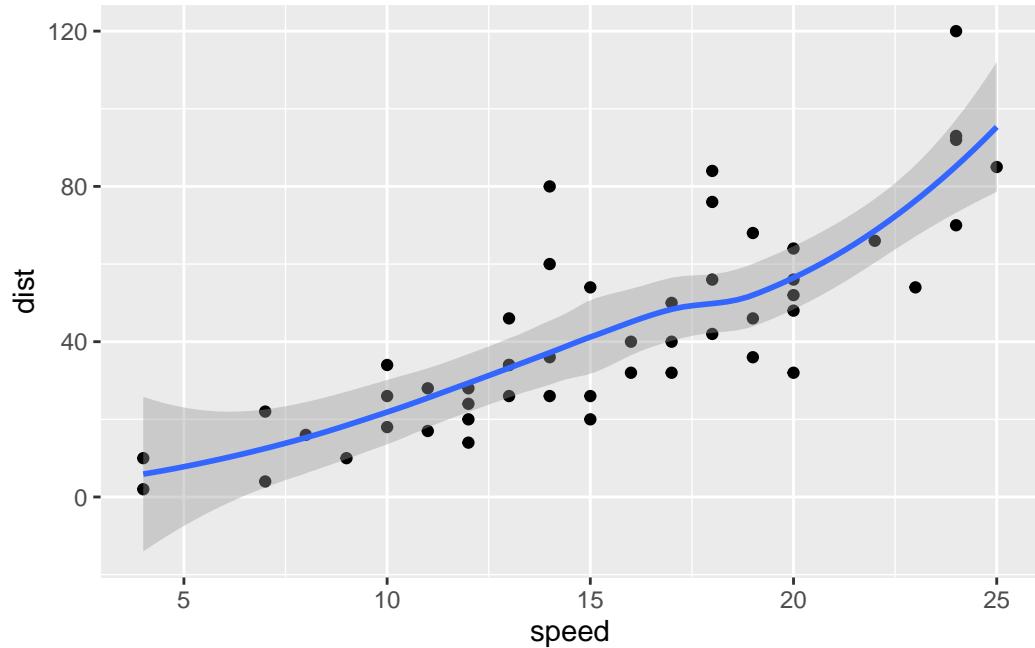
```
ggplot(cars) + aes(x=speed, y=dist)
```



```
ggplot(cars) + aes(x=speed, y=dist) + geom_point()
```

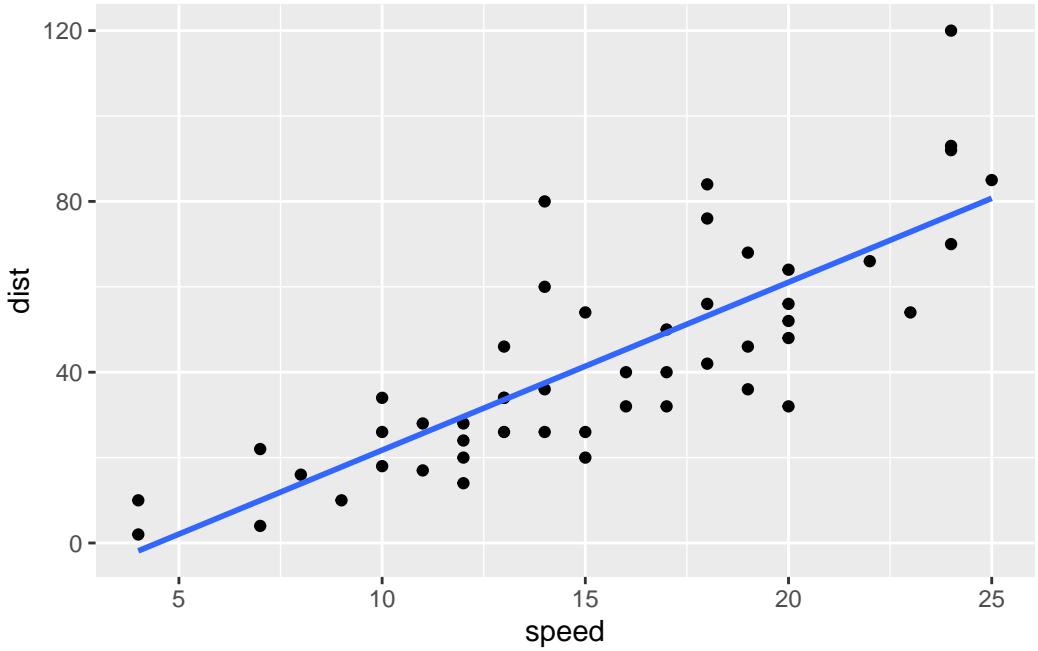


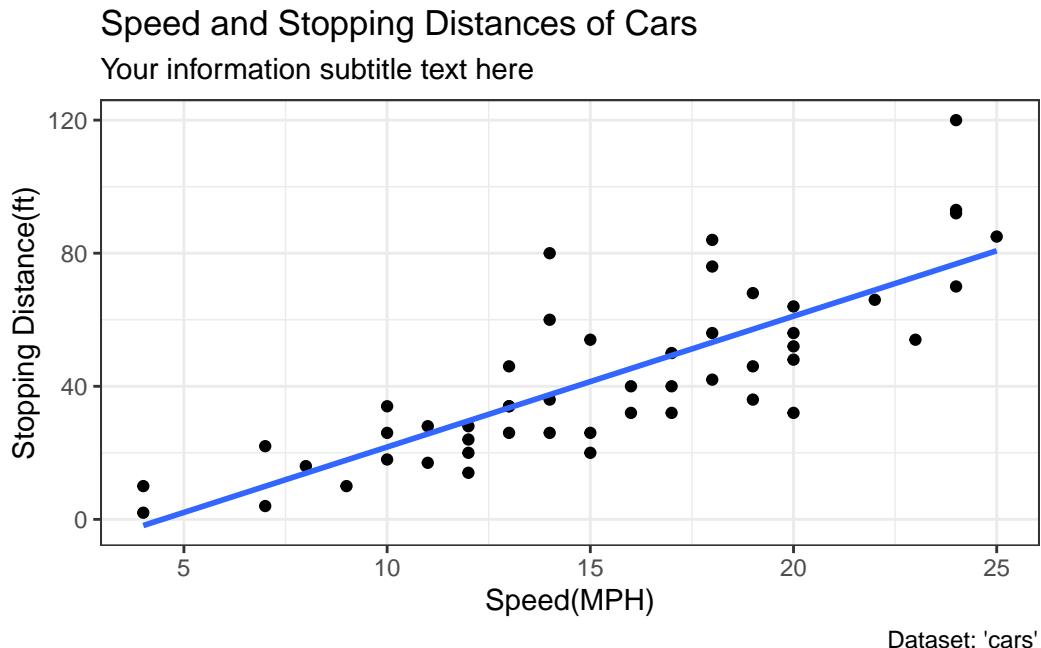
```
ggplot(cars) + aes(x=speed, y=dist) + geom_point() + geom_smooth()  
  
`geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Argue with `geom_smooth()` to add a straight line from a linear model without the shaded standard error region

```
ggplot(cars) + aes(x=speed, y=dist) + geom_point() + geom_smooth(method="lm", se=FALSE)  
  
`geom_smooth()` using formula 'y ~ x'
```





Adding more plot aesthetics through (aes)

```
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

Q6: Use the nrow() function to find out how many genes are in this dataset. What is your answer?

```
nrow(genes)
```

[1] 5196

Q7: Use the colnames() function and the ncol() function on the genes data frame to find out what the column names are (we will need these later) and how many columns there are. How many columns did you find?

```
colnames(genes)
```

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

```
ncol(genes)
```

```
[1] 4
```

Q8: Use the table() function on the State column of this data.frame to find out how many ‘up’ regulated genes there are. What is your answer?

```
table(genes$State)
```

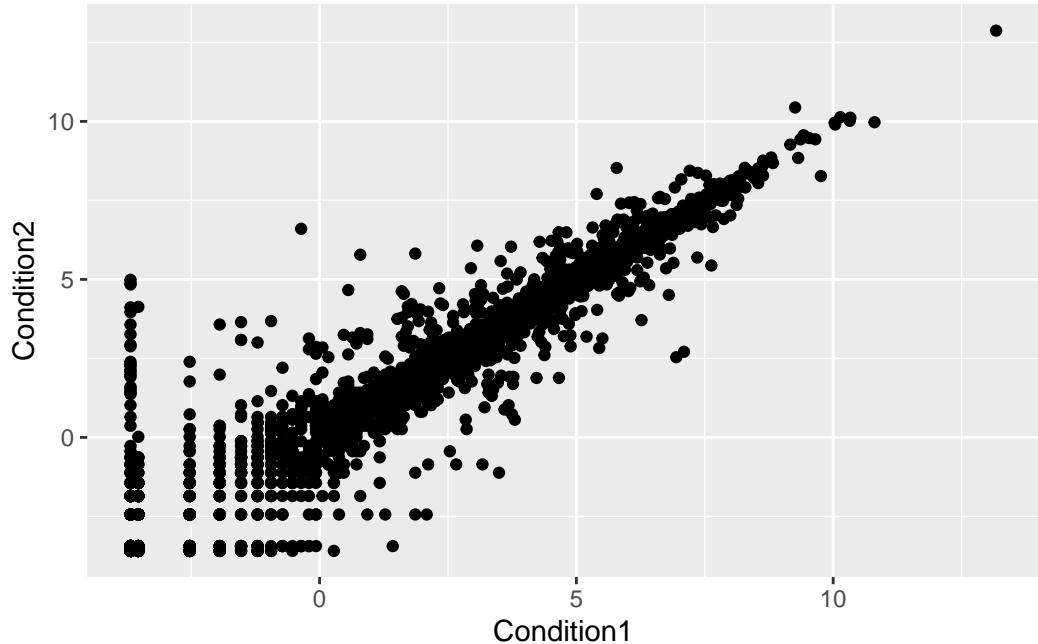
	down	unchanging	up
	72	4997	127

Q9: Using your values above and 2 significant figures. What fraction of total genes is up-regulated in this dataset?

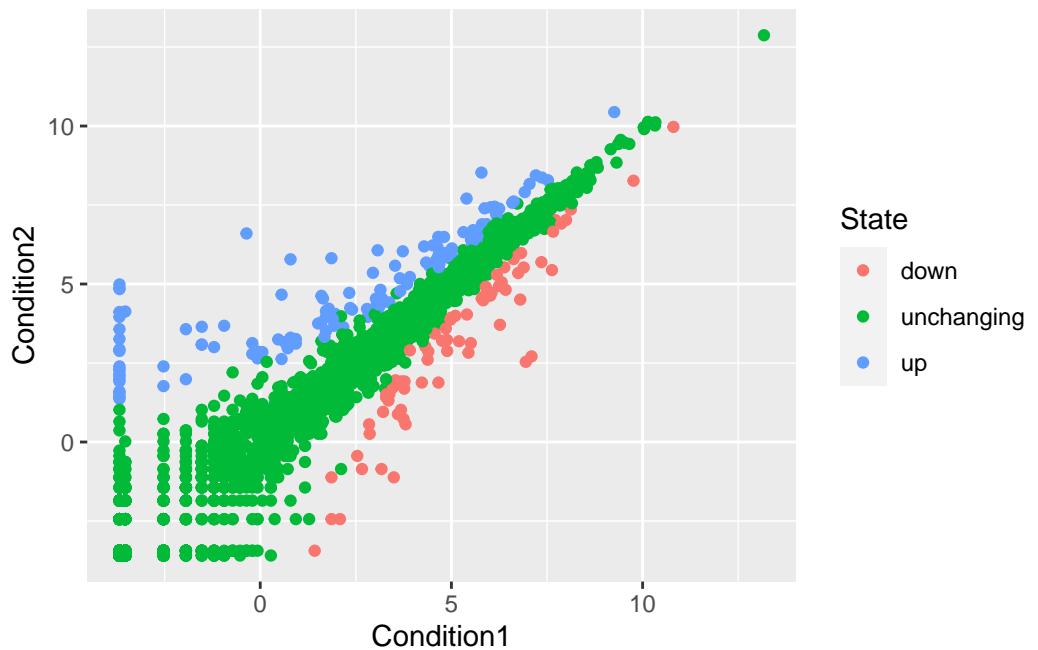
```
round(table(genes$State)/nrow(genes) * 100, 2)
```

	down	unchanging	up
	1.39	96.17	2.44

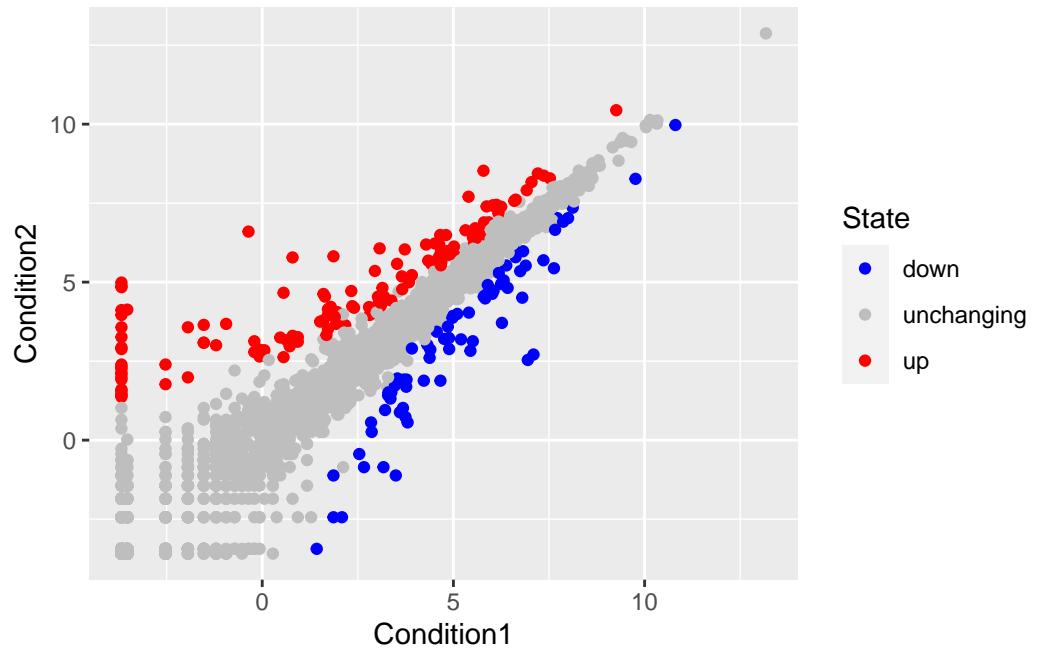
```
ggplot(genes) + aes(x=Condition1, y=Condition2) + geom_point()
```



```
p <- ggplot(genes) + aes(x=Condition1, y=Condition2, col=State) + geom_point()  
p
```

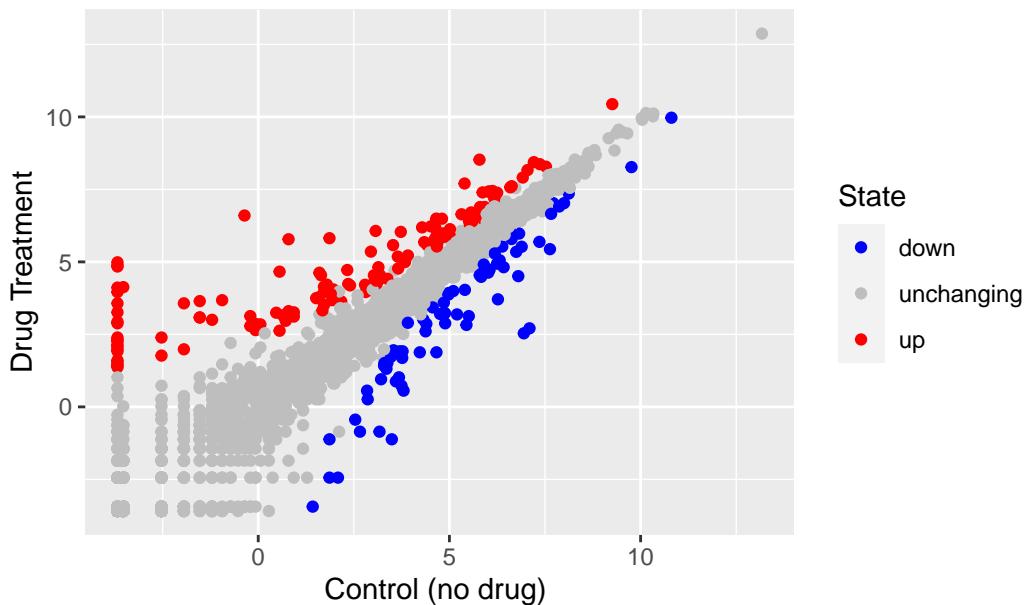


```
p + scale_colour_manual(values=c("blue","gray","red") )
```



```
p + scale_colour_manual(values=c("blue","gray","red")) + labs(title="Gene Expression Change")
```

## Gene Expression Changes Upon Drug Treatment



```
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.csv"
gapminder <- read.delim(url)
```

```
library(dplyr)
```

```
Attaching package: 'dplyr'
```

```
The following objects are masked from 'package:stats':
```

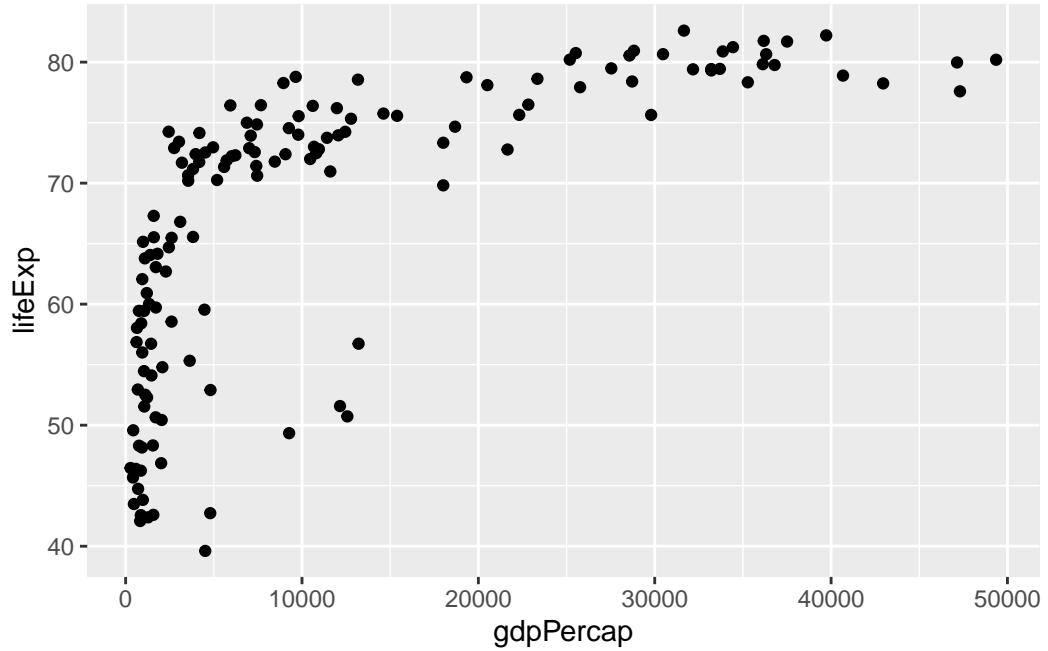
```
filter, lag
```

```
The following objects are masked from 'package:base':
```

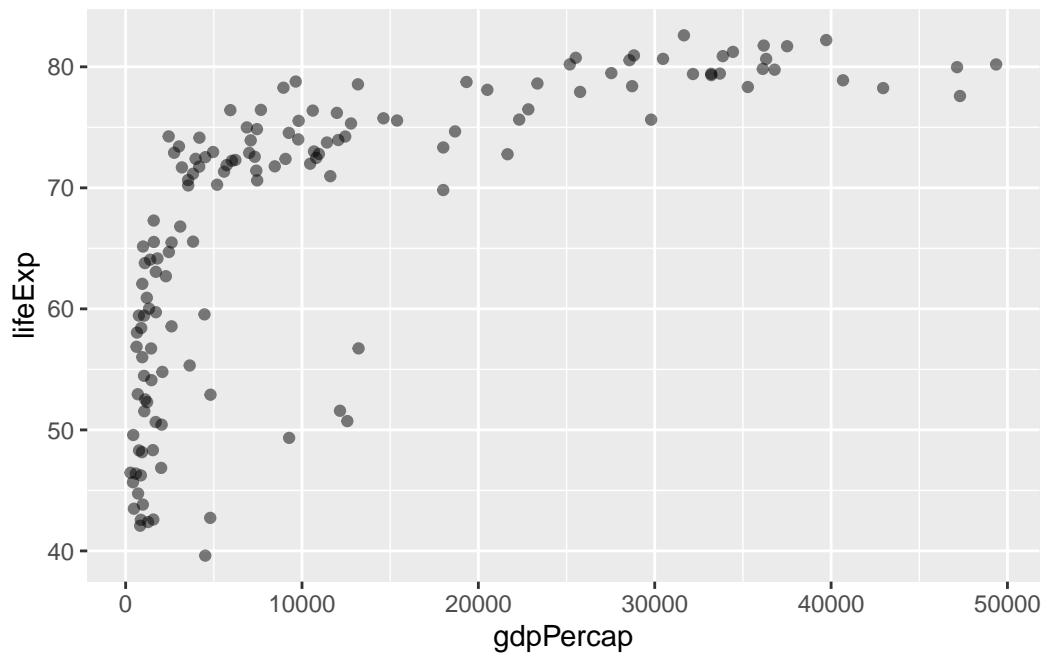
```
intersect, setdiff, setequal, union
```

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

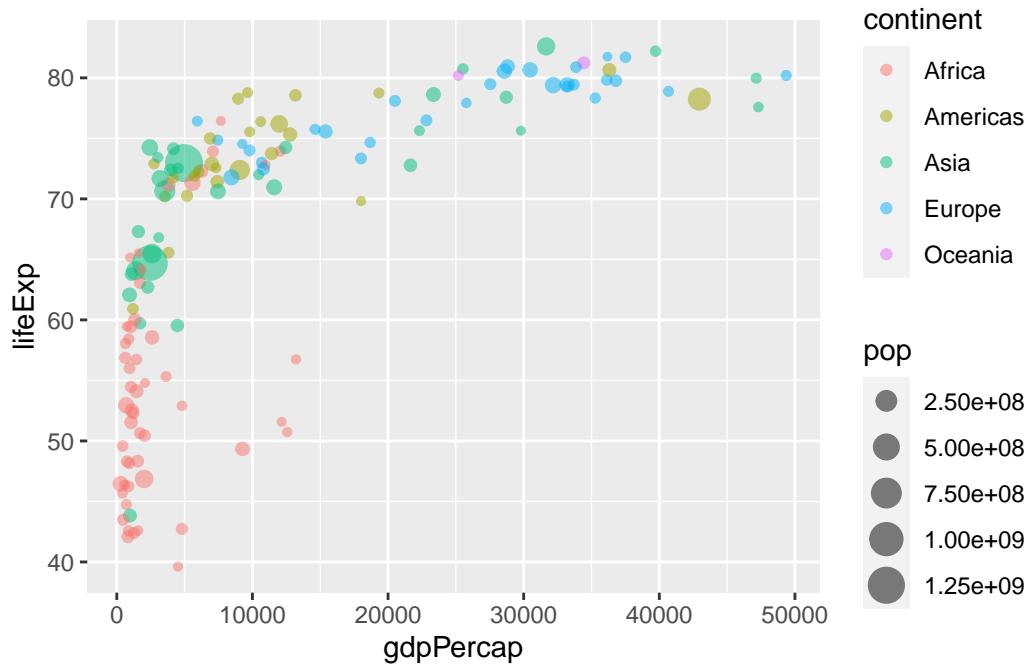
```
ggplot(gapminder_2007) + aes(x=gdpPercap, y=lifeExp) + geom_point()
```



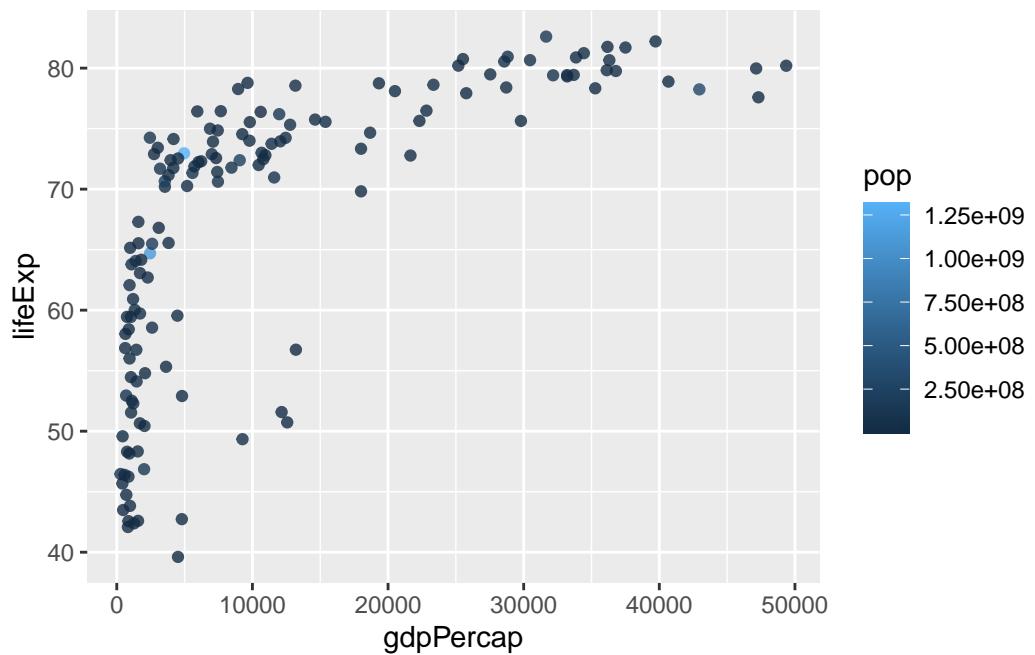
```
ggplot(gapminder_2007) + aes(x=gdpPercap, y=lifeExp) + geom_point(alpha=0.5)
```



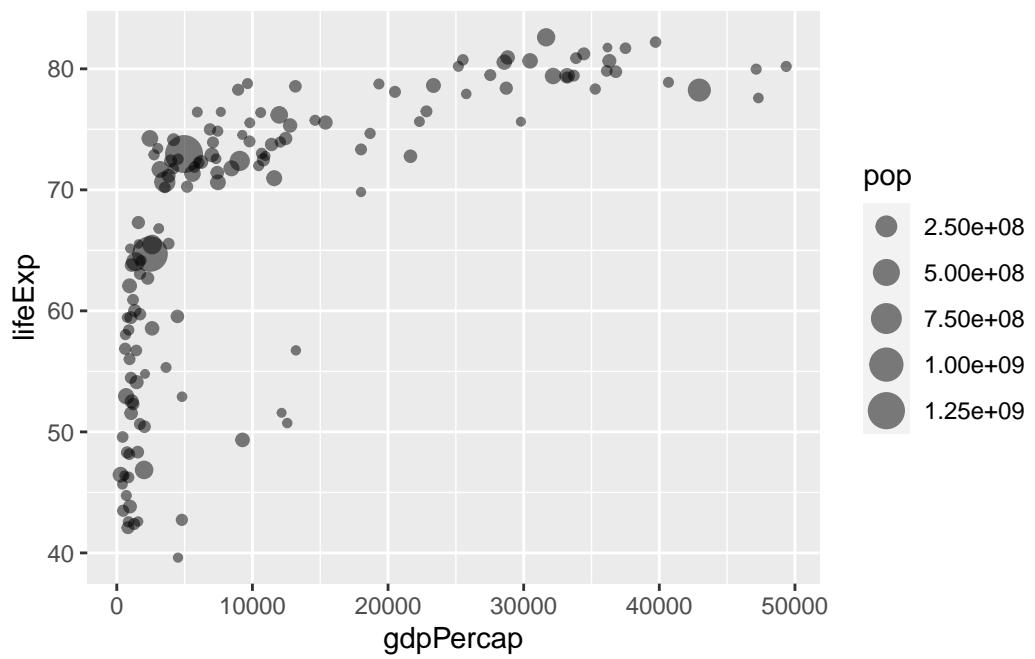
```
ggplot(gapminder_2007) + aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) + geom_poi
```



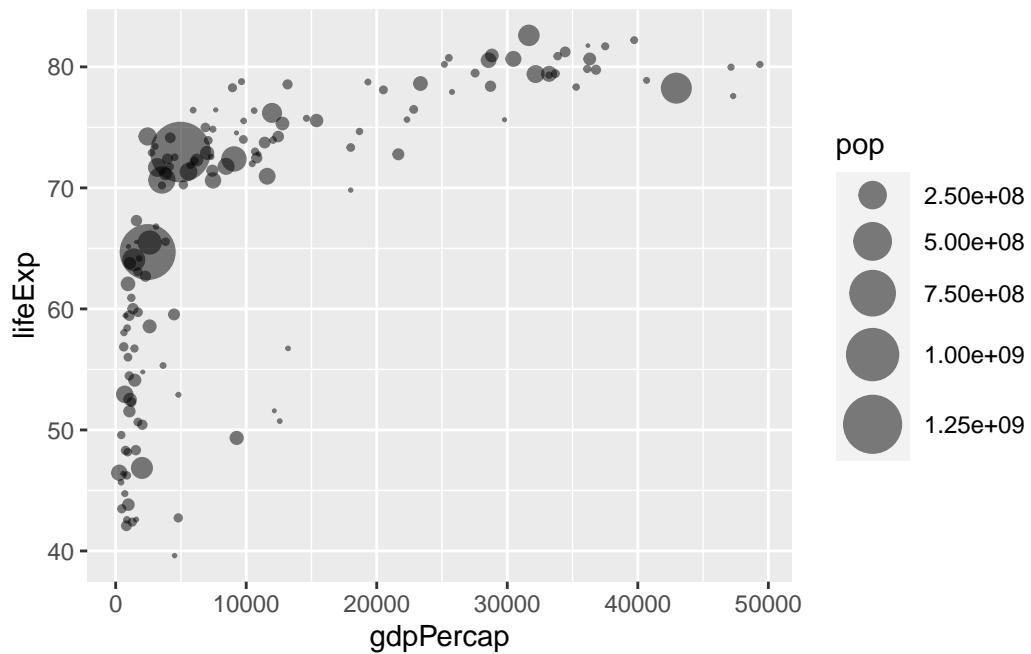
```
ggplot(gapminder_2007) + aes(x = gdpPercap, y = lifeExp, color = pop) + geom_point(alpha=0
```



```
ggplot(gapminder_2007) + aes(x = gdpPercap, y = lifeExp, size = pop) + geom_point(alpha=0.5)
```

