

# World Happiness Report

Mahtab Moradi

2022-12-20

## Happiness report

The World Happiness Report is a landmark survey of the state of global happiness. The first report was published in 2012, the second in 2013, the third in 2015, and the fourth in the 2016 Update. The World Happiness 2017, which ranks 155 countries by their happiness levels, was released at the United Nations at an event celebrating International Day of Happiness on March 20th. The report continues to gain global recognition as governments, organizations and civil society increasingly use happiness indicators to inform their policy-making decisions. Leading experts across fields – economics, psychology, survey analysis, national statistics, health, public policy and more – describe how measurements of well-being can be used effectively to assess the progress of nations. The reports review the state of happiness in the world today and show how the new science of happiness explains personal and national variations in happiness. The original data can be found at <https://www.kaggle.com/datasets/unsdsn/world-happiness> wherein extra explanation regarding different studied components is also given.

Below, we provide some maps based on different items considered in the happiness report during 2019. Throughout, the R packages `sf` and `ggplot2` are used. The happiness data is given as a csv file, and the maps of countries as shape files. We display maps for the entire world, europe, and south-america.

Note: Countries appeared in grey are those we have no data for.

## World

```
Happiness <- read.csv("2019.csv",sep = ";")  
head(Happiness)
```

```
##   Overall.rank      Country.or.region Score GDP.per.capita Social.support  
## 1          156           South Sudan  2.853     0.306       0.575  
## 2          155 Central African Republic  3.083     0.026       0.000  
## 3          154        Afghanistan  3.203     0.350       0.517  
## 4          153          Tanzania  3.231     0.476       0.885  
## 5          152            Rwanda  3.334     0.359       0.711  
## 6          151            Yemen  3.380     0.287       1.163  
##   Healthy.life.expectancy Freedom.to.make.life.choices Generosity  
## 1                  0.295                 0.010      0.202  
## 2                  0.105                 0.225      0.235  
## 3                  0.361                 0.000      0.158  
## 4                  0.499                 0.417      0.276  
## 5                  0.614                 0.555      0.217  
## 6                  0.463                 0.143      0.108  
##   Perceptions.of.corruption
```

```

## 1          0.091
## 2          0.035
## 3          0.025
## 4          0.147
## 5          0.411
## 6          0.077

library(sf)

World <- read_sf("World_Countries.shp")
head(World)

## Simple feature collection with 6 features and 1 field
## Geometry type: MULTIPOLYGON
## Dimension:      XY
## Bounding box:  xmin: -70.05966 ymin: 12.41111 xmax: 74.91574 ymax: 42.66034
## Geodetic CRS:  WGS 84
## # A tibble: 6 x 2
##   COUNTRY                                geometry
##   <chr>                                     <MULTIPOLYGON [°]>
## 1 Aruba (Netherlands) (((-69.88223 12.41111, -69.94695 12.43667, -70.05904 12.5~)
## 2 Antigua and Barbuda (((-61.73889 17.54055, -61.75195 17.54944, -61.81556 17.5~)
## 3 Afghanistan ((61.27656 35.60725, 61.29638 35.62853, 61.37506 35.6361~)
## 4 Algeria (((-5.152135 30.18047, -5.139167 30.19236, -5.080973 30.2~)
## 5 Azerbaijan (((46.54037 38.87559, 46.49554 38.95735, 46.50221 38.9998~)
## 6 Albania (((20.79192 40.43154, 20.78722 40.39472, 20.75861 40.3119~
```

There are some countries in the maps which are not included in the happiness report, below we extract the list of such countries.

```
World$COUNTRY[!World$COUNTRY %in% Happiness$Country.or.region]
```

```

## [1] "Aruba (Netherlands)"
## [2] "Antigua and Barbuda"
## [3] "Andorra"
## [4] "Angola"
## [5] "American Samoa (US)"
## [6] "Anguilla (UK)"
## [7] "Antarctica"
## [8] "Barbados"
## [9] "Bermuda (UK)"
## [10] "Bahamas"
## [11] "Belize"
## [12] "Solomon Islands"
## [13] "Bouvet Island (Norway)"
## [14] "Brunei"
## [15] "Congo"
## [16] "Democratic Republic of the Congo"
## [17] "Cayman Islands (UK)"
## [18] "Cocos (Keeling) Islands (Australia)"
## [19] "Northern Mariana Islands (US)"
## [20] "Cuba"
```

```

## [21] "Cape Verde"
## [22] "Cook Islands (New Zealand)"
## [23] "Djibouti"
## [24] "Dominica"
## [25] "Jarvis Island (US)"
## [26] "Equatorial Guinea"
## [27] "Eritrea"
## [28] "French Guiana (France)"
## [29] "Fiji"
## [30] "Falkland Islands (UK)"
## [31] "Federated States of Micronesia"
## [32] "Faroe Islands (Denmark)"
## [33] "French Polynesia (France)"
## [34] "Baker Island (US)"
## [35] "French Southern & Antarctic Lands (France)"
## [36] "Gibraltar (UK)"
## [37] "Grenada"
## [38] "Guernsey (UK)"
## [39] "Greenland (Denmark)"
## [40] "Glorioso Islands (France)"
## [41] "Guadeloupe (France)"
## [42] "Guam (US)"
## [43] "Guyana"
## [44] "Palestine"
## [45] "Heard Island & McDonald Islands (Australia)"
## [46] "Howland Island (US)"
## [47] "Isle of Man (UK)"
## [48] "British Indian Ocean Territory (UK)"
## [49] "Jersey (UK)"
## [50] "Jan Mayen (Norway)"
## [51] "Johnston Atoll (US)"
## [52] "Juan De Nova Island (France)"
## [53] "North Korea"
## [54] "Kiribati"
## [55] "Christmas Island (Australia)"
## [56] "Liechtenstein"
## [57] "Martinique (France)"
## [58] "Mayotte (France)"
## [59] "Montserrat (UK)"
## [60] "Macedonia"
## [61] "Monaco"
## [62] "Midway Islands (US)"
## [63] "Oman"
## [64] "Maldives"
## [65] "New Caledonia (France)"
## [66] "Niue (New Zealand)"
## [67] "Norfolk Island (Australia)"
## [68] "Vanuatu"
## [69] "Nauru"
## [70] "Suriname"
## [71] "Pitcairn Islands (UK)"
## [72] "Paracel Islands (China)"
## [73] "Spratly Islands (Disputed)"
## [74] "Papua New Guinea"

```

```

## [75] "Palau (US)"
## [76] "Guinea-Bissau"
## [77] "Reunion (France)"
## [78] "Marshall Islands"
## [79] "Puerto Rico (US)"
## [80] "St. Pierre and Miquelon (France)"
## [81] "St. Kitts and Nevis"
## [82] "Seychelles"
## [83] "St. Helena (UK)"
## [84] "San Marino"
## [85] "St. Lucia"
## [86] "Sudan"
## [87] "Svalbard (Norway)"
## [88] "South Georgia and the South Sandwich Is (UK)"
## [89] "Trinidad and Tobago"
## [90] "Turks and Caicos Islands (UK)"
## [91] "Tokelau (New Zealand)"
## [92] "Tonga"
## [93] "Sao Tome and Principe"
## [94] "Tuvalu"
## [95] "St. Vincent and the Grenadines"
## [96] "British Virgin Islands(UK)"
## [97] "American Virgin Islands (US)"
## [98] "Wallis and Futuna (France)"
## [99] "Western Sahara"
## [100] "Wake Island (US)"
## [101] "Western Samoa"
## [102] "East Timor"
## [103] "Curacao (Netherlands)"
## [104] "Bonaire (Netherlands)"

```

Below we extract the happiness data corresponding to different countries, and for those countries which there is no data, we set NA.

```

Happ_World <- Happiness[Happiness$Country.or.region %in% World$COUNTRY [
  World$COUNTRY %in% Happiness$Country.or.region],]
NA_rows <- as.data.frame(matrix(NA, nrow=13, ncol=9))
names(NA_rows) <- names(Happ_World)
Happ_World <- rbind(Happ_World, NA_rows)

```

Next, we merge the happiness data to our map of the world.

```
World_happ <- cbind(World, Happ_World[match(World$COUNTRY, Happ_World$Country.or.region),])
```

