HM06_final

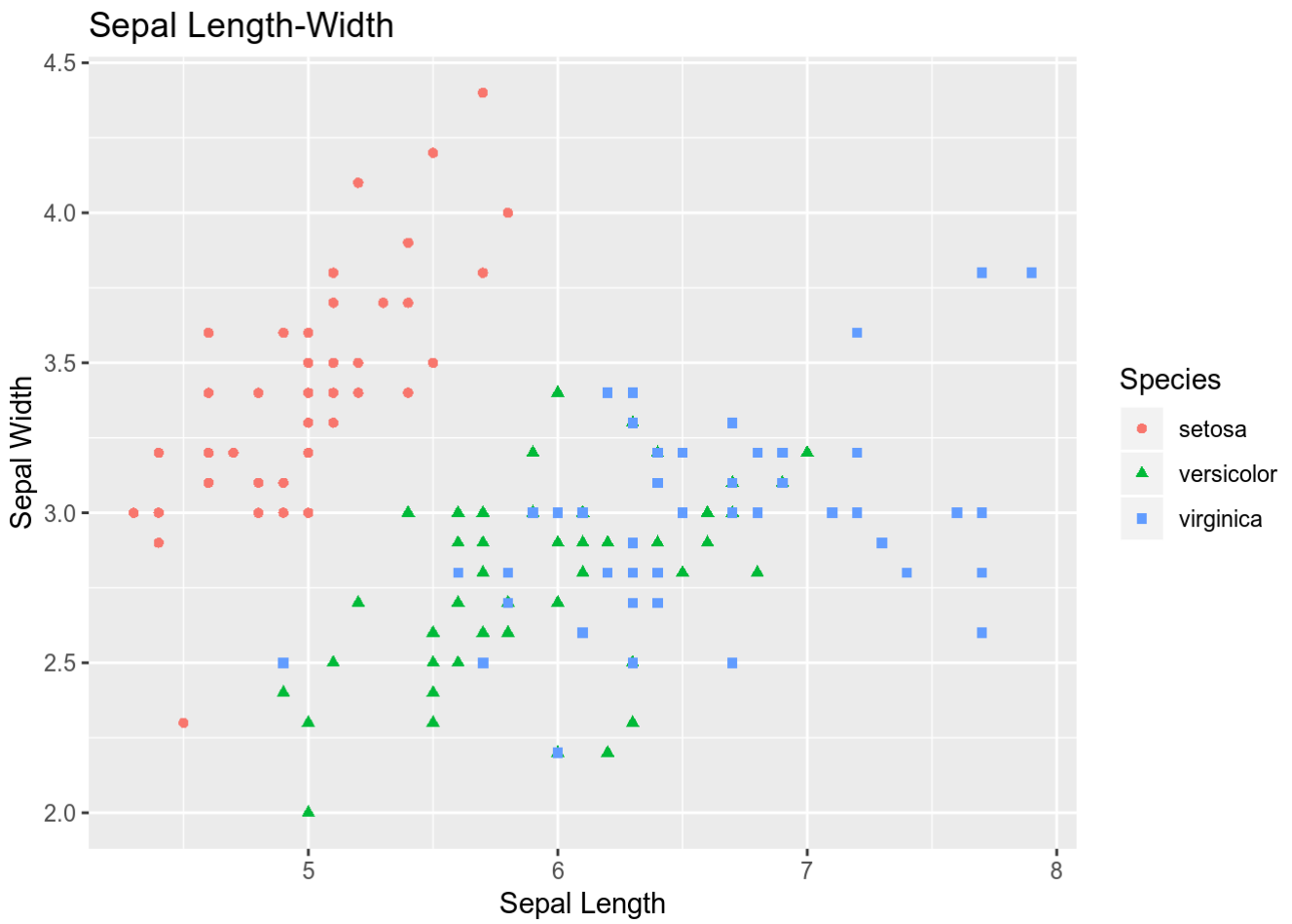
```
#Data organizing:
library(dplyr)
library(data.table)
library(tidyr)

#Palettes and visualization:
library(ggplot2)

#Data
library(gapminder)
```

```
#The iris data are organized like this:
head(iris)
```

	Sepal.Length <dbl>	Sepal.Width <dbl>	Petal.Length <dbl>	Petal.Width <dbl>	Species <fctr>
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

6 rows

```
#plot iris data
scatter <- ggplot(data=iris, aes(x = Sepal.Length, y = Sepal.Width))
scatter + geom_point(aes(color=Species, shape=Species)) +
  xlab("Sepal Length") +  ylab("Sepal Width") +
  ggtitle("Sepal Length-Width")
```




```
#But in order to capitalize on ggplot functionality, we need to reorganize the data s
o each row only has data for a single trait, this is known as "long" format, where ea
ch row only has a single trait observation. To make this data conversion, we use the
 the gather function:
# data: The dataset to be modified
# key: the name of the new "naming" variable
# value: the name of the new "result" variable
#exclude:-z

iris_long <- gather(data = iris,
                    key = trait,
                    value = measurement,
                    -Species)  #exclude
head(iris_long)
```