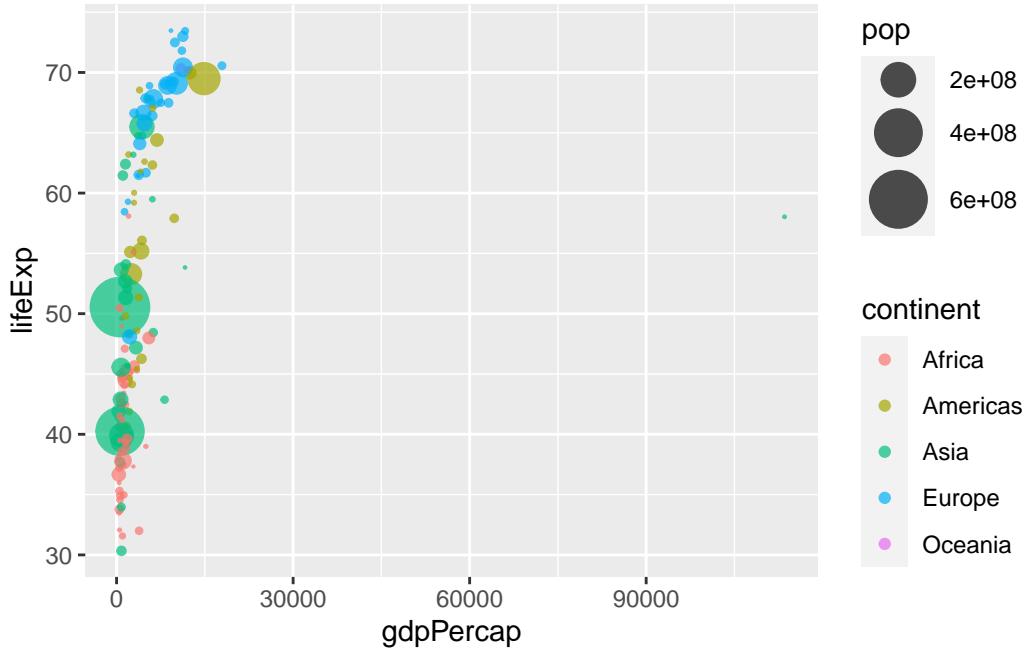


```
ggplot(gapminder_2007) + geom_point(aes(x = gdpPercap, y = lifeExp, size = pop), alpha=0.5)
```



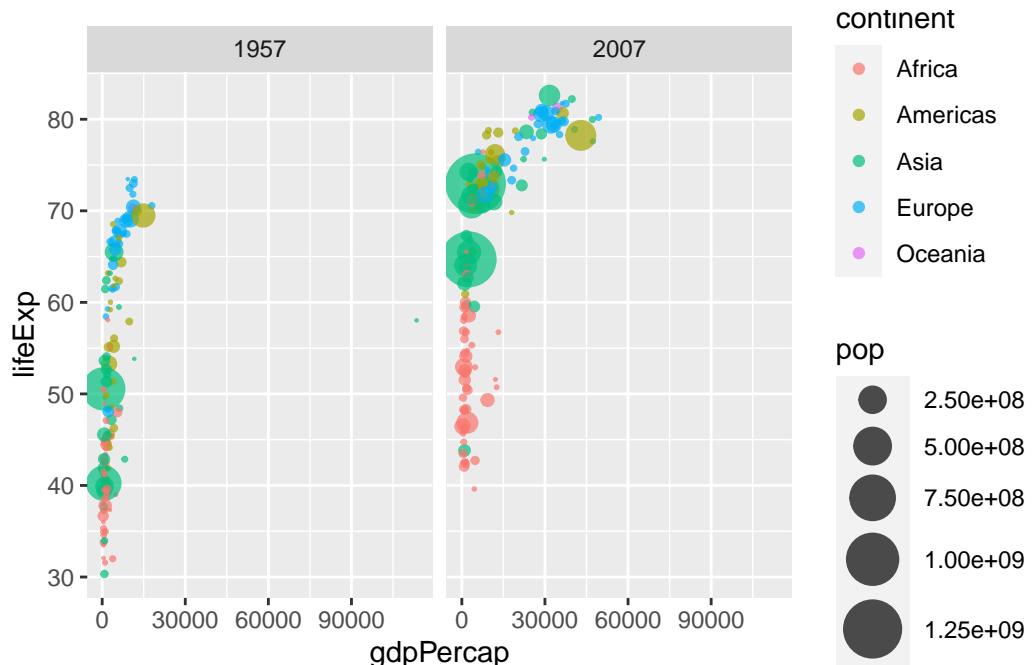
```
gapminder_1957 <- gapminder %>% filter(year==1957)
```

```
ggplot(gapminder_1957) + aes(x = gdpPercap, y = lifeExp, color=continent, size = pop) + ge
```



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

ggplot(gapminder_1957) + geom_point(aes(x = gdpPerCap, y = lifeExp, color=continent, size
```

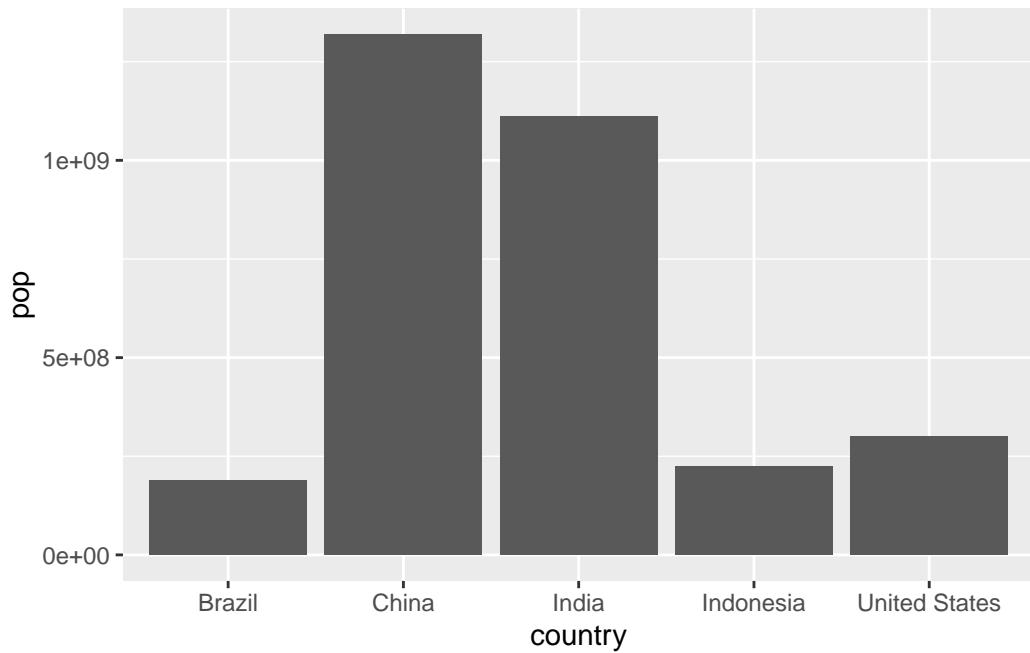


```
gapminder_top5 <- gapminder %>%
  filter(year==2007) %>%
  arrange(desc(pop)) %>%
  top_n(5, pop)

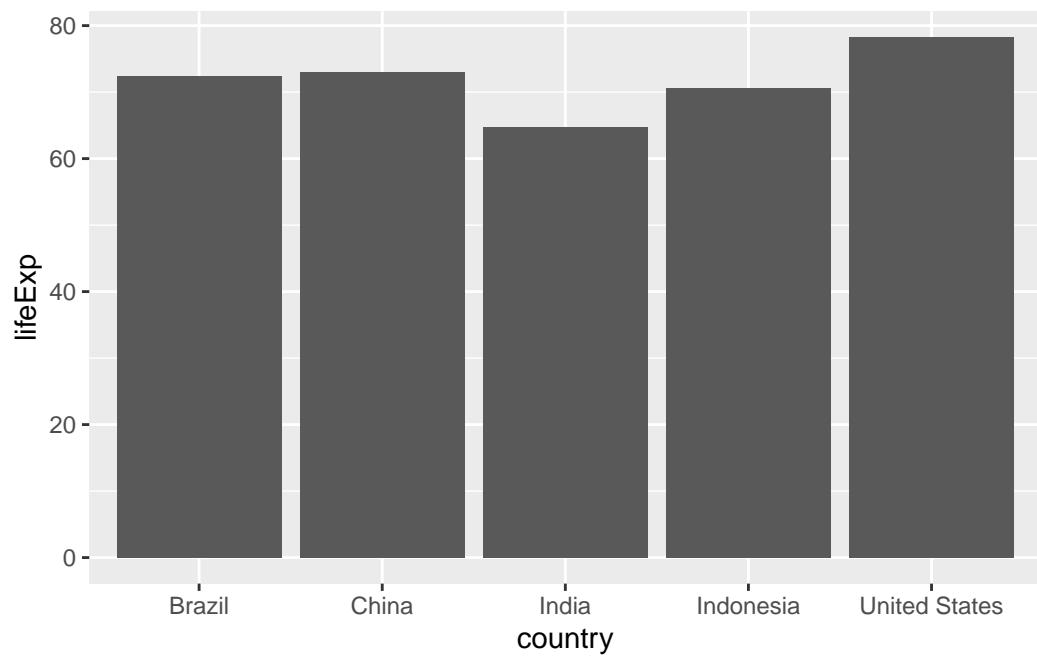
gapminder_top5
```

	country	continent	year	lifeExp	pop	gdpPercap
1	China	Asia	2007	72.961	1318683096	4959.115
2	India	Asia	2007	64.698	1110396331	2452.210
3	United States	Americas	2007	78.242	301139947	42951.653
4	Indonesia	Asia	2007	70.650	223547000	3540.652
5	Brazil	Americas	2007	72.390	190010647	9065.801

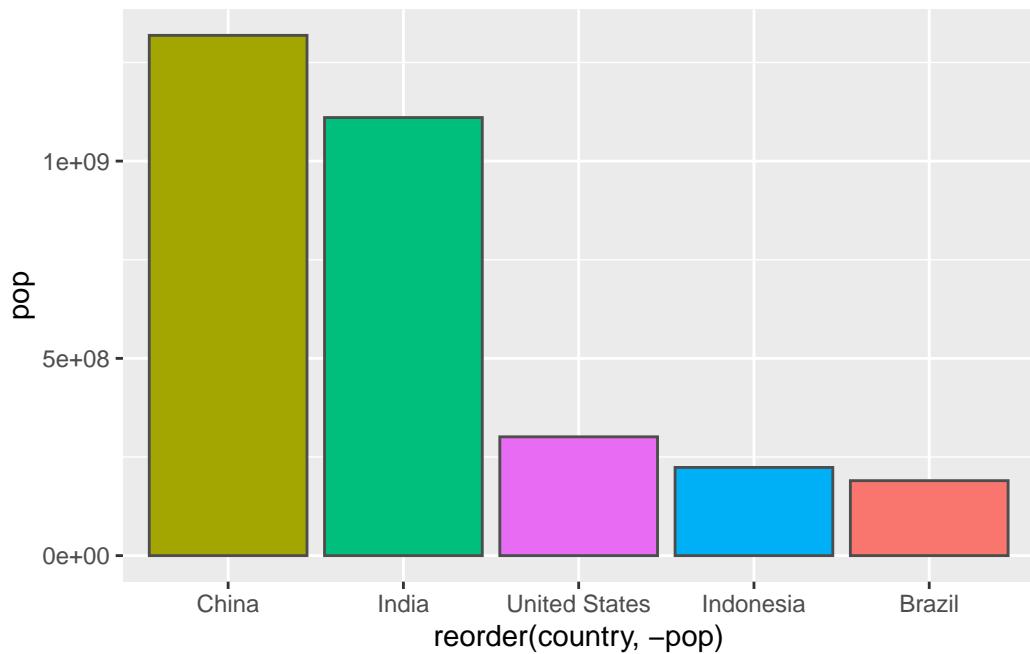
```
ggplot(gapminder_top5) + geom_col(aes(x = country, y = pop))
```



```
ggplot(gapminder_top5) + geom_col(aes(x = country, y = lifeExp))
```



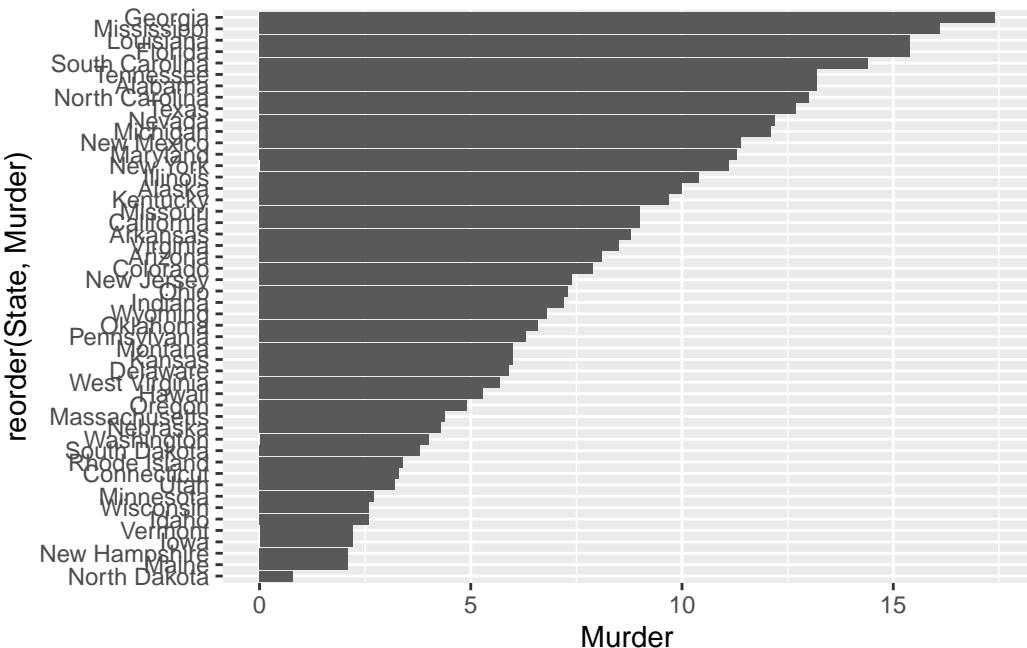
```
ggplot(gapminder_top5) + aes(x=reorder(country, -pop), y=pop, fill=country) + geom_col(col
```



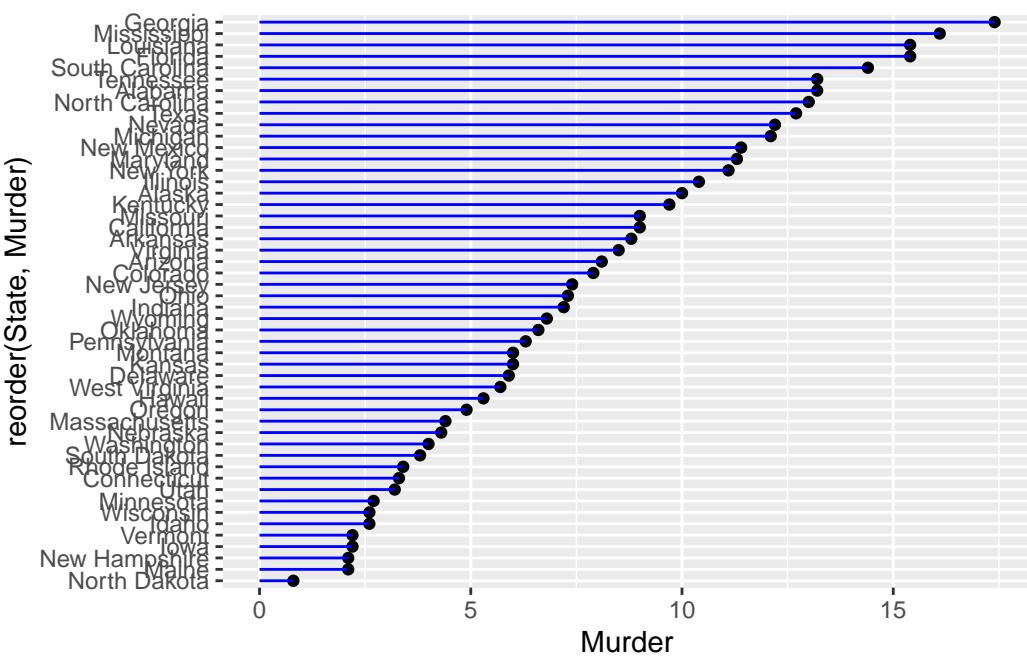
```
head(USArrests)
```

	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

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



```
ggplot(USArrests) + aes(x=reorder(State,Murder), y=Murder) + geom_point() + geom_segment(a
```



```
library(gapminder)
```

Attaching package: 'gapminder'

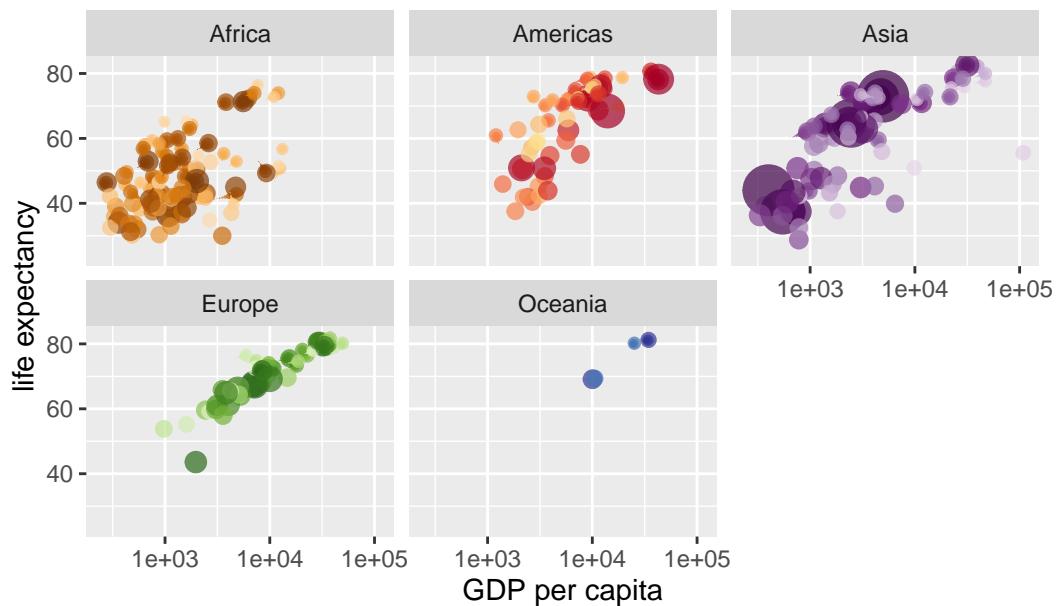
The following object is masked `_by_` '.GlobalEnv':

```
gapminder
```

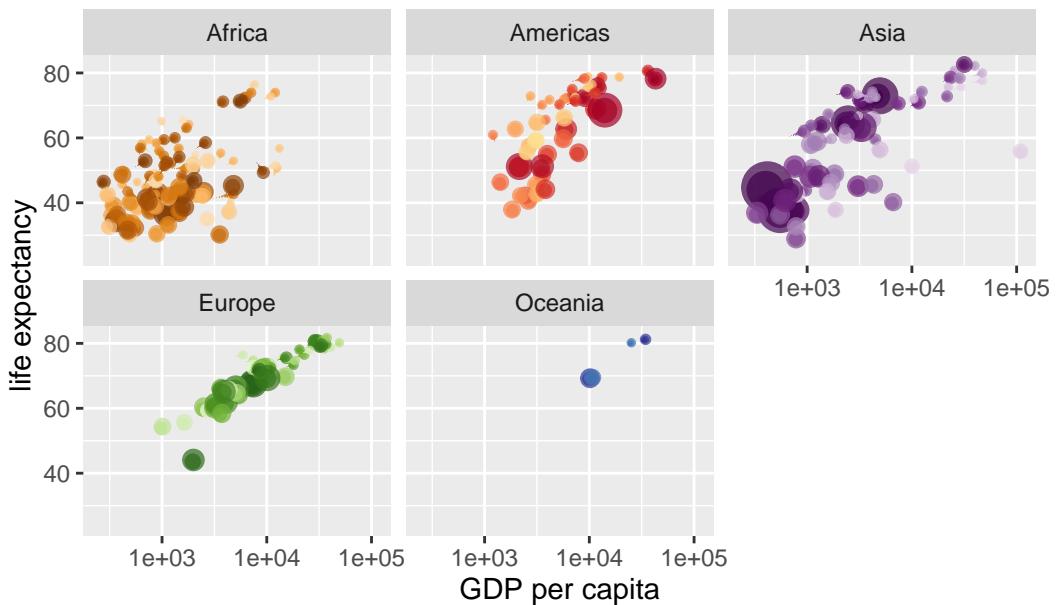
```
library(gganimate)
```