There were few countries which their names were written in different ways in the two sources of data, i.e. world map and happiness data. Below, we update the rows corresponding to such countries.

```

World_happ[World_happ$COUNTRY=="Trinidad & Tobago",] <- cbind(World_happ[
  World_happ$COUNTRY=="Trinidad & Tobago", 1], Happiness[
    Happiness$Country.or.region=="Trinidad and Tobago",])
World_happ[World_happ$COUNTRY=="Macedonia",] <- cbind(World_happ[
  World_happ$COUNTRY=="Macedonia", 1], Happiness[
    Happiness$Country.or.region=="North Macedonia",])

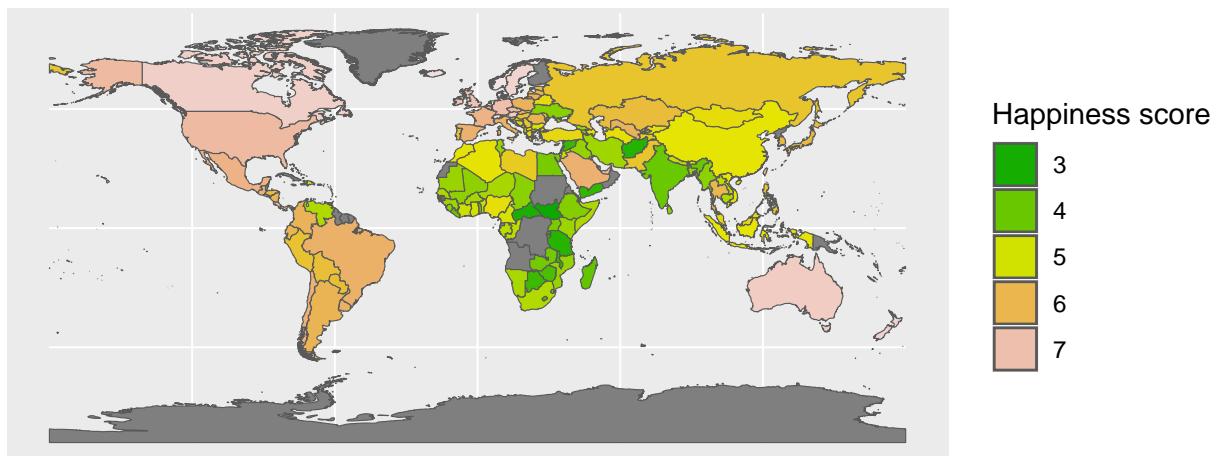
```

```
World_happ[World_happ$COUNTRY=="Congo",] <- cbind(World_happ[
  World_happ$COUNTRY=="Congo",1],Happiness[
    Happiness$Country.or.region=="Congo (Brazzaville)",])
```

We now display different maps using ggplot2 package in which colours stand for the values of studied items.

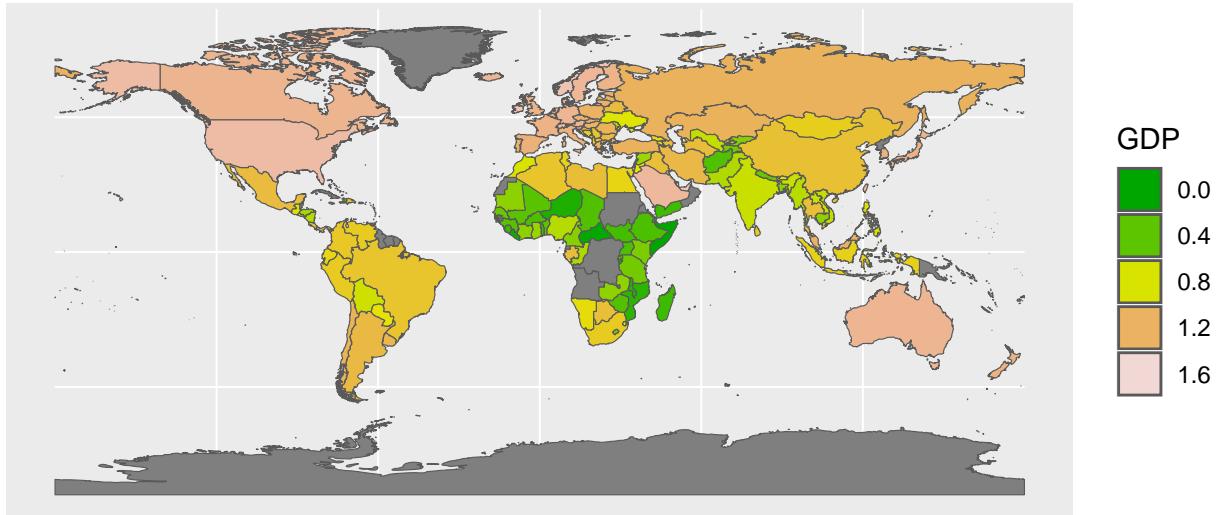
```
library(ggplot2)
ggplot(World_happ) +
  geom_sf(aes(fill = Score)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(2.7,7.7))+ 
  guides(fill=guide_legend(title="Happiness score"))
```

Happniess score



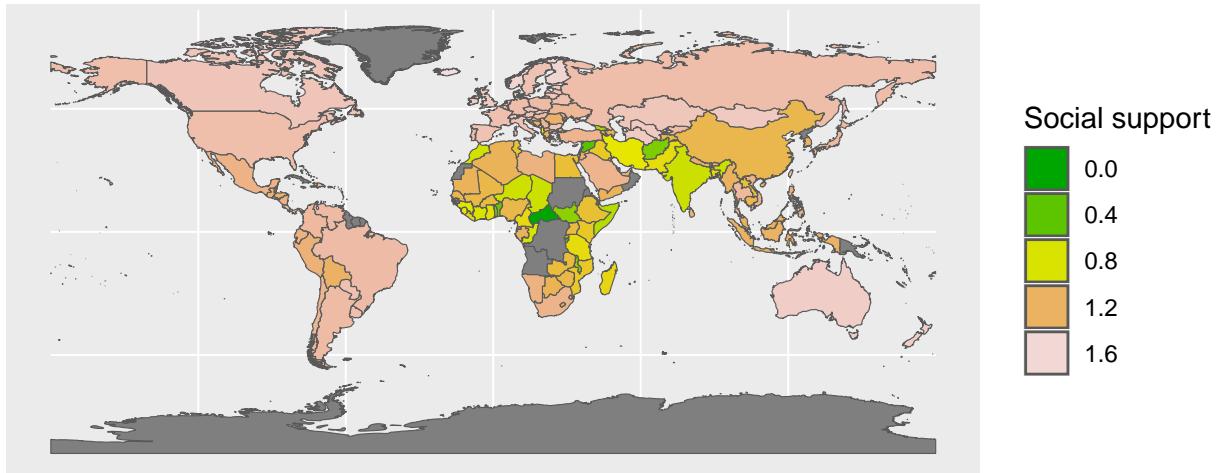
```
ggplot(World_happ) +
  geom_sf(aes(fill = GDP.per.capita)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(0,1.7))+ 
  guides(fill=guide_legend(title="GDP"))
```

GDP per capita



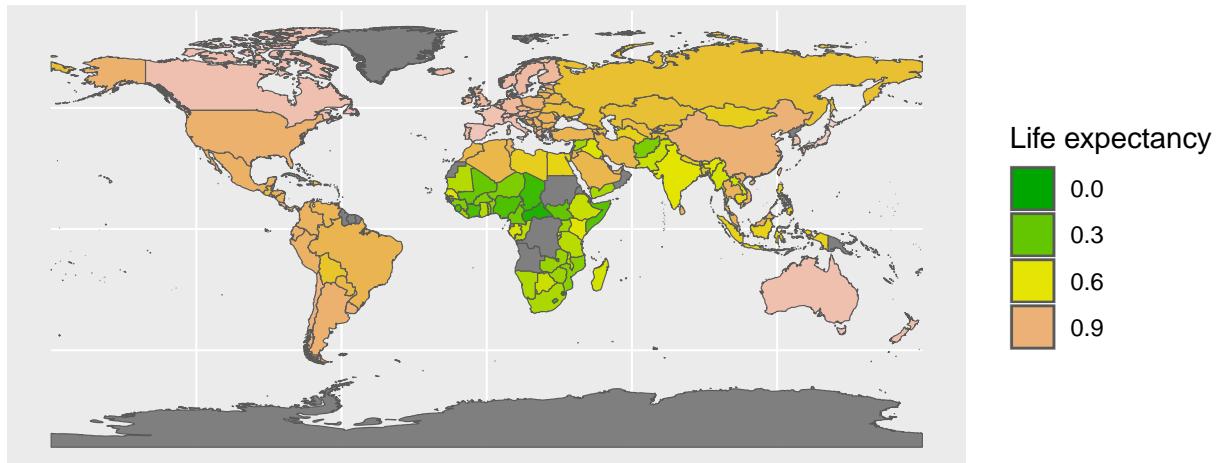
```
ggplot(World_happ) +
  geom_sf(aes(fill = Social.support)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(0,1.7))+
```

