

Assignment 3 ST302

your name

2023-03-30

```
#install.packages(c("RColorBrewer", "gapminder", "ggrepel", "colorspace", "ISLR2"))
#install.packages(c("cvd_emulator"))
library(tidyverse)
library(gapminder)
library(ggrepel)
library(colorspace)
library(ISLR2)
library(ggplot2)
#library(cvd_emulator)
```

Question 1.

- a. Here is figure 6.27 in the notes (shown below), and beside it how the figure appears in the most common type of colour blindness (Deuteranope). Is the choice of colours in the original plot successful for colour blind people?

Wheats RGB Colour Code is: (227,199,166) Blue's Colour Code is: (0,0,256)

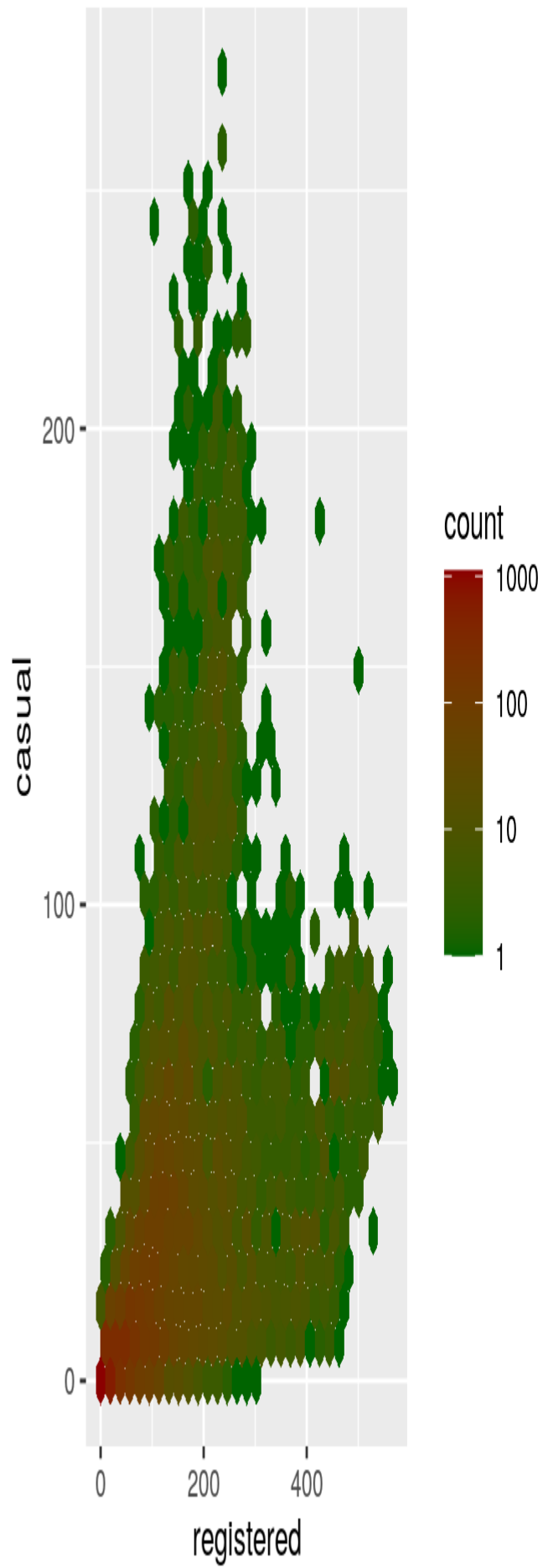
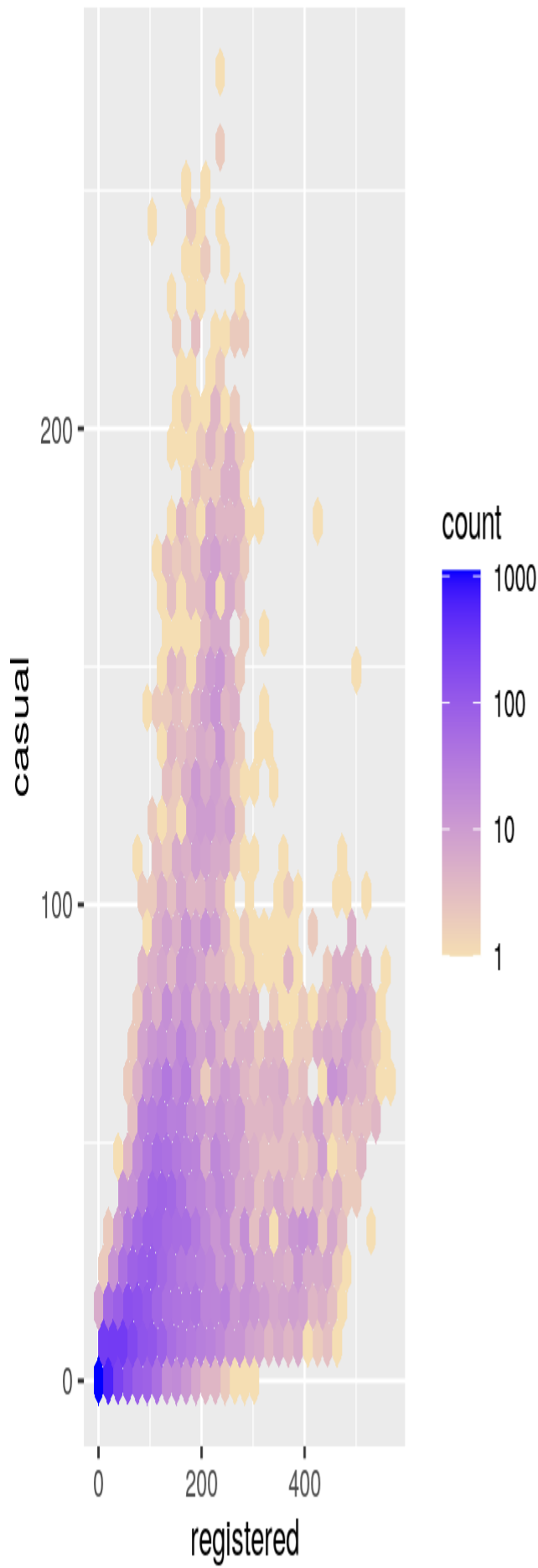
The colours in the plot are is successful for colorblind people.