```
ggplot(gapminder, aes(gdpPerCap, lifeExp, size = pop, colour = country)) + geom_point(alpha = .5)
```

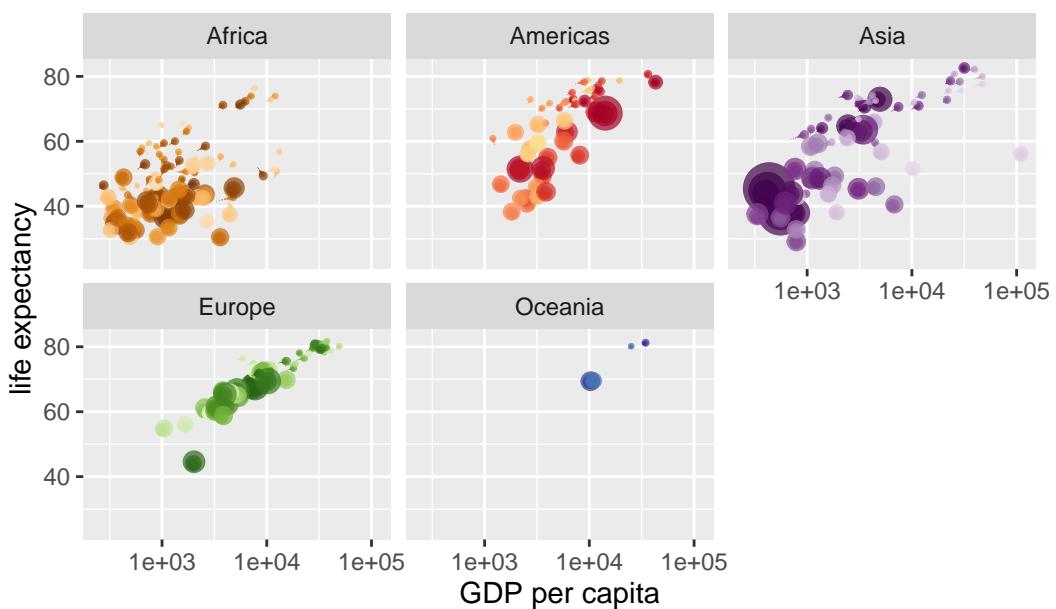
Year: 1952



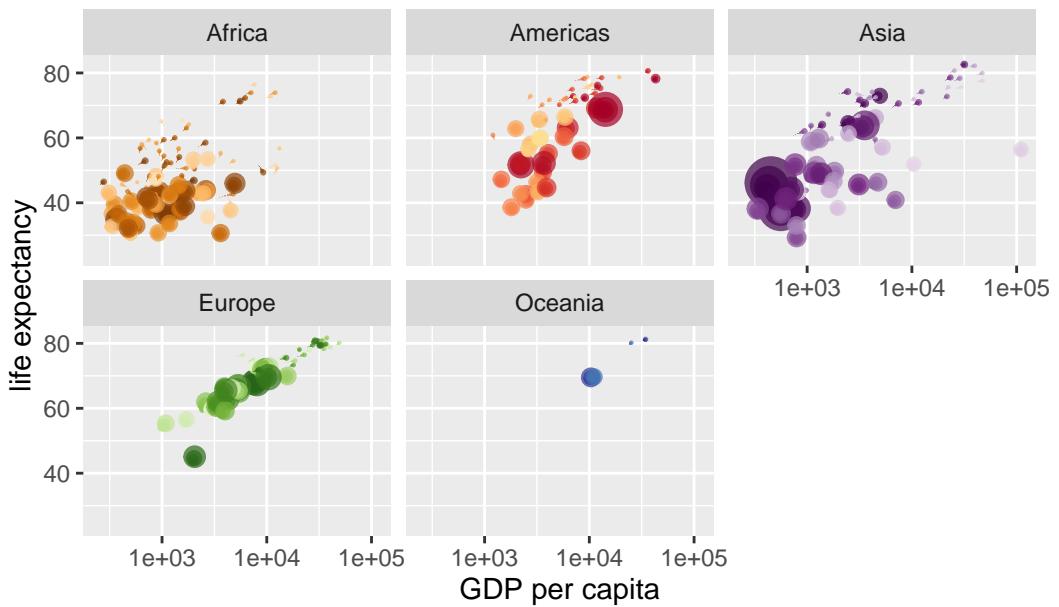
Year: 1953



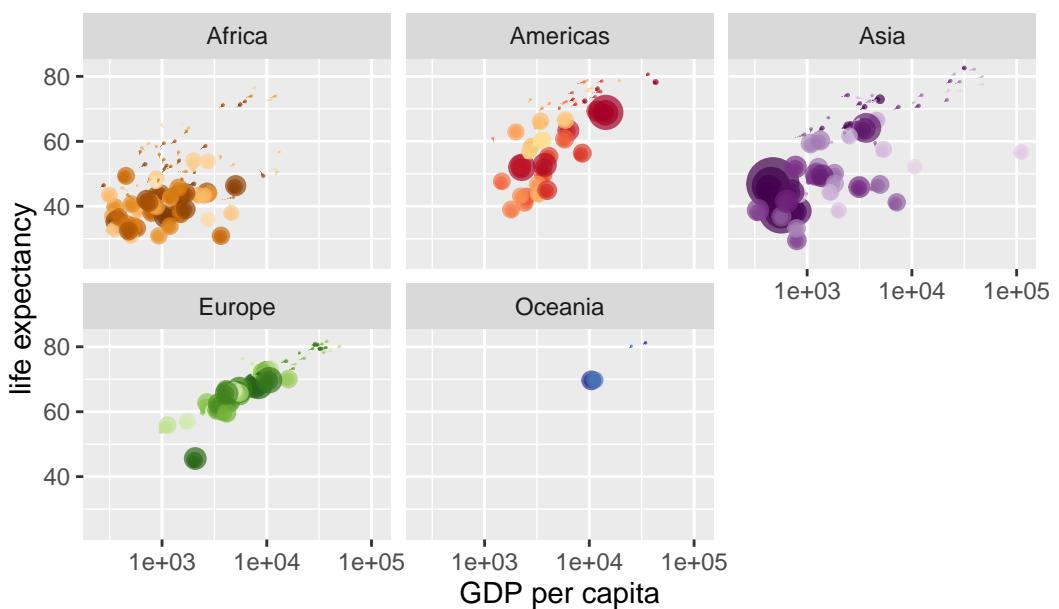
Year: 1953



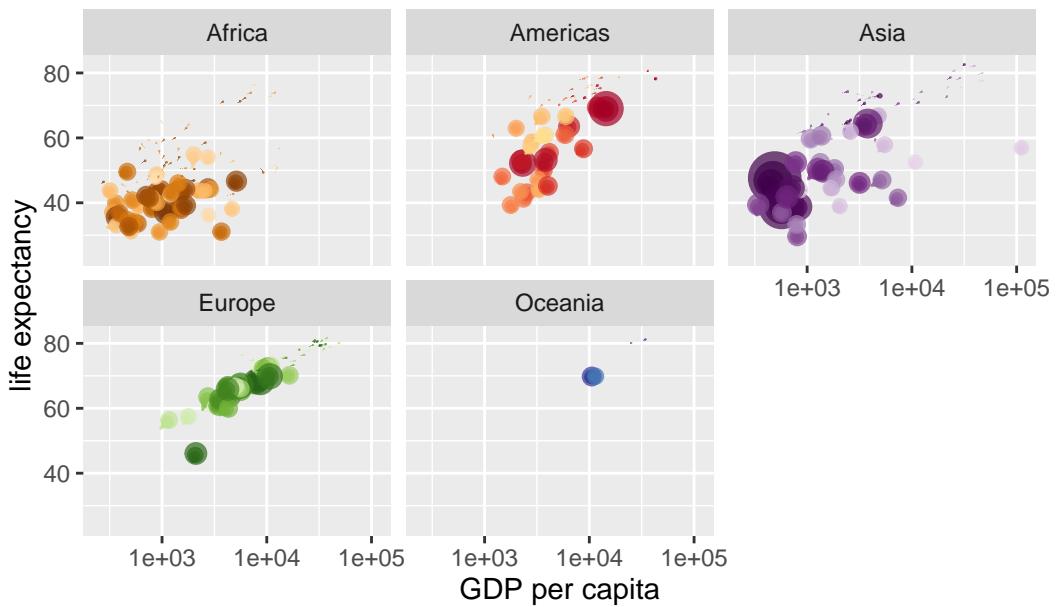
Year: 1954



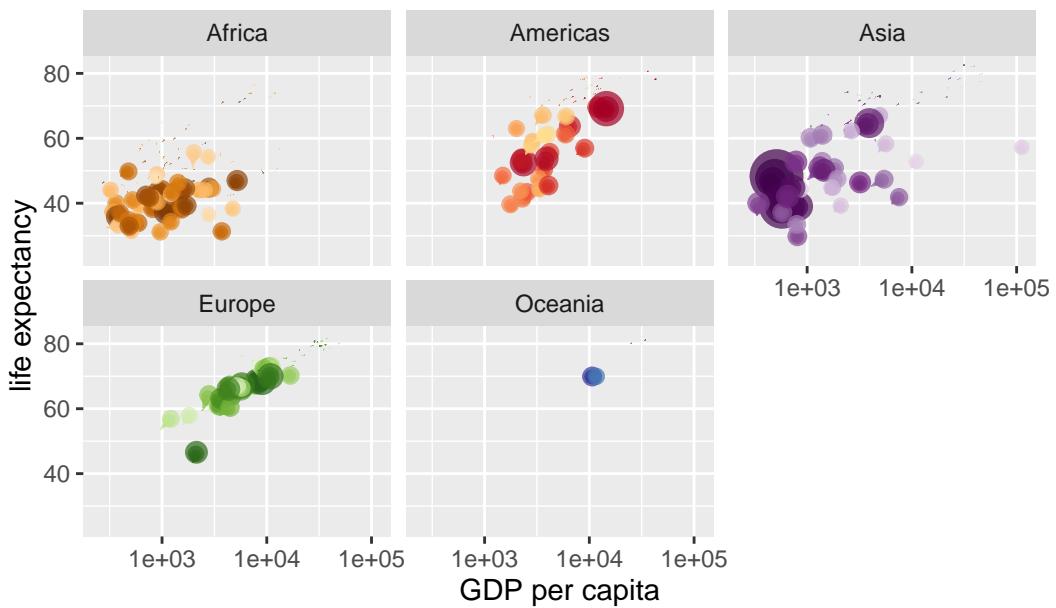
Year: 1954



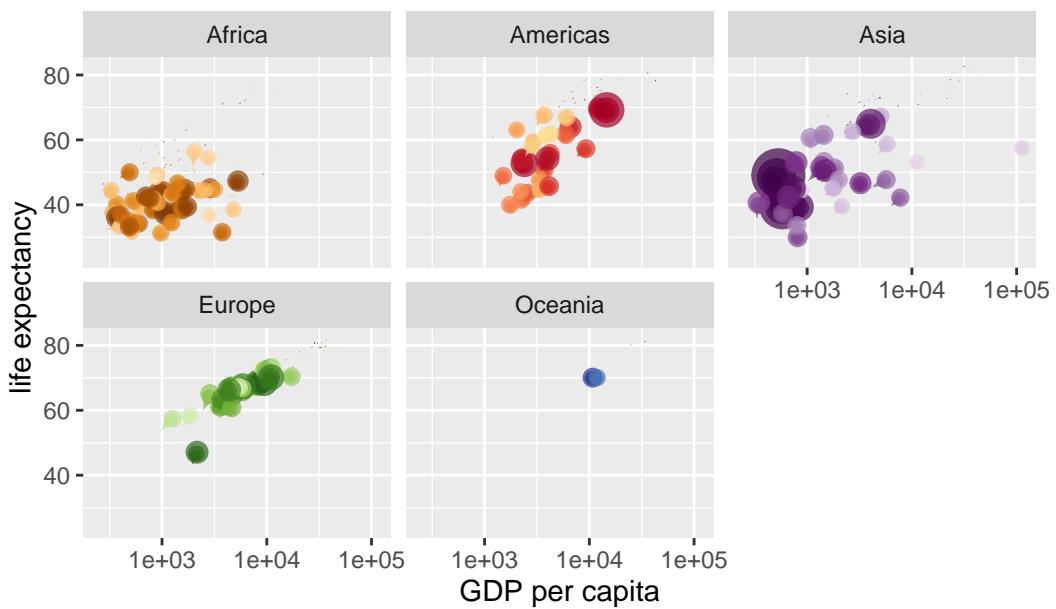
Year: 1955



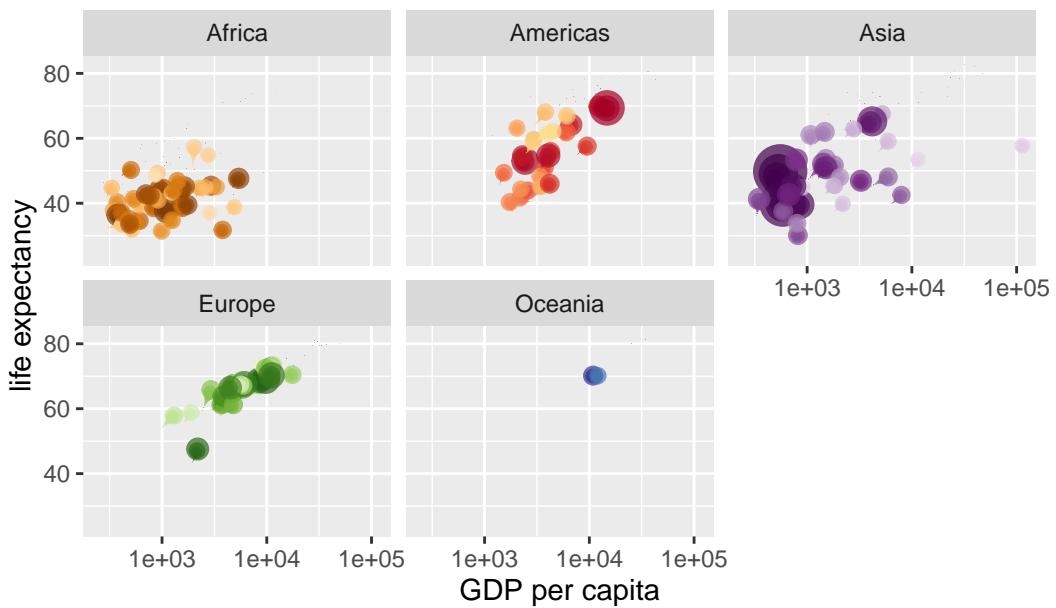
Year: 1955



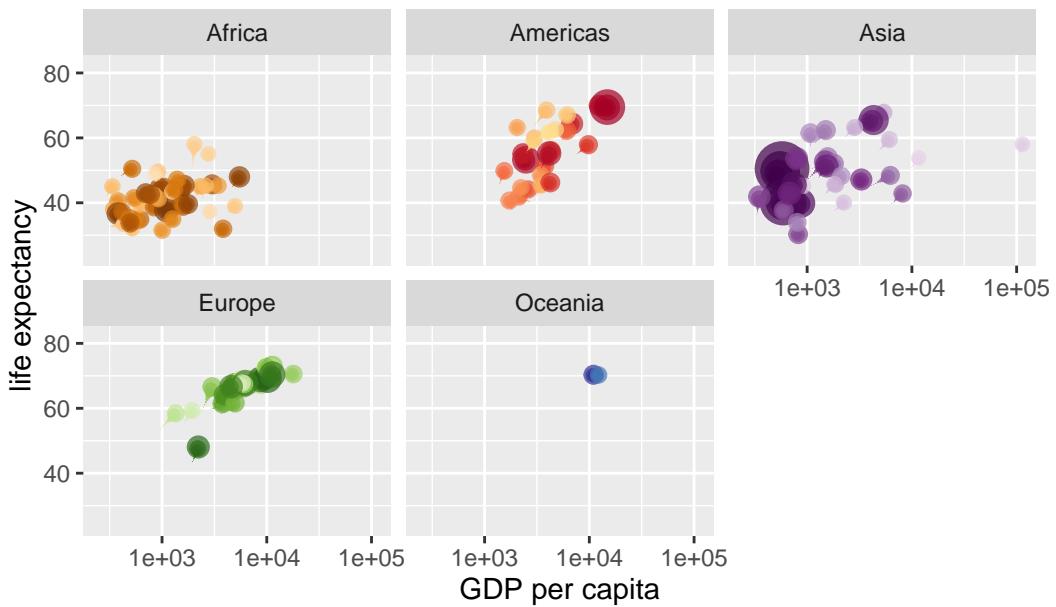
Year: 1956



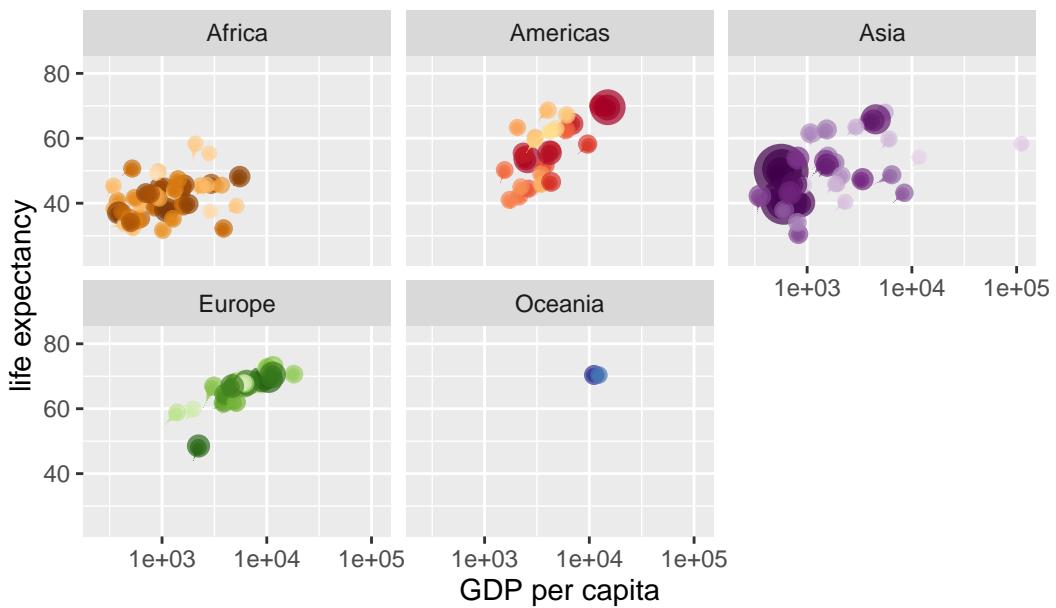
Year: 1956



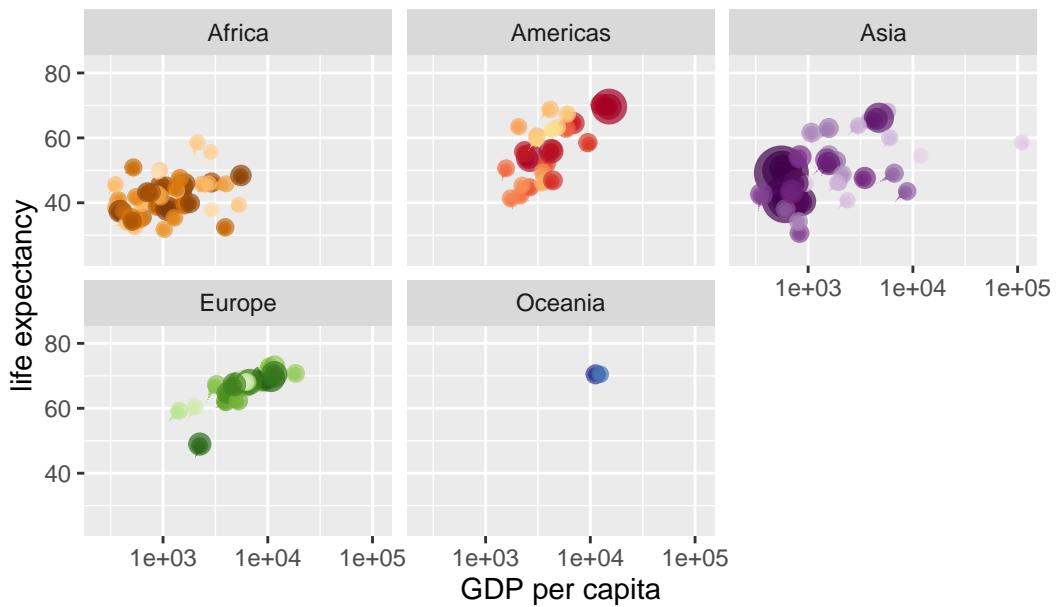
Year: 1957



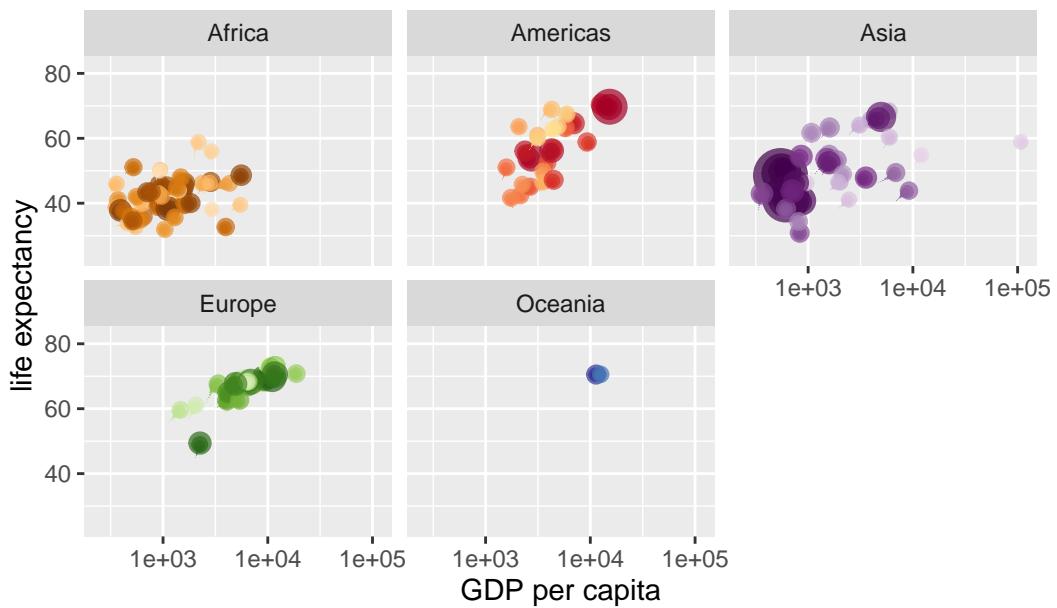
Year: 1958



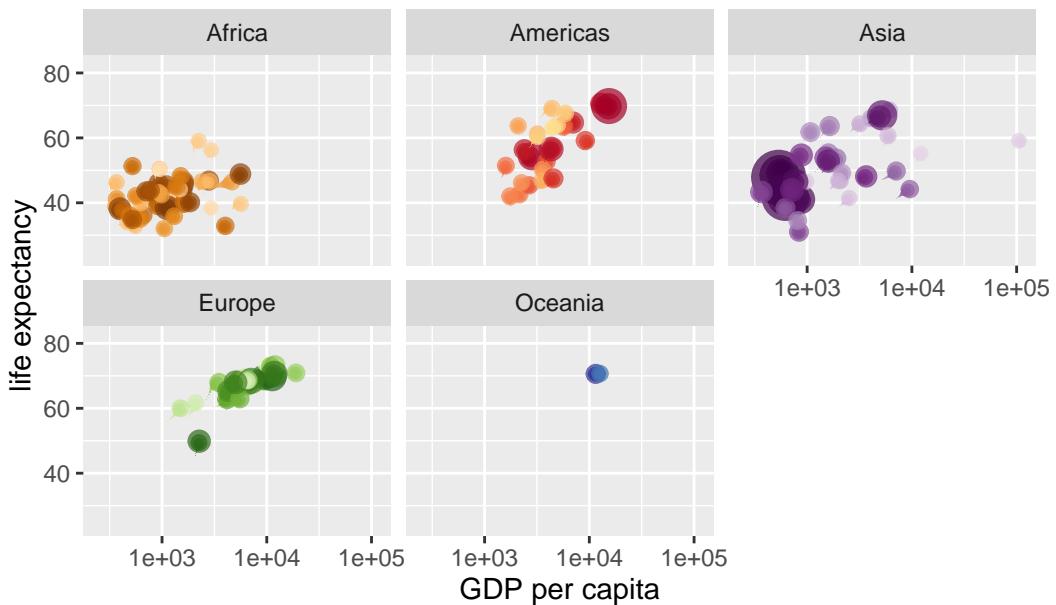
Year: 1958



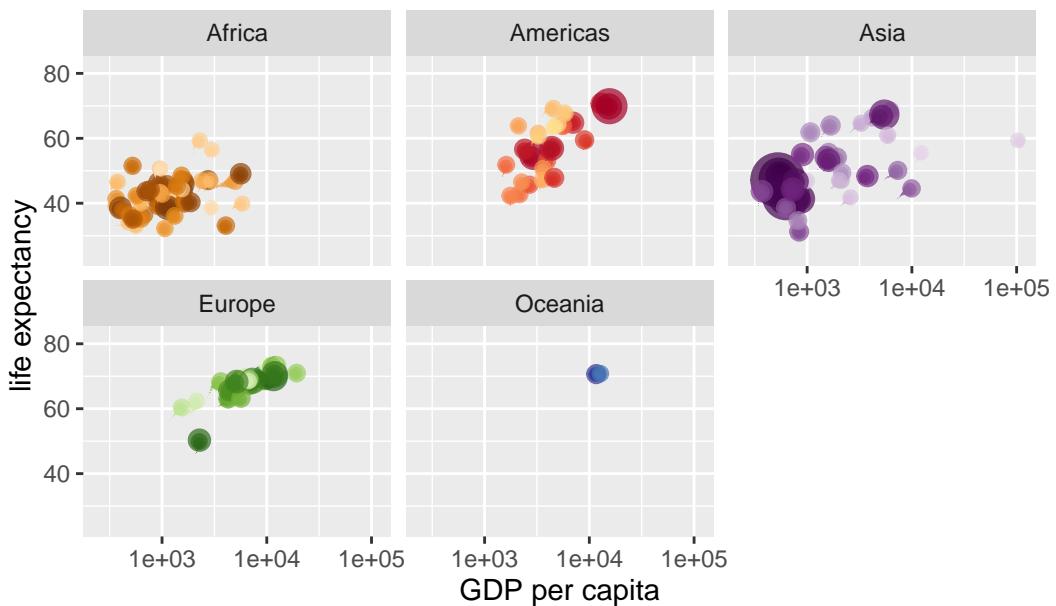
Year: 1959



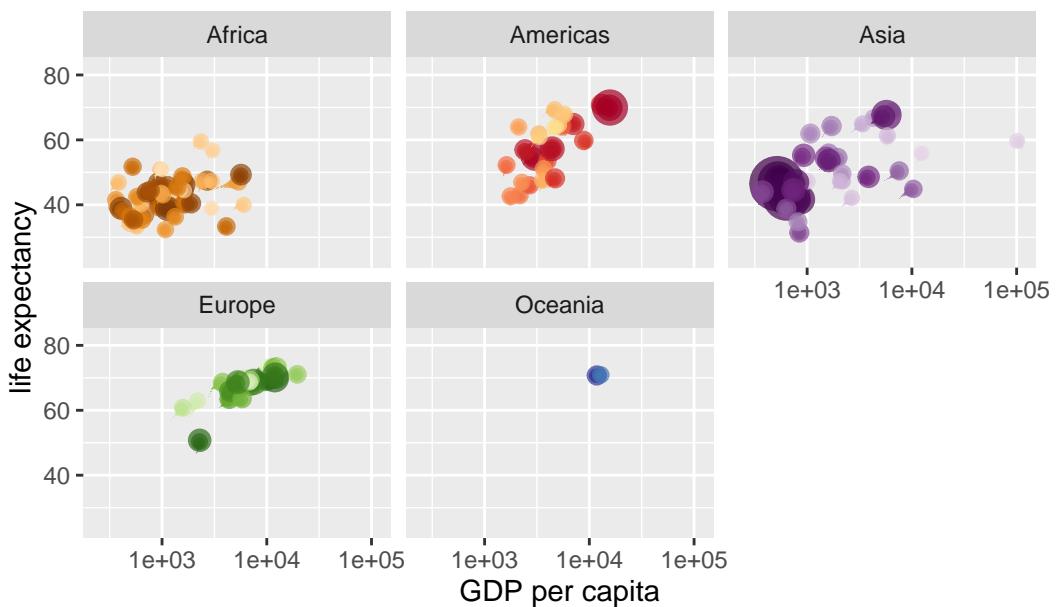
Year: 1959



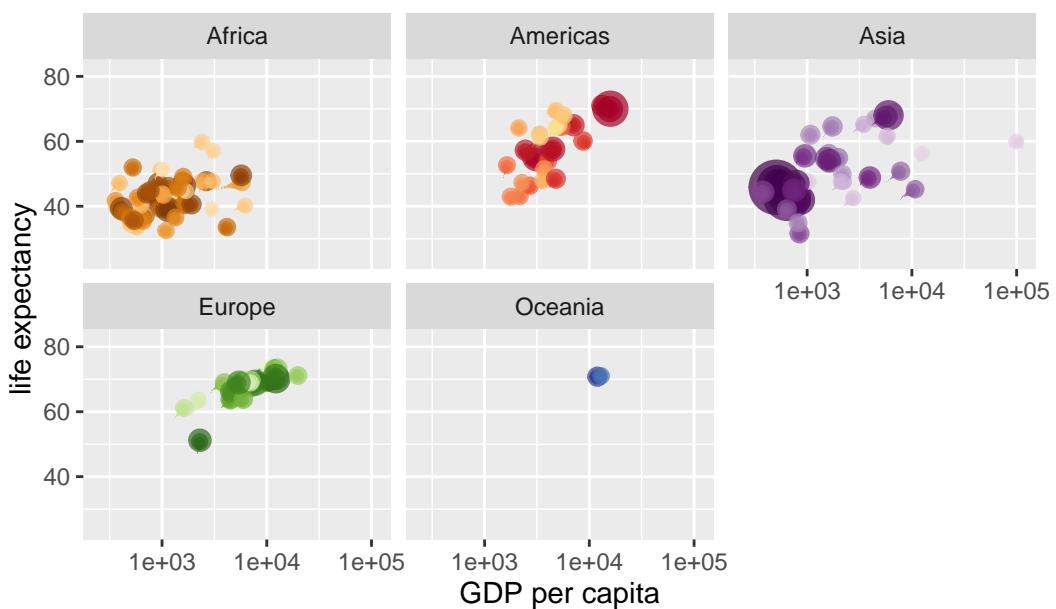
Year: 1960



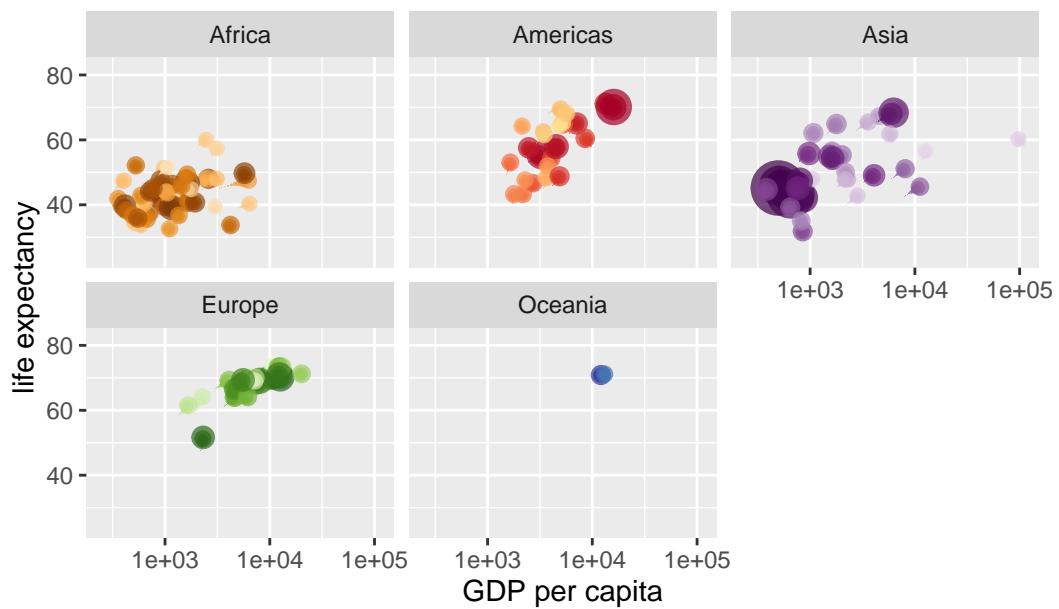
Year: 1960



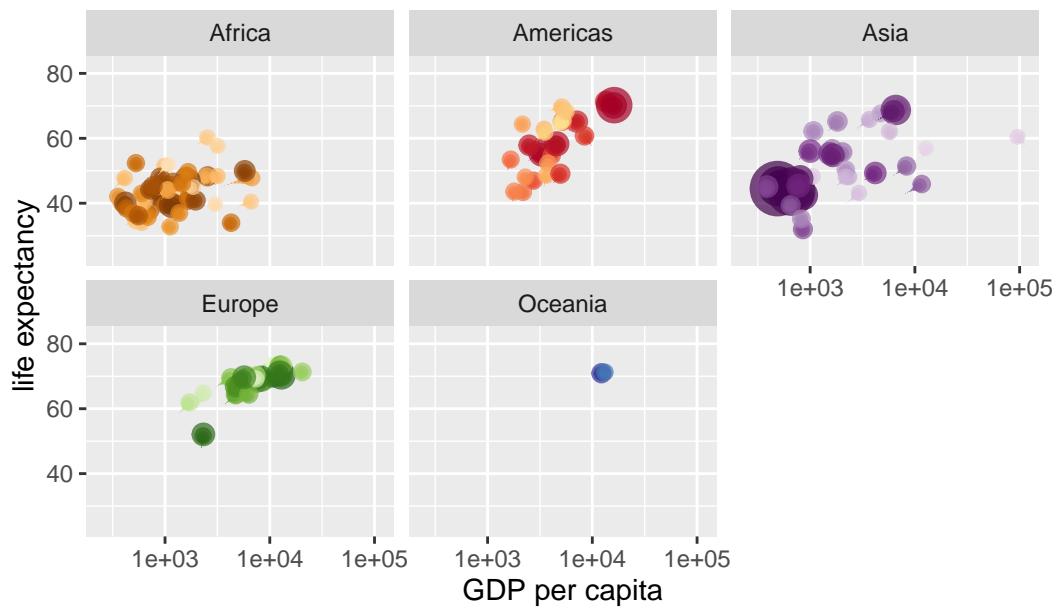
Year: 1961

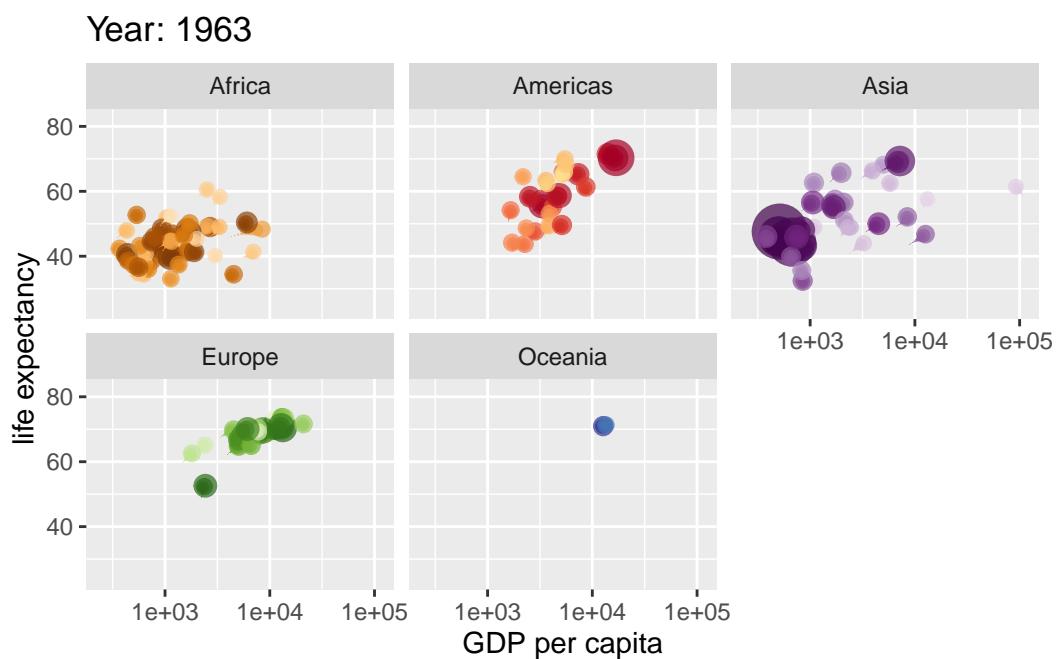