### Social support



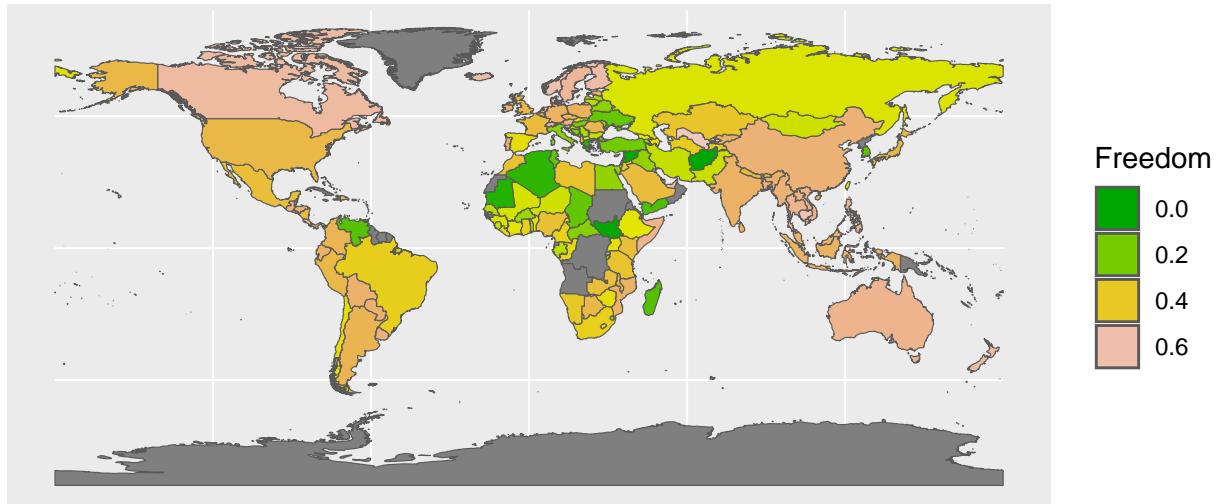
```
ggplot(World_happ) +
  geom_sf(aes(fill = Healthy.life.expectancy)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(0,1.2))+
```

## Life expectancy



```
ggplot(World_happ) +  
  geom_sf(aes(fill = Freedom.to.make.life.choices)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +  
  scale_fill_gradientn(colours = terrain.colors(100), limits=c(0,0.7)) +  
  guides(fill=guide_legend(title="Freedom"))
```

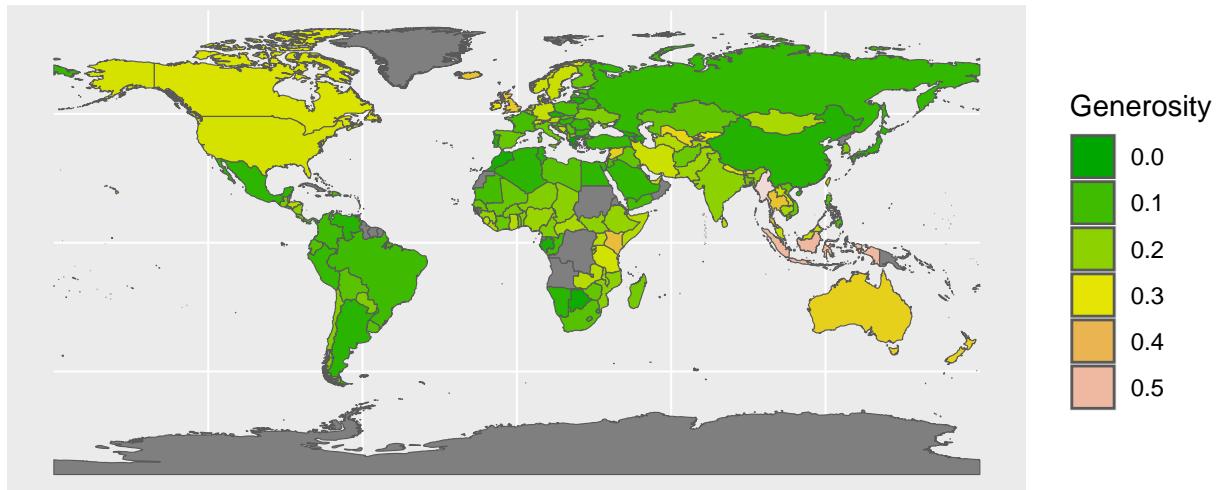
## Freedom



```
ggplot(World_happ) +  
  geom_sf(aes(fill = Generosity)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +
```

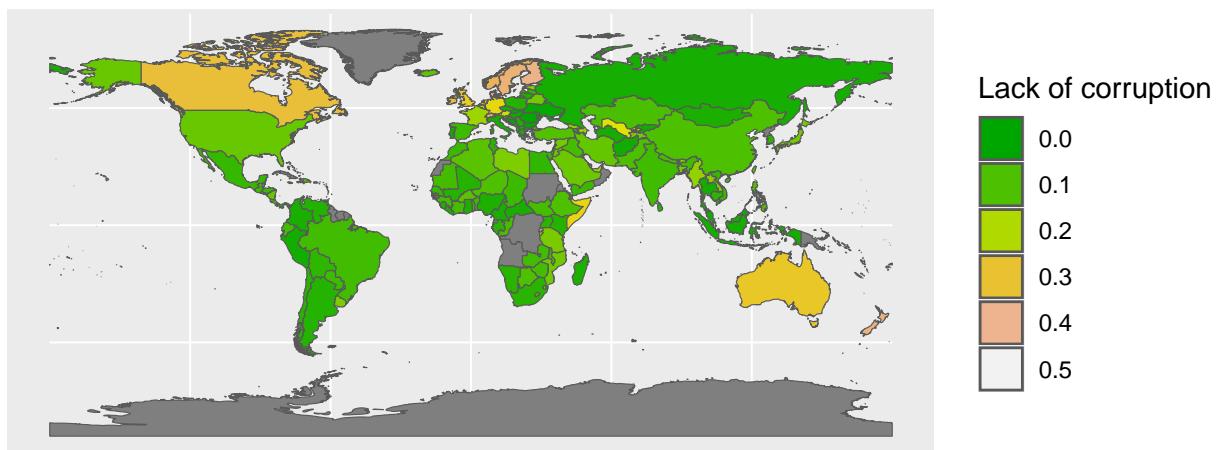
```
scale_fill_gradientn(colours = terrain.colors(100),limits=c(0,0.6))+  
guides(fill=guide_legend(title="Generosity"))
```

Generosity



```
ggplot(World_happ) +  
  geom_sf(aes(fill = Perceptions.of.corruption)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank())+  
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(0,0.5))+  
  guides(fill=guide_legend(title="Lack of corruption"))
```

Lack of corruption



Europe

```

EU <- read_sf("Europe.shp")
head(EU)

## Simple feature collection with 6 features and 2 fields
## Geometry type: MULTIPOLYGON
## Dimension: XY
## Bounding box: xmin: 1.422104 ymin: 39.64836 xmax: 21.06846 ymax: 51.50544
## Geodetic CRS: WGS 84
## # A tibble: 6 x 3
##   NAME      ORGN_NAME      geometry
##   <chr>    <chr>          <MULTIPOLYGON [°]>
## 1 Albania  Shqipëria    (((19.50115 40.9623, 19.50563 40.9612, ~
## 2 Andorra  Andorra        (((1.439922 42.60649, 1.450409 42.6059~
## 3 Austria  Österreich    (((16 48.77775, 16 48.78252, 16 00811 ~
## 4 Belgium  België / Belgique (((5 49.79374, 4.997239 49.79696, 4.99~
## 5 Bosnia Herzegovina Bosna i Hercegovina (((19.22947 43.53458, 19.22925 43.5359~
## 6 Croatia  Hrvatska       (((14.30038 44.50156, 14.28972 44.5053~
```

```

Happ_eu <- Happiness[Happiness$Country.or.region %in% EU$NAME[
  EU$NAME %in% Happiness$Country.or.region],]
NA_rows <- as.data.frame(matrix(NA,nrow=13,ncol=9))
names(NA_rows) <- names(Happ_eu)
Happ_eu <- rbind(Happ_eu,NA_rows)
```

```
EU_happ <- cbind(EU,Happ_eu[match(EU$NAME,Happ_eu$Country.or.region),])
```

