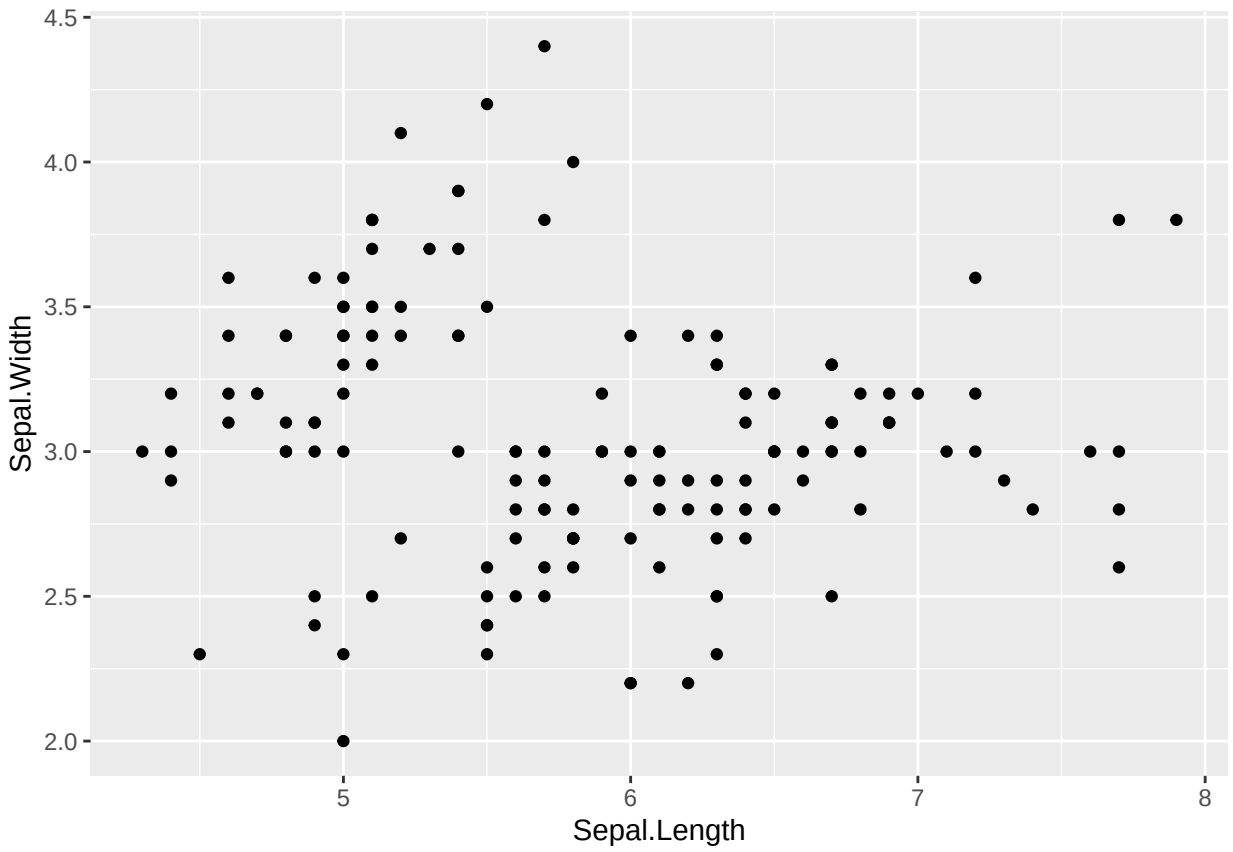
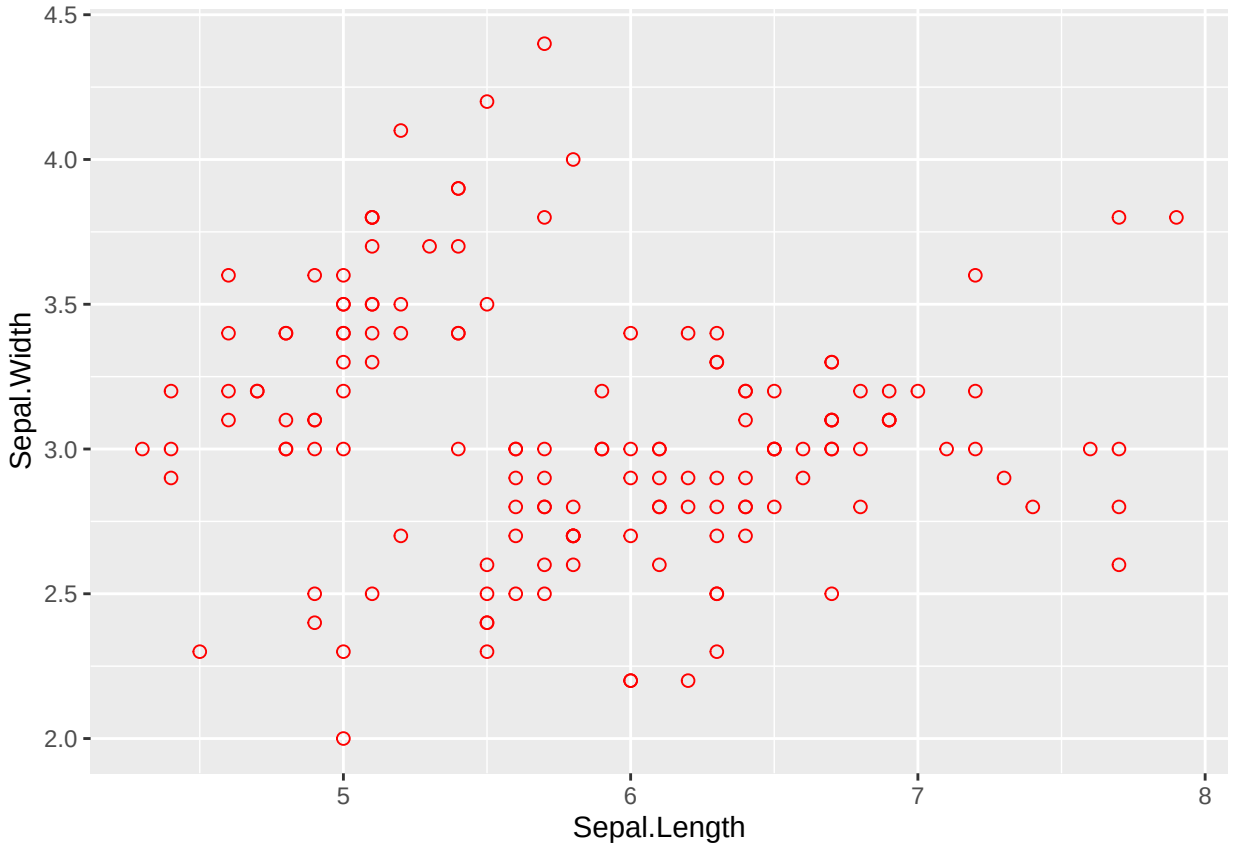
ggplot2map

```
#install.packages(ggplot2)
#install.packages("ggplot2")
library(ggplot2)
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width))+
  geom_point()
```




```
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width))+
  geom_point(size = 1.9, color = "red", shape = 21)
```




```
library(ggplot2)
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
```

```
require(maps)
```

```
## Loading required package: maps
```

```
require(viridis)
```

```
## Loading required package: viridis

## Loading required package: viridisLite
```

```r
theme_set(
  theme_void()
)


  library(ggplot2)
library(dplyr)
#installed.packages("viridis")
require(maps)
require(viridis)
theme_set(
  theme_void()
)
monde_map <- map_data("world")
ggplot(monde_map, aes(x = long, y = lat, group = group)) + geom_polygon(fill="lightgray", colour = "whi
```



```r
# Afficher quelque pays de l'Asie
some.asie.countries <- c(
  "Russia", "China", "Iran", "Mongolia", "India",
  "Australia", "Kazakhstan", "North Korea", "South Korea",
  "Nepal", "Pakistan","Japan","Iraq"
)
#turkey,syria,afghanistan,syria
# Recuperer la map
some.asie.maps <- map_data("world", region = some.asie.countries)

# Utilisé comme coordonnée étiquette pour les noms de pays
```

```
region.lab.data <- some.asie.maps %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(some.asie.maps, aes(x = long, y = lat)) +
  geom_polygon(aes( group = group, fill = region))+
  geom_text(aes(label = region), data = region.lab.data,  size = 3, hjust = 0.5)+
  scale_fill_viridis_d()+
  theme_void()+
  theme(legend.position = "left")
```