If you suffer from the most common type of colour blindness (Deuteranope), the choice of colours in the original plot is successful for people with this type of color blindness as every shade is easily distinguishable from each other. This is because Wheat & Blue's color values are very different from each other. . Additionally, the log scale ensures that there are differences in values while using only 10 colours.

B.

```
p <- ggplot(data=Bikeshare, aes(x=registered, y=casual))
```

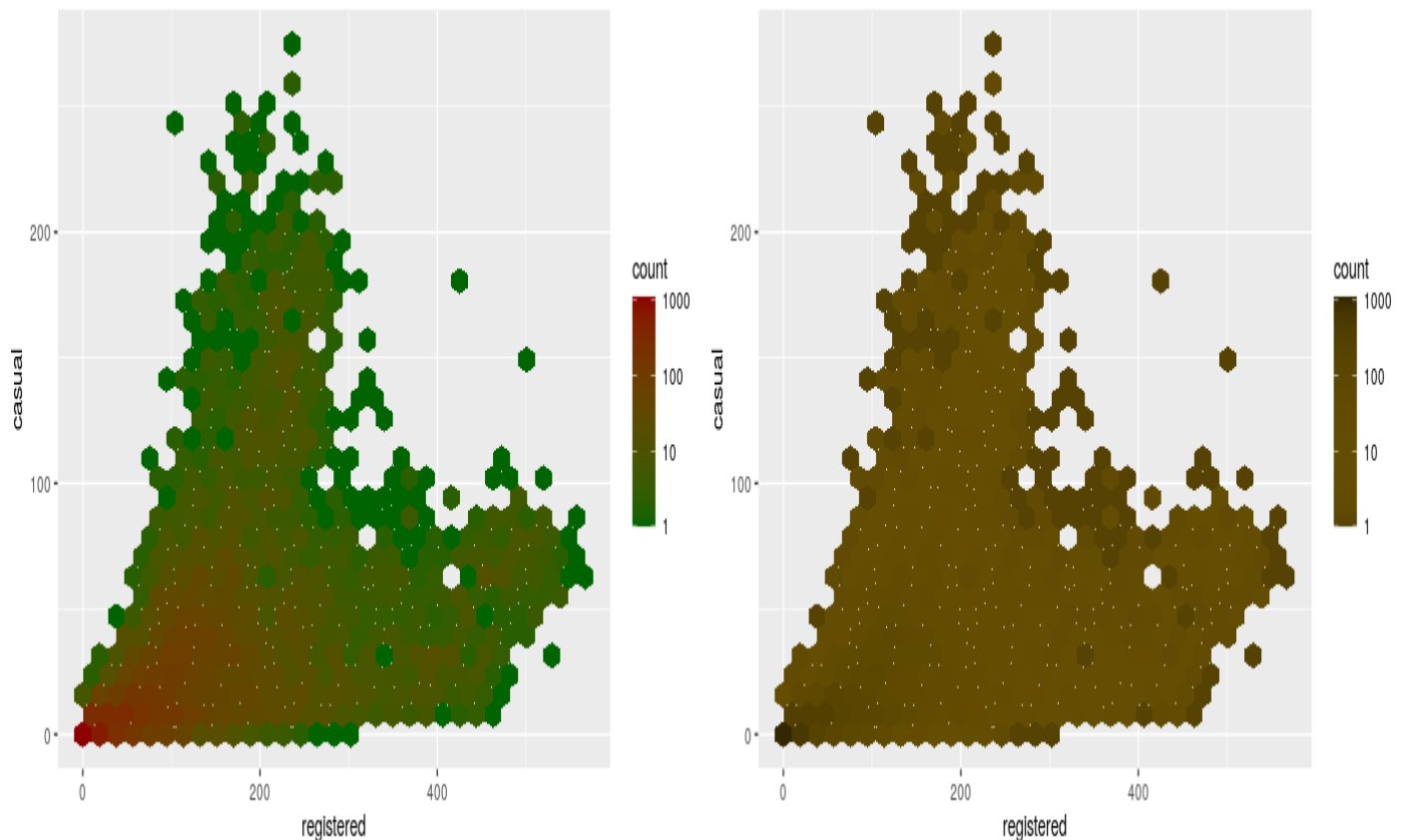
```
require(gridExtra) #Plot 1 is how it normally looks, Plot 2 is how the colorblind see it.
plot1 <- p + geom_hex() + scale_fill_gradient(low = "wheat", high = "blue", trans="log10")
plot2 <- p + geom_hex() + scale_fill_gradient(low = "darkgreen", high = "darkred", trans = "log10")
grid.arrange(plot1, plot2, ncol=2)
```



```
cvd_emulator(here::here("hwk", "figs", "q1b.png"), overwrite=TRUE)
```

```
## Convert deutan    -> /users/students/20325583/ST302/hwk/figs/deutan_q1b.png  
## Convert protan    -> /users/students/20325583/ST302/hwk/figs/protan_q1b.png  
## Convert tritan     -> /users/students/20325583/ST302/hwk/figs/tritan_q1b.png  
## Convert desaturate -> /users/students/20325583/ST302/hwk/figs/desaturate_q1b.png
```

```
knitr::include_graphics(here::here("hwk", "figs", "q1b.png"))  
knitr::include_graphics(here::here("hwk", "figs", "deutan_q1b.png"))
```