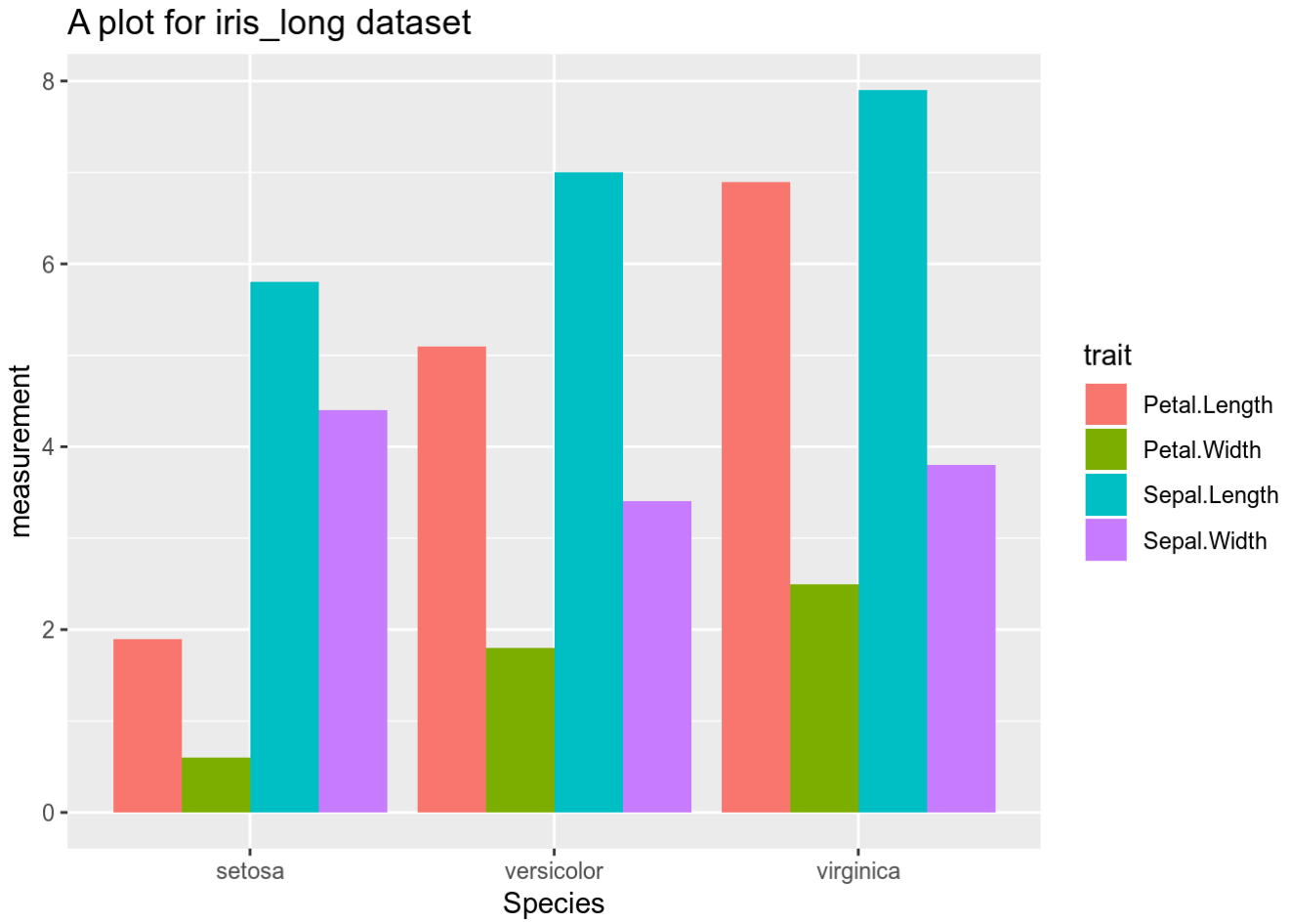
	Species <fctr>	trait <chr>	measurement <dbl>
1	setosa	Sepal.Length	5.1
2	setosa	Sepal.Length	4.9
3	setosa	Sepal.Length	4.7
4	setosa	Sepal.Length	4.6
5	setosa	Sepal.Length	5.0
6	setosa	Sepal.Length	5.4

6 rows

```
colnames(iris_long)
```

```
## [1] "Species"     "trait"       "measurement"
```

```
## Plot the data:
bar <- ggplot(data=iris_long, aes(x=Species, y=measurement, fill=trait))
bar + geom_bar(stat="identity", position="dodge")+
  ggtitle("A plot for iris_long dataset")
```




```
## look at the dataset:
head(gapminder)
```

country <fctr>	continent <fctr>	year <int>	lifeExp <dbl>	pop <int>	gdpPercap <dbl>
Afghanistan	Asia	1952	28.801	8425333	779.4453
Afghanistan	Asia	1957	30.332	9240934	820.8530
Afghanistan	Asia	1962	31.997	10267083	853.1007
Afghanistan	Asia	1967	34.020	11537966	836.1971
Afghanistan	Asia	1972	36.088	13079460	739.9811
Afghanistan	Asia	1977	38.438	14880372	786.1134