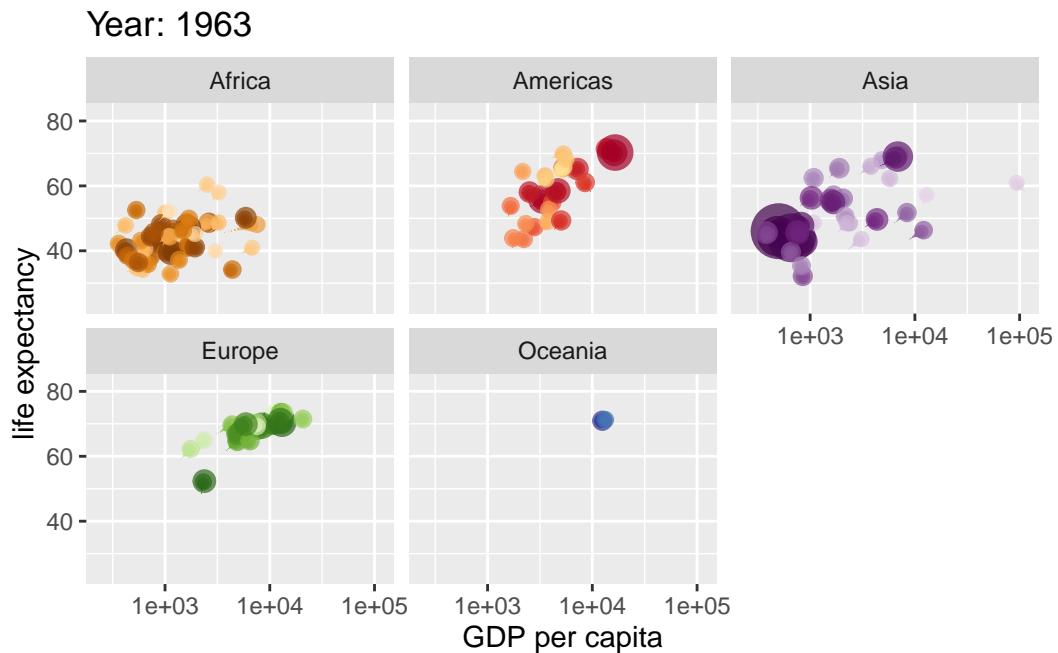


Year: 1961

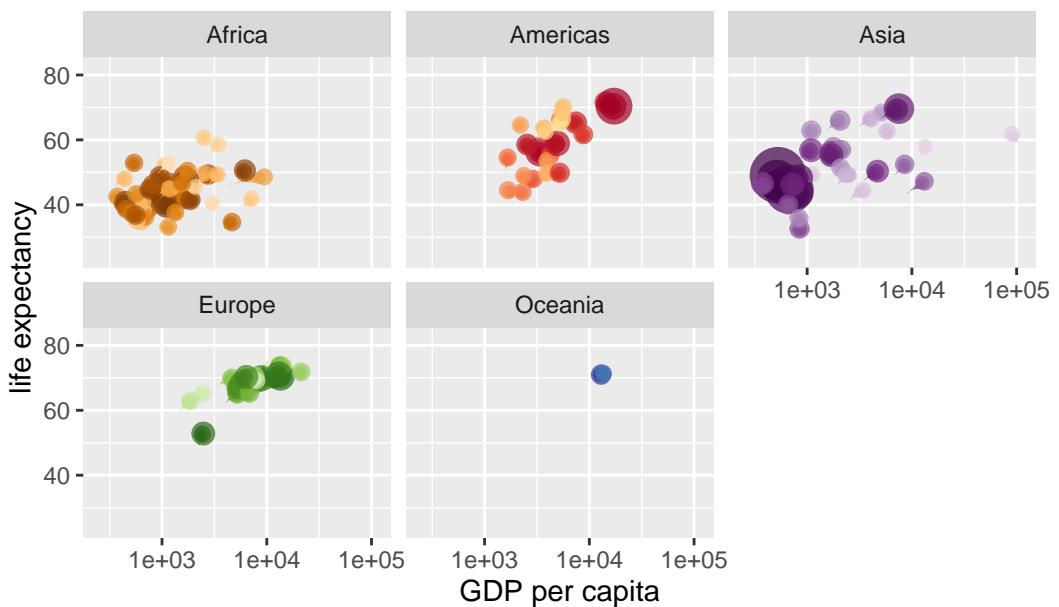


Year: 1962

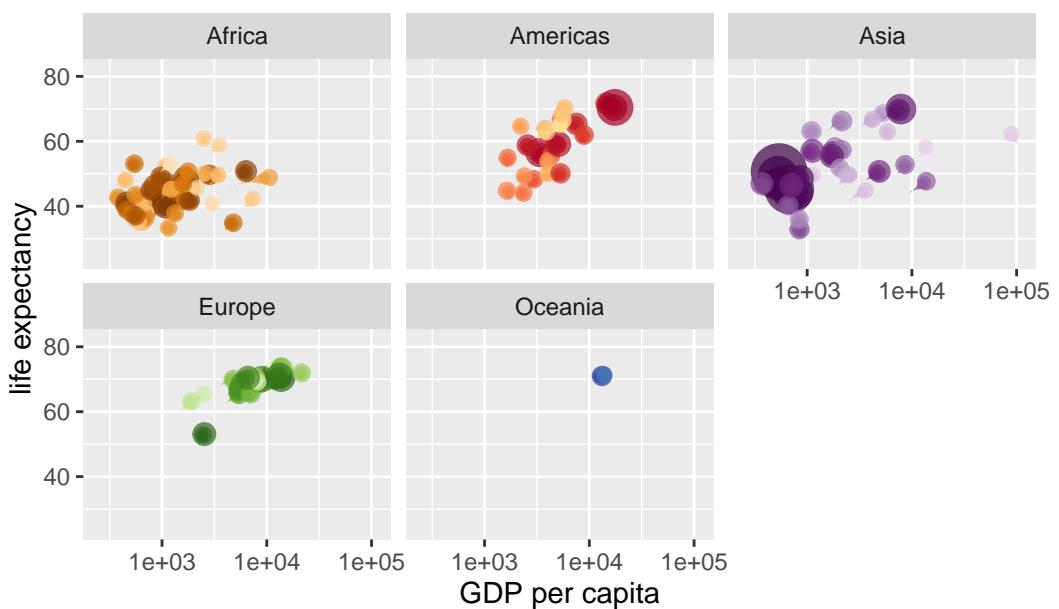




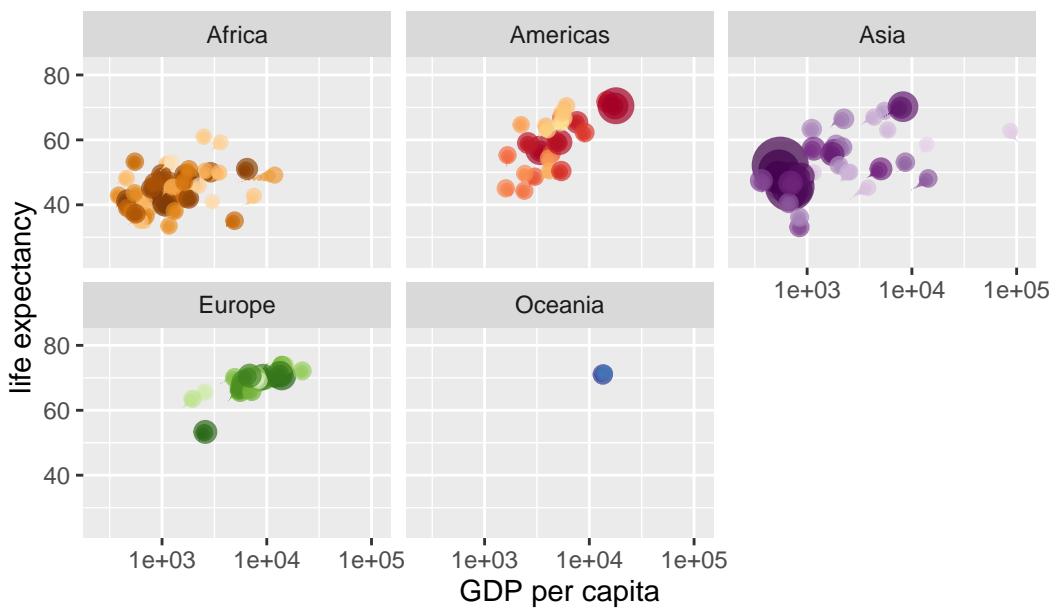
Year: 1964



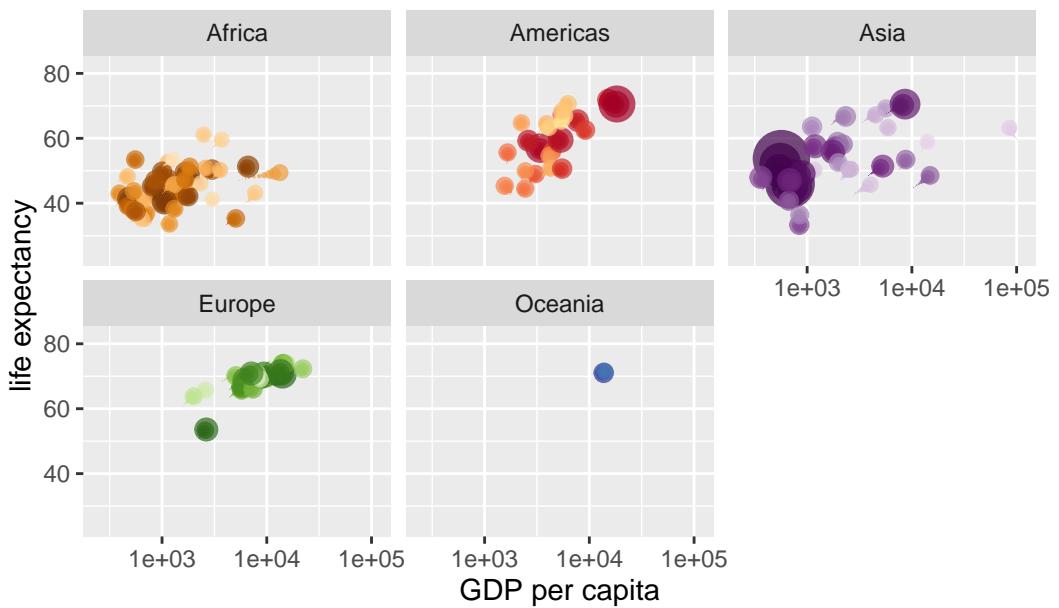
Year: 1964



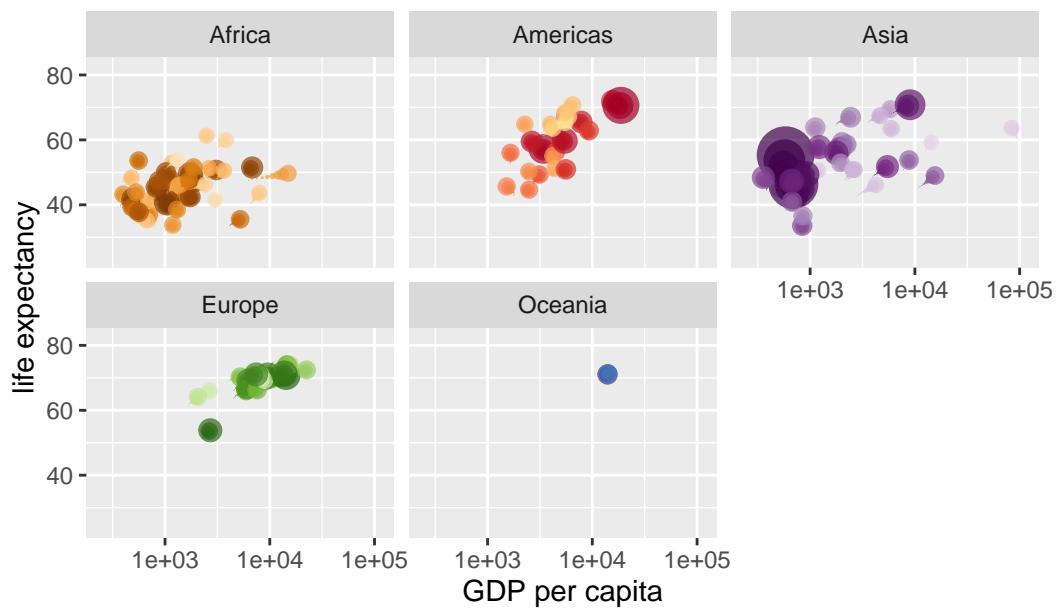
Year: 1965



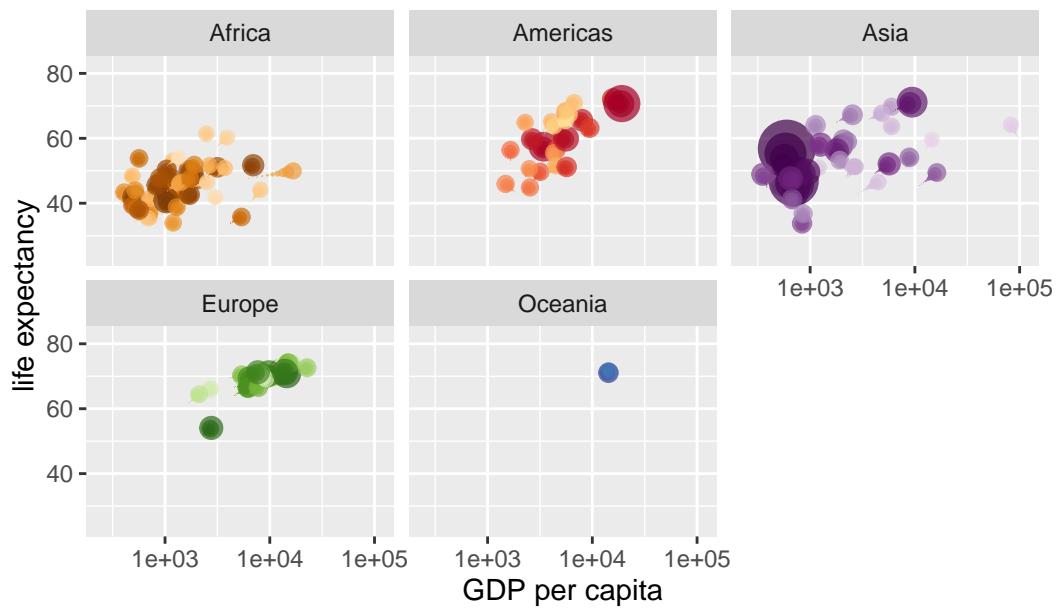
Year: 1965



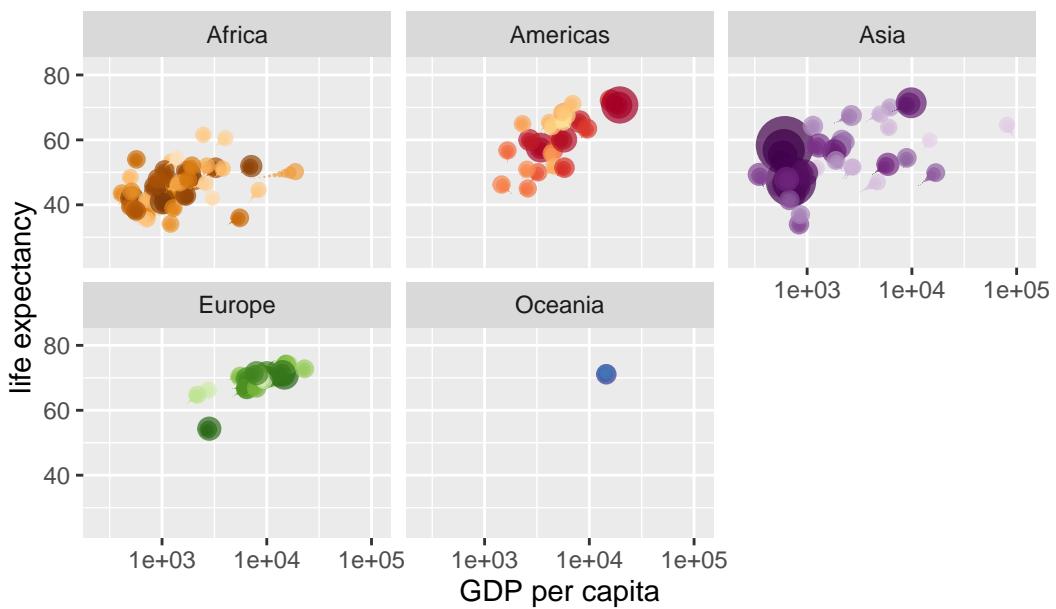
Year: 1966



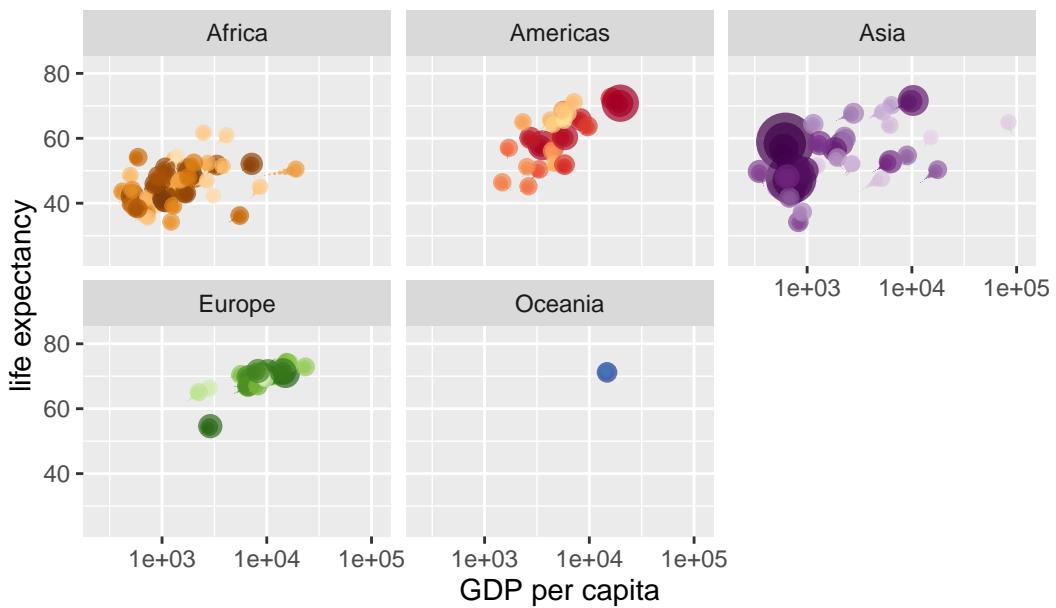
Year: 1966



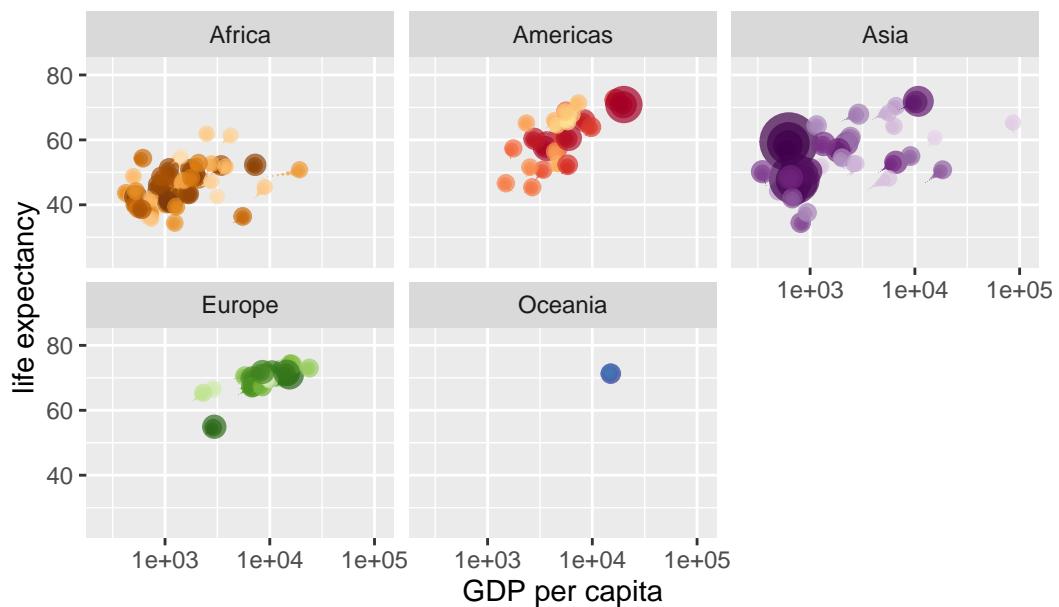
Year: 1967



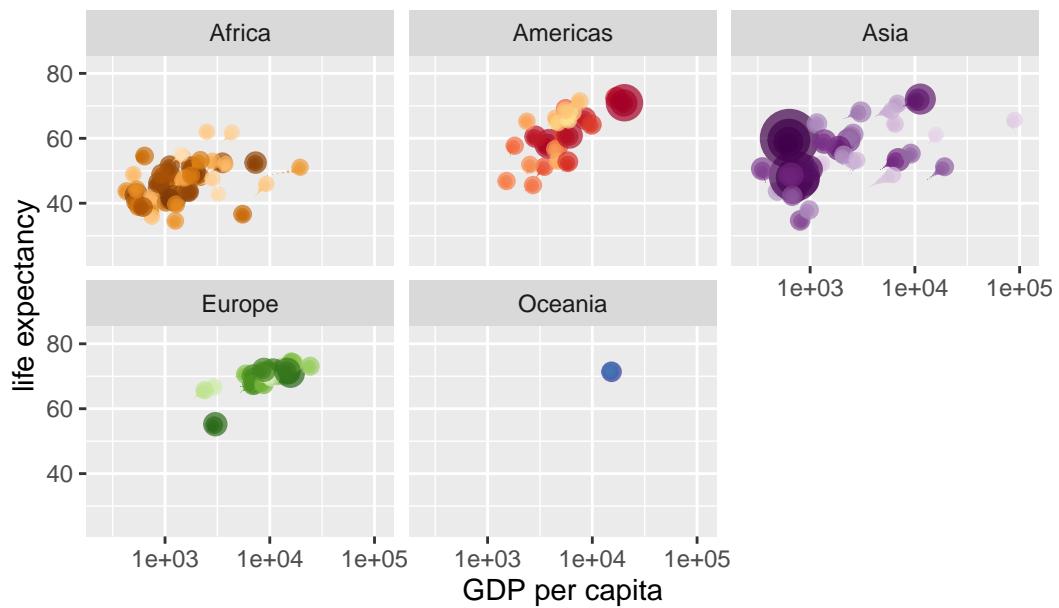
Year: 1968



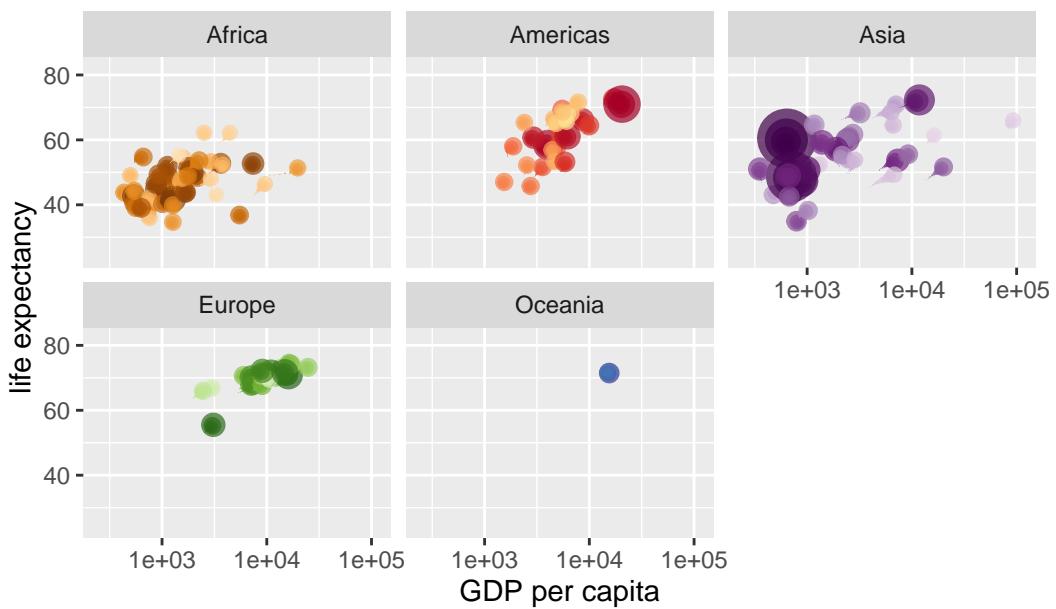
Year: 1968



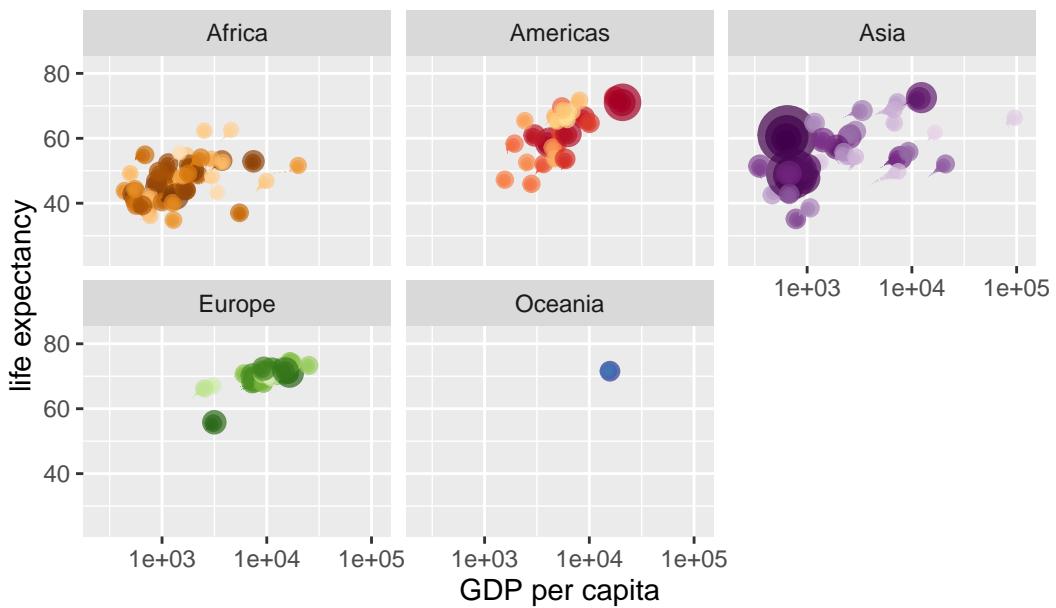
Year: 1969



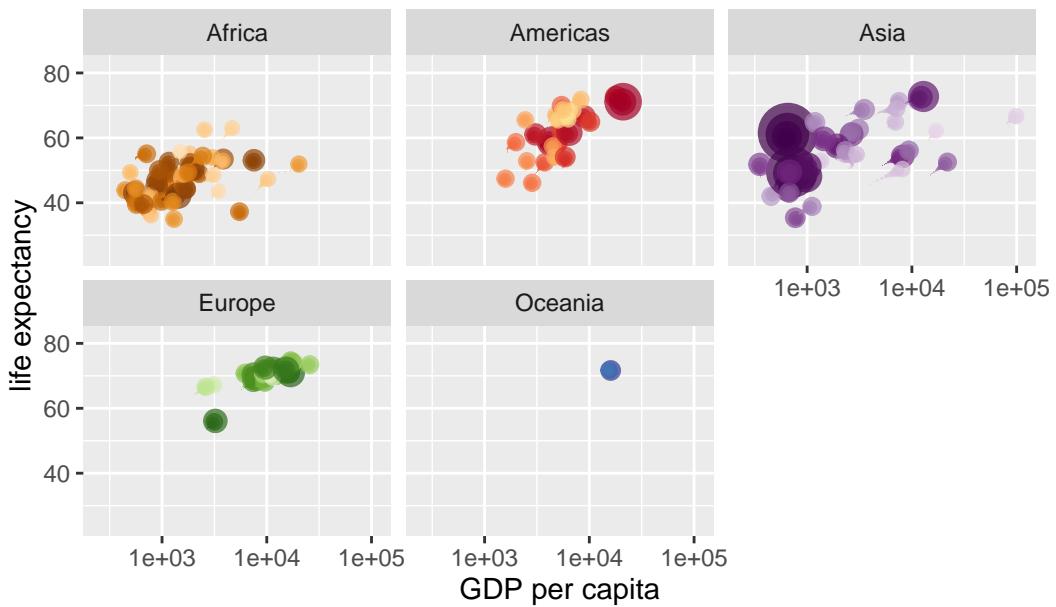
Year: 1969



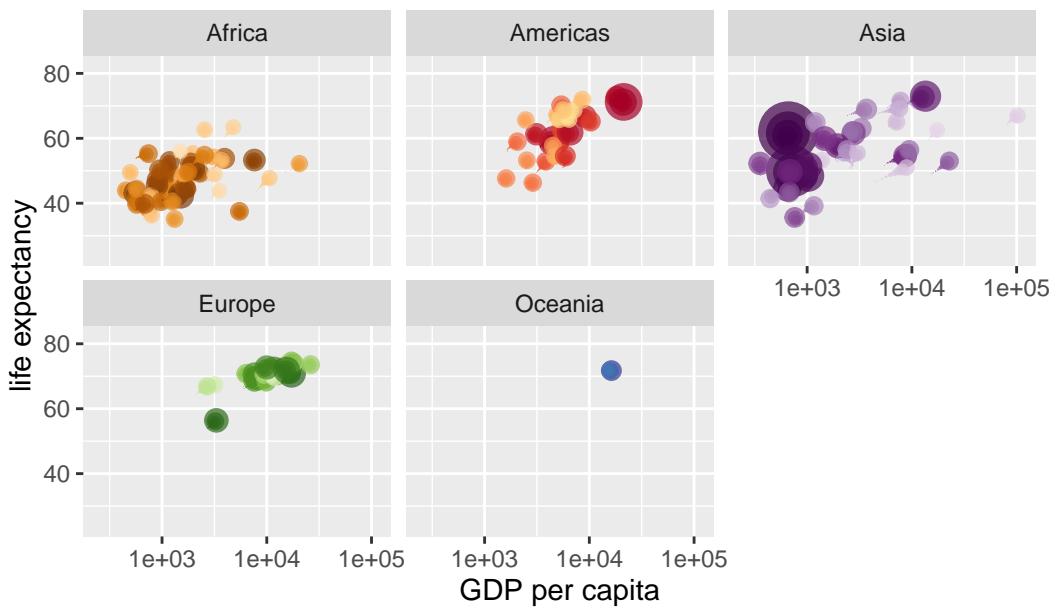
Year: 1970



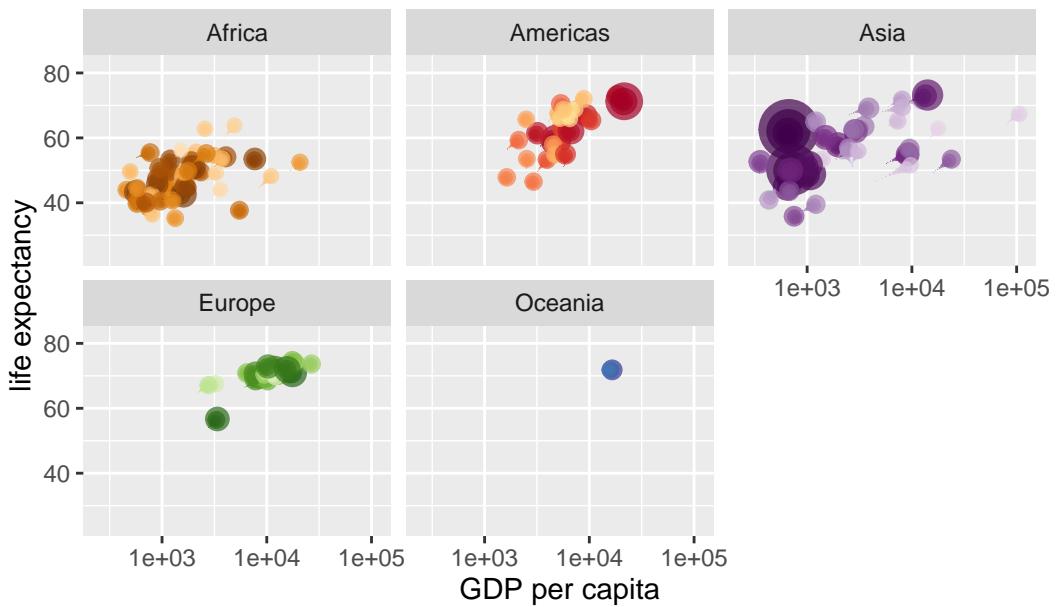
Year: 1970



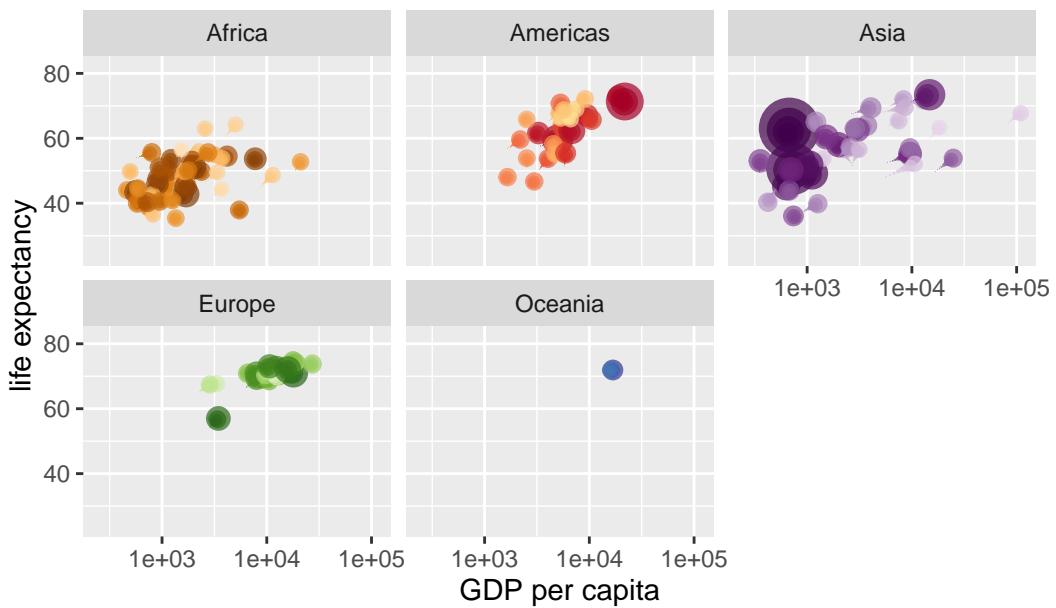
Year: 1971



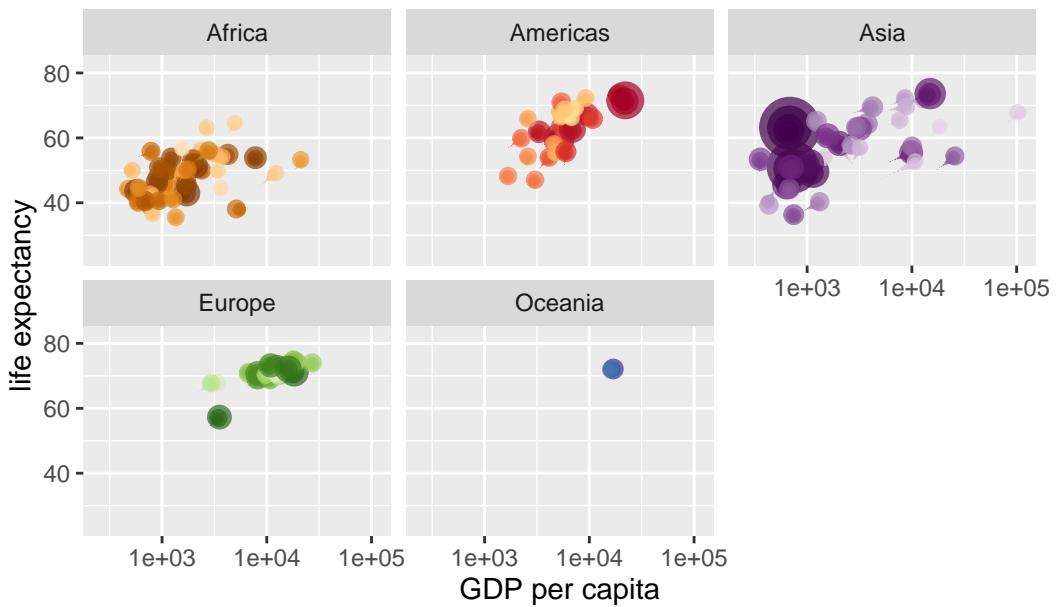
Year: 1971



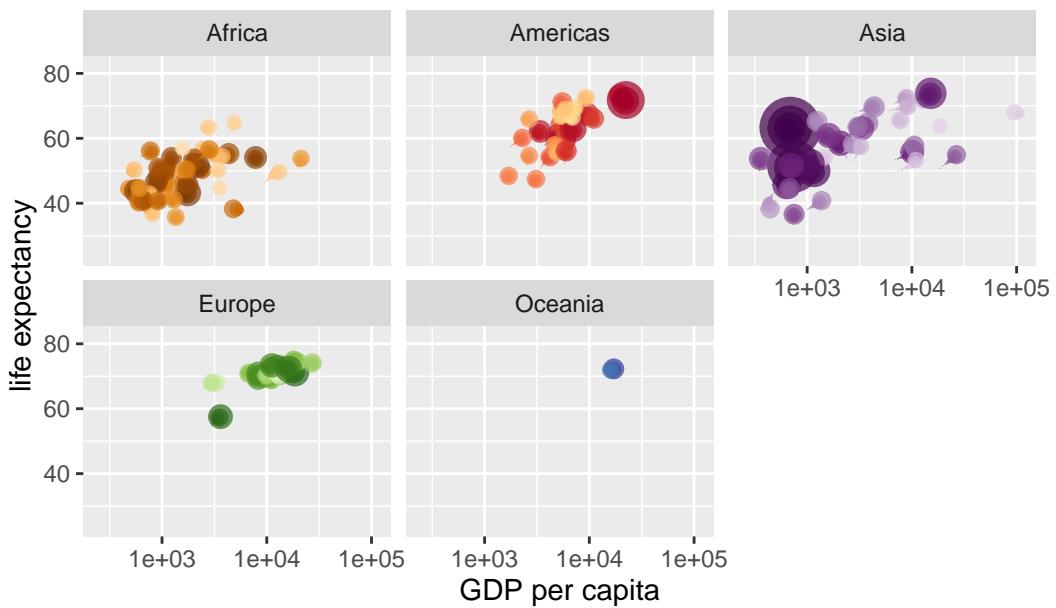
Year: 1972



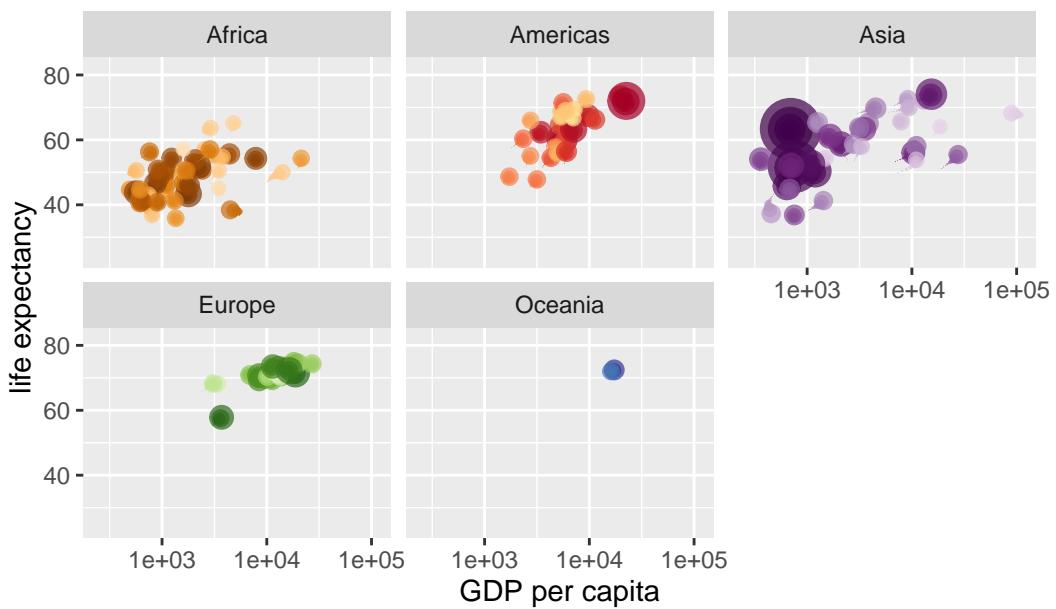
Year: 1973