B. Explain why this plot does not work for a color blind person.

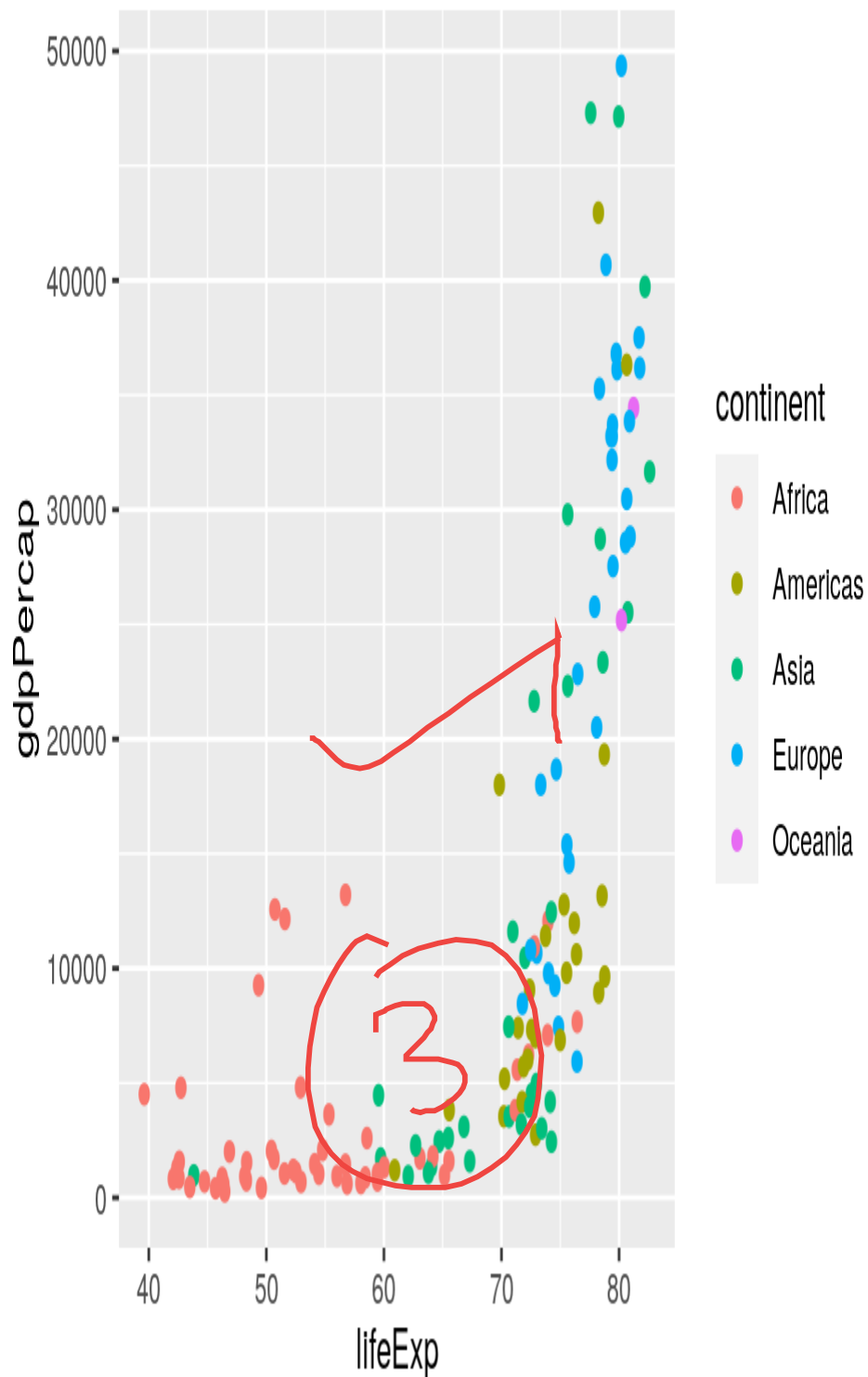
This plot may not work for a color blind person as dark green and dark red are problematic colors for people with Deuteranope. It is very common to mix up both light shades of red and green, and dark shades of red and green. Even by looking at the plot, it is fairly challenging to differentiate between some of the shades of red and green in the plot.

Question 2

- a. For the gapminder data, plot lifeExp versus gdpPercap for year=2007. Use colour to identify the points, using the default ggplot colours. Comment on the relationship between the two variables.

```
gapminder2007 <- gapminder %>% filter(year== 2007)
```

```
ggplot(data = gapminder2007, aes(x = lifeExp, y = gdpPercap, color = continent)) + geom_point()
```



#There is a positive correlation between Life Expectancy and GDP per capita.

#As life expectancy increases, GDP per capita increases as well.

#The continent with the lowest Life Expectancy and GDP per capita is Africa.

#European countries having the highest GDP per capita and life expectancy, with Europe having the country with the highest Life Expectancy and GDP per capita.