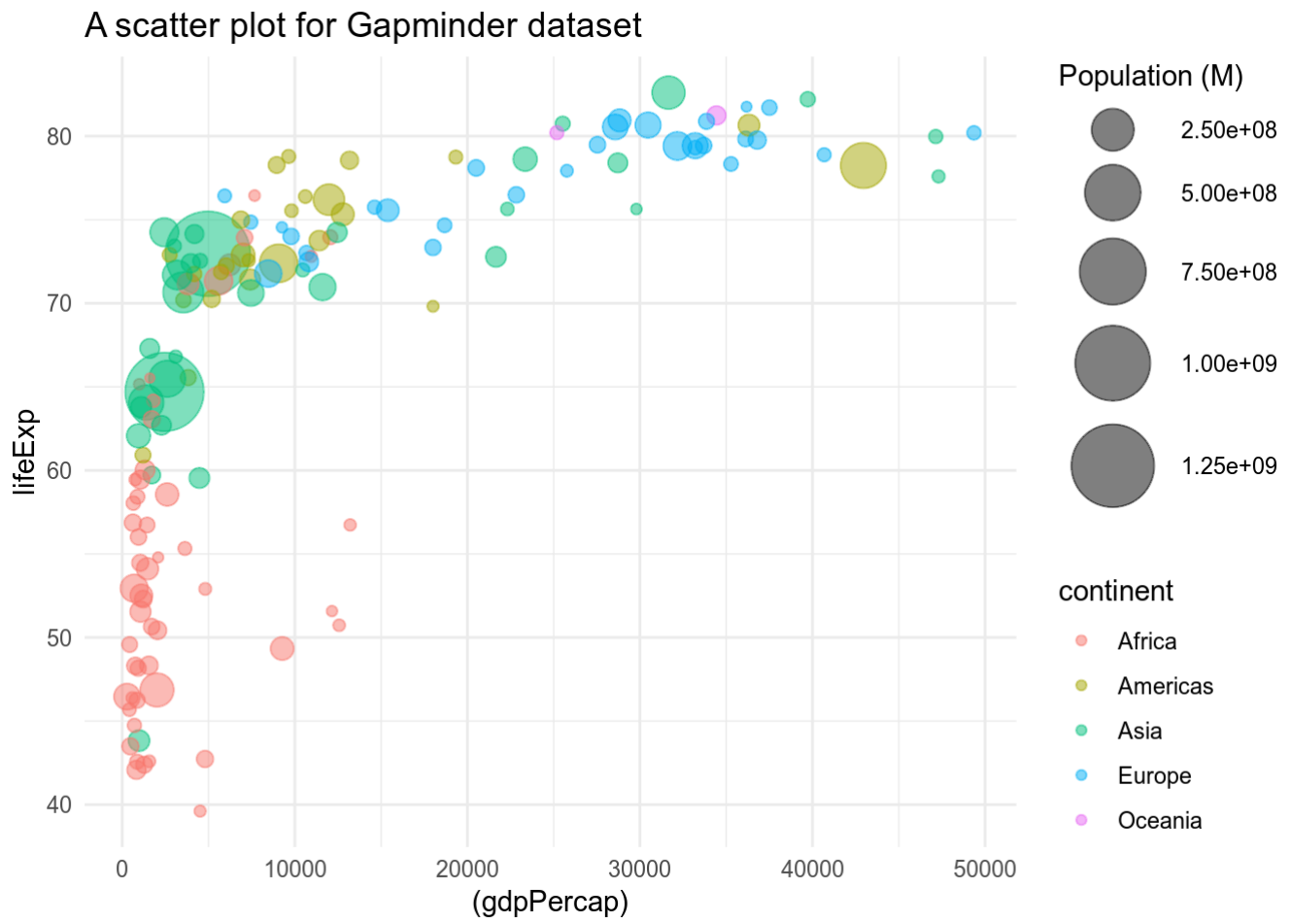
6 rows

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

gapminder_2007
```

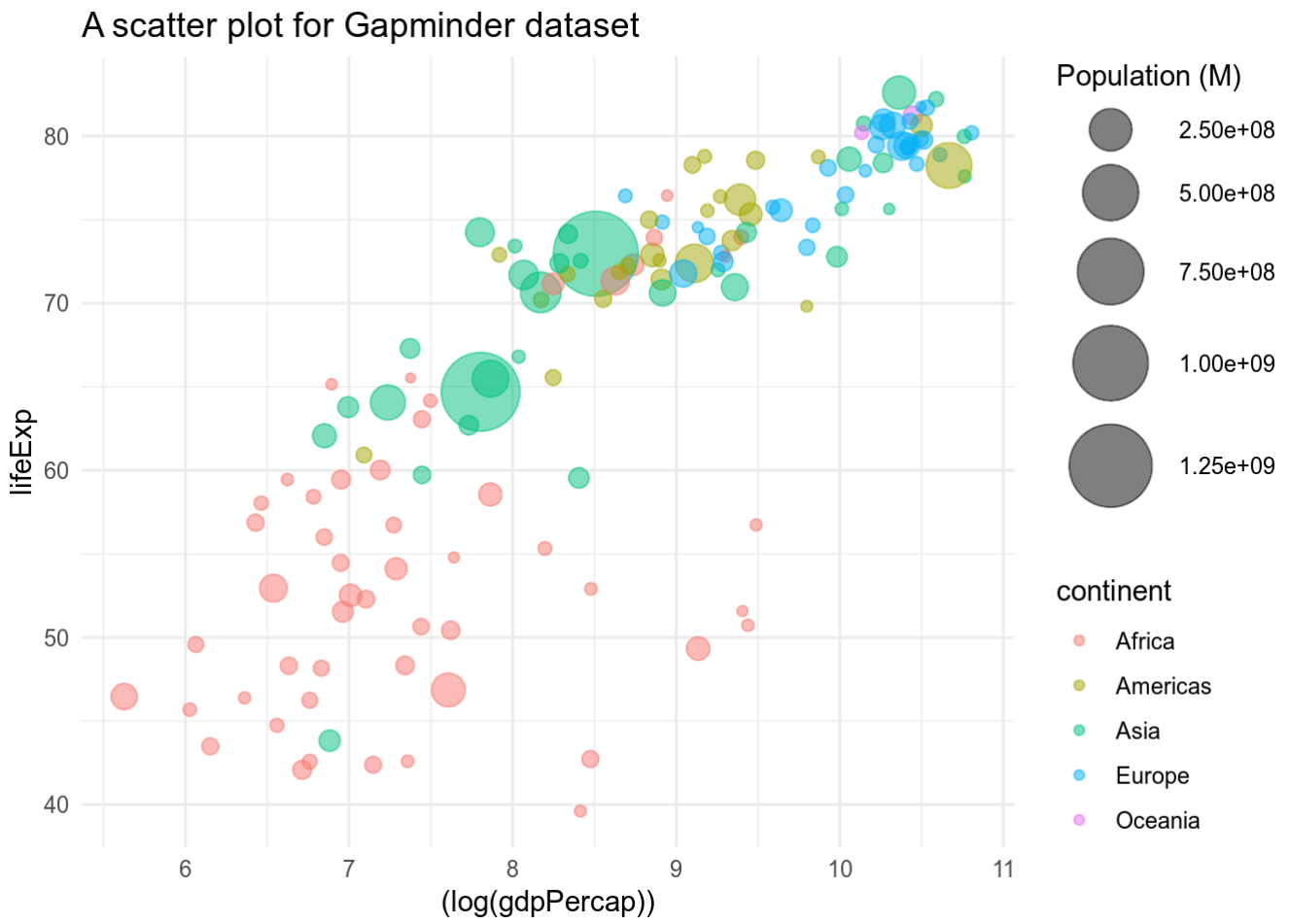
country <fctr>	continent <fctr>	year <int>	lifeExp <dbl>	pop <int>	gdpPercap <dbl>
Afghanistan	Asia	2007	43.828	31889923	974.5803
Albania	Europe	2007	76.423	3600523	5937.0295
Algeria	Africa	2007	72.301	33333216	6223.3675
Angola	Africa	2007	42.731	12420476	4797.2313
Argentina	Americas	2007	75.320	40301927	12779.3796
Australia	Oceania	2007	81.235	20434176	34435.3674
Austria	Europe	2007	79.829	8199783	36126.4927
Bahrain	Asia	2007	75.635	708573	29796.0483
Bangladesh	Asia	2007	64.062	150448339	1391.2538
Belgium	Europe	2007	79.441	10392226	33692.6051

1-10 of 142 rows Previous **1** 2 3 4 5 6 … 15 Next

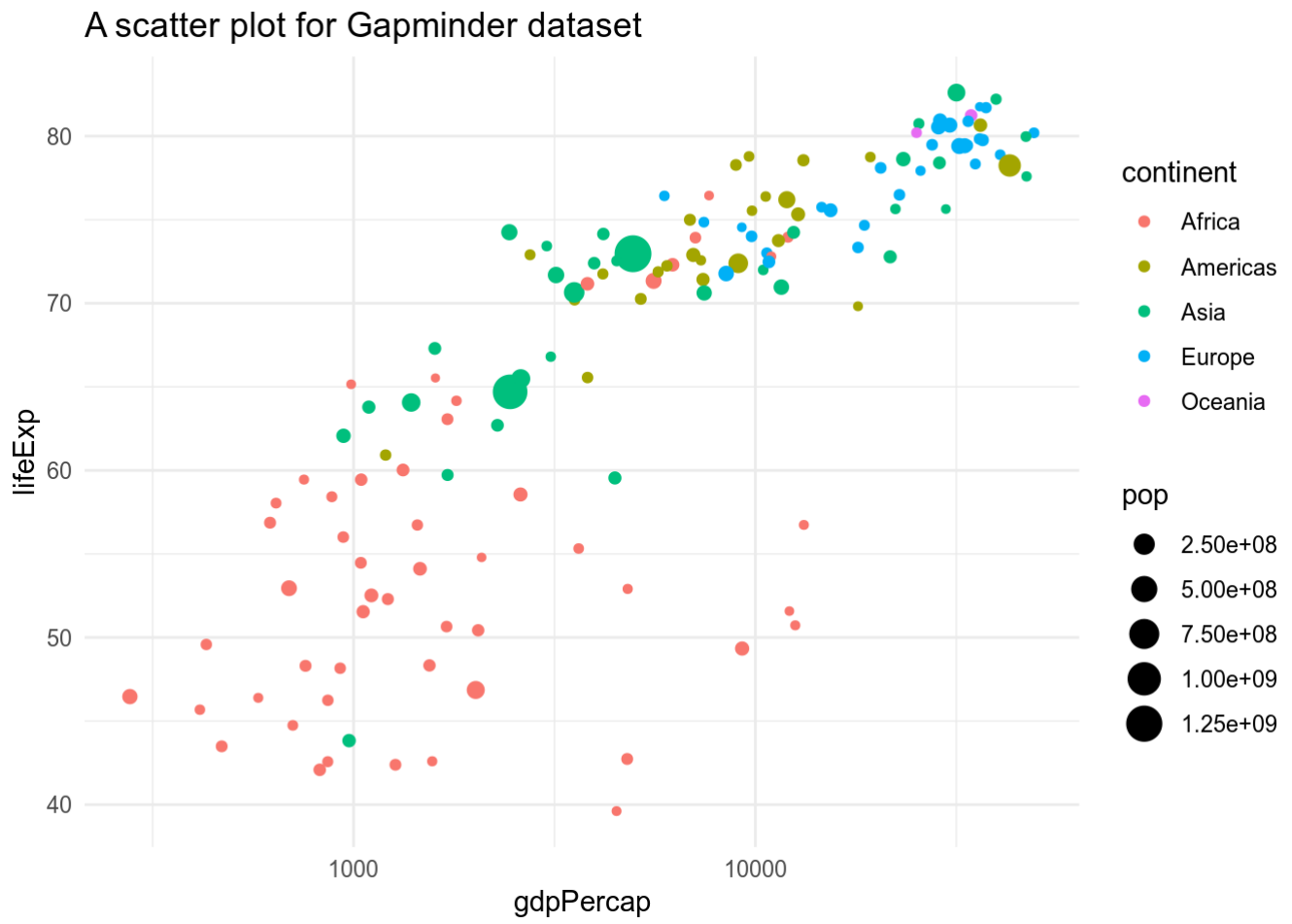
```
## A scatter plot:
ggplot( data= gapminder_2007, mapping = aes(x=(gdpPercap), y=lifeExp, size = pop)) +
    geom_point(aes(color = continent), alpha = 0.5, position = "jitter") +
    scale_size(range = c(1.4, 15), name="Population (M)") +
    theme(legend.position="right")+ theme_minimal()+
  #theme_void()+
  ggtitle("A scatter plot for Gapminder dataset")
```




```
# It is better to use a log scale, when one variable is very densely distributed
ggplot( data= gapminder_2007, mapping = aes(x=(log(gdpPercap)), y=lifeExp, size = po
p)) +
    geom_point(aes(color = continent), alpha = 0.5, position = "jitter") +
    scale_size(range = c(1.4, 15), name="Population (M)") +
    theme(legend.position="right")+ theme_minimal()+
  ggtitle("A scatter plot for Gapminder dataset")
```




```
# Swe can make it also like this
ggplot(gapminder_2007, aes(x = gdpPercap, y = lifeExp, color = continent, size = po
p)) +
geom_point()+
  theme(legend.position="right")+ theme_minimal()+
  scale_x_log10(breaks = c(1000, 10000))+ #Experiment with a different scale
  #theme_void()+
  ggtitle("A scatter plot for Gapminder dataset")
```