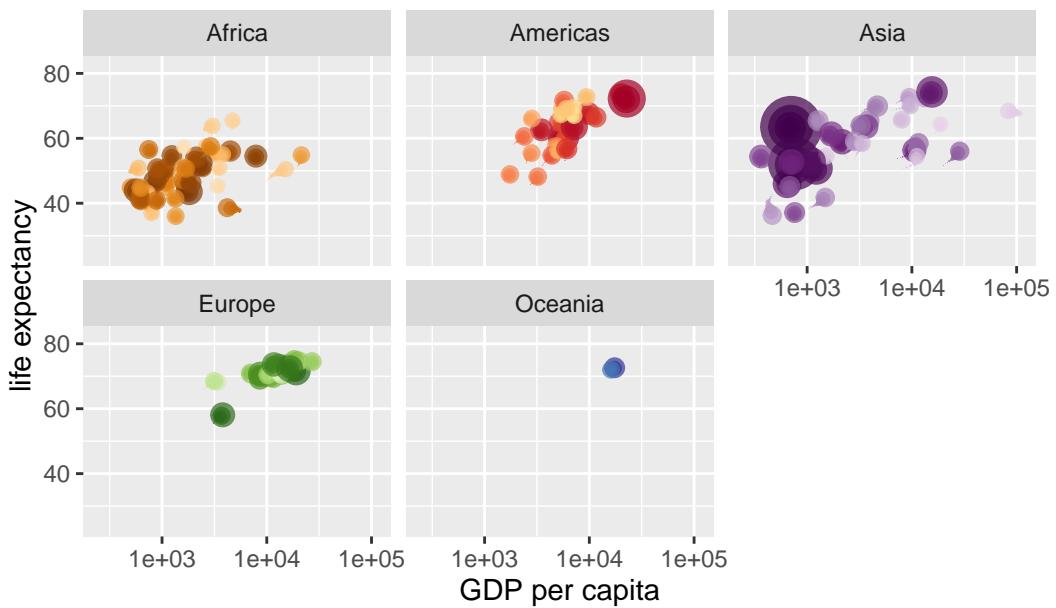
Year: 1973



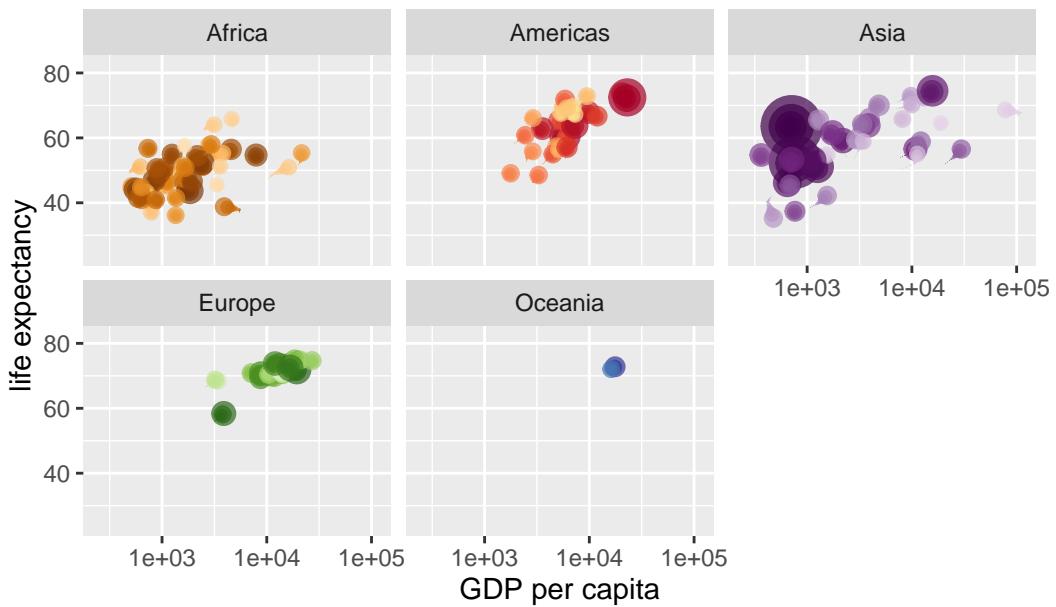
Year: 1974



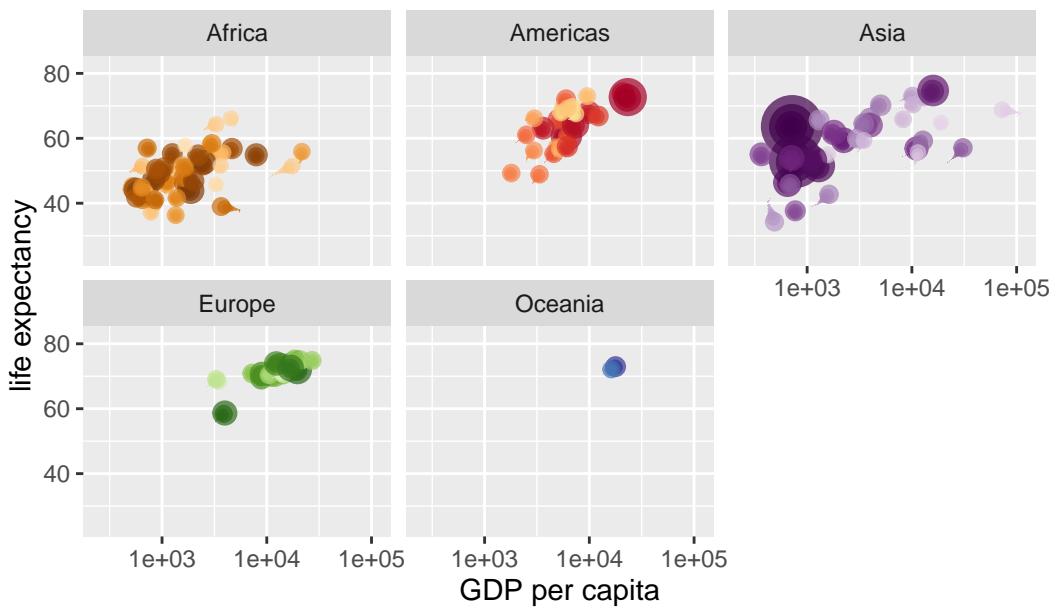
Year: 1974



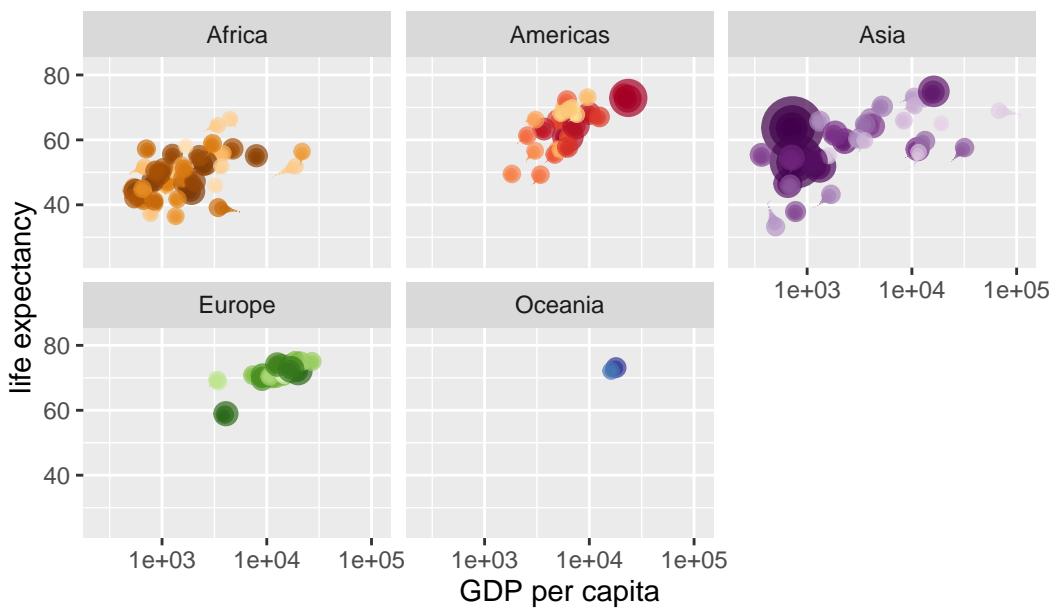
Year: 1975



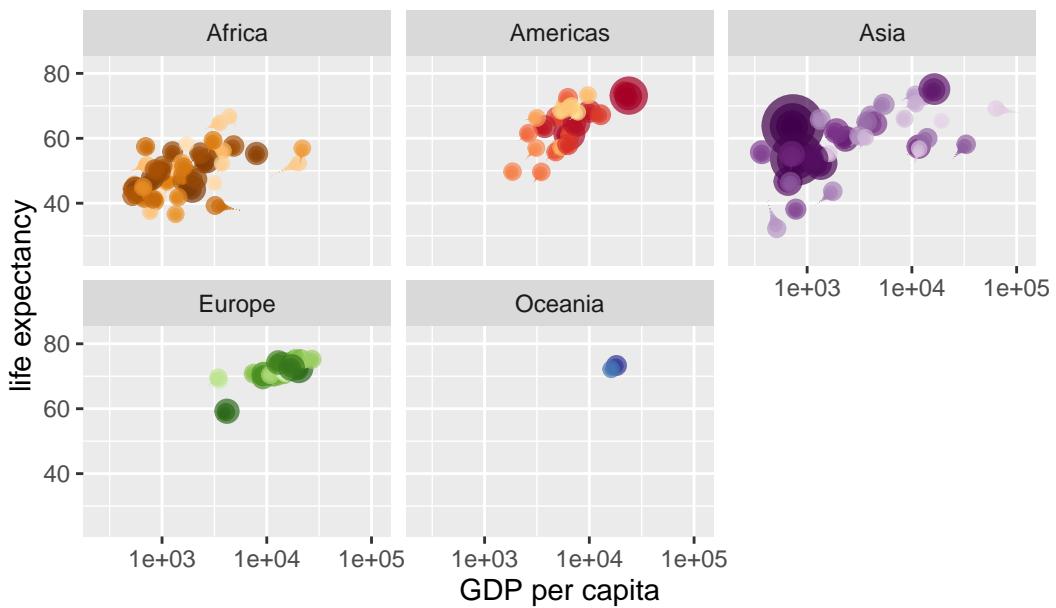
Year: 1975



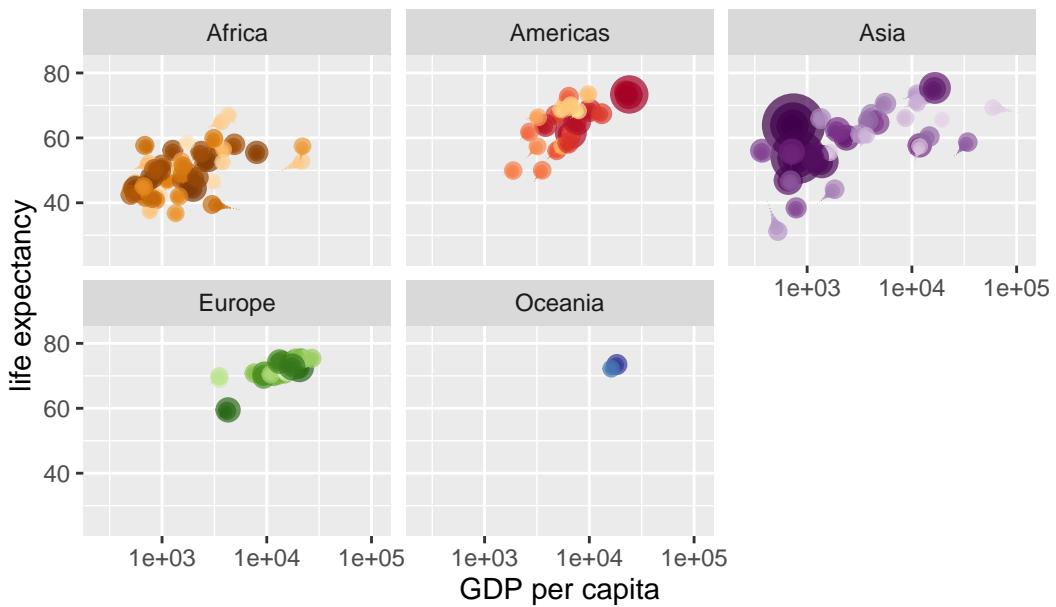
Year: 1976



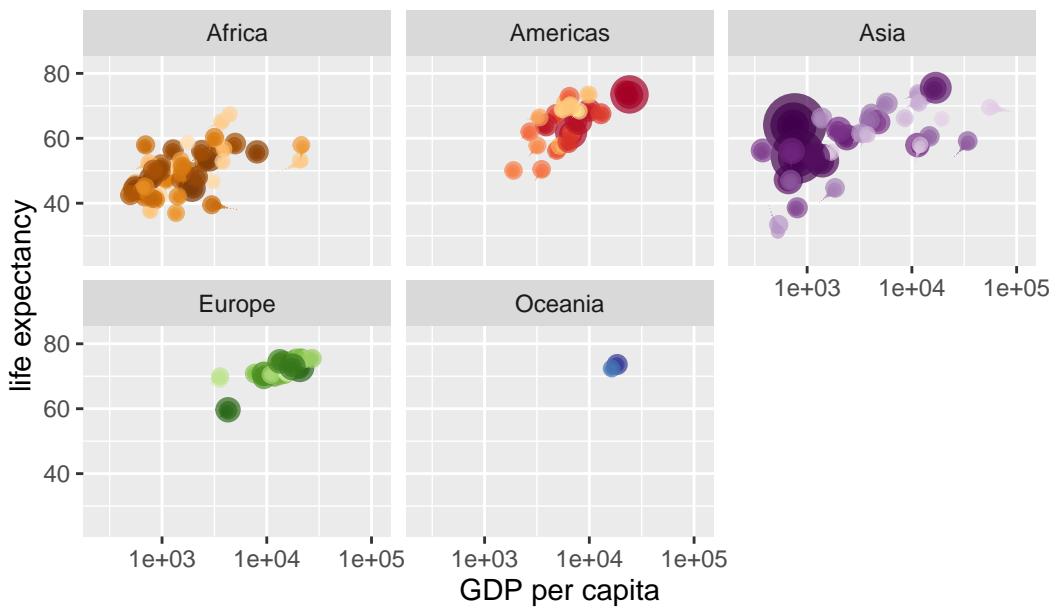
Year: 1976



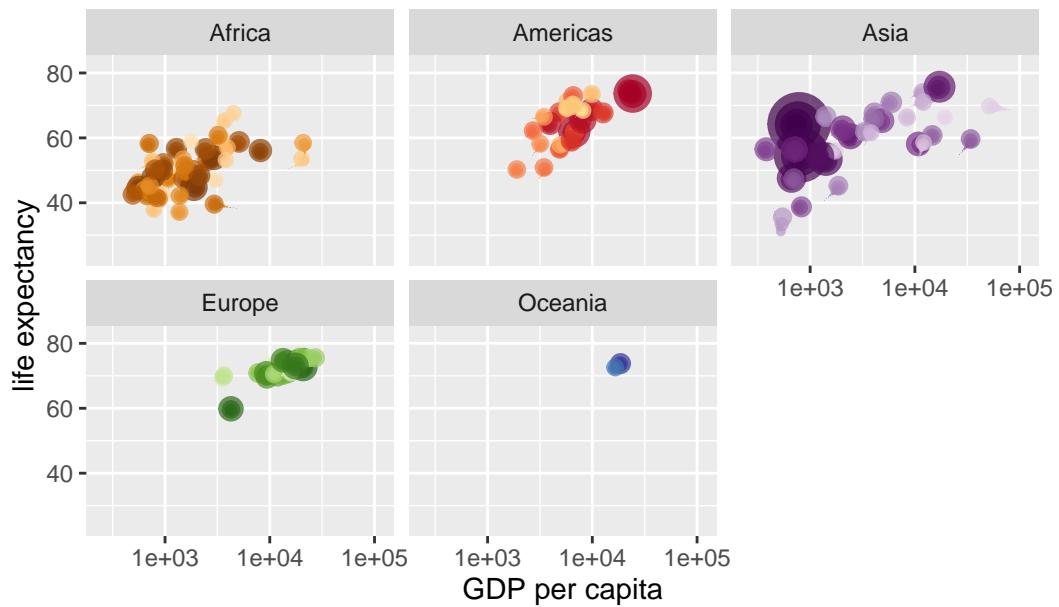
Year: 1977



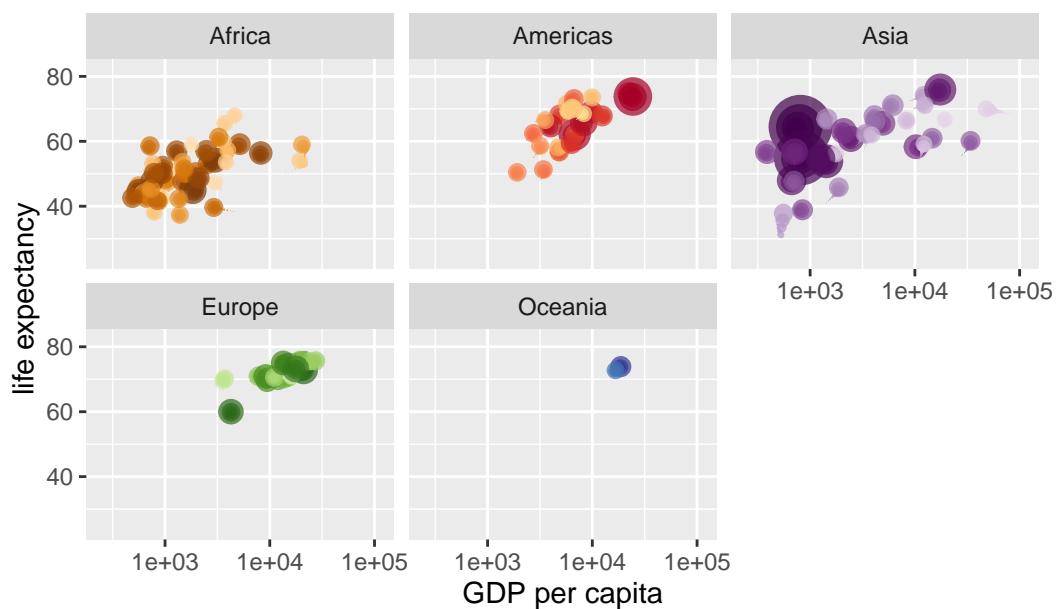
Year: 1978



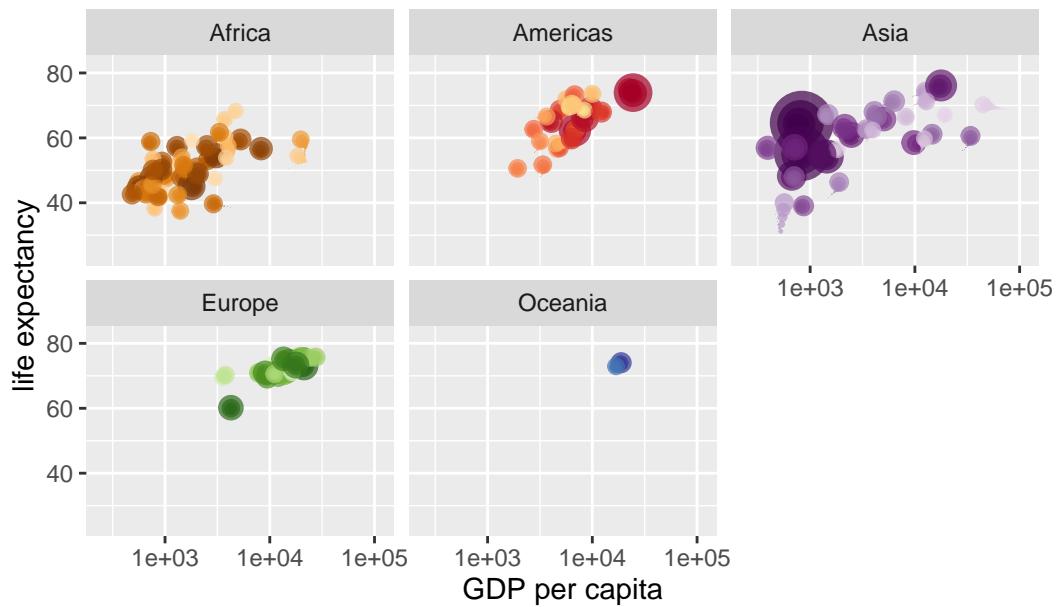
Year: 1978



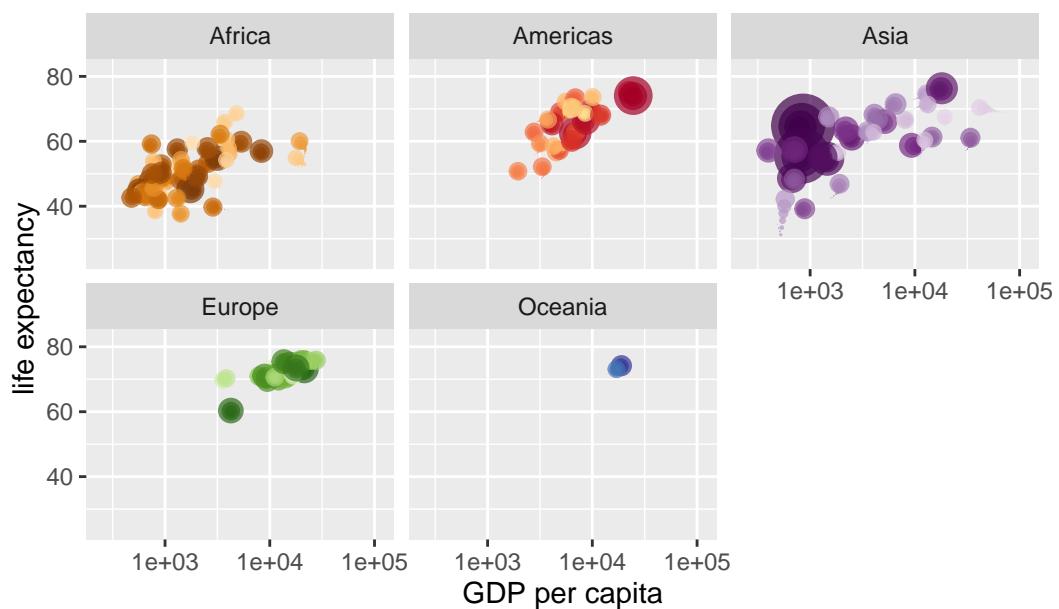
Year: 1979



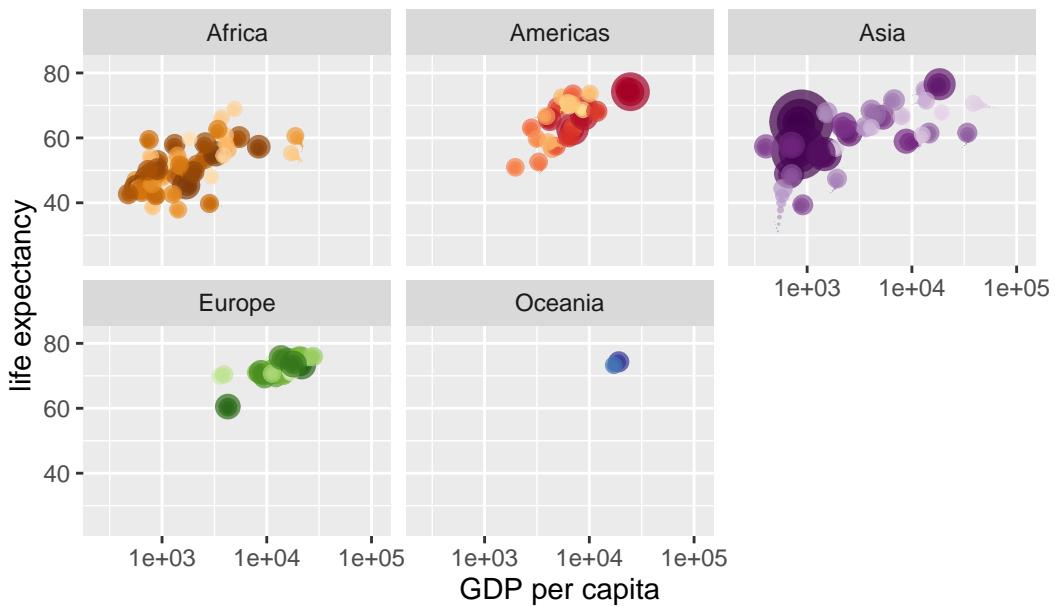
Year: 1979



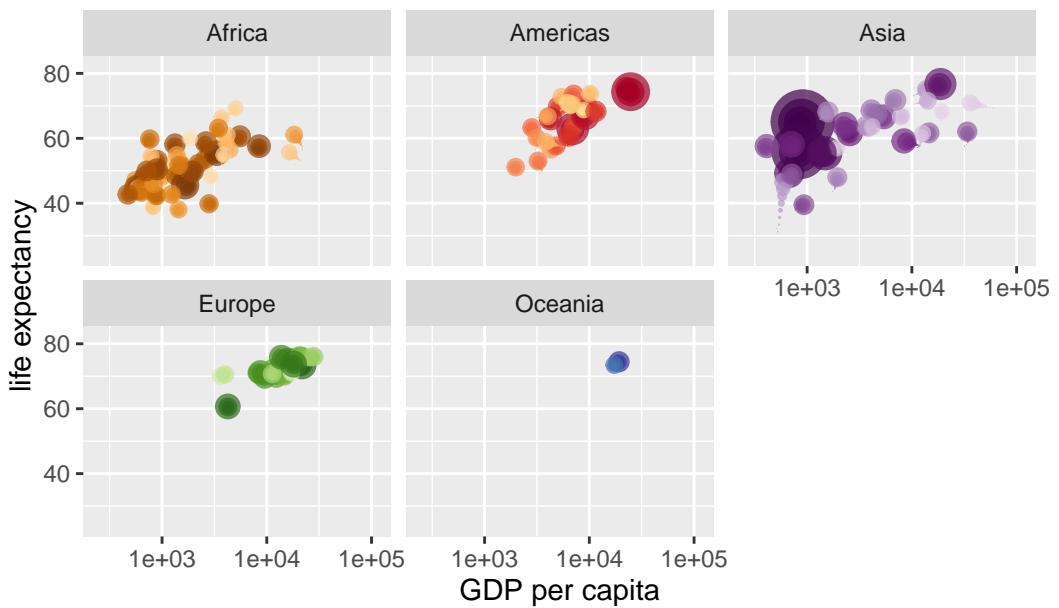
Year: 1980



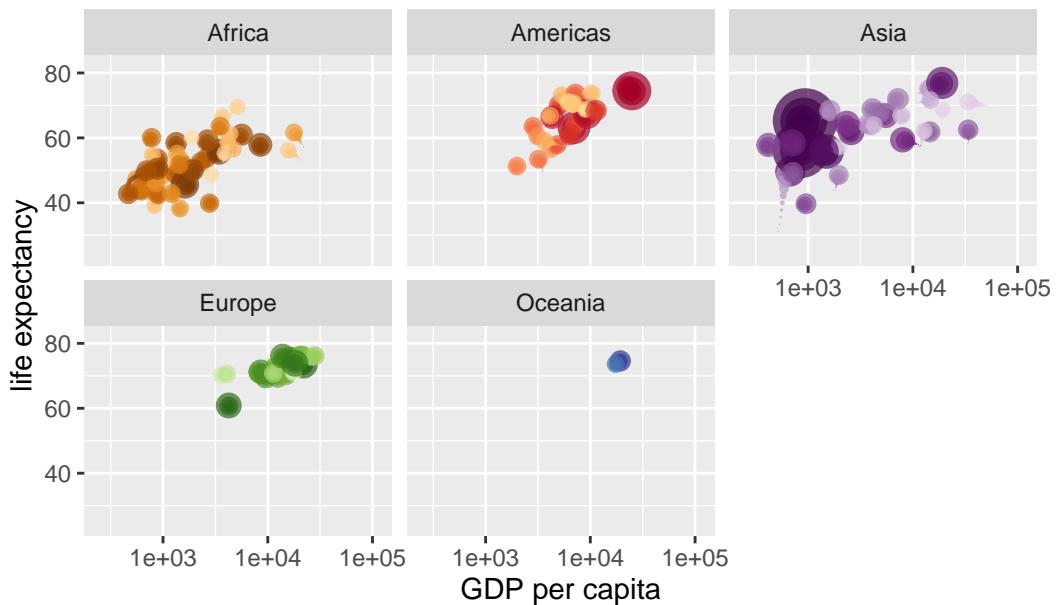
Year: 1980



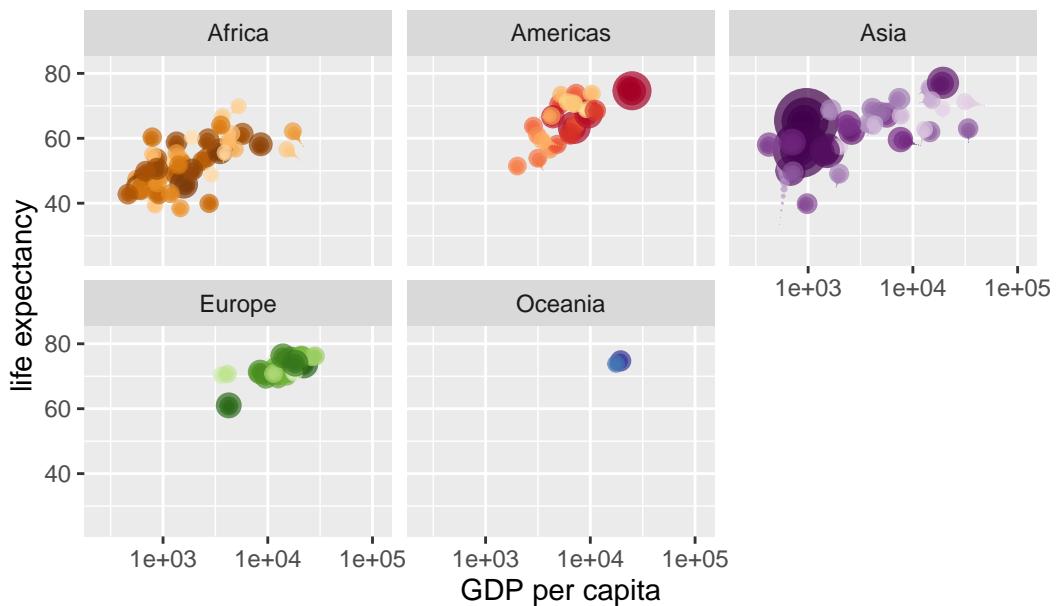
Year: 1981



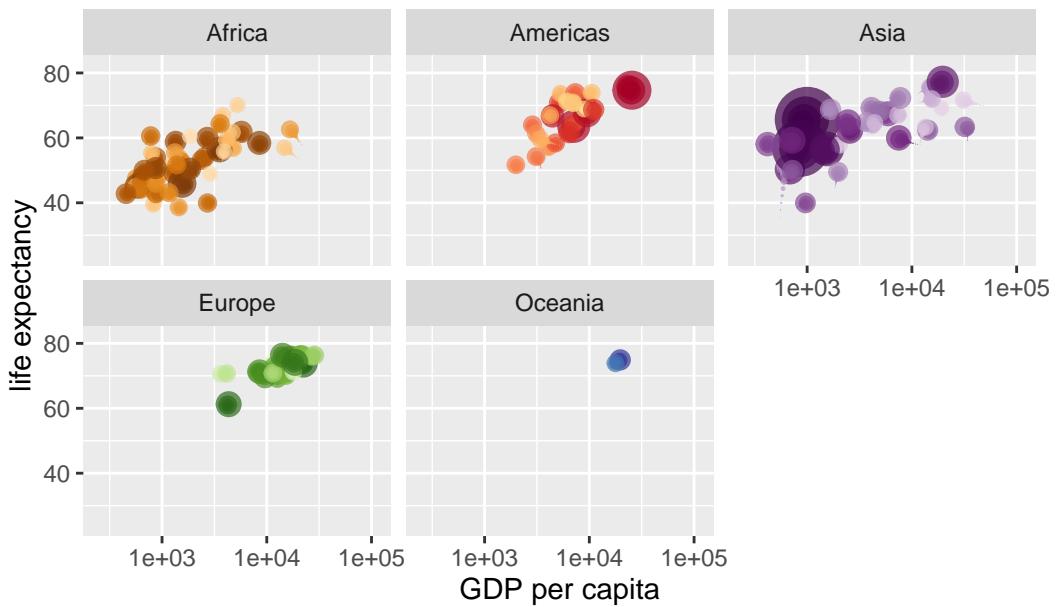
Year: 1981



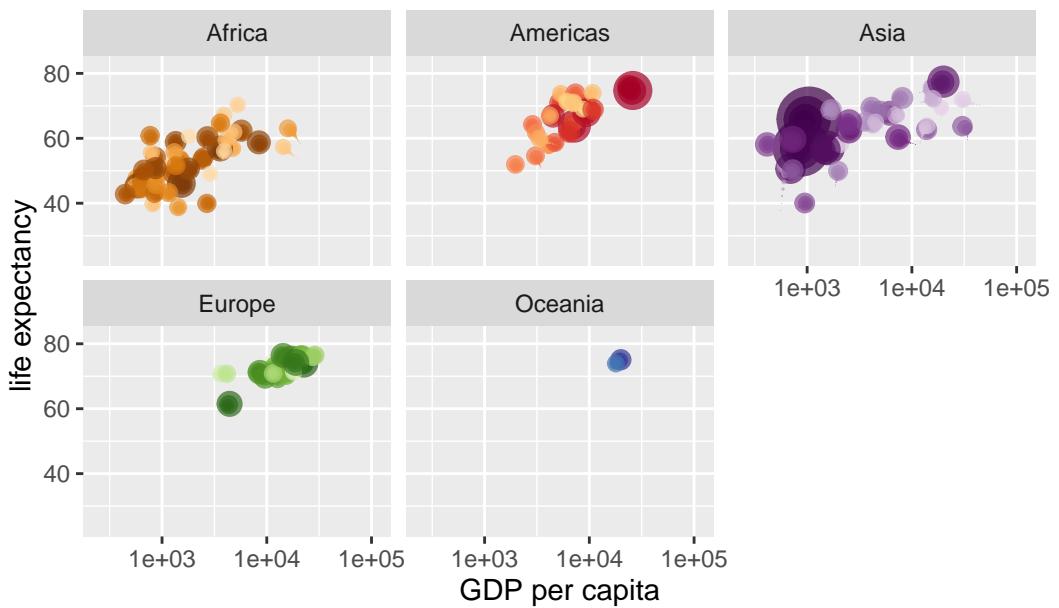
Year: 1982

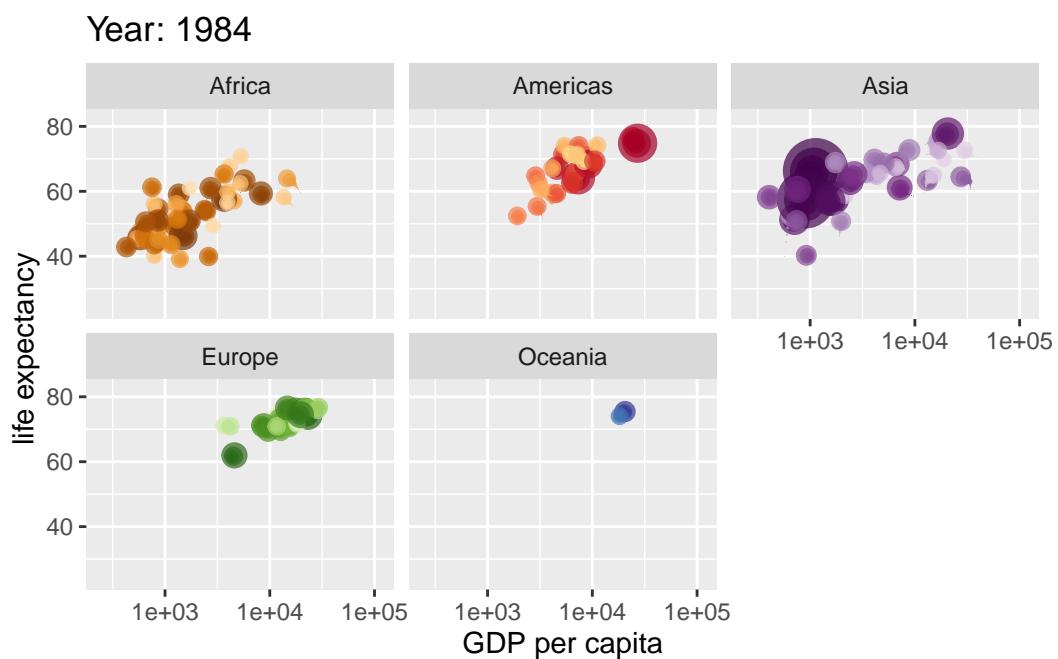
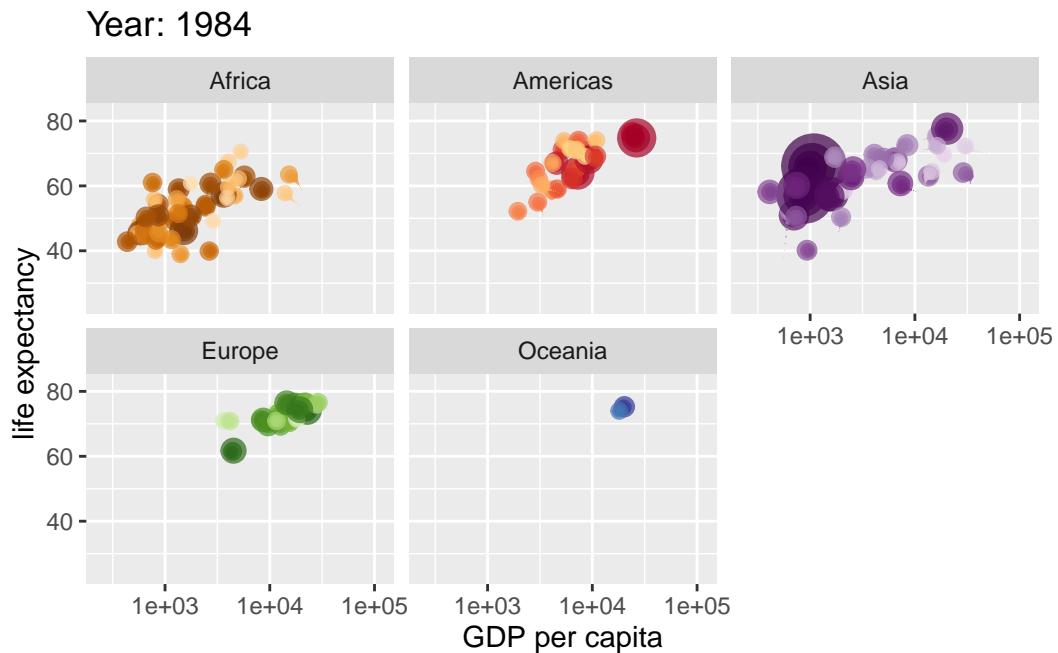


Year: 1983

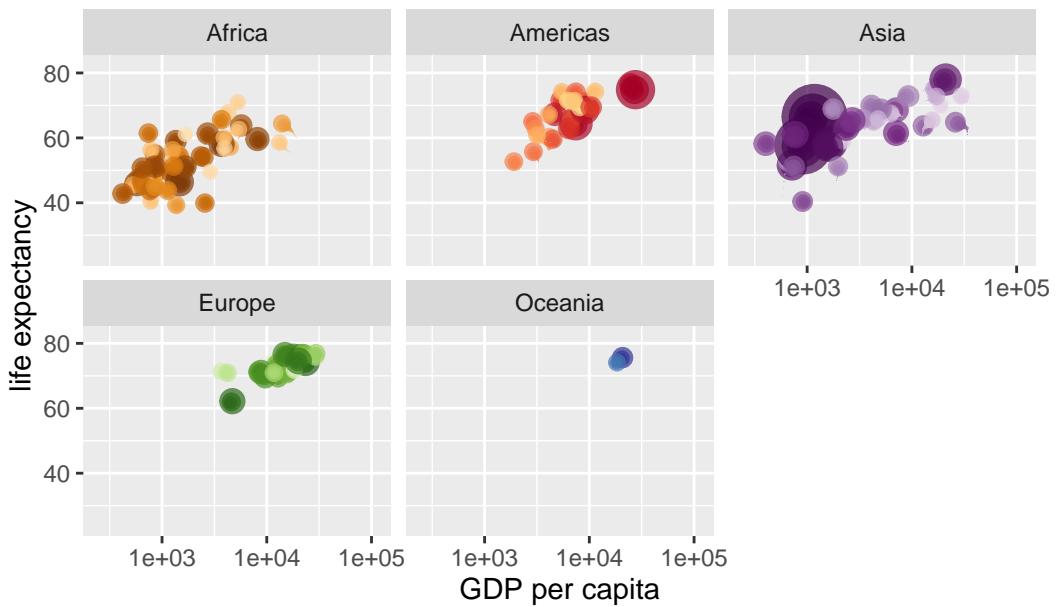


Year: 1983

