Gapminder - Modeling the relationship between GDP and life expectancy

Abigail Castro, Courtney Kennedy, Aditi jain, Qinyuan Jiang, Tanushri Roy

Instructions: Divide the task among yourself. Each team member should contribute to at least one part of the assignment. The person whose name starts last in the alphabetic order shares screen and compiles the Rmd file. All team members should submit the Rmd and pdf files.

The gapminder data summarizes the progression of countries over time, looking at statistics like life expectancy and GDP. Use RDS 25.2 as starting point and answer the following question: How well does GDP predict life expectancy in each country and continent?

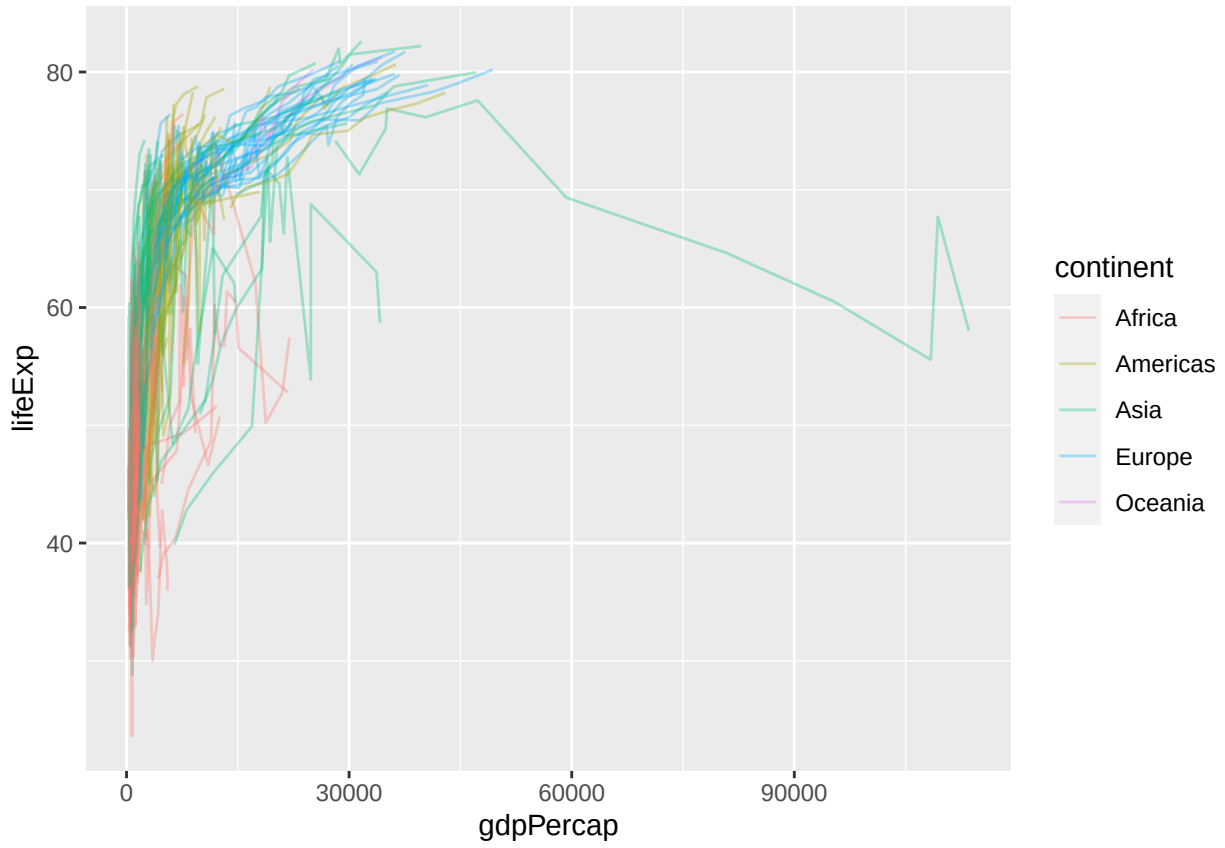
```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.6     v dplyr   1.0.3
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(modelr)
library(gapminder)
str(gapminder)
```

```
## tibble [1,704 x 6] (S3: tbl_df/tbl/data.frame)
##  $ country  : Factor w/ 142 levels "Afghanistan",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ continent: Factor w/ 5 levels "Africa","Americas",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ year     : int [1:1704] 1952 1957 1962 1967 1972 1977 1982 1987 1992 1997 ...
##  $ lifeExp  : num [1:1704] 28.8 30.3 32 34 36.1 ...
##  $ pop      : int [1:1704] 8425333 9240934 10267083 11537966 13079460 14880372 12881816 13867957 163
##  $ gdpPercap: num [1:1704] 779 821 853 836 740 ...
```

```
gapminder %>%
  ggplot(aes(gdpPercap, lifeExp, group = country, color = continent)) +
    geom_line(alpha = 1/3)
```