```
#Take the mean across all years for each country: (Scatter+Line at one plot for mean
 LifeExp, and one for total Pop over years)
gapminder_mean <- gapminder %>%
  group_by(year, continent) %>%
  summarise(m_lifeExp = mean(lifeExp), totalPop = sum(pop) / 1000000)
gapminder_mean
```

year <int>	continent <fctr>	m_lifeExp <dbl>	totalPop <dbl>
1952	Africa	39.13550	237.64050
1952	Americas	53.27984	345.15245
1952	Asia	46.31439	1395.35735
1952	Europe	64.40850	418.12085
1952	Oceania	69.25500	10.68601
1957	Africa	41.26635	264.83774
1957	Americas	55.96028	386.95392
1957	Asia	49.31854	1562.78060
1957	Europe	66.70307	437.89035
1957	Oceania	70.29500	11.94198

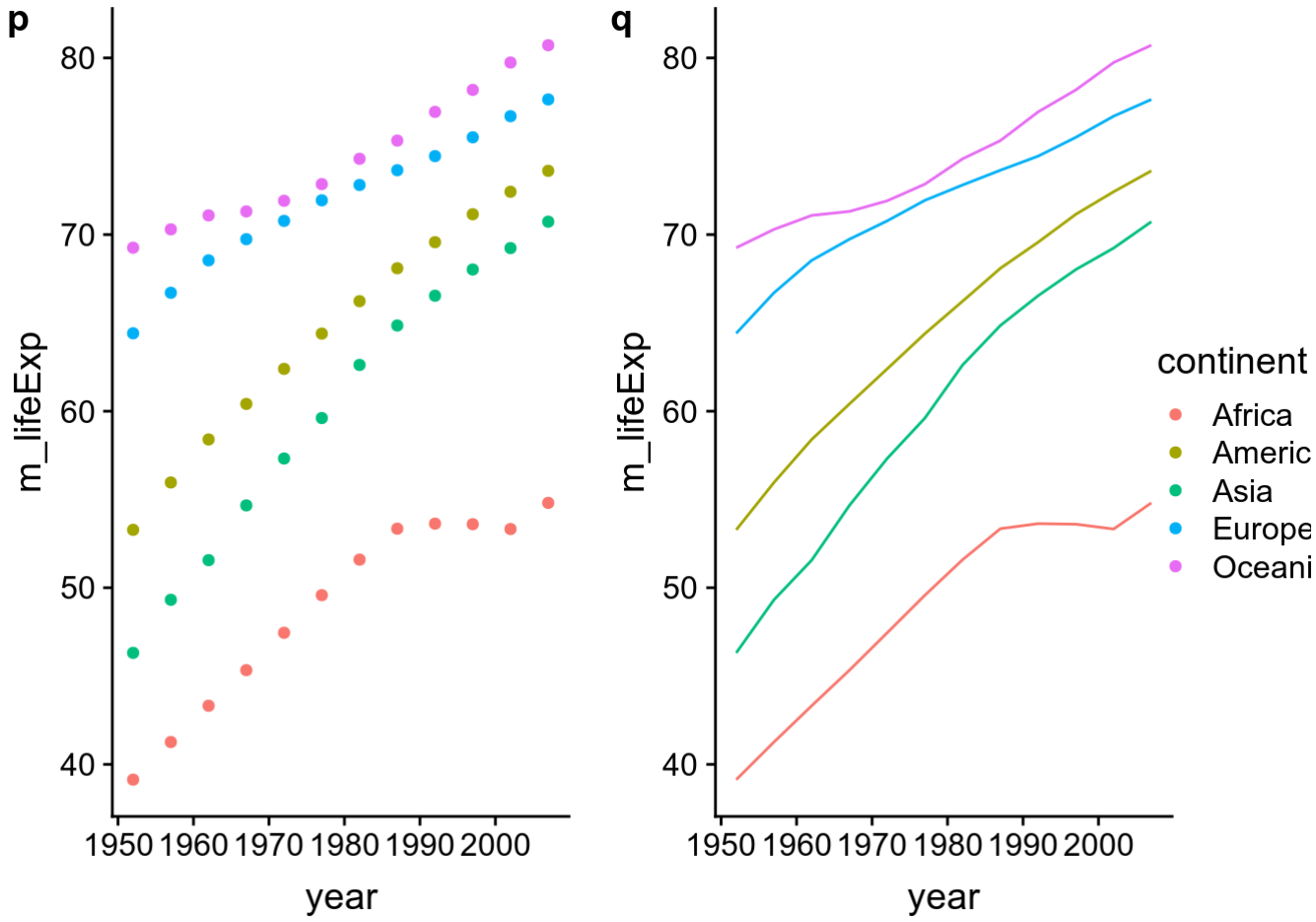
1-10 of 60 rows Previous **1** 2 3 4 5 6 Next