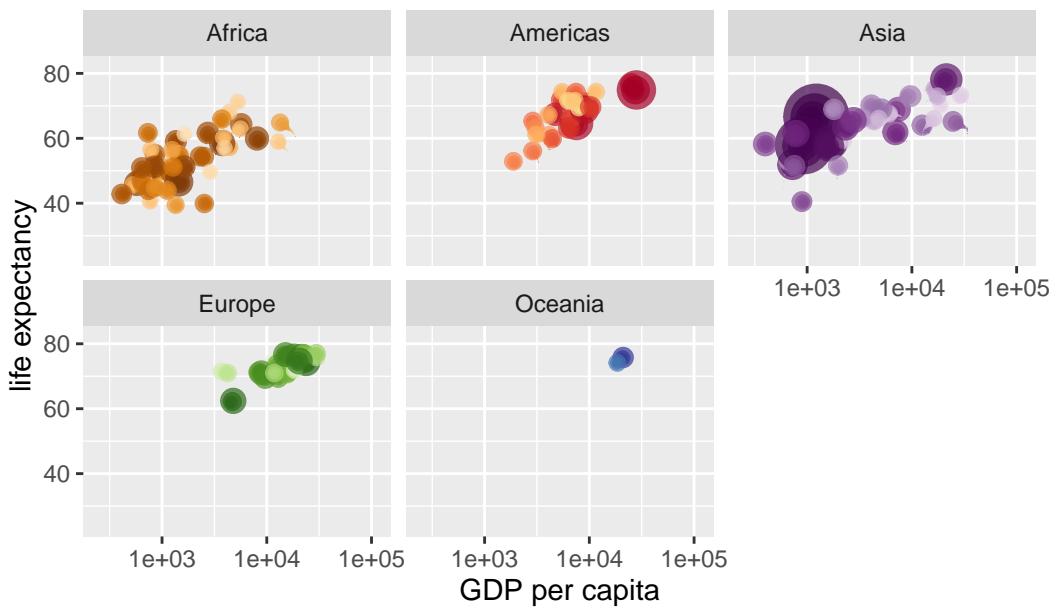




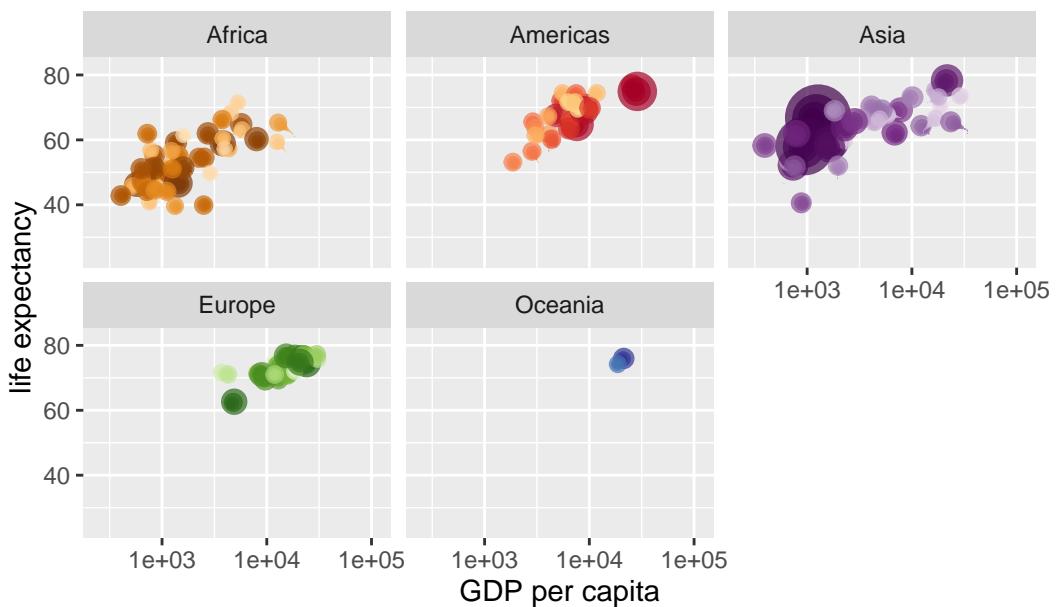
Year: 1985



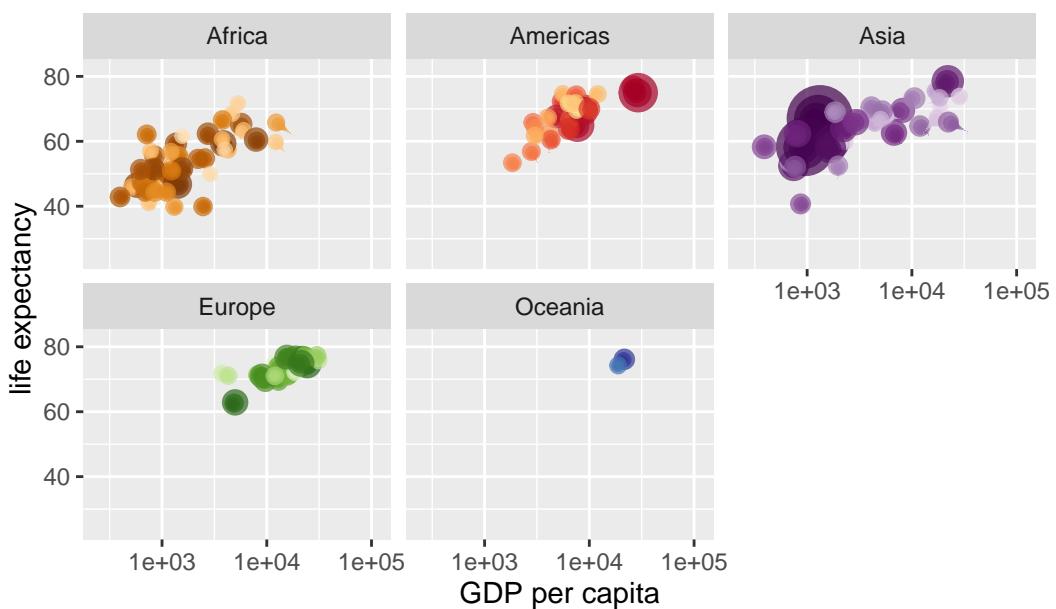
Year: 1985



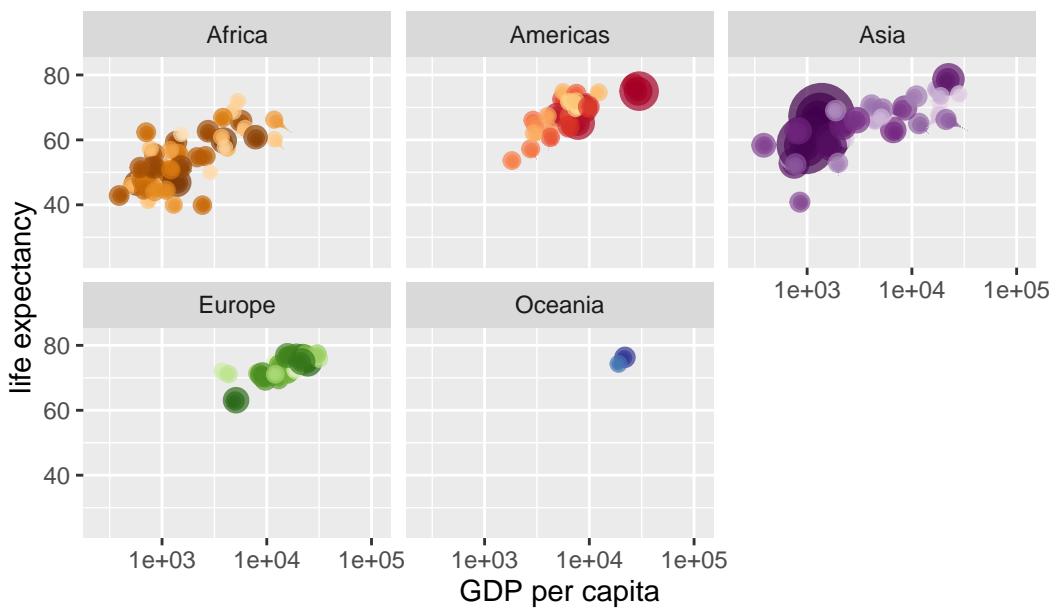
Year: 1986



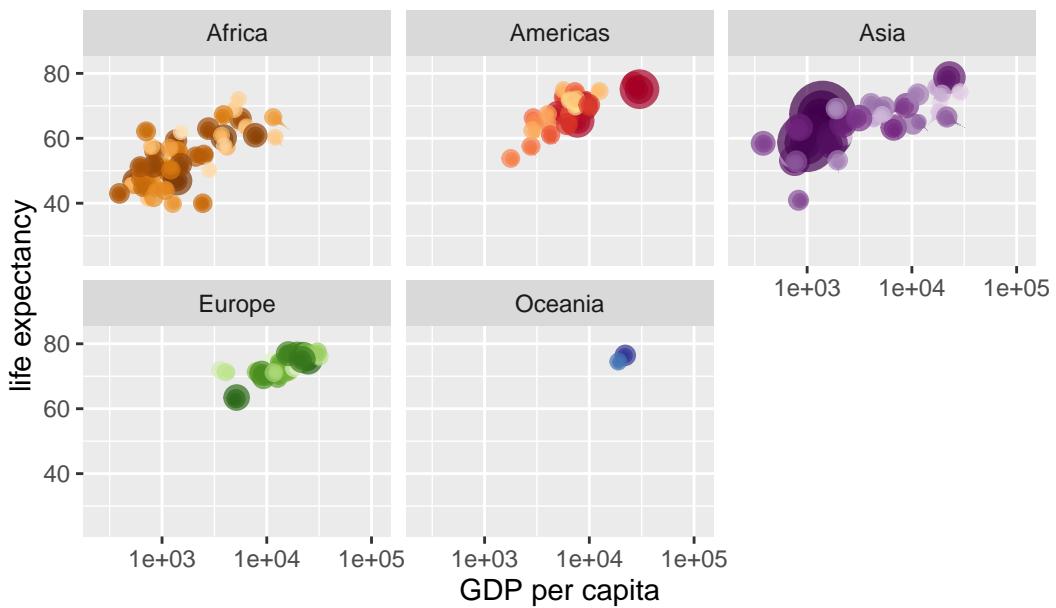
Year: 1986



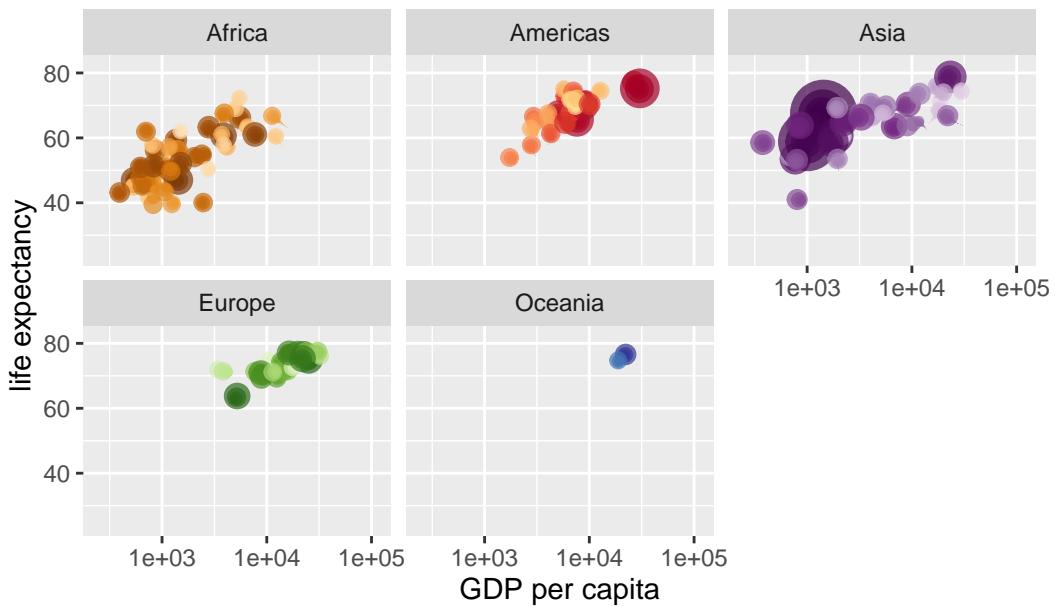
Year: 1987



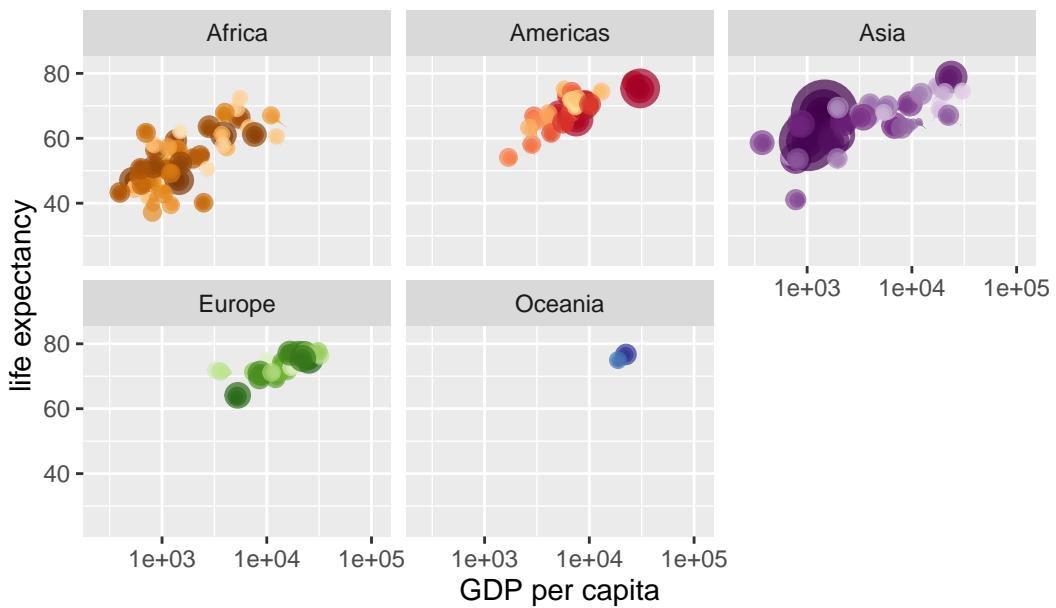
Year: 1988



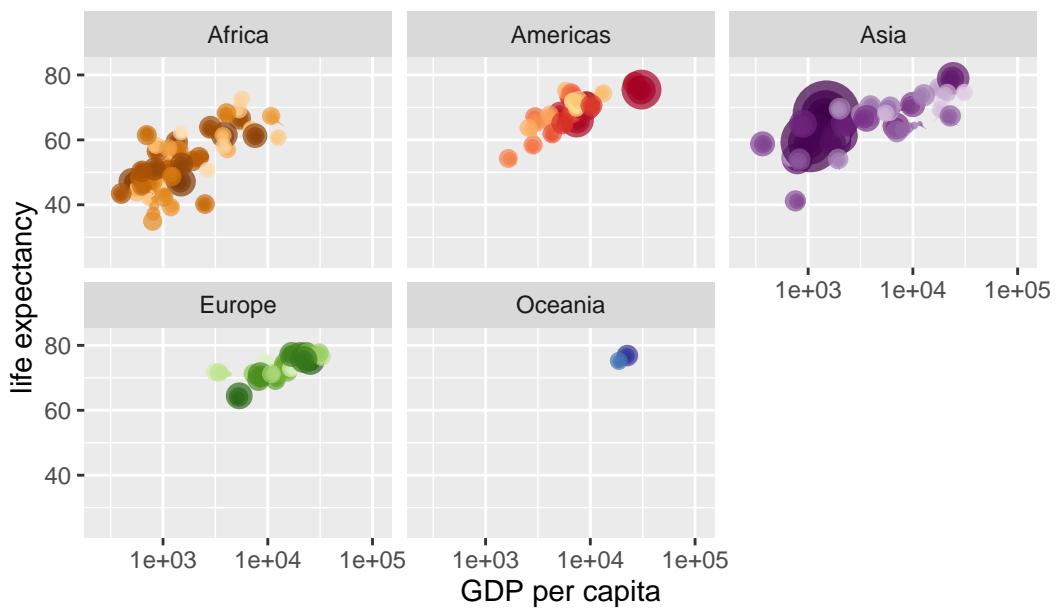
Year: 1988



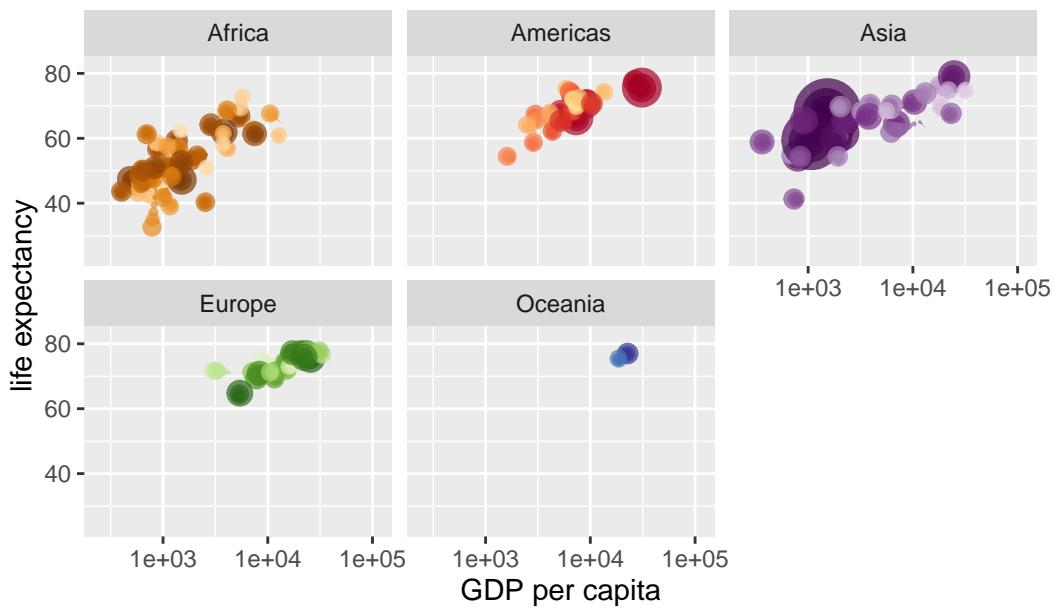
Year: 1989



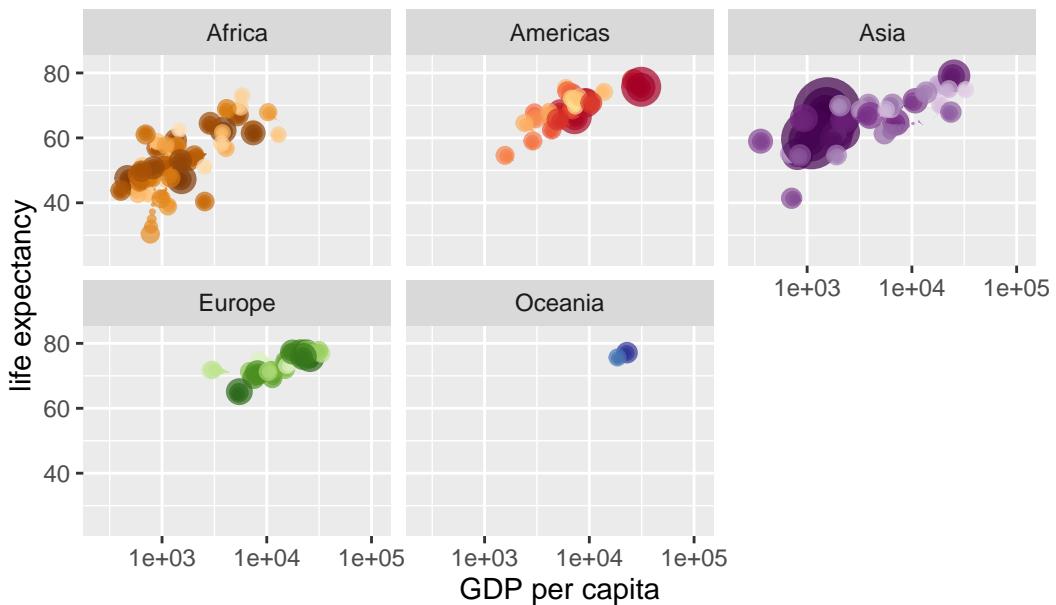
Year: 1989



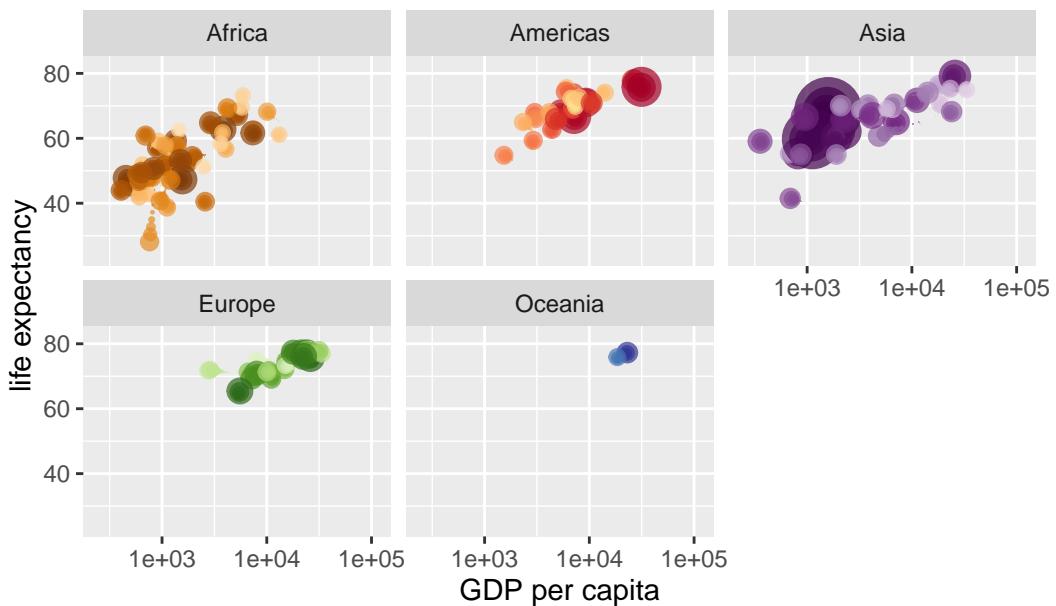
Year: 1990



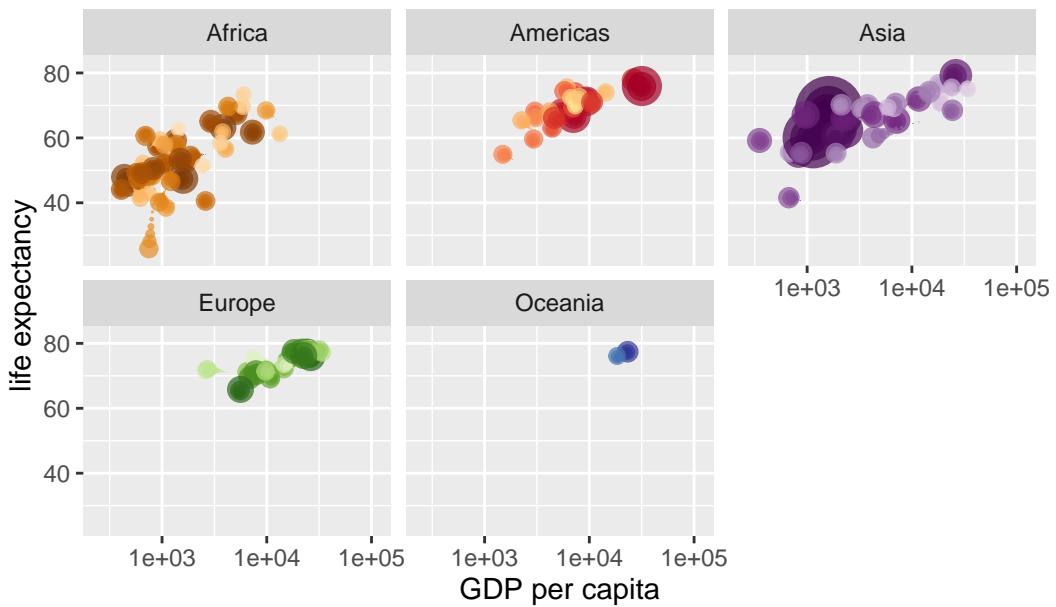
Year: 1990



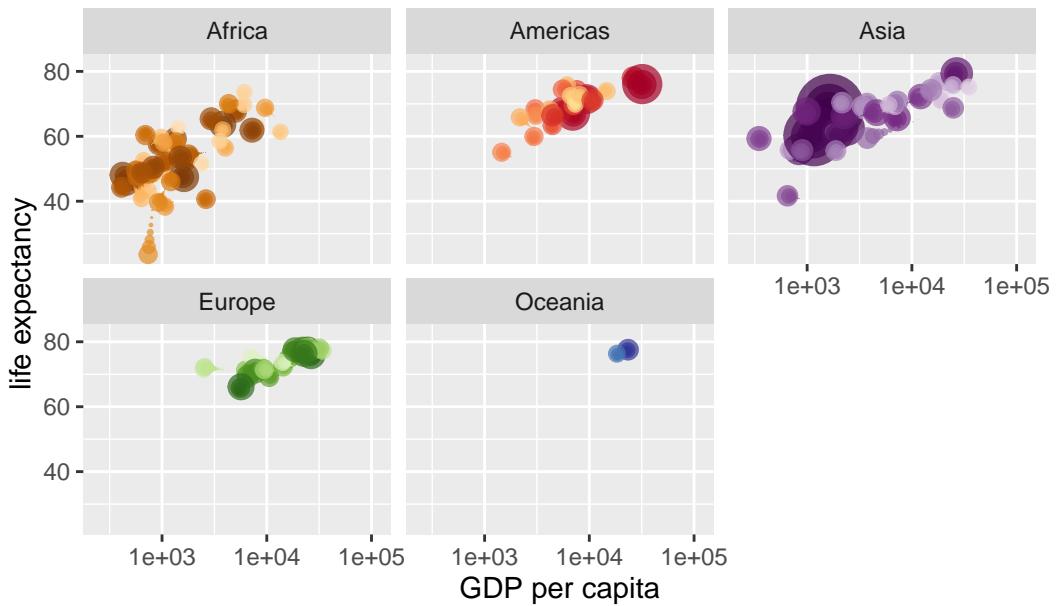
Year: 1991



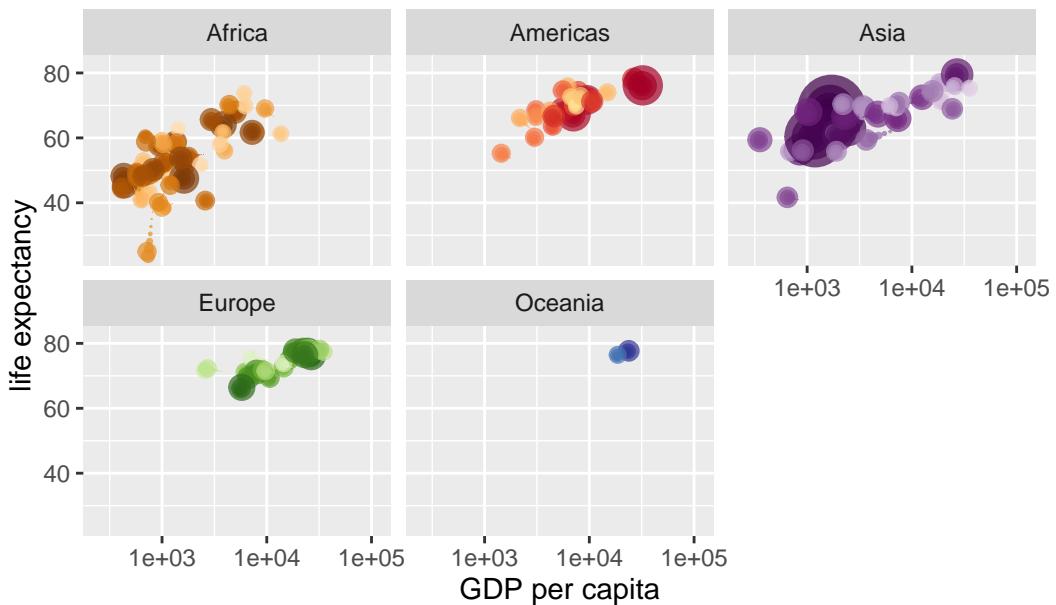
Year: 1991



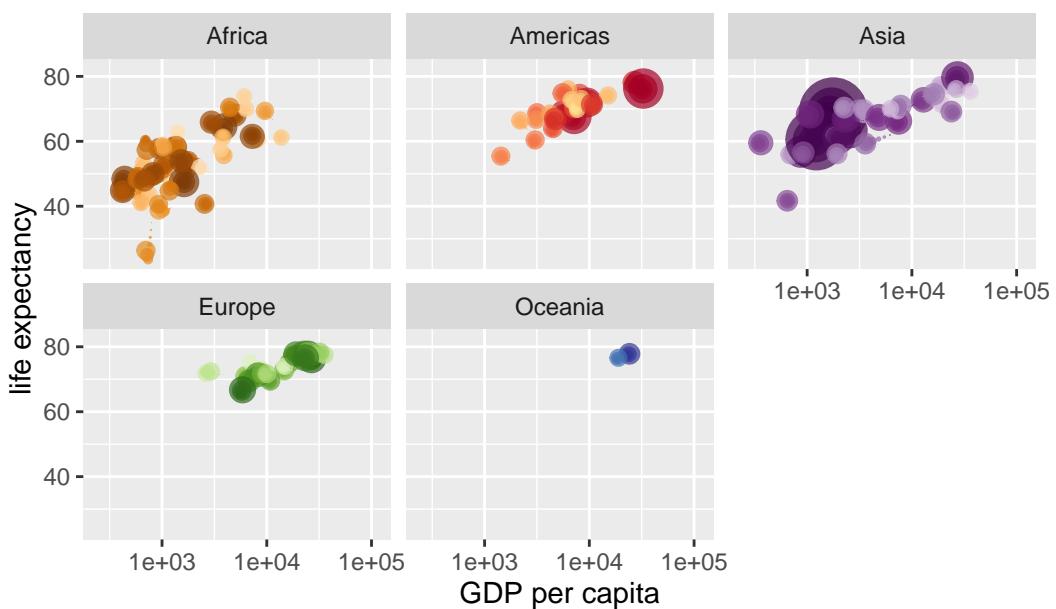
Year: 1992



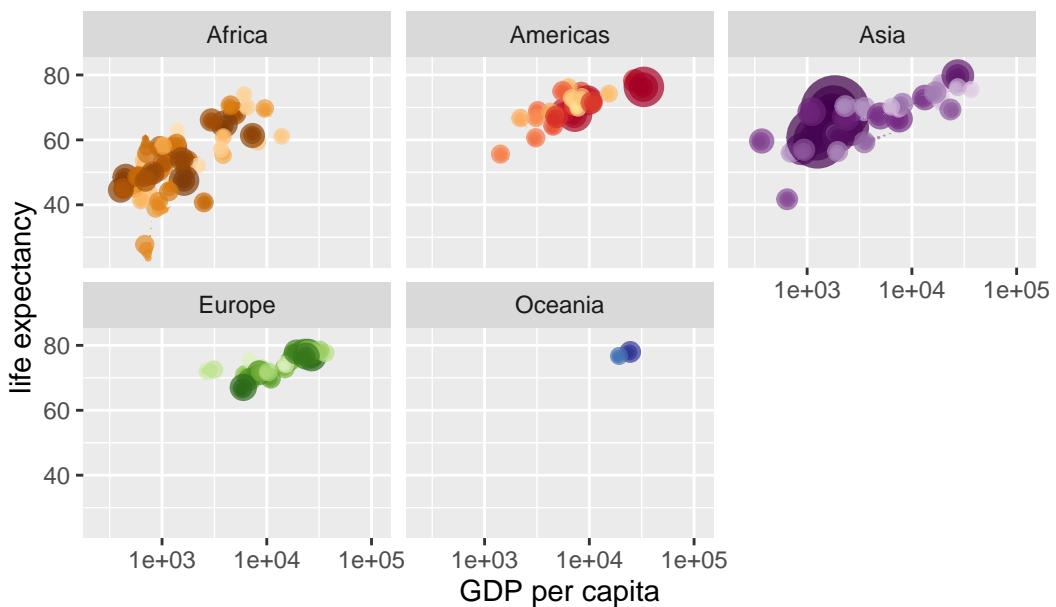
Year: 1993



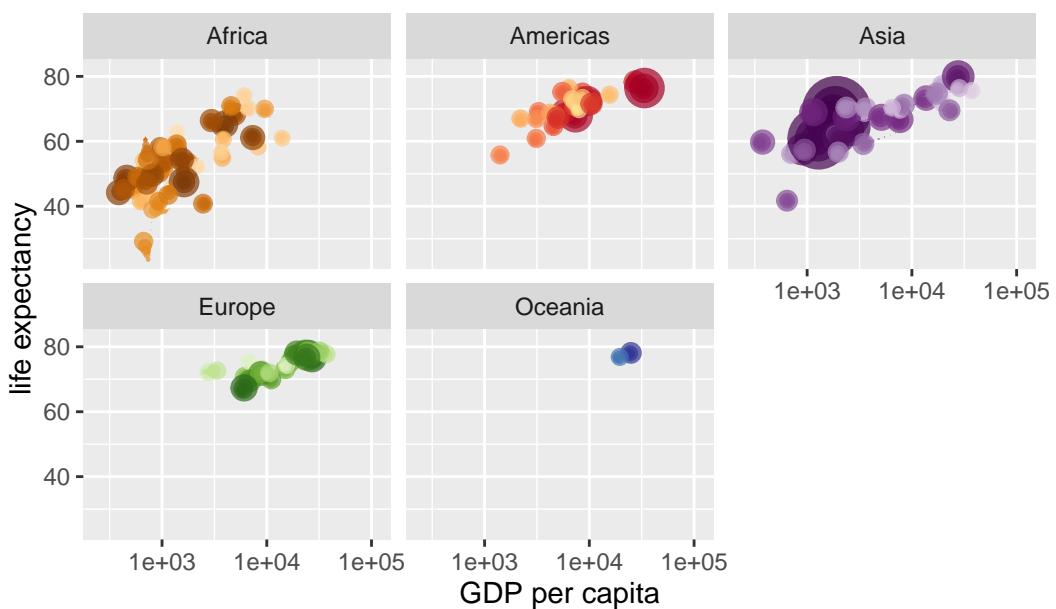
Year: 1993



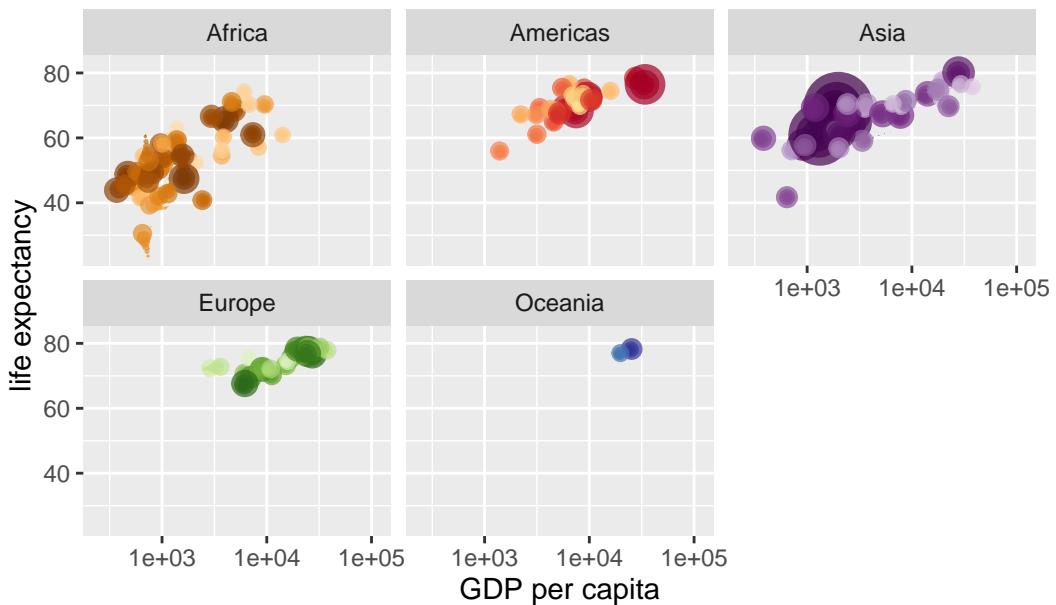
Year: 1994



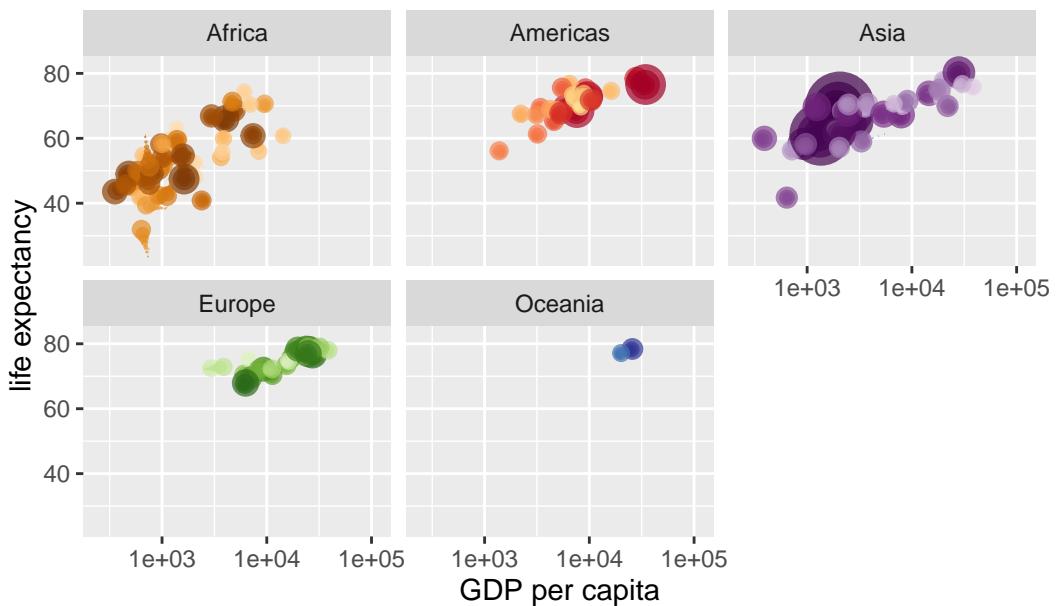
Year: 1994



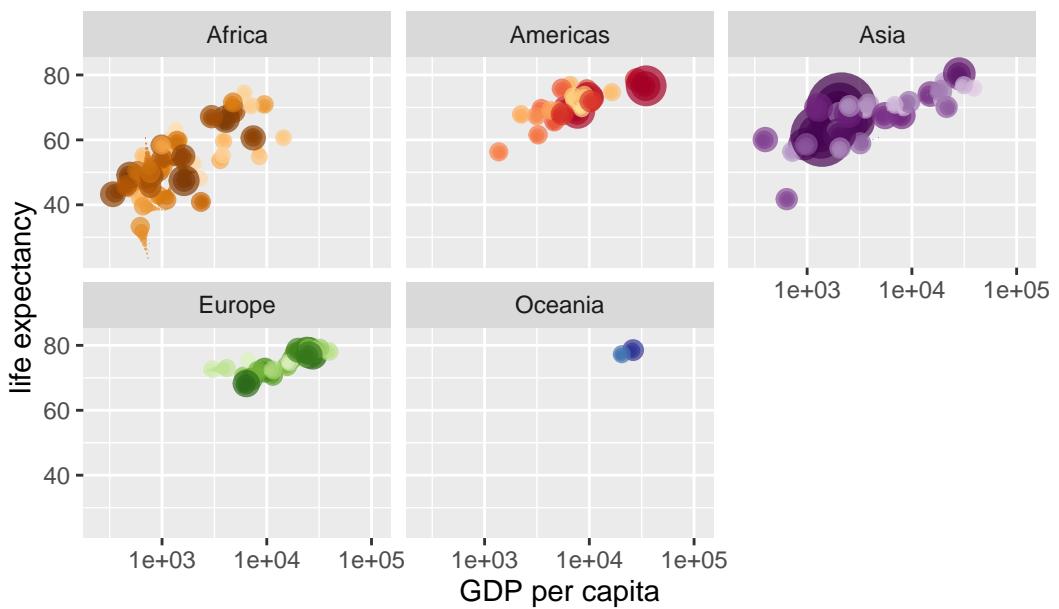
Year: 1995



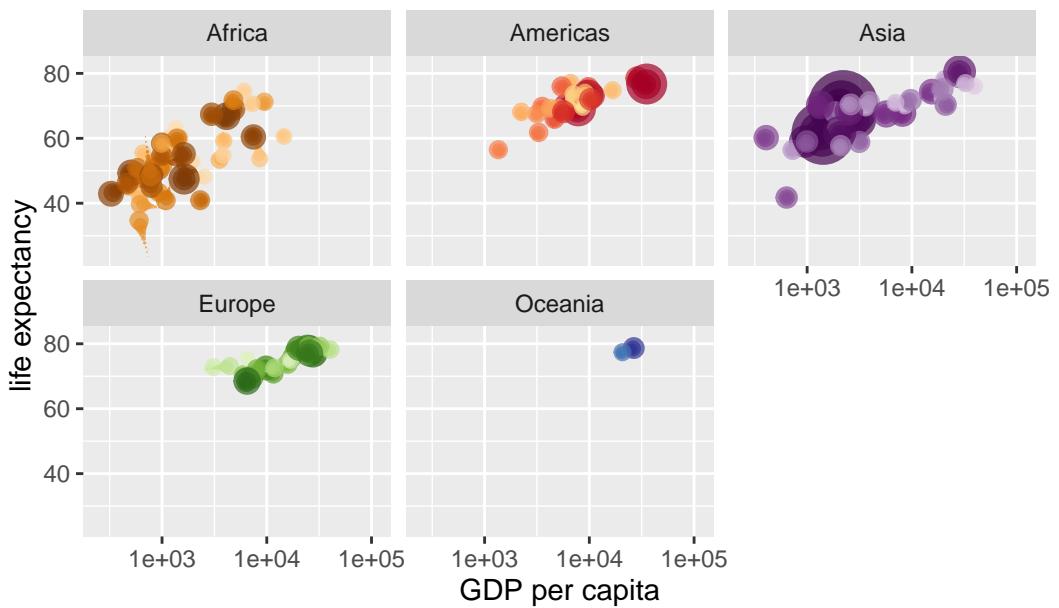