1. Use Linear modeling

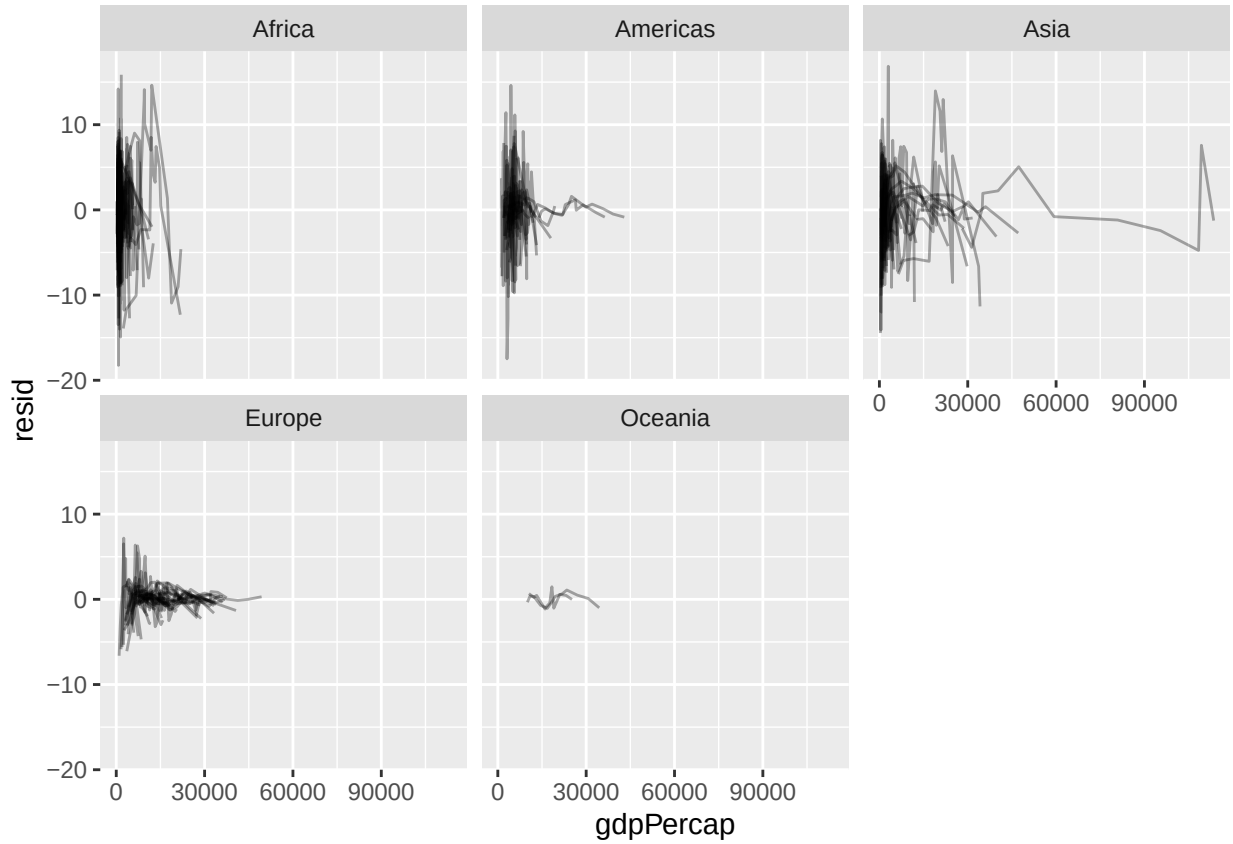
Follow the steps in RDS 25.2

```
by_country <- gapminder %>%
  group_by(country, continent) %>%
  nest()

country_model <- function(df) {
  lm(lifeExp ~ gdpPercap, data = df)
}

models <- map(by_country$data, country_model)

by_country <- by_country %>%
  mutate(model = map(data, country_model))

by_country %>%
  arrange(continent, country)
```

```
## # A tibble: 142 x 4
## # Groups:   country, continent [142]
##    country                  continent data             model
##    <fct>                    <fct>     <list>           <list>
```

```
##  1 Algeria                  Africa    <tibble [12 x 4]> <lm>
##  2 Angola                   Africa    <tibble [12 x 4]> <lm>
##  3 Benin                    Africa    <tibble [12 x 4]> <lm>
##  4 Botswana                 Africa    <tibble [12 x 4]> <lm>
##  5 Burkina Faso             Africa    <tibble [12 x 4]> <lm>
##  6 Burundi                  Africa    <tibble [12 x 4]> <lm>
##  7 Cameroon                 Africa    <tibble [12 x 4]> <lm>
##  8 Central African Republic Africa    <tibble [12 x 4]> <lm>
##  9 Chad                     Africa    <tibble [12 x 4]> <lm>
## 10 Comoros                  Africa    <tibble [12 x 4]> <lm>
## # ... with 132 more rows
```

```
by_country <- by_country %>%
  mutate(
    resids = map2(data, model, add_residuals)
  )

resids <- unnest(by_country, resids)

resids %>%
  ggplot(aes(gdpPercap, resid, group = country)) +
    geom_line(alpha = 1 / 3) +
    facet_wrap(~continent)
```