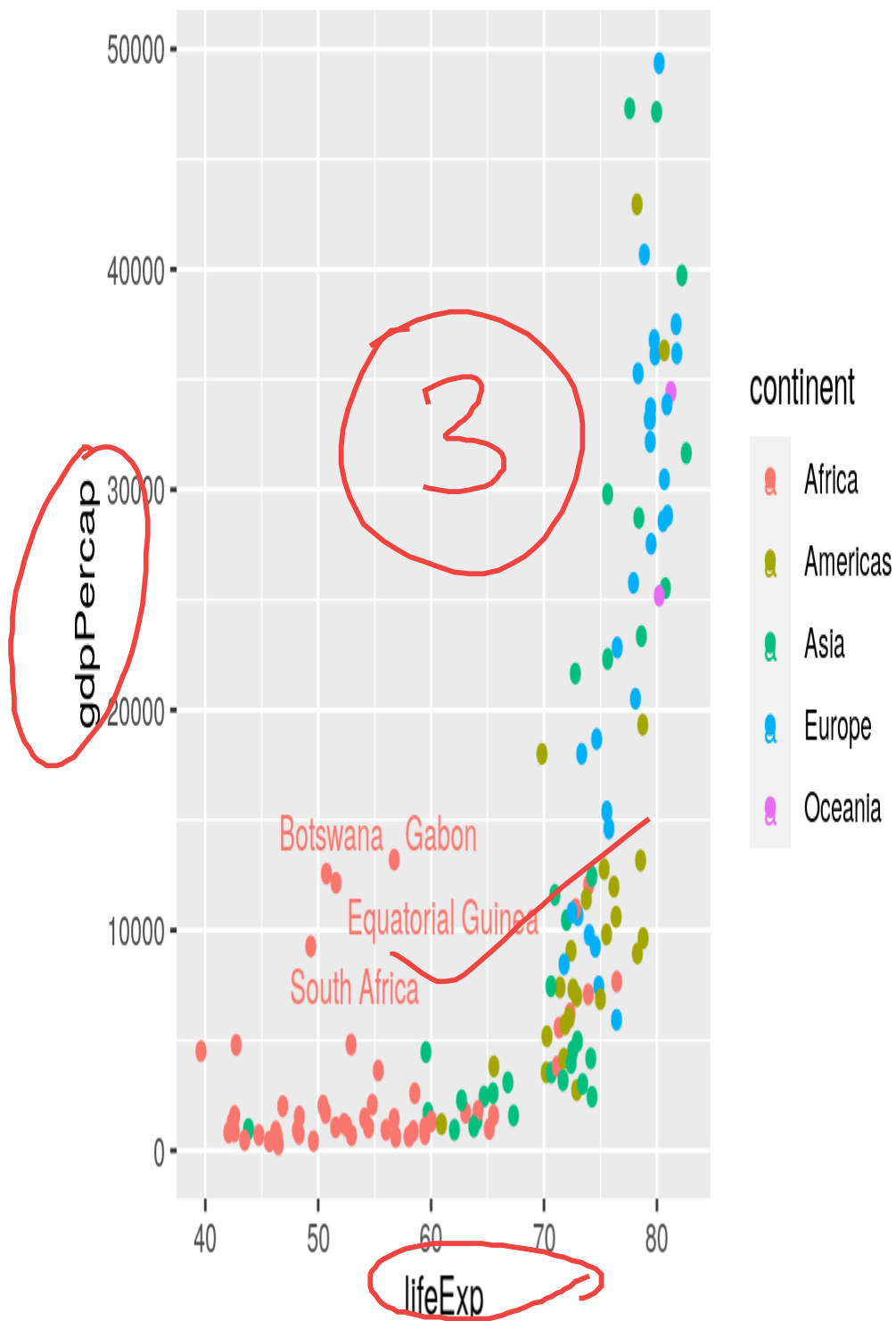
- b. What continent do the outlier countries belong to? What is unusual about their values? Add labels to the four outlier countries and place the labels nicely so they do not overlap the points.

#If you are referring to the gapminder filtered to 2007, Botswana, Equatorial Guinea, Gabon, and South Africa are the values that are the outliers.

#They are from Africa.

```
subset2007 <- filter(gapminder2007, gdpPercap > 8000, lifeExp < 60)
```