```
#install.packages("cowplot")
library(cowplot)
```

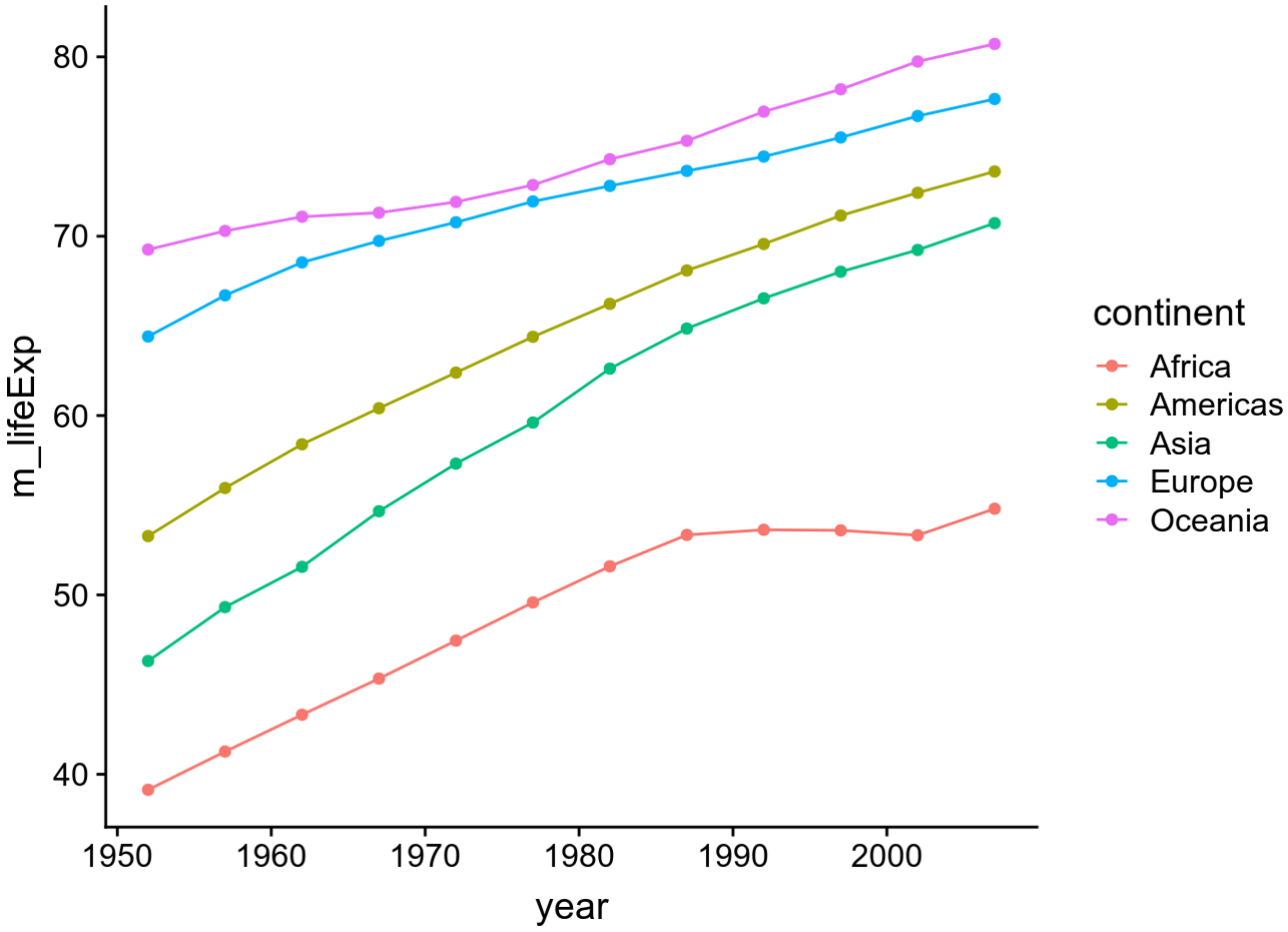
```
##
## Attaching package: 'cowplot'
```

```
## The following object is masked from 'package:ggplot2':
##
##     ggsave
```

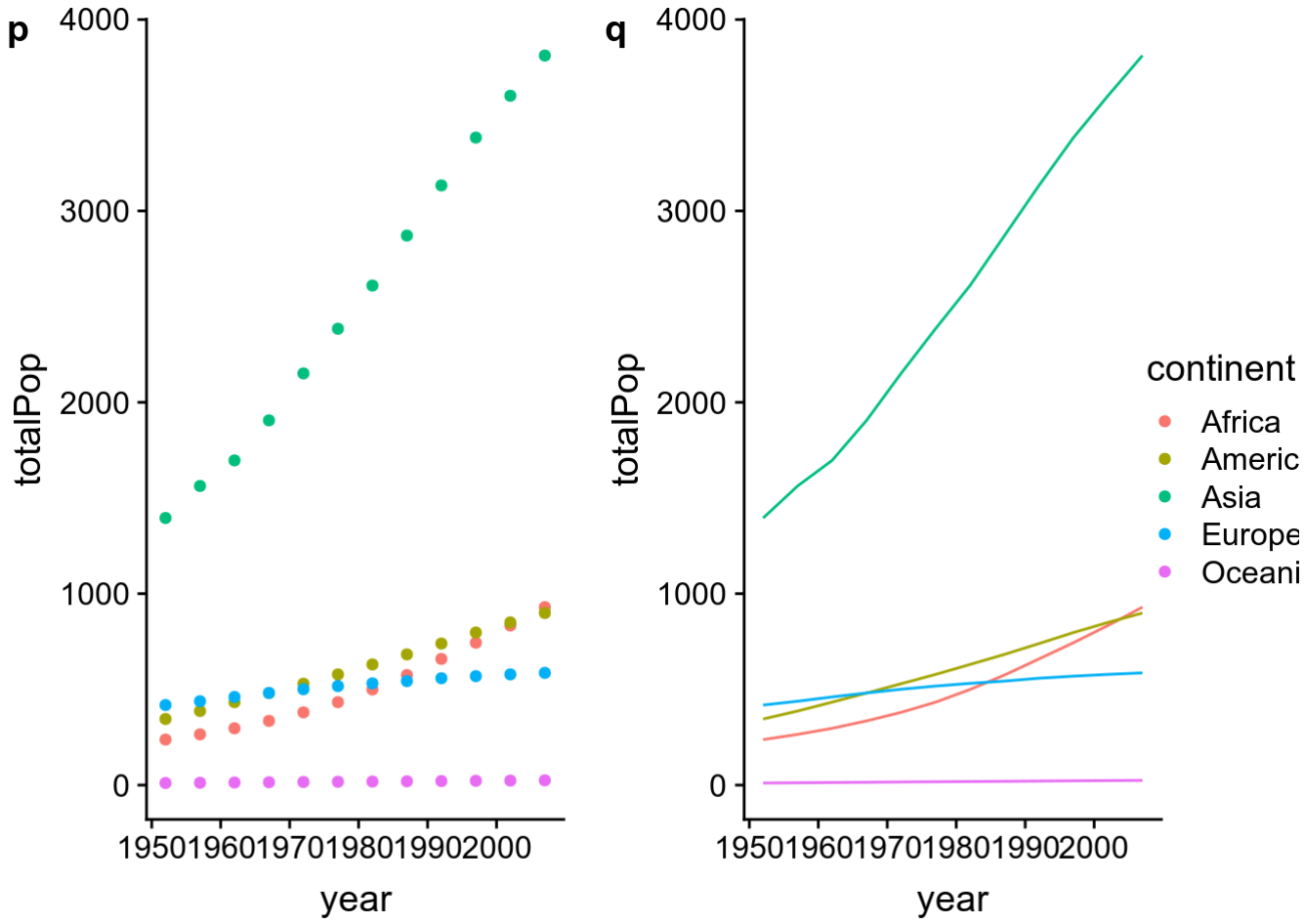
```
p <- ggplot(gapminder_mean, aes(x = year, color = continent)) +
  geom_point(aes(y = m_lifeExp))
q <- ggplot(gapminder_mean, aes(x = year, color = continent)) +
  geom_line(aes(y = m_lifeExp))