2. Try different model families :

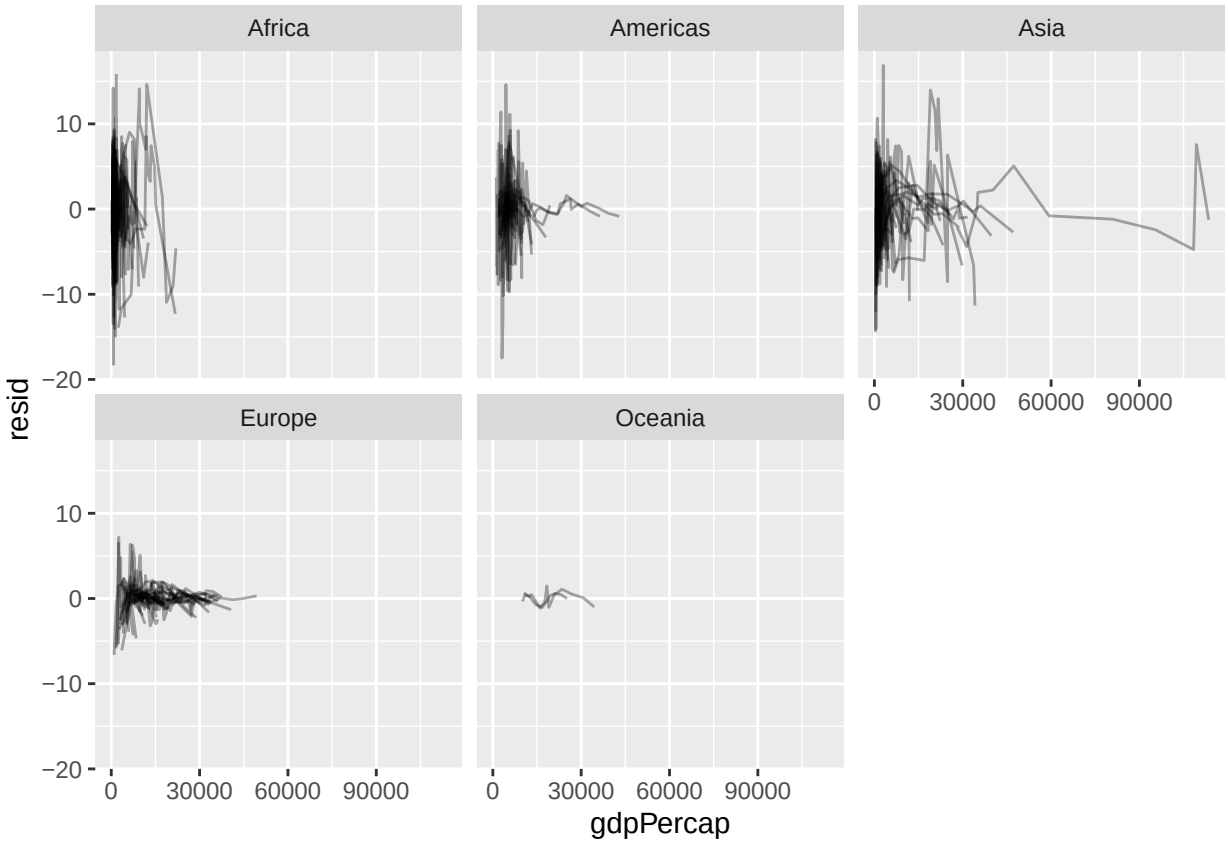
(See RDS 23.6)