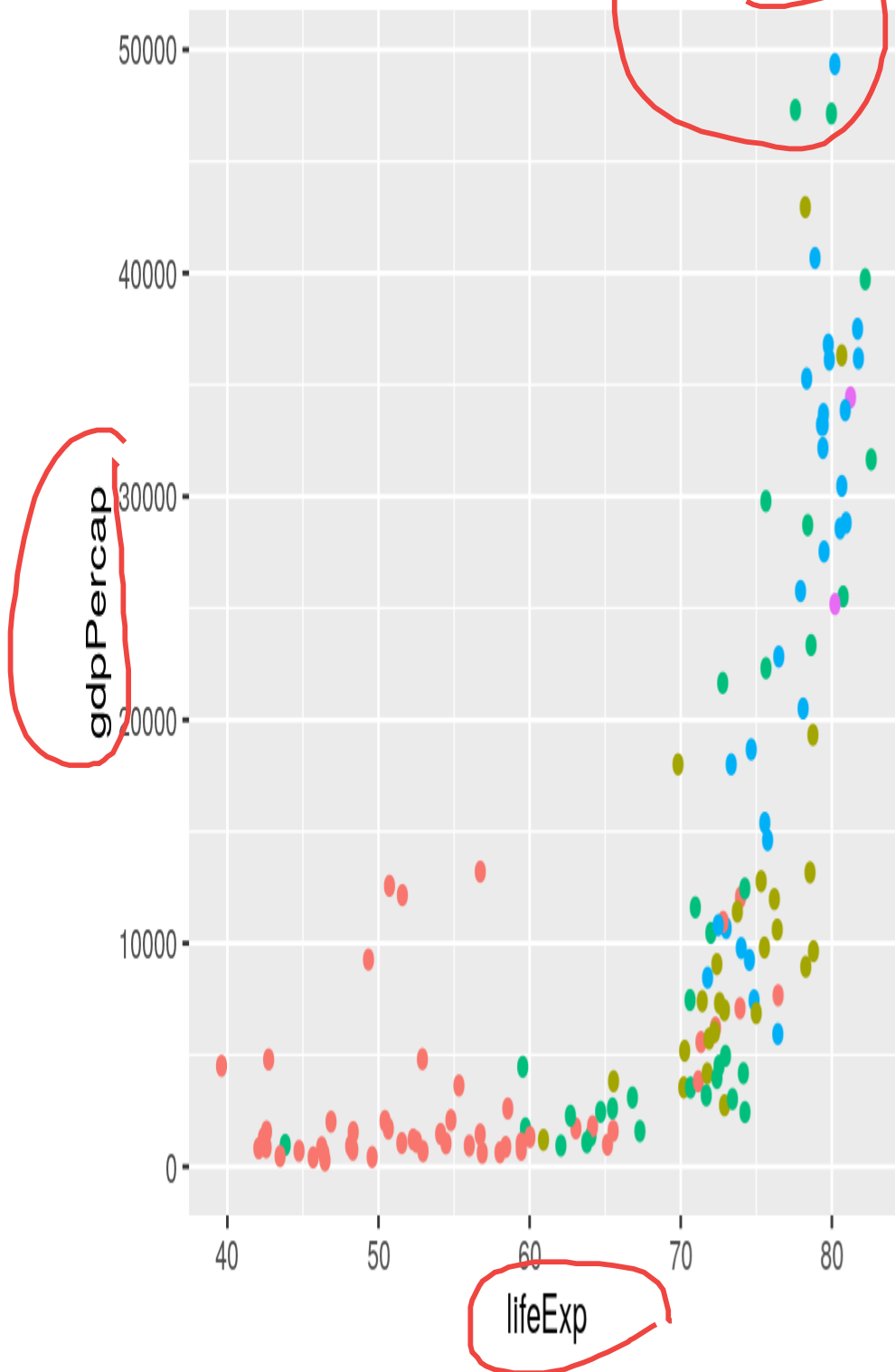
```
ggplot(data = gapminder2007, aes(x = lifeExp, y = gdpPercap, color = continent)) +  
  geom_point() +  
  geom_text_repel(data = subset2007, aes(label = country), size = 3.5, nudge_x = 2, nudge_y = -2)
```



c. The gapminder package comes with a colour vector called `continent_colors`. Redo the plot with these colours. (The help page for `continent_colors` gives some examples of how to do this.) Are these colours an improvement over the default colours?

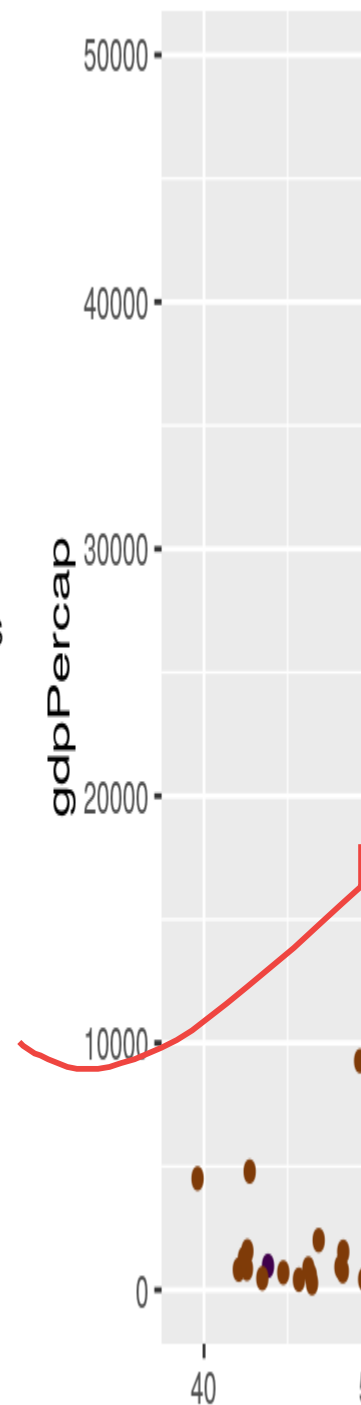
No, the colors here are a lot darker in color. Oceania in particular was very hard to find without looking at the original plot. I also believe the contrast in colors in the original is a lot more noticeable. I don't believe the `continent_colors` is better than the original data in any way shape or form.

```
require(gridExtra)
plot5 <- ggplot(gapminder2007, aes(x = lifeExp, y = gdpPercap, color = continent))+geom_point()
plot6 <- ggplot(gapminder2007, aes(x= lifeExp, y = gdpPercap, color = continent))+geom_point()
+scale_colour_manual(values = continent_colors)
grid.arrange(plot5, plot6, ncol=2)
```



continent

- Africa
- Americas
- Asia
- Europe
- Oceania



d. What happens if you colour by country instead using country_colors? Do not include the plot!