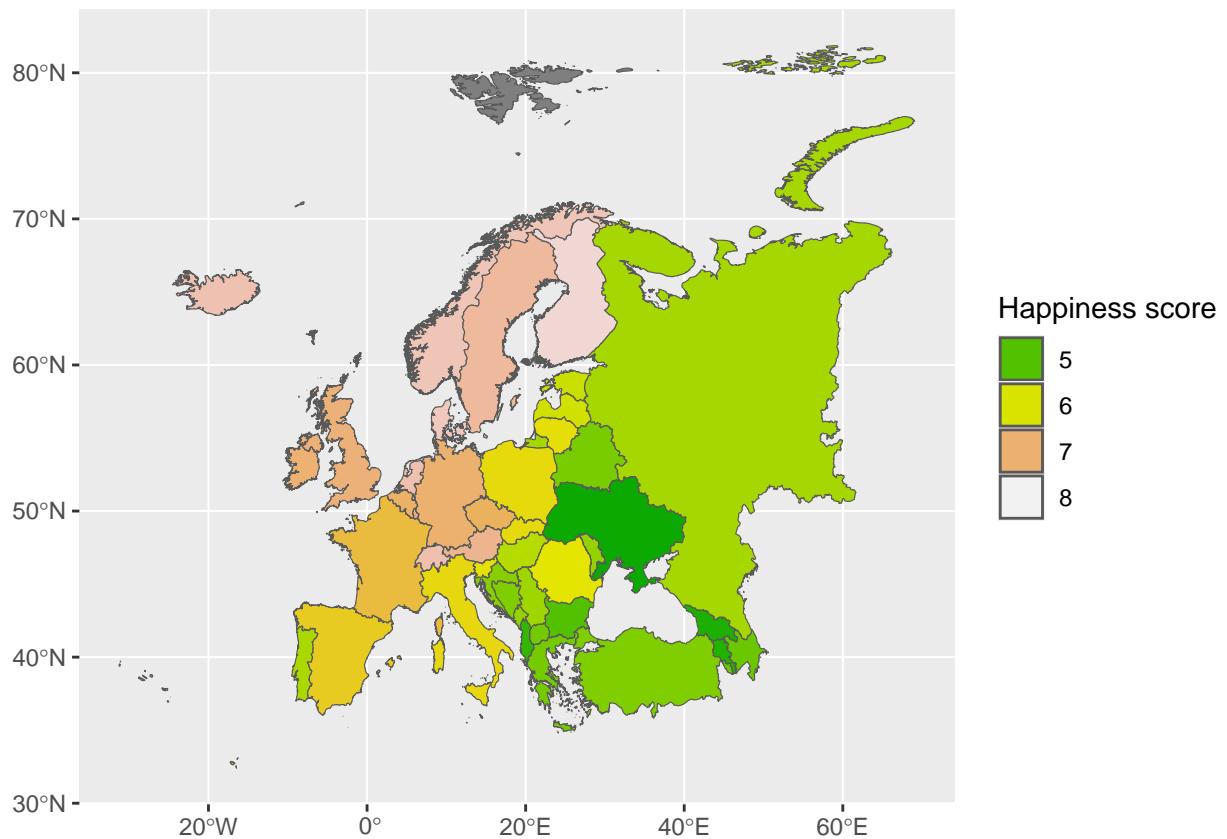
```

EU_happ[EU_happ$NAME=="Bosnia Herzegovina",] <- cbind(EU_happ[
  EU_happ$NAME=="Bosnia Herzegovina",c(1:2)],Happiness[
  Happiness$Country.or.region=="Bosnia and Herzegovina",])
EU_happ[EU_happ$NAME=="Macedonia",] <- cbind(EU_happ[
  EU_happ$NAME=="Macedonia",c(1:2)],Happiness[
  Happiness$Country.or.region=="North Macedonia",])
```

```

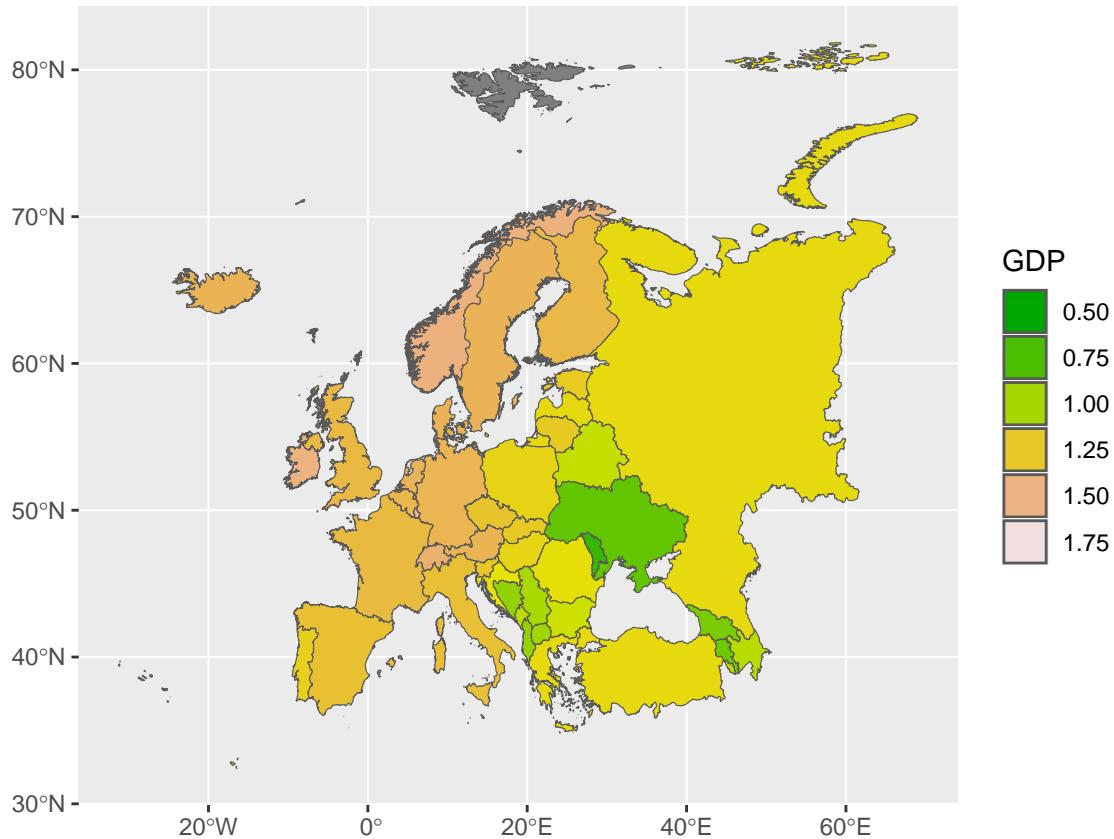
ggplot(EU_happ) +
  geom_sf(aes(fill = Score)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(4.2,8))+
```

Happniess score



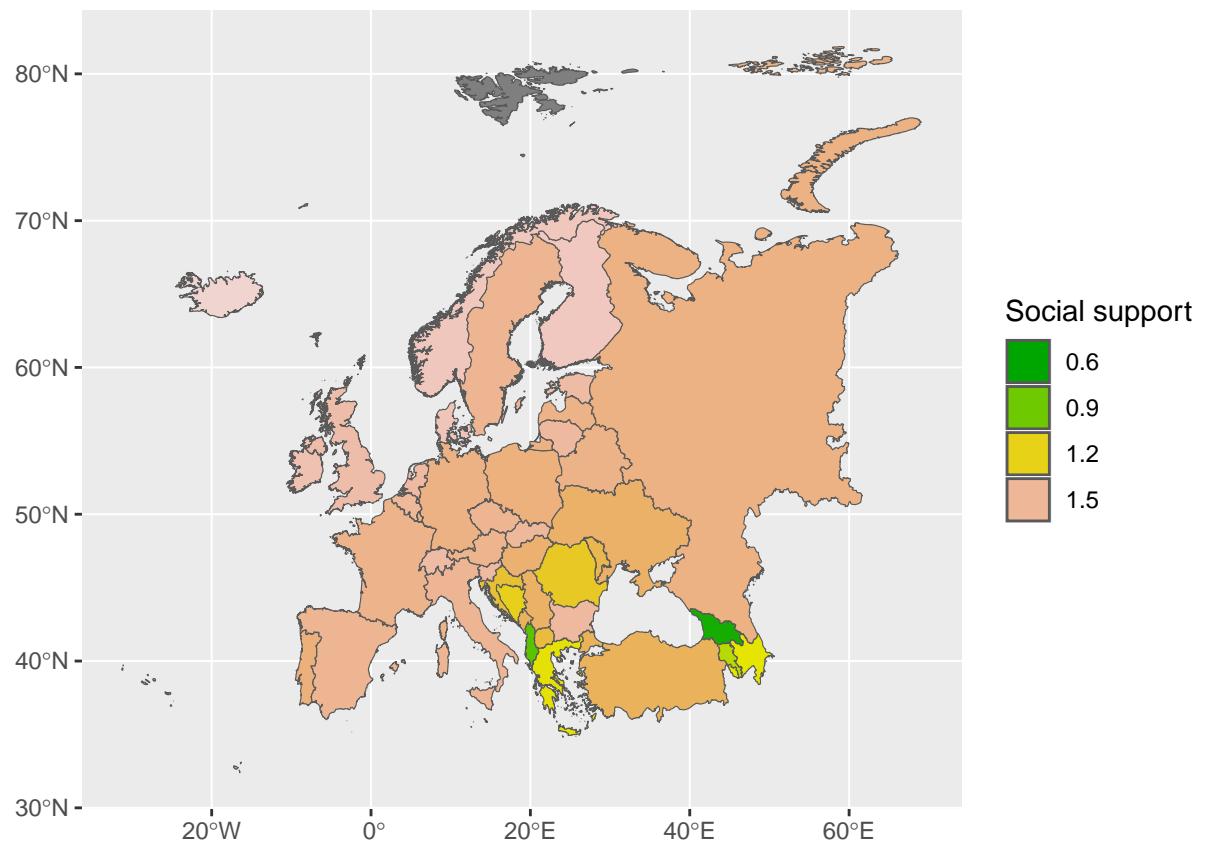
```
ggplot(EU_happ) +  
  geom_sf(aes(fill = GDP.per.capita)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +  
  scale_fill_gradientn(colours = terrain.colors(100), limits=c(0.5,1.8)) +  
  guides(fill=guide_legend(title="GDP"))
```