2.1 Generalized linear models

```
by_country1 <- gapminder %>%
  group_by(country, continent) %>%
  nest()

country_model1 <- function(df) {
  stats::glm(lifeExp ~ gdpPercap, data = df)
}

models1 <- map(by_country1$data, country_model1)

by_country1 <- by_country1 %>%
  mutate(model = map(data, country_model1))

by_country1 %>%
  arrange(continent, country)
```

```
## # A tibble: 142 x 4
## # Groups:   country, continent [142]
##    country                  continent data              model
##    <fct>                    <fct>     <list>            <list>
##  1 Algeria                  Africa    <tibble [12 x 4]> <glm>
##  2 Angola                   Africa    <tibble [12 x 4]> <glm>
##  3 Benin                    Africa    <tibble [12 x 4]> <glm>
##  4 Botswana                 Africa    <tibble [12 x 4]> <glm>
##  5 Burkina Faso             Africa    <tibble [12 x 4]> <glm>
##  6 Burundi                  Africa    <tibble [12 x 4]> <glm>
##  7 Cameroon                 Africa    <tibble [12 x 4]> <glm>
##  8 Central African Republic Africa    <tibble [12 x 4]> <glm>
##  9 Chad                     Africa    <tibble [12 x 4]> <glm>
## 10 Comoros                  Africa    <tibble [12 x 4]> <glm>
## # ... with 132 more rows
```

```
by_country1 <- by_country1 %>%
  mutate(
    resids1 = map2(data, model, add_residuals)
  )

resids1 <- unnest(by_country1, resids1)

resids1 %>%
  ggplot(aes(gdpPercap, resid, group = country)) +
    geom_line(alpha = 1 / 3) +
    facet_wrap(~continent)
```



2.4 Robust linear models

```
by_country2 <- gapminder %>%
  group_by(country, continent) %>%
  nest()

country_model2 <- function(df) {
  MASS::rlm(lifeExp ~ gdpPercap, data = df)
}

models2 <- map(by_country2$data, country_model2)
```

```
## Warning in rlm.default(x, y, weights, method = method, wt.method = wt.method, :
## 'rlm' failed to converge in 20 steps

## Warning in rlm.default(x, y, weights, method = method, wt.method = wt.method, :
## 'rlm' failed to converge in 20 steps

## Warning in rlm.default(x, y, weights, method = method, wt.method = wt.method, :
## 'rlm' failed to converge in 20 steps
```