Year: 1995



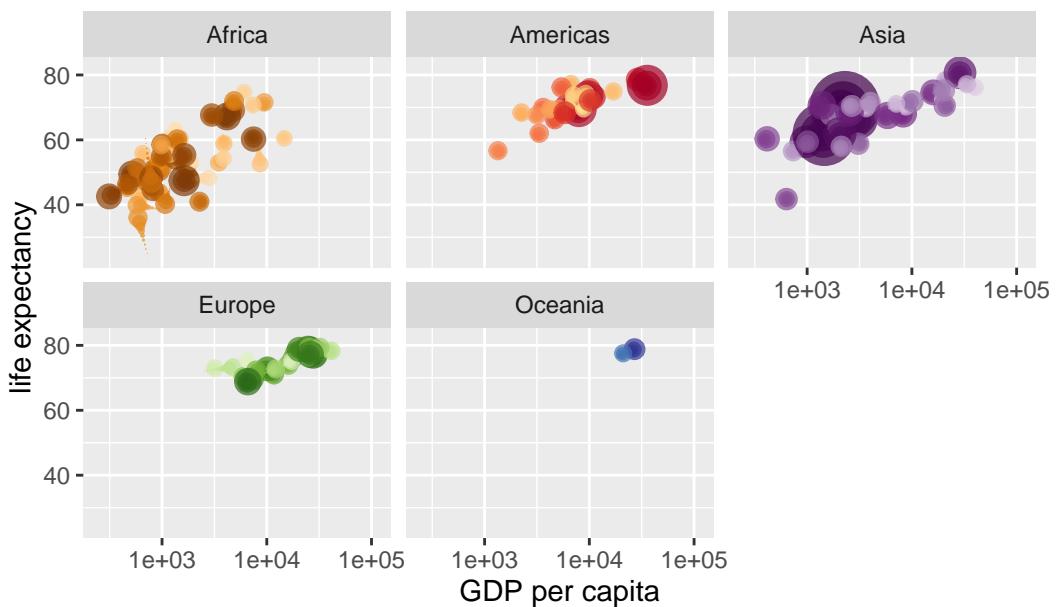
Year: 1996



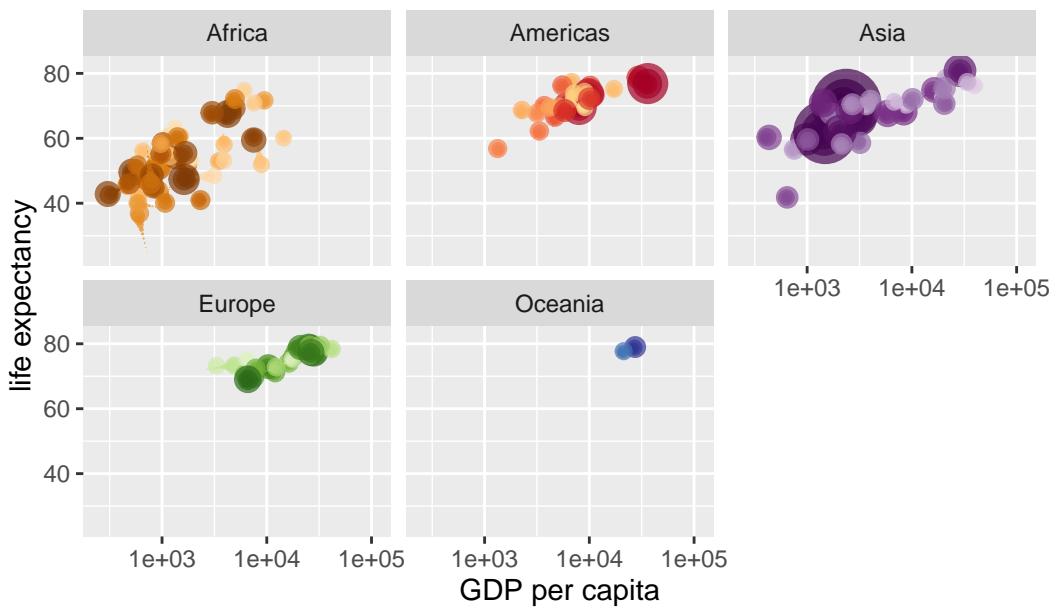
Year: 1996



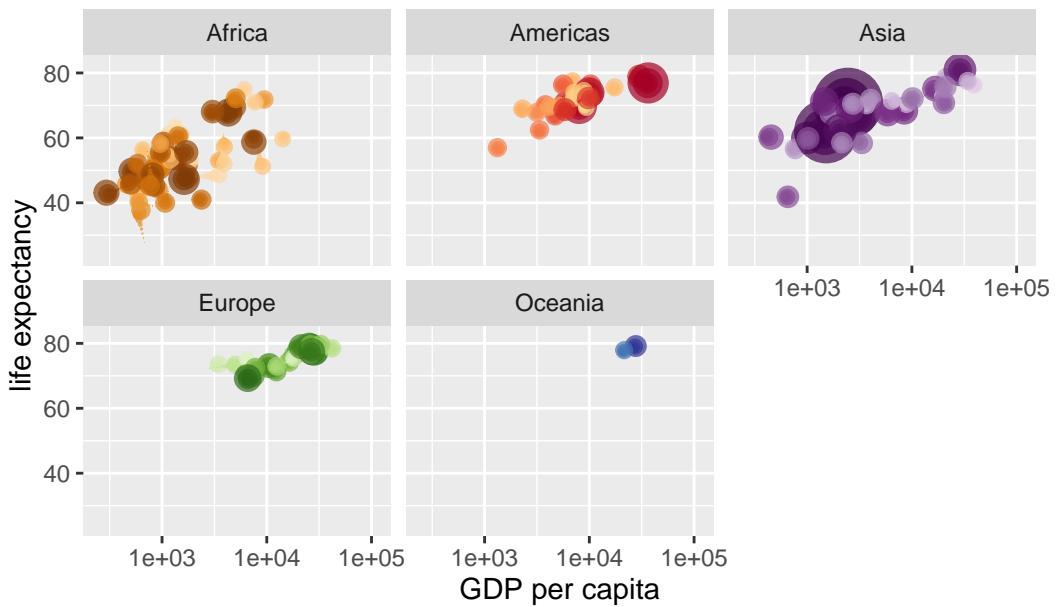
Year: 1997



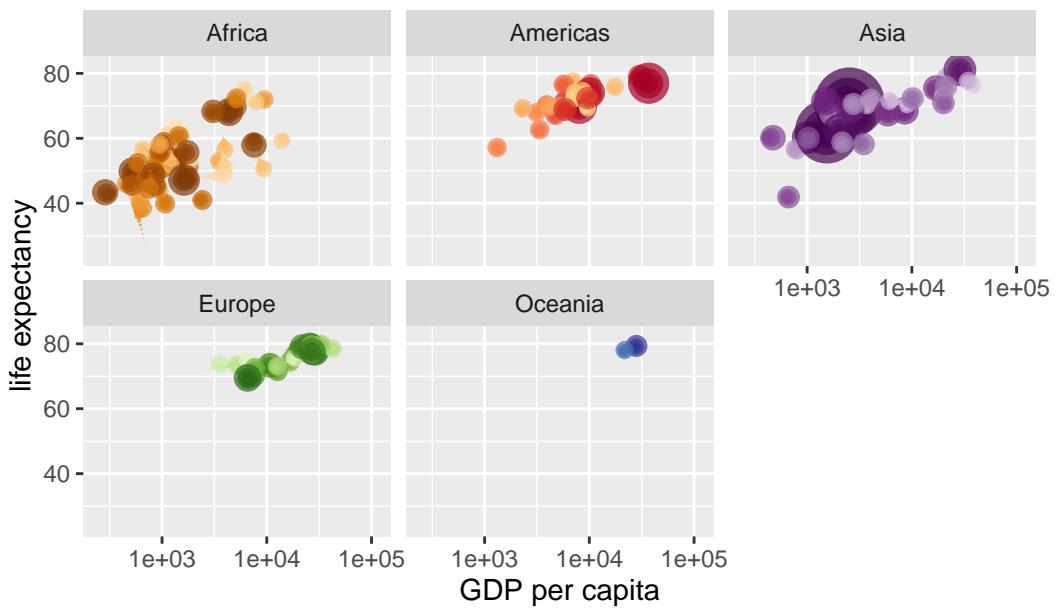
Year: 1998



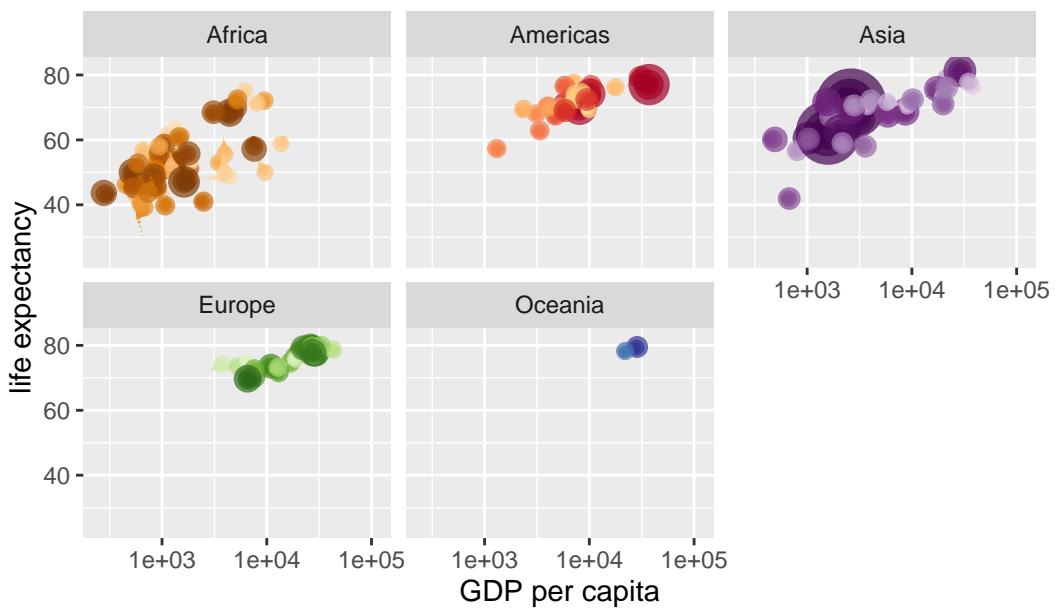
Year: 1998



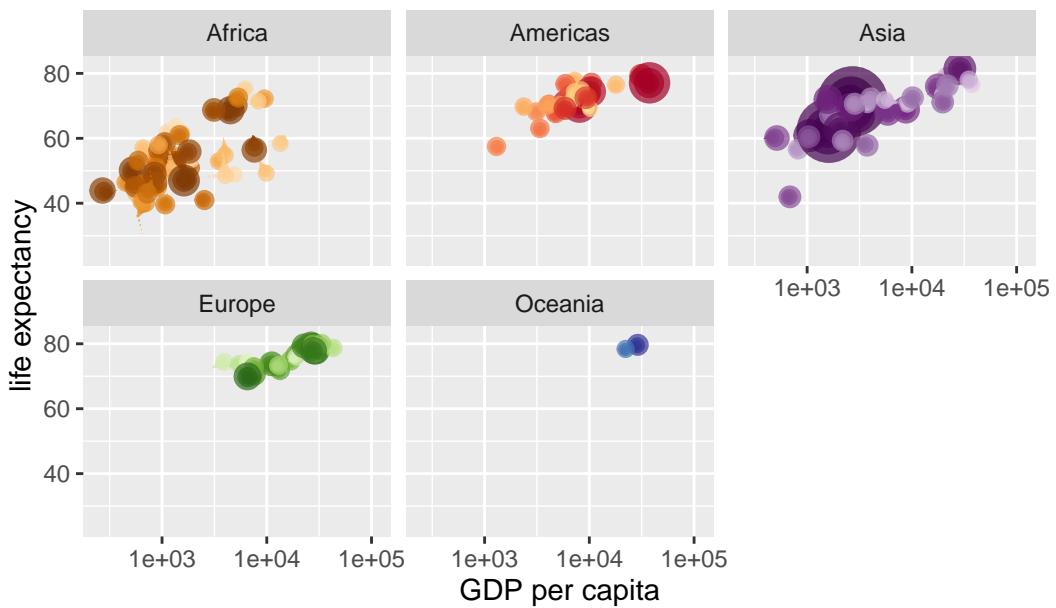
Year: 1999



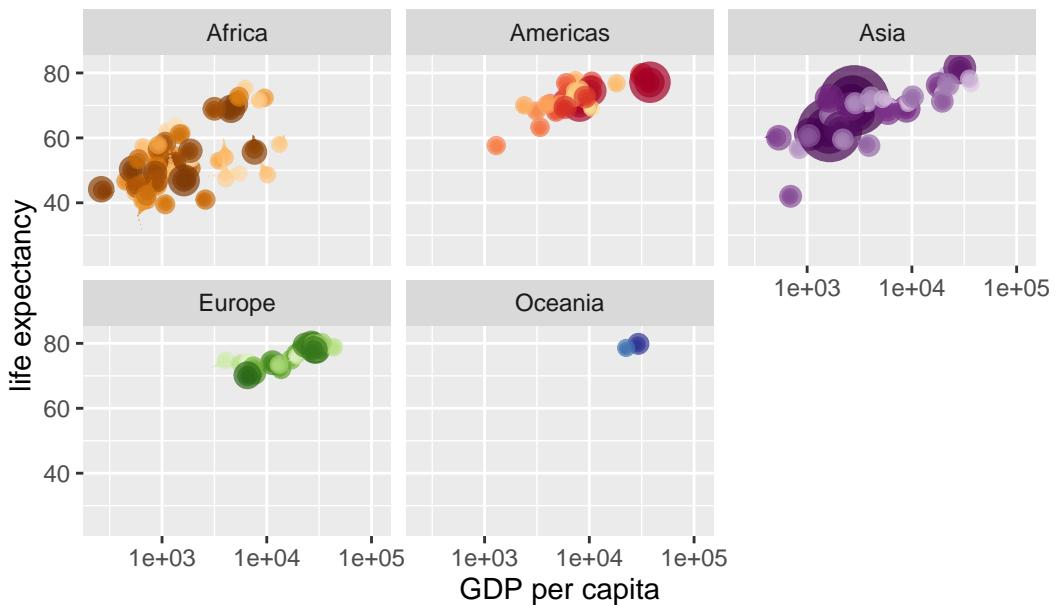
Year: 1999



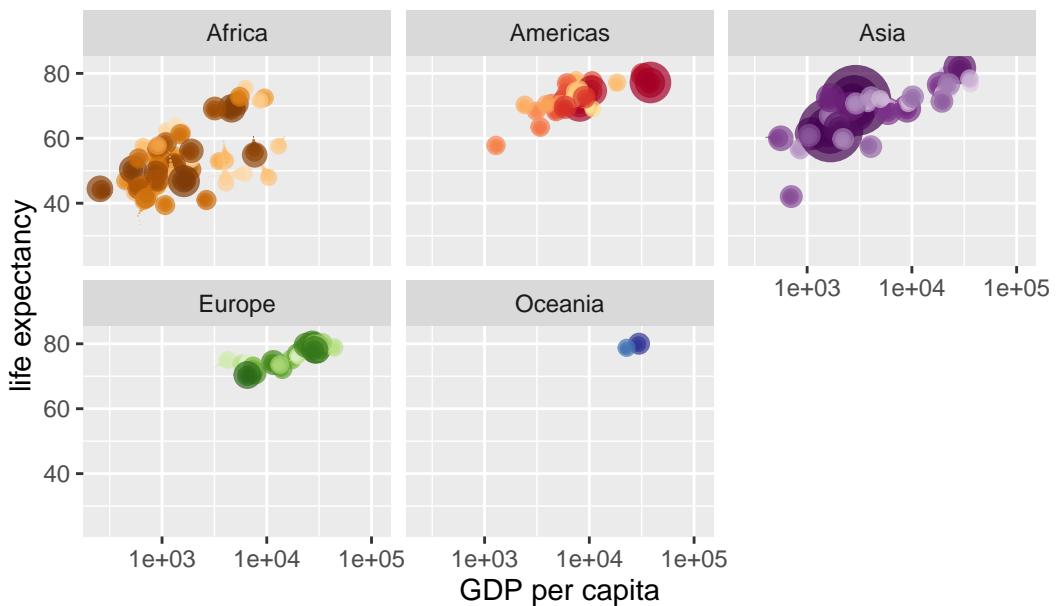
Year: 2000



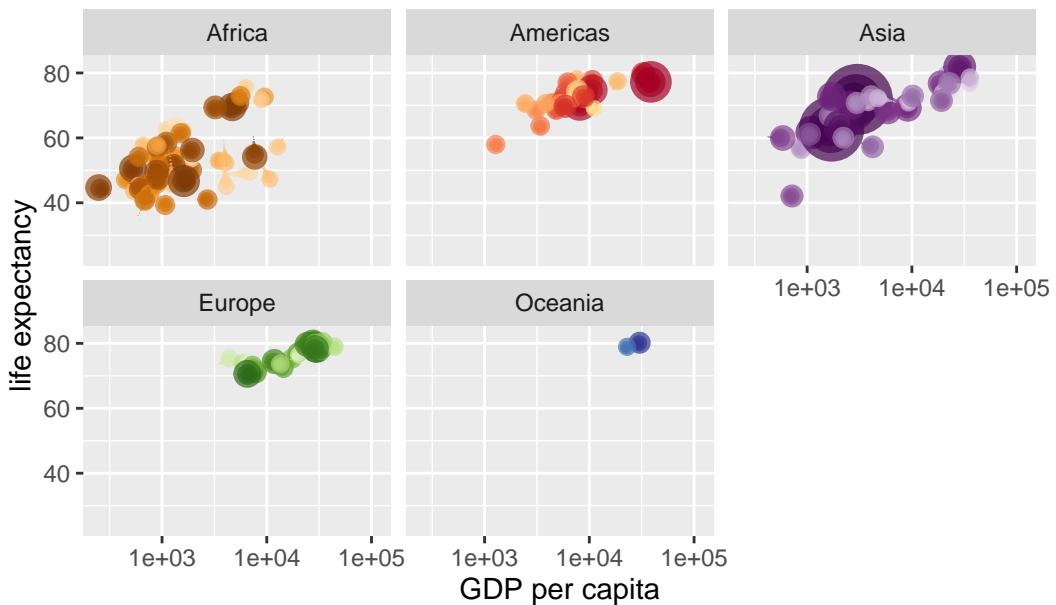
Year: 2000



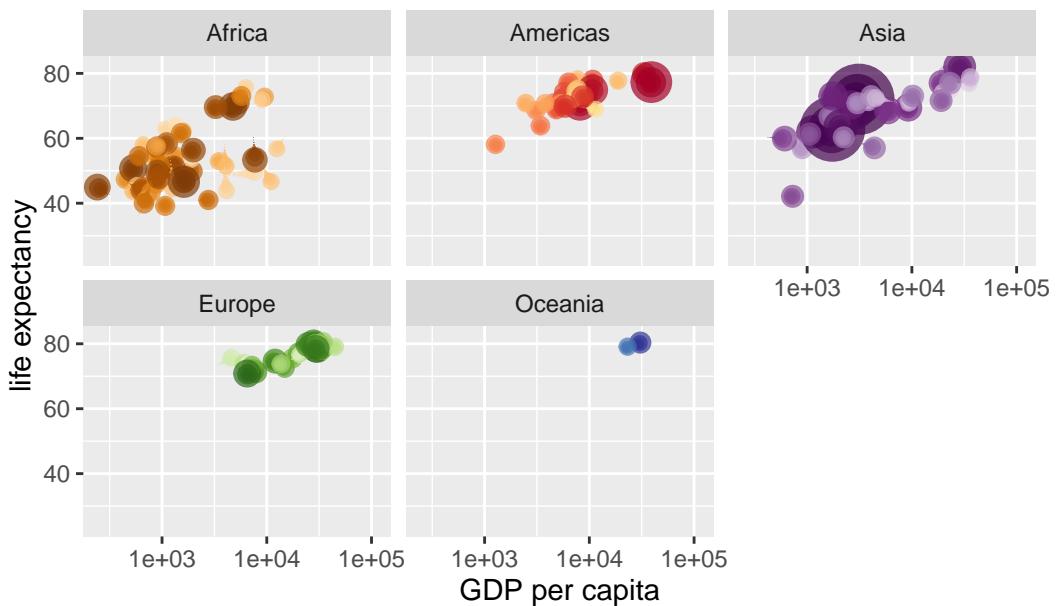
Year: 2001



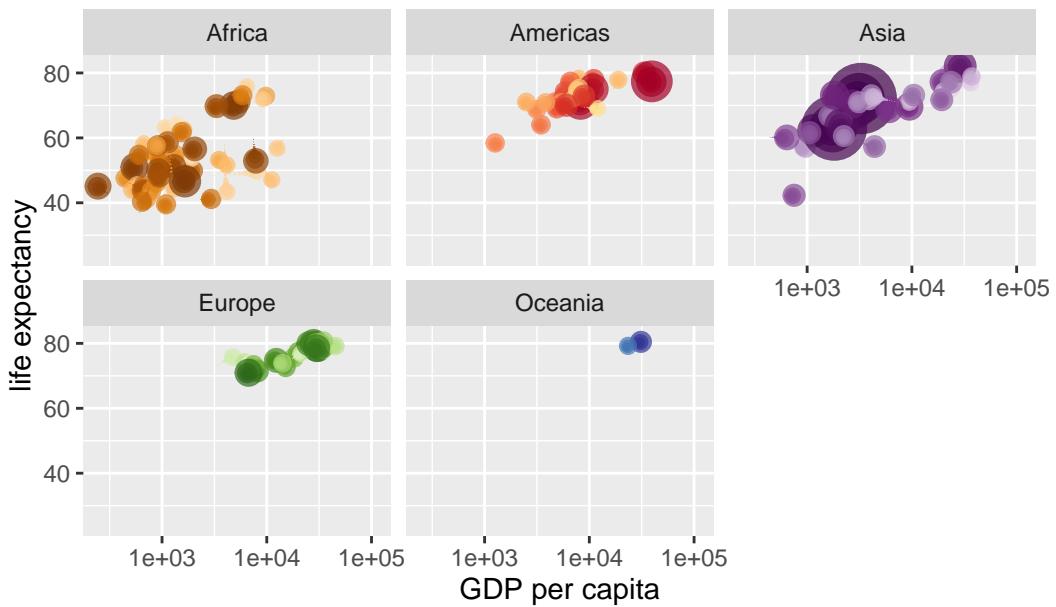
Year: 2001



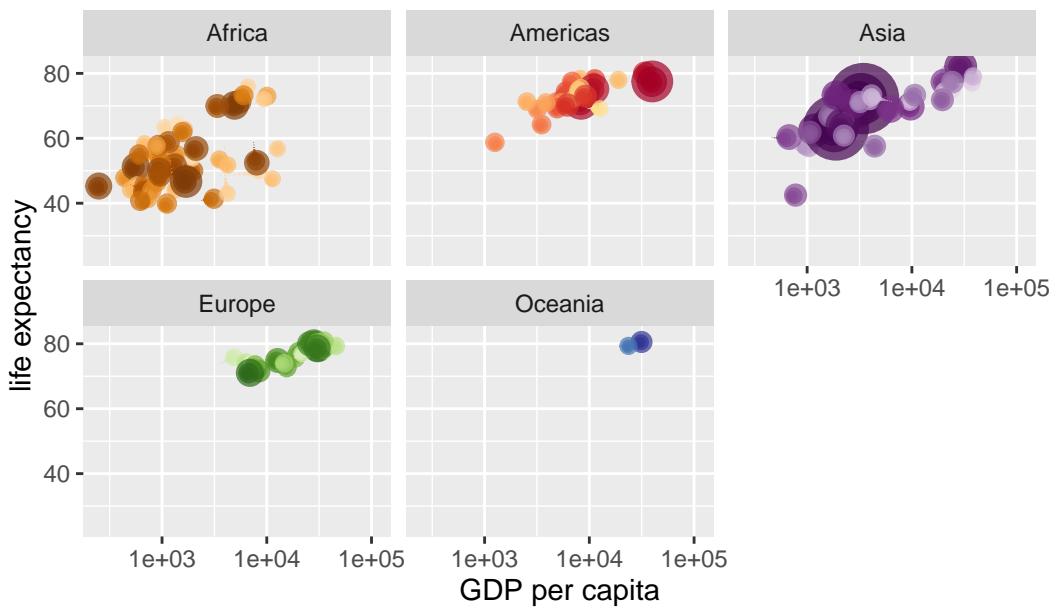
Year: 2002



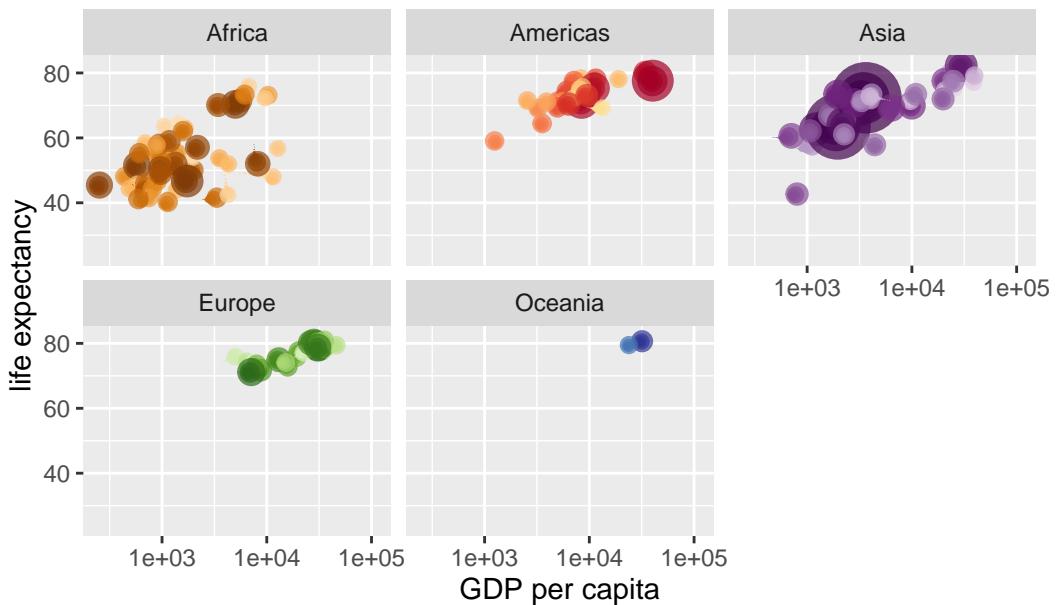
Year: 2003



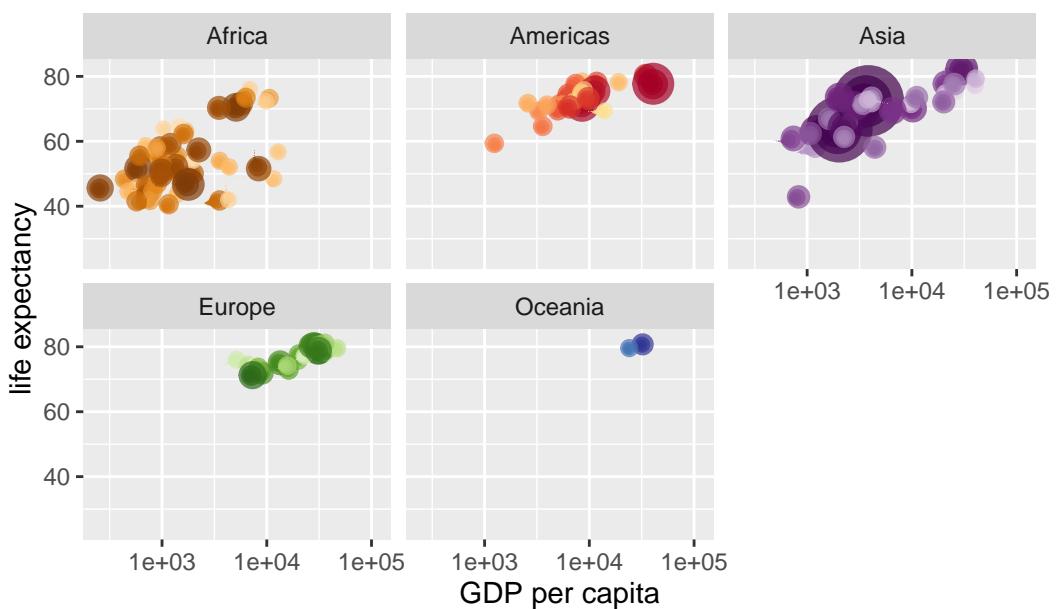
Year: 2003



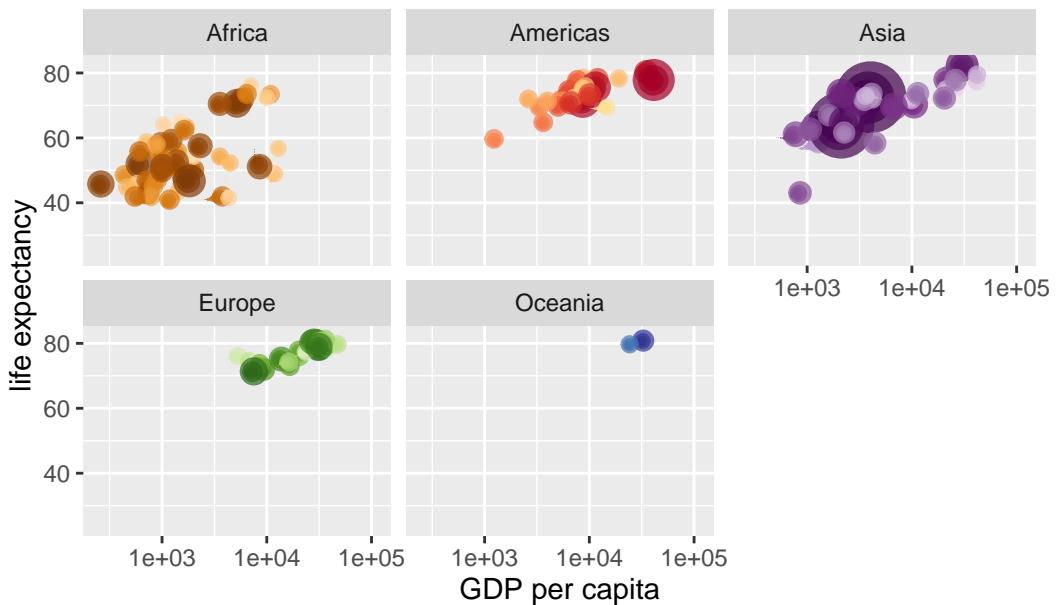
Year: 2004



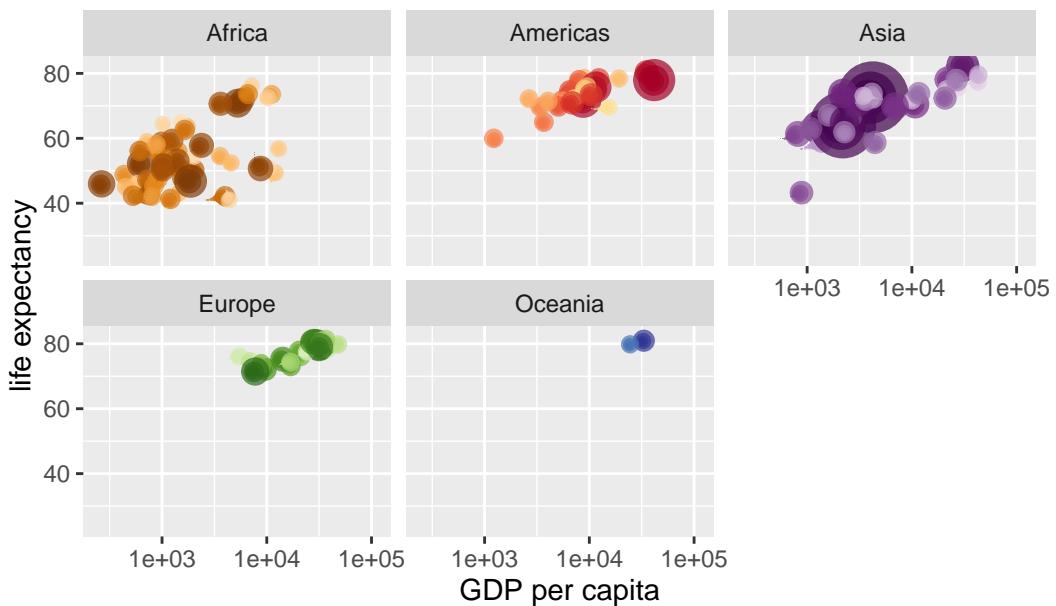
Year: 2004



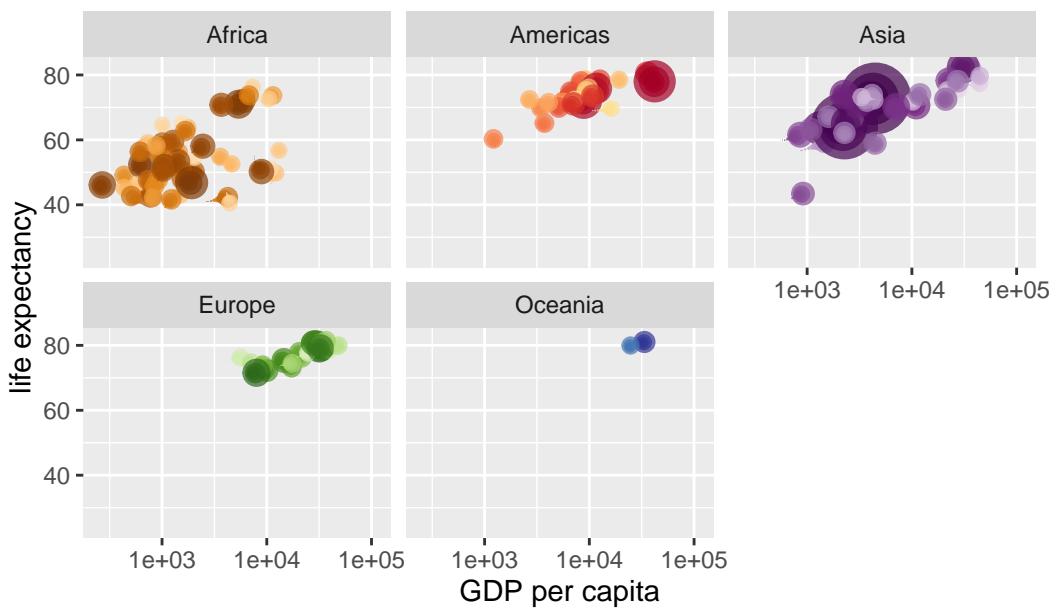
Year: 2005



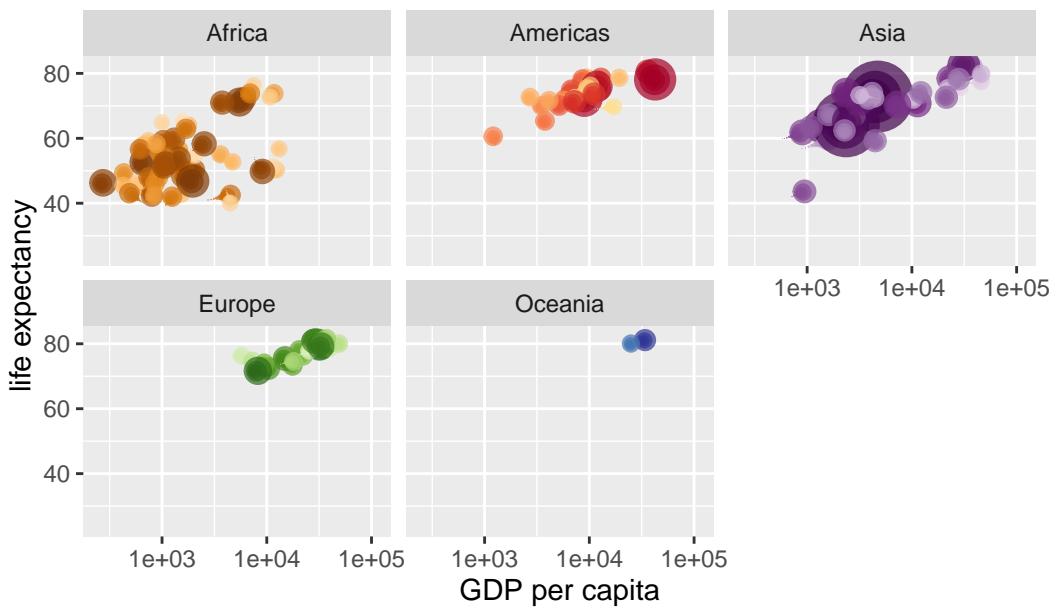
Year: 2005

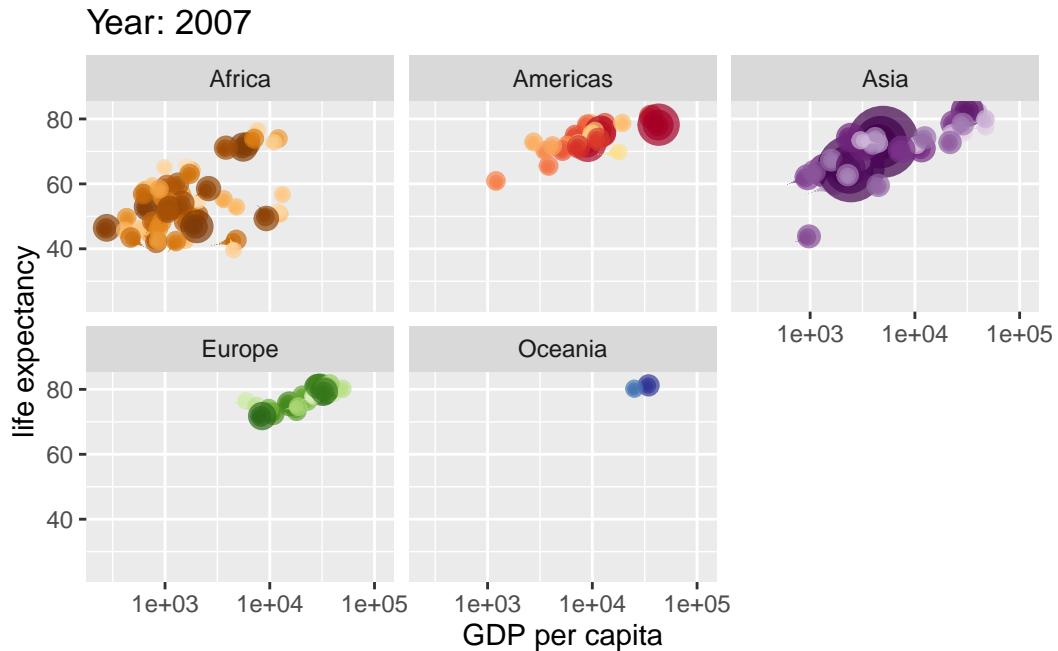


Year: 2006



Year: 2006