You get the name of every country with their corresponding color.

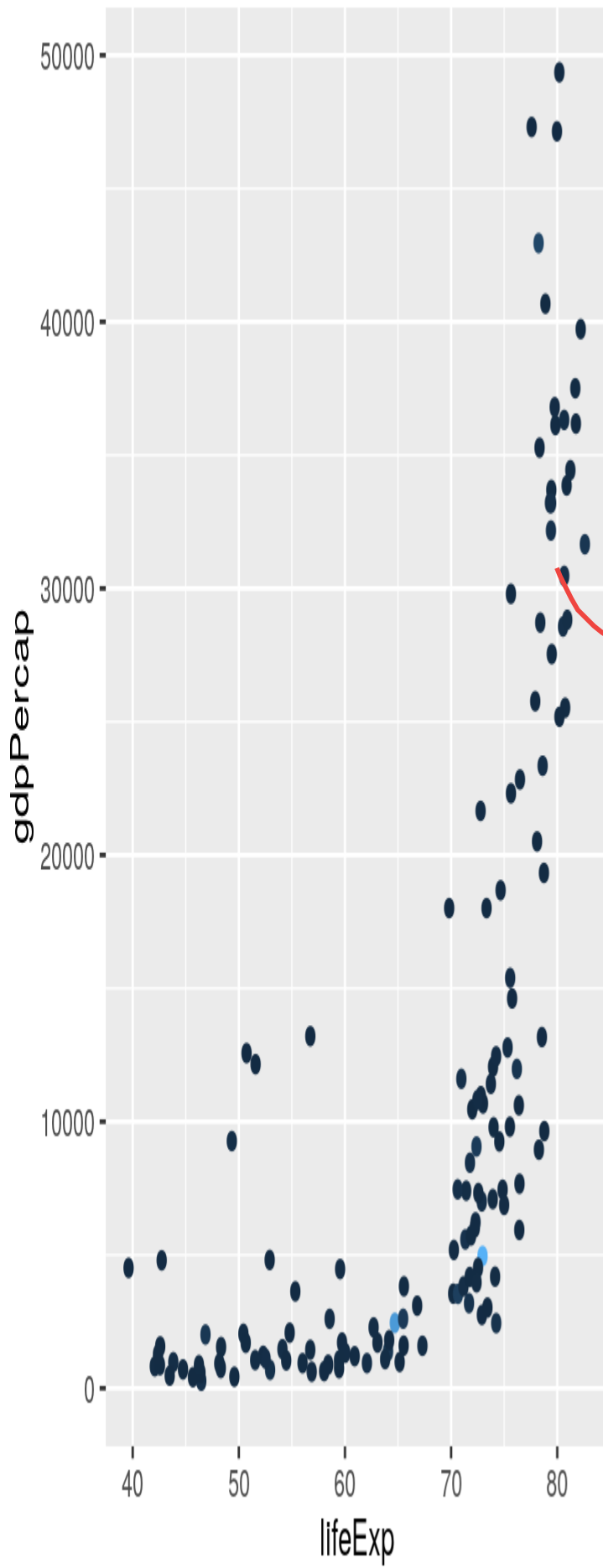
and there colours are not an improvement

```
ggplot(gapminder, aes(x = gdpPercap, y = lifeExp, color = country))+geom_point()  
+scale_colour_manual(values = country_colors)
```