GDP per capita



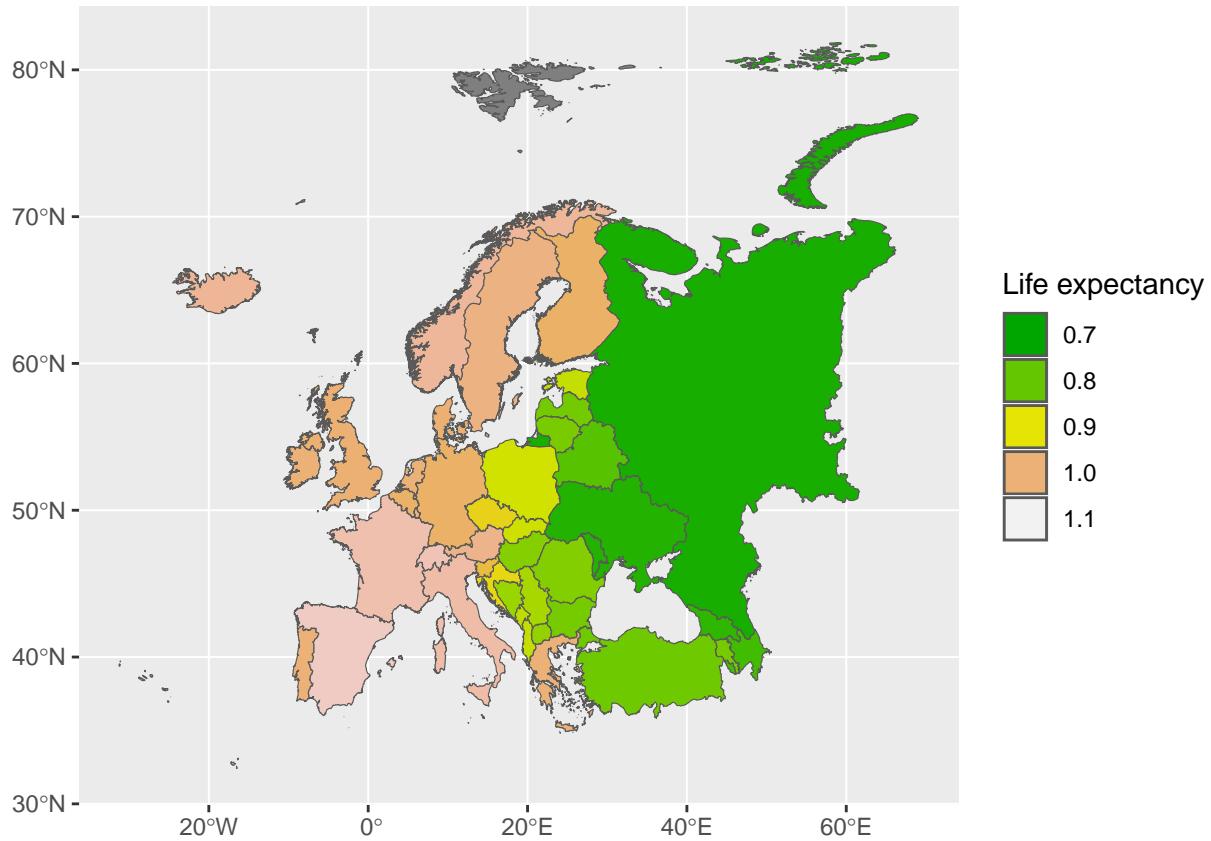
```
ggplot(EU_happ) +  
  geom_sf(aes(fill = Social.support)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +  
  scale_fill_gradientn(colours = terrain.colors(100), limits=c(0.6,1.7)) +  
  guides(fill=guide_legend(title="Social support"))
```

## Social support



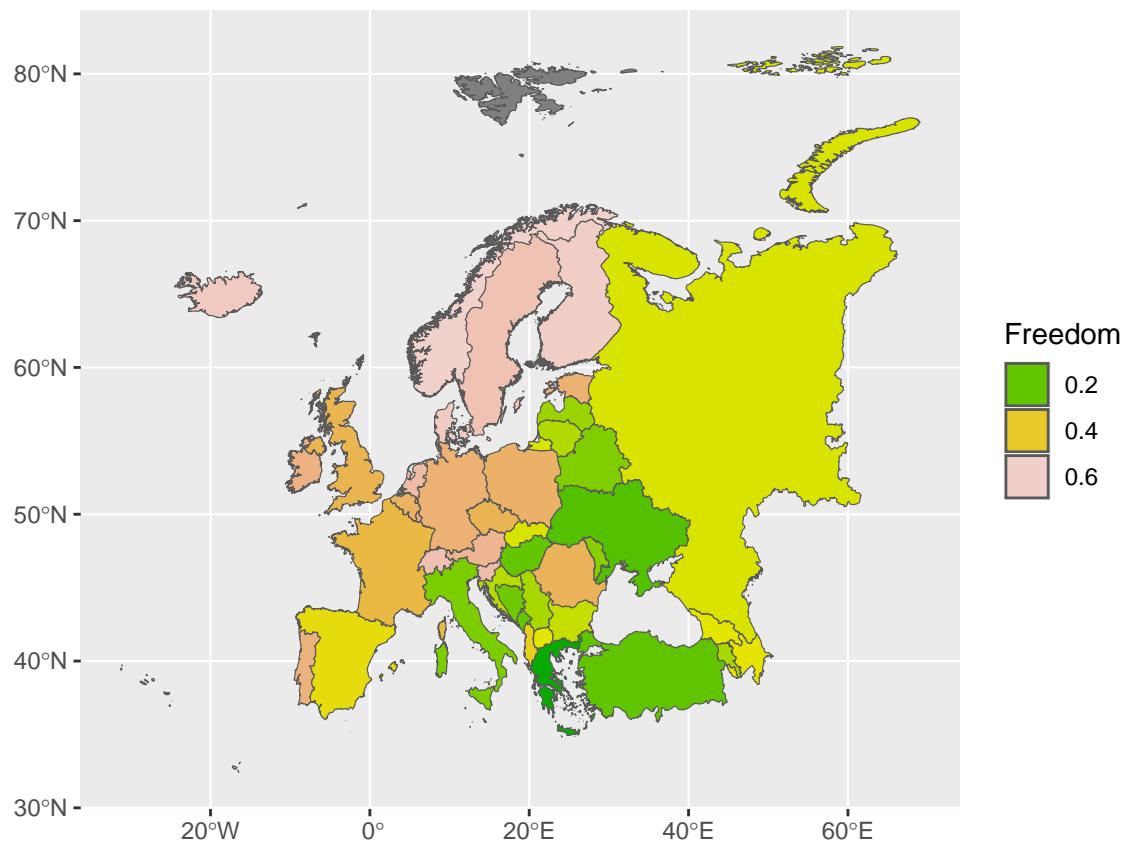
```
ggplot(EU_happ) +  
  geom_sf(aes(fill = Healthy.life.expectancy)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +  
  scale_fill_gradientn(colours = terrain.colors(100), limits=c(0.7,1.1)) +  
  guides(fill=guide_legend(title="Life expectancy"))
```

Life expectancy



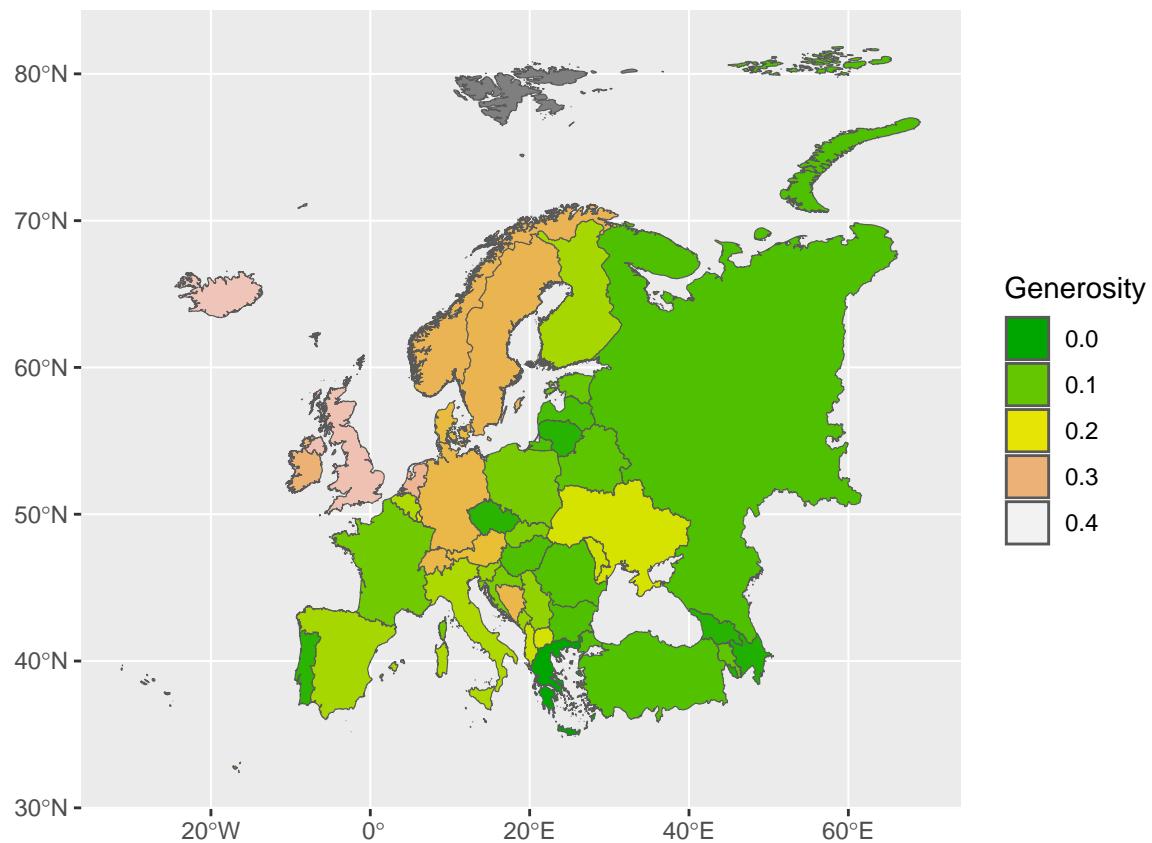
```
ggplot(EU_happ) +  
  geom_sf(aes(fill = Freedom.to.make.life.choices)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +  
  scale_fill_gradientn(colours = terrain.colors(100), limits=c(0.05,0.65)) +  
  guides(fill=guide_legend(title="Freedom"))
```