```
sessionInfo()
```

```
R version 4.2.1 (2022-06-23)
Platform: aarch64-apple-darwin20 (64-bit)
Running under: macOS Monterey 12.1

Matrix products: default
BLAS:      /Library/Frameworks/R.framework/Versions/4.2-arm64/Resources/lib/libRblas.0.dylib
LAPACK:    /Library/Frameworks/R.framework/Versions/4.2-arm64/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] ganimate_1.0.8  gapminder_0.3.0 dplyr_1.0.10    ggplot2_3.3.6

loaded via a namespace (and not attached):
[1] pillar_1.8.1     compiler_4.2.1   prettyunits_1.1.1 progress_1.2.2
[5] tools_4.2.1      digest_0.6.29    lattice_0.20-45   nlme_3.1-157
```

```
[9] jsonlite_1.8.2      evaluate_0.16      lifecycle_1.0.3    tibble_3.1.8
[13] gtable_0.3.1       mgcv_1.8-40       pkgconfig_2.0.3   rlang_1.0.6
[17] Matrix_1.4-1       cli_3.4.1        yaml_2.3.5       xfun_0.33
[21] fastmap_1.1.0      withr_2.5.0       stringr_1.4.1    knitr_1.40
[25] hms_1.1.2          generics_0.1.3    vctrs_0.4.2       grid_4.2.1
[29] tidyselect_1.2.0    glue_1.6.2        R6_2.5.1         gifski_1.6.6-1
[33] fansi_1.0.3        rmarkdown_2.16    tweenr_2.0.2     farver_2.1.1
[37] magrittr_2.0.3      ellipsis_0.3.2    splines_4.2.1    scales_1.2.1
[41] htmltools_0.5.3    colorspace_2.0-3  labeling_0.4.2   utf8_1.2.2
[45] stringi_1.7.8      munsell_0.5.0     crayon_1.5.2
```