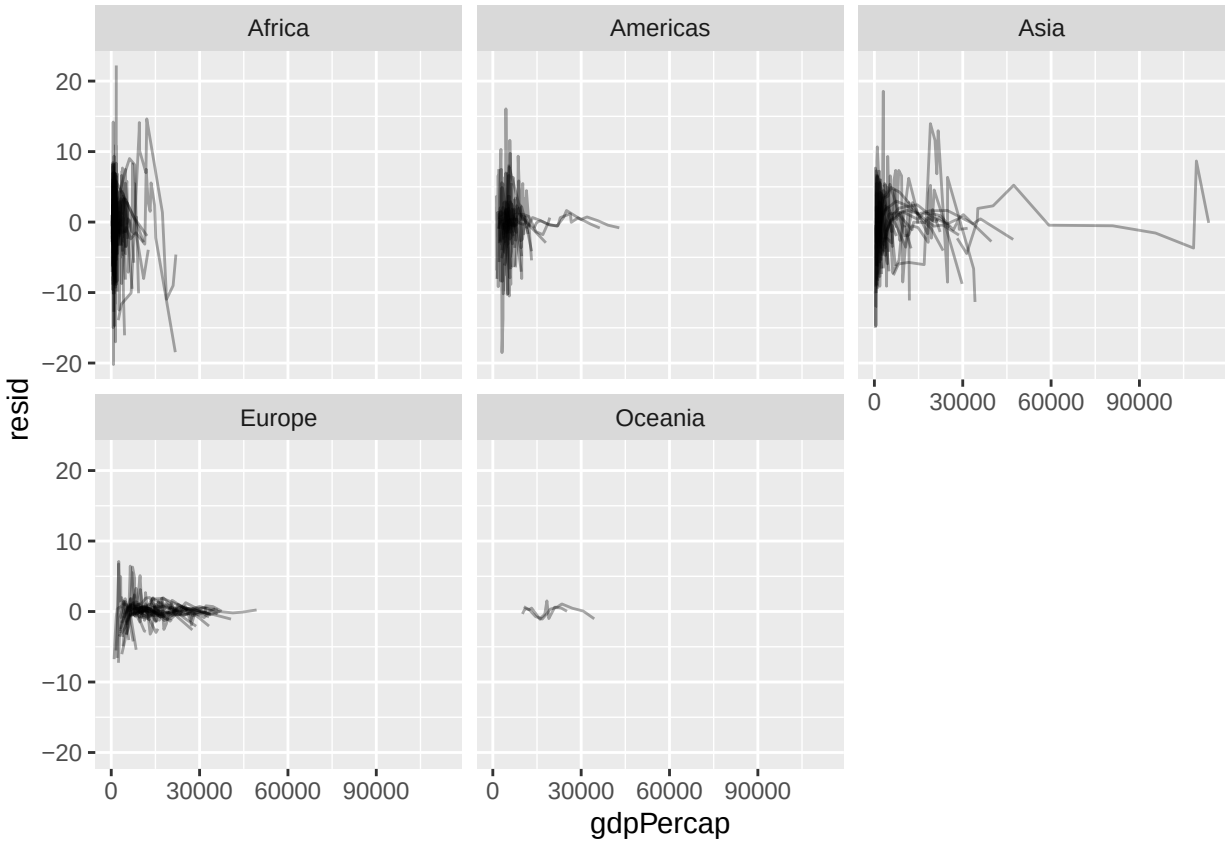
```
by_country2 <- by_country2 %>%
  mutate(model = map(data, country_model2))
```

```
## Warning: Problem with `mutate()` input `model`.
## i 'rlm' failed to converge in 20 steps
## i Input `model` is `map(data, country_model2)`.
## i The error occurred in group 51: country = "Guatemala", continent = "Americas".

## Warning: Problem with `mutate()` input `model`.
## i 'rlm' failed to converge in 20 steps
## i Input `model` is `map(data, country_model2)`.
## i The error occurred in group 111: country = "Senegal", continent = "Africa".

## Warning: Problem with `mutate()` input `model`.
## i 'rlm' failed to converge in 20 steps
## i Input `model` is `map(data, country_model2)`.
## i The error occurred in group 132: country = "Turkey", continent = "Europe".
```

```
by_country2 %>%
  arrange(continent, country)
```

```
## # A tibble: 142 x 4
## # Groups:   country, continent [142]
##    country                  continent data              model
##    <fct>                    <fct>     <list>            <list>
##  1 Algeria                  Africa    <tibble [12 x 4]> <rlm>
##  2 Angola                   Africa    <tibble [12 x 4]> <rlm>
##  3 Benin                    Africa    <tibble [12 x 4]> <rlm>
##  4 Botswana                 Africa    <tibble [12 x 4]> <rlm>
##  5 Burkina Faso             Africa    <tibble [12 x 4]> <rlm>
##  6 Burundi                  Africa    <tibble [12 x 4]> <rlm>
##  7 Cameroon                 Africa    <tibble [12 x 4]> <rlm>
##  8 Central African Republic Africa    <tibble [12 x 4]> <rlm>
##  9 Chad                     Africa    <tibble [12 x 4]> <rlm>
## 10 Comoros                  Africa    <tibble [12 x 4]> <rlm>
## # ... with 132 more rows
```

```
by_country2 <- by_country2 %>%
  mutate(
    resids2 = map2(data, model, add_residuals)
  )

resids2 <- unnest(by_country2, resids2)

resids2 %>%
  ggplot(aes(gdpPercap, resid, group = country)) +
    geom_line(alpha = 1 / 3) +
    facet_wrap(~continent)
```