# arrange 2 plots in a single row
prow <- plot_grid( p + theme(legend.position="none"),
           q + theme(legend.position="none"),
           align = 'vh',
           labels = c("p", "q"),
           hjust = -1,
           nrow = 1
           )

#Legend to the side:They have the same legend, so we can pick one.
legend <- get_legend(p)
p <- plot_grid( prow, legend, rel_widths = c(2, .2))
p
```



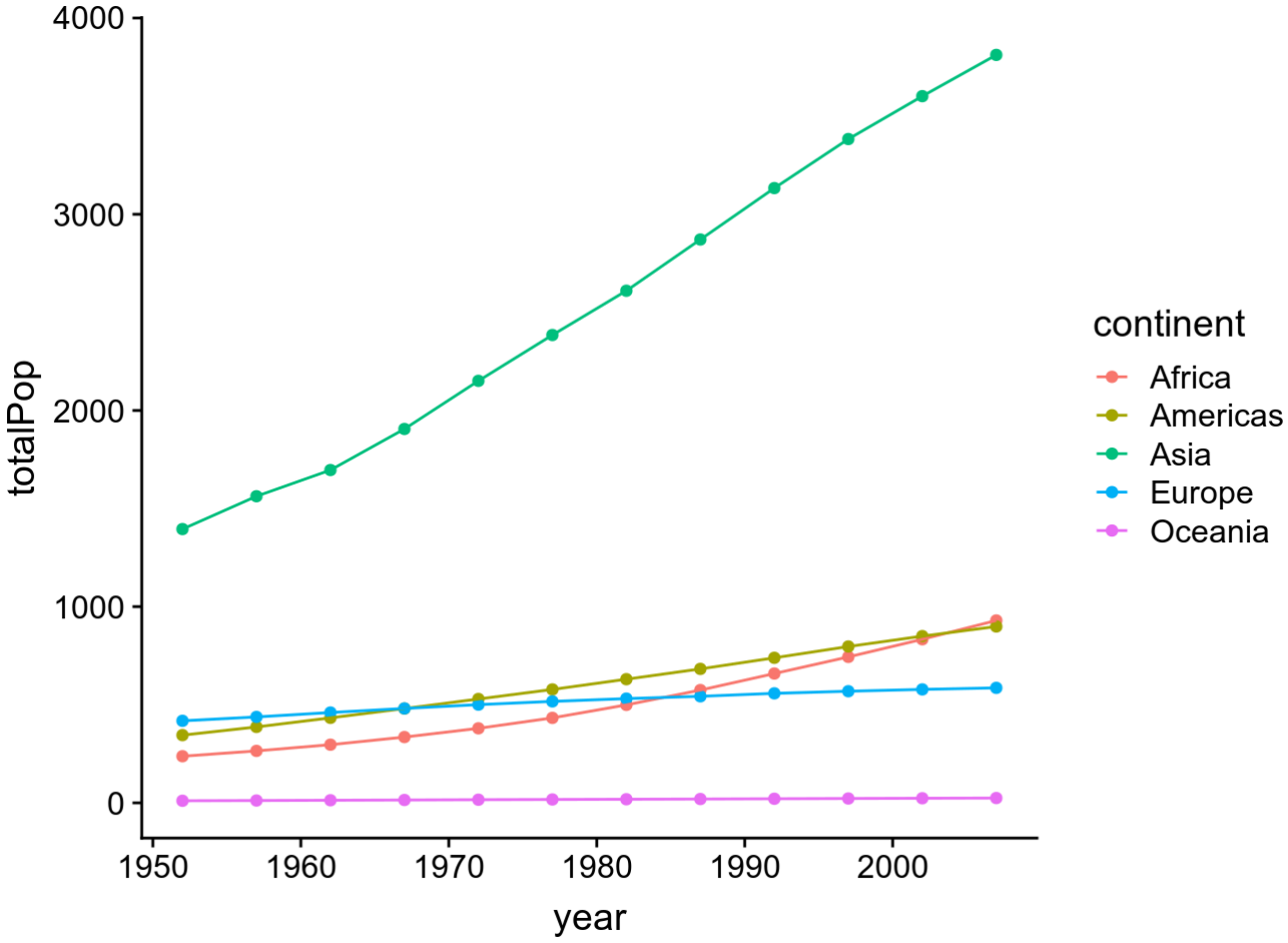