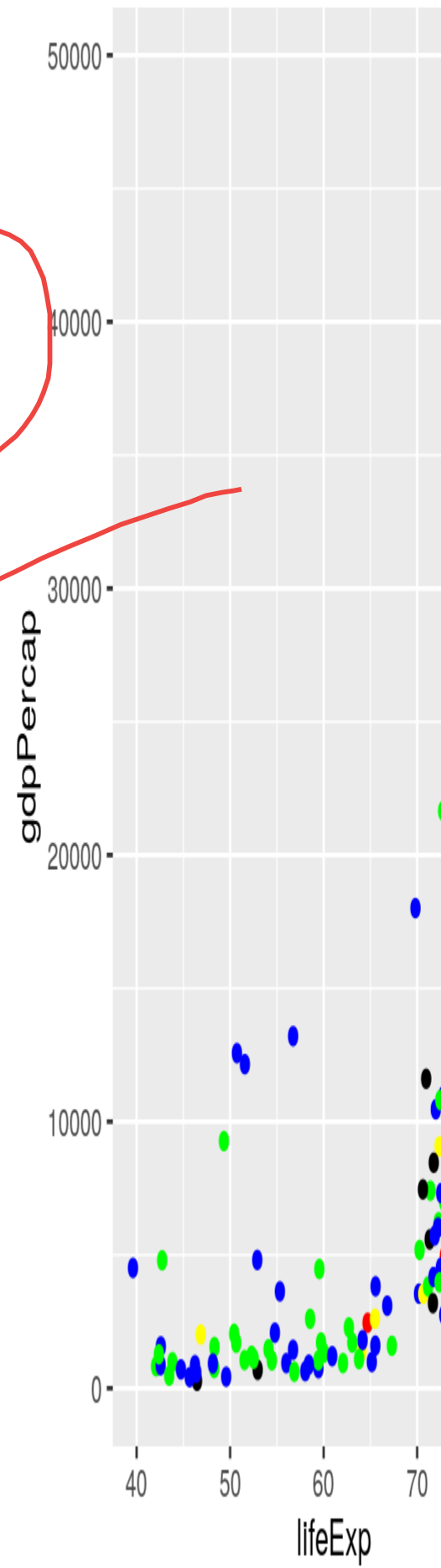
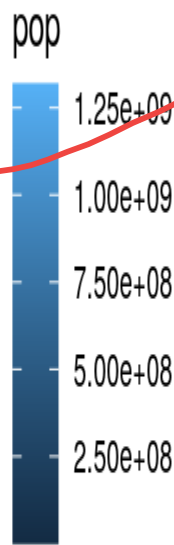
- e. Instead of colouring by continent, colour by population. Present two plots, one with the default colour scheme, and the second a colour scheme which is more informative. Explain why the default colour scheme is not helpful. Explain the steps you used to improve this and give the reasons for your final choice. What do you learn from your final choice?

```
gapminder2007$cutdown <- cut(gapminder2007$pop, breaks = c(0, 10000000, 25000000, 50000000,  
75000000, 100000000, Inf),  
labels = c("Under 10 Million", "10-25 Million", "25-50 Million", "50-75 Million", "75-100  
Million", "Over 100 Million"))  
BasicColors <- c("blue", "green", "black", "yellow", "magenta", "red")
```

```
require(gridExtra)  
plot3 <- ggplot(gapminder2007, aes(x = lifeExp, y = gdpPercap, color = pop))+geom_point()  
plot4 <- ggplot(gapminder2007, aes(x = lifeExp, y = gdpPercap, color = cutdown)) +geom_point() +  
scale_color_manual(values = BasicColors)  
grid.arrange(plot3, plot4, ncol=2)
```

4



The default color scheme is not helpful because it uses a continuous color scale to represent the levels of population. They are hard to distinguish from each other as all the values blend in together. This makes it harder to interpret between each other & there isn't a clear distinction for the population values.

The color scheme was improved by deviding up each category of population with an assigned color. This makes it easier for someone to see the relationship between the 3 variables used. The colors used are a lot more distinguishable compared to the one in the default as well.

Question 3

a. What does this graph tell you? Write a few sentences.

Here we can see the average life expectancy of a person in each country in Europe in the years 1957 and 2007. The highest extension in life expectancy for a country is Turkey. The shortest extension in life expectancy for a country is Denmark. Ireland is notably highlighted in green as this is the country the module is being taught in. 7 Countries are above above 80. There are 30 countries listed (When counting Bosnia and Herzegovinia as one) Despite having the highest jumps in life expectancy, Turkey has the lowest average life expectancy of 2007. That countries in Western Europe have a higher average life expectancy than countries in Eastern Europe.

b. Write code to reconstruct this graph.

```
gapminder %>% filter(continent=="Europe", year == c(1952,2007)) %>% mutate(yr = factor(year))
%>%
ggplot(aes(x = lifeExp, y = reorder(country,lifeExp,max), color = yr))+geom_point()+
geom_line(aes(group = country), color = "yellow")+
geom_line(data = filter(gapminder, country == "Ireland", year == c(1952,2007)), color = "green")
+ylab(" ")
```