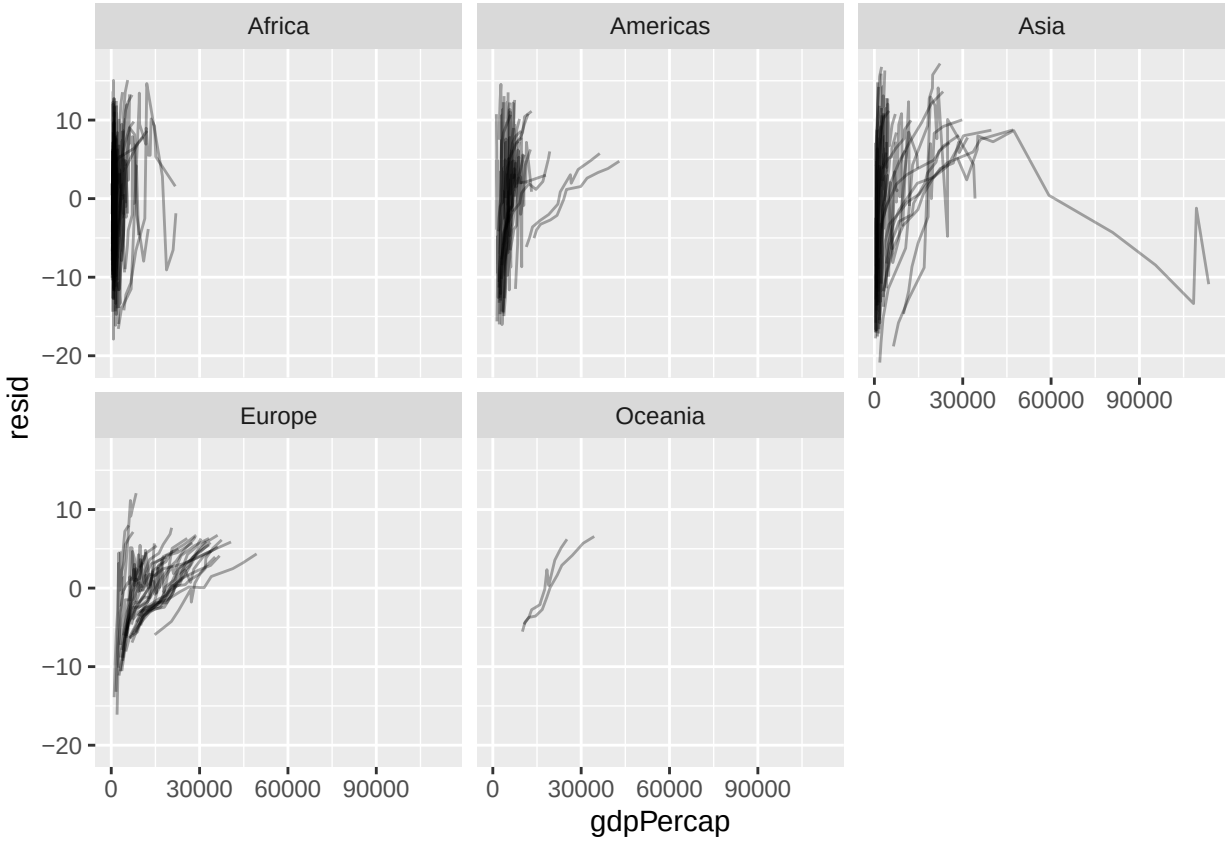
2.5 Trees

```
by_country3 <- gapminder %>%
  group_by(country, continent) %>%
  nest()

country_model3 <- function(df) {
  rpart::rpart(lifeExp ~ gdpPercap, data = df)
}

models3 <- map(by_country3$data, country_model3)

by_country3 <- by_country3 %>%
  mutate(model = map(data, country_model3))

by_country3 %>%
  arrange(continent, country)
```

```
## # A tibble: 142 x 4
## # Groups:   country, continent [142]
##    country                  continent data              model
##    <fct>