```
p <- ggplot(gapminder_mean, aes(x = year, color = continent)) +
  geom_point(aes(y = m_lifeExp)) +
geom_line(aes(y = m_lifeExp))
p
```



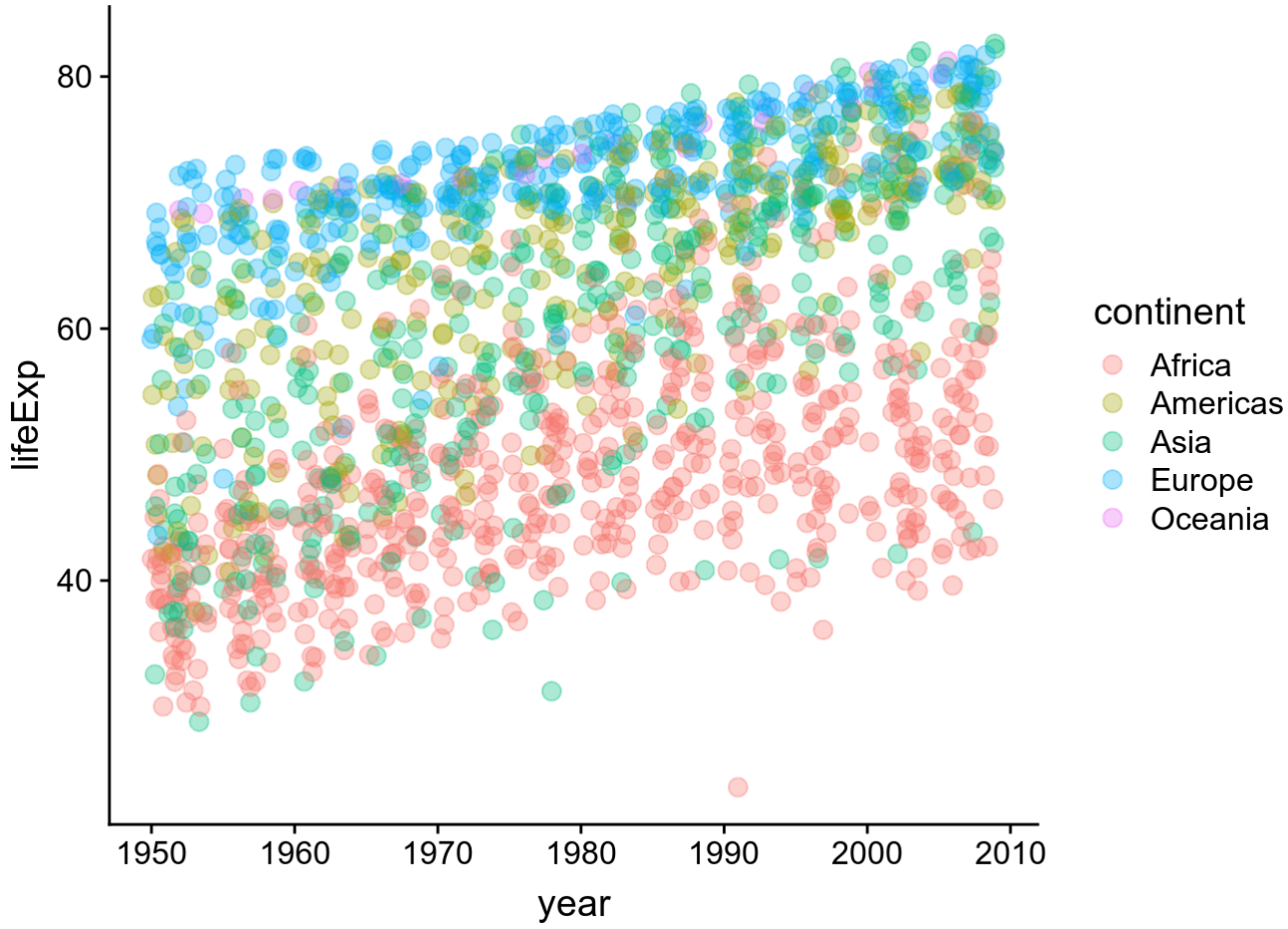

```
X <- ggplot(gapminder_mean, aes(x = year, color = continent)) +
  geom_point(aes(y =  totalPop))
Y <- ggplot(gapminder_mean, aes(x = year, color = continent)) +
  geom_line(aes(y =  totalPop))

# arrange 2 plots in a single row
prow <- plot_grid( X + theme(legend.position="none"),
           Y + theme(legend.position="none"),
           align = 'vh',
           labels = c("p", "q"),
           hjust = -1,
           nrow = 1
           )

#Legend to the side:They have the same legend, so we can pick one.
legend <- get_legend(X)
p <- plot_grid( prow, legend, rel_widths = c(2, .2))
p
```




```
ggplot(gapminder_mean, aes(x = year, color = continent)) +
  geom_point(aes(y = totalPop)) +
  geom_line(aes(y = totalPop))
```




```
#plot lifeExp against year

ggplot(gapminder, aes(x = year, y = lifeExp,
                      color = continent)) +
  geom_jitter(alpha = 1/3, size = 3)
```




```
#life expectancy distributions in 2007
gapminder_2007 <- gapminder %>%
    filter(year == 2007)

gapminder_2007
```

country <fctr>	continent <fctr>	year <int>	lifeExp <dbl>	pop <int>	gdpPercap <dbl>
Afghanistan	Asia	2007	43.828	31889923	974.5803
Albania	Europe	2007	76.423	3600523	5937.0295
Algeria	Africa	2007	72.301	33333216	6223.3675
Angola	Africa	2007	42.731	12420476	4797.2313
Argentina	Americas	2007	75.320	40301927	12779.3796
Australia	Oceania	2007	81.235	20434176	34435.3674
Austria	Europe	2007	79.829	8199783	36126.4927
Bahrain	Asia	2007	75.635	708573	29796.0483
Bangladesh	Asia	2007	64.062	150448339	1391.2538
Belgium	Europe	2007	79.441	10392226	33692.6051

1-10 of 142 rows Previous **1** 2 3 4 5 6 ... 15 Next

```
ggplot(gapminder_2007, aes(x=continent , y=lifeExp, fill = continent))+
  geom_boxplot(alpha= 0.8)+
  ggtitle("life expectancy distributions in 2007")+
  theme(legend.position="right")
```