Freedom



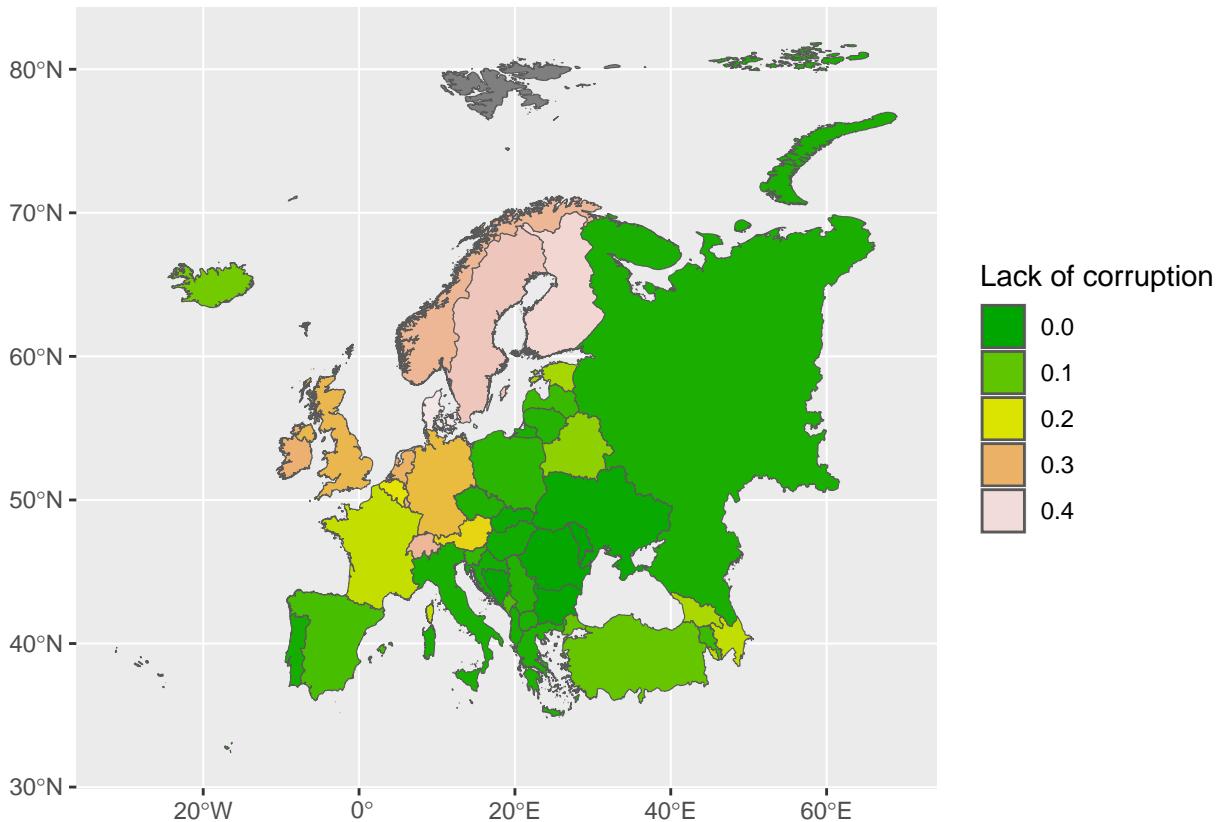
```
ggplot(EU_happ) +  
  geom_sf(aes(fill = Generosity)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +  
  scale_fill_gradientn(colours = terrain.colors(100), limits=c(0,0.4)) +  
  guides(fill=guide_legend(title="Generosity"))
```

Generosity



```
ggplot(EU_happ) +
  geom_sf(aes(fill = Perceptions.of.corruption)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(0,0.42))+
```

### Lack of corruption



## South America

```

SA <- read_sf("South_America.shp")
head(SA)

## Simple feature collection with 6 features and 1 field
## Geometry type: MULTIPOLYGON
## Dimension: XY
## Bounding box: xmin: -109.4461 ymin: -55.90223 xmax: -34.79292 ymax: 12.59028
## Geodetic CRS: WGS 84
## # A tibble: 6 x 2
##   COUNTRY                                geometry
##   <chr>                                     <MULTIPOLYGON [°]>
## 1 Argentina (((-71.01648 -36.47591, -70.98195 -36.48361, -70.95278 -36.48695, -~)
## 2 Bolivia   (((-62.19884 -20.47139, -62.26945 -20.56223, -62.26778 -20.62195, -~)
## 3 Brazil    (((-70.62862 -9.948488, -70.62889 -9.895834, -70.62556 -9.825973, -~)
## 4 Chile     (((-73.61806 -51.6339, -73.60494 -51.62841, -73.59862 -51.6025, -73~)
## 5 Colombia  (((-81.71306 12.49028, -81.72014 12.49632, -81.72015 12.54528, -81.~
## 6 Ecuador   (((-78.70903 -4.584787, -78.72348 -4.58889, -78.76723 -4.603889, -7~
```

```
SA$COUNTRY[!SA$COUNTRY %in% Happiness$Country.or.region]
```

```
## [1] "French Guiana (France)"
## [2] "Falkland Islands (UK)"
```

```

## [3] "Guyana"
## [4] "Suriname"
## [5] "South Georgia and the South Sandwich Is (UK)"

Happ_SA <- Happiness[Happiness$Country.or.region %in% SA$COUNTRY[
  SA$COUNTRY %in% Happiness$Country.or.region],]
NA_rows <- as.data.frame(matrix(NA,nrow=13,ncol=9))
names(NA_rows) <- names(Happ_SA)
Happ_SA <- rbind(Happ_SA,NA_rows)

SA_happ <- cbind(SA,Happ_SA[match(SA$COUNTRY,Happ_SA$Country.or.region),])

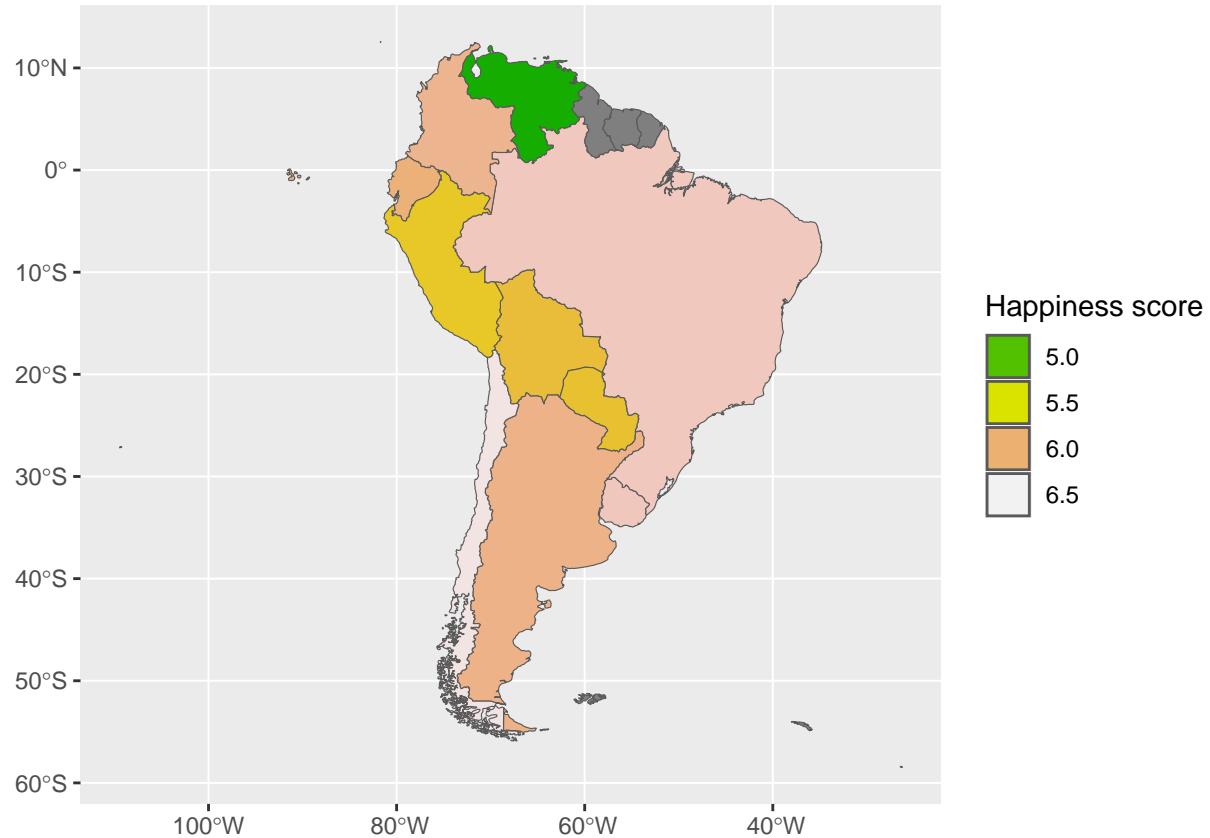
```

```

ggplot(SA_happ) +
  geom_sf(aes(fill = Score)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits = c(4.6,6.5))+ 
  guides(fill=guide_legend(title="Happiness score"))

```

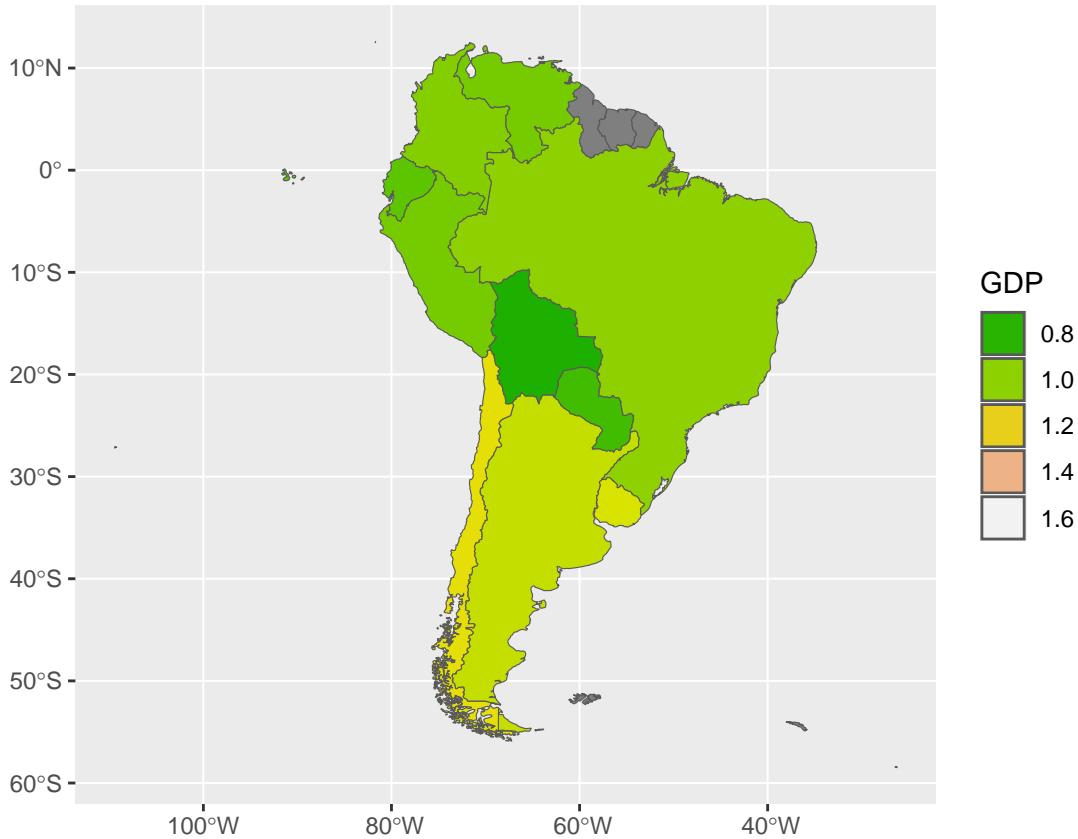
Happniess score



```
ggplot(SA_happ) +
  geom_sf(aes(fill = GDP.per.capita)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(0.7,1.6))+
```

guides(fill=guide\_legend(title="GDP"))

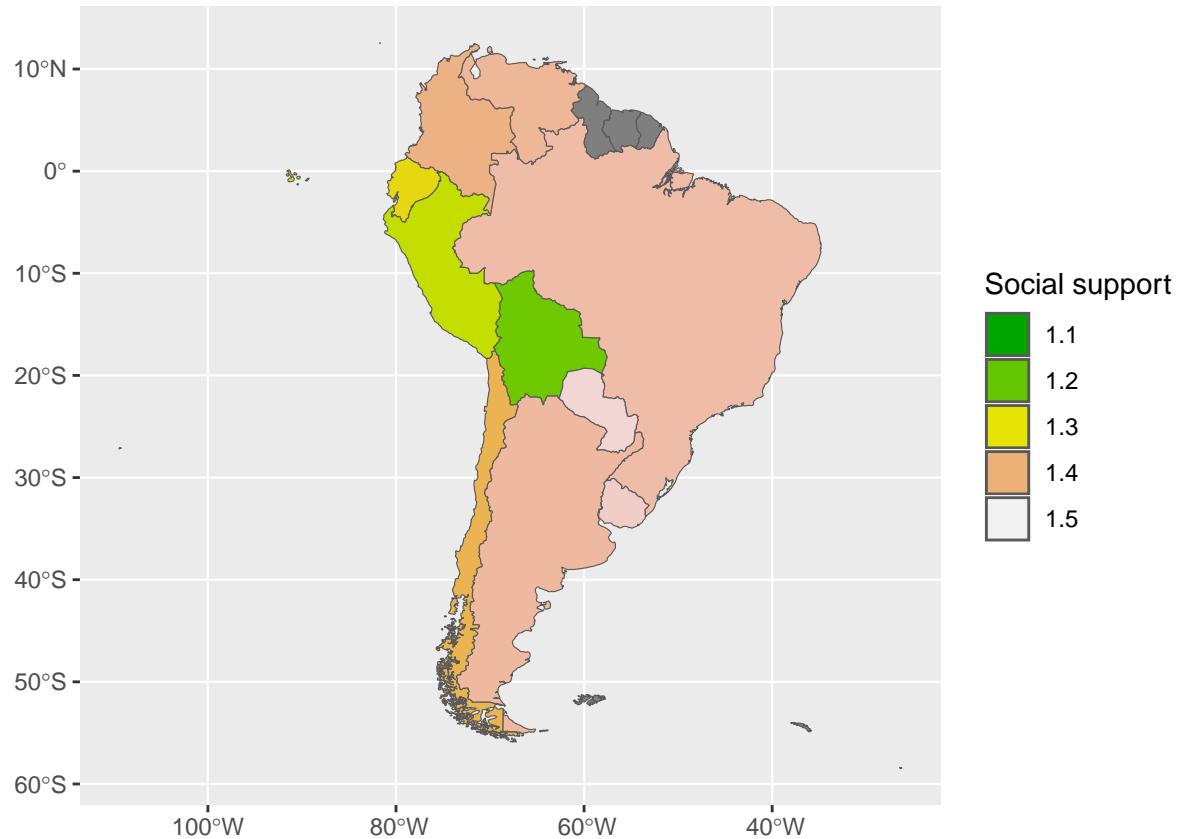
GDP per capita



```
ggplot(SA_happ) +
  geom_sf(aes(fill = Social.support)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(1.1,1.5))+
```

guides(fill=guide\_legend(title="Social support"))

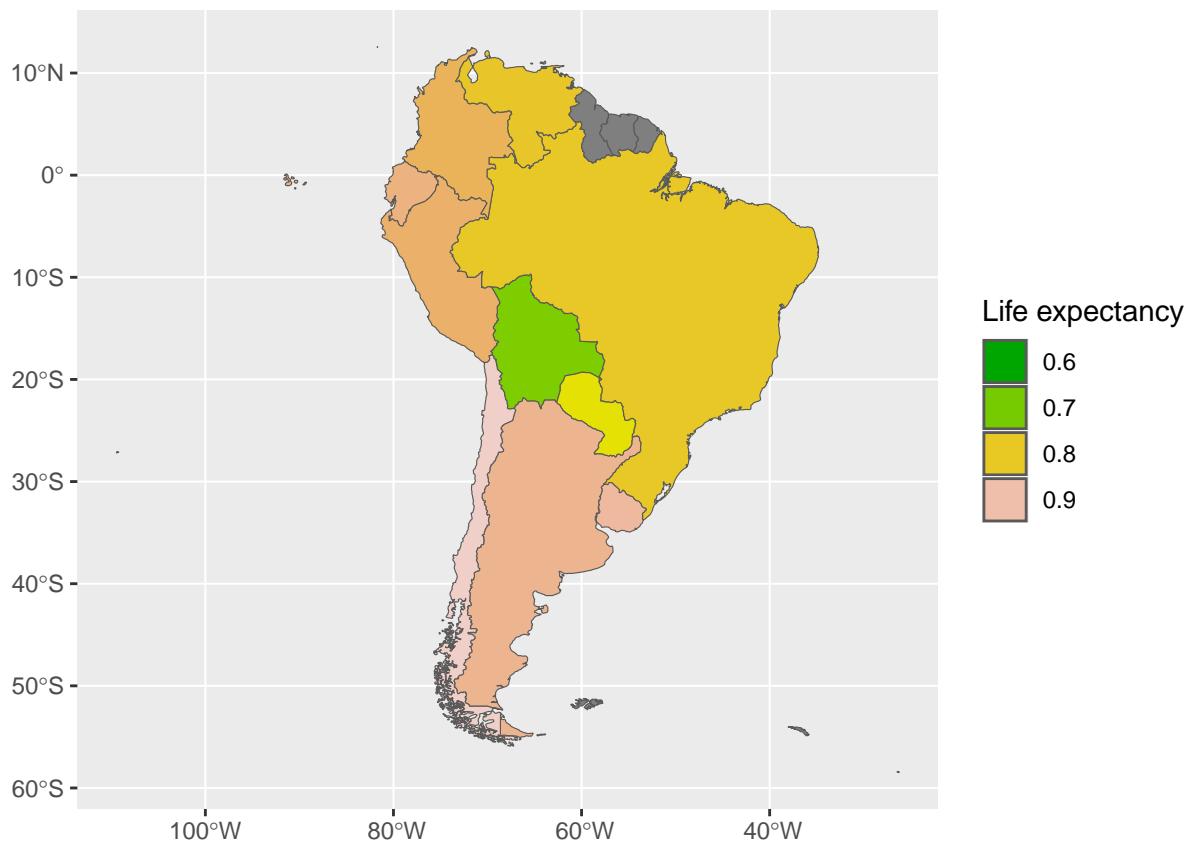
Social support



```
ggplot(SA_happ) +
  geom_sf(aes(fill = Healthy.life.expectancy)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(0.6,0.95))+
```

guides(fill=guide\_legend(title="Life expectancy"))

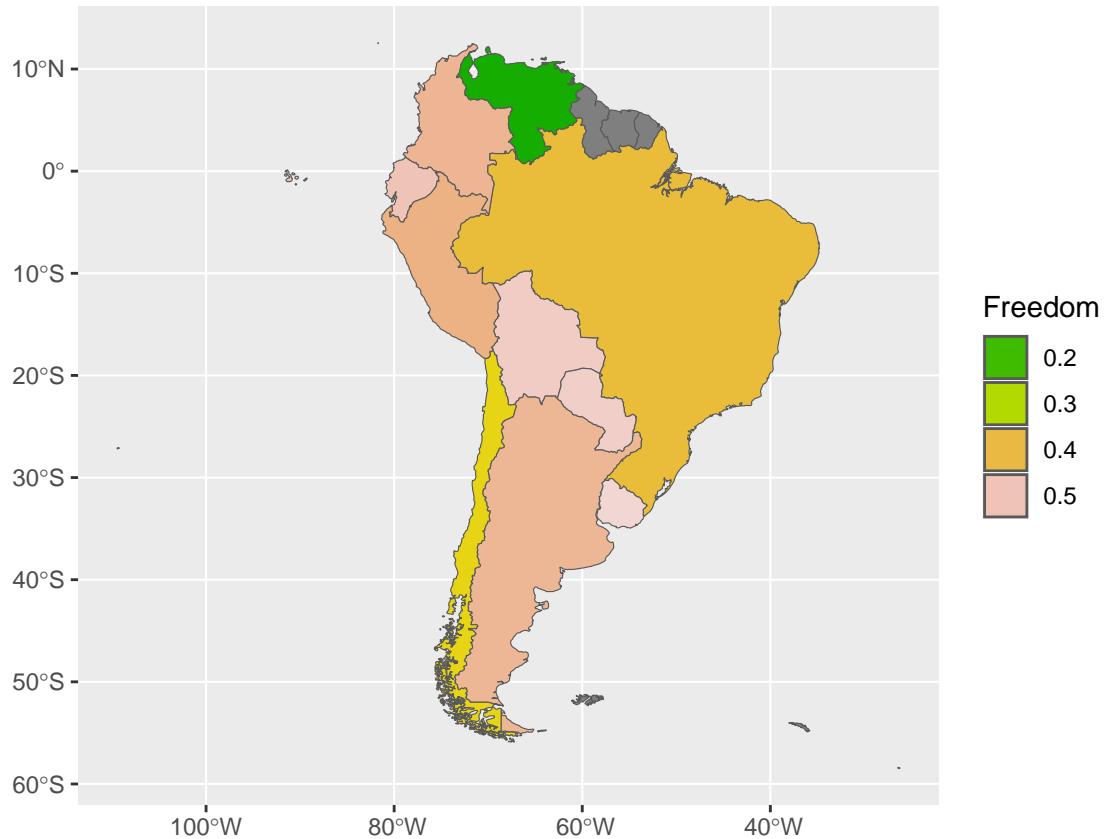
Life expectancy



```
ggplot(SA_happ) +
  geom_sf(aes(fill = Freedom.to.make.life.choices)) +
  theme(axis.title.x = element_blank(),
        axis.title.y = element_blank())+
  scale_fill_gradientn(colours = terrain.colors(100),limits=c(0.13,0.55))+
```

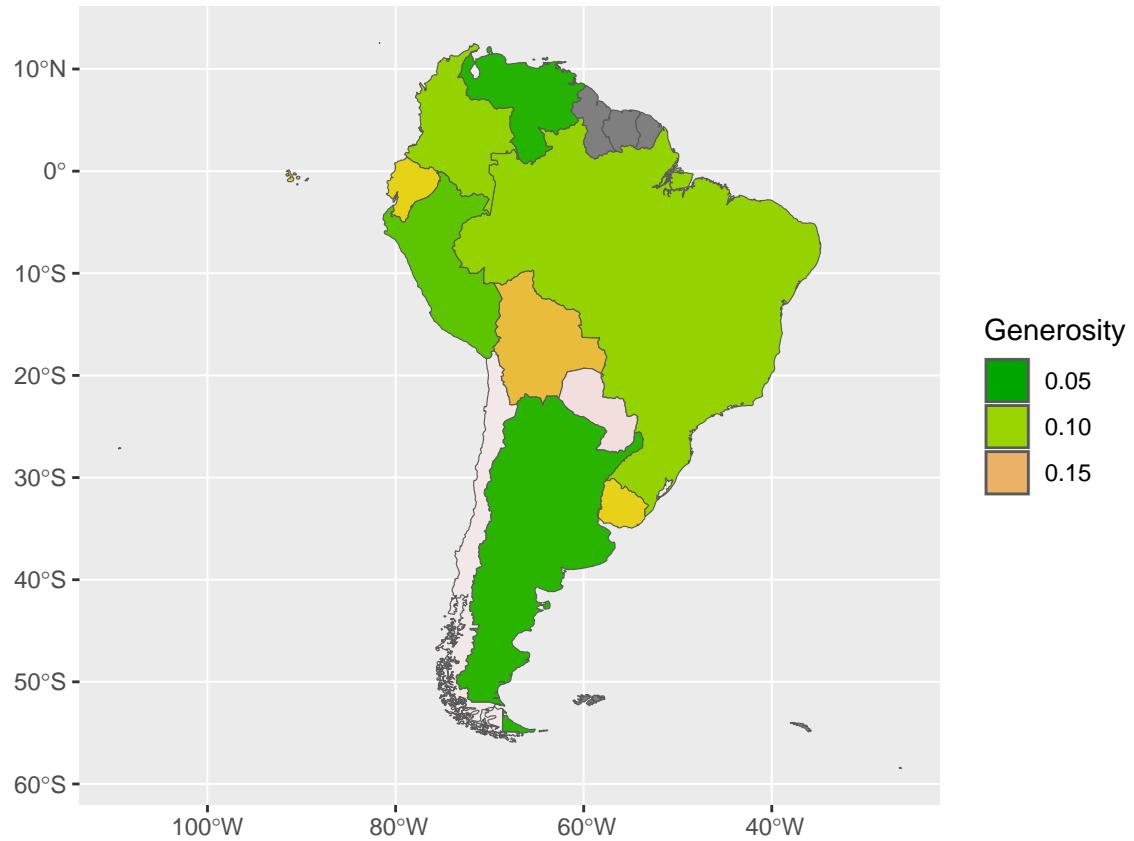
guides(fill=guide\_legend(title="Freedom"))

**Freedom**



```
ggplot(SA_happ) +  
  geom_sf(aes(fill = Generosity)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +  
  scale_fill_gradientn(colours = terrain.colors(100), limits=c(0.05,0.19)) +  
  guides(fill=guide_legend(title="Generosity"))
```

Generosity



```
ggplot(SA_happ) +  
  geom_sf(aes(fill = Perceptions.of.corruption)) +  
  theme(axis.title.x = element_blank(),  
        axis.title.y = element_blank()) +  
  scale_fill_gradientn(colours = terrain.colors(100), limits=c(0.02,0.16)) +  
  guides(fill=guide_legend(title="Lack of corruption"))
```

## Lack of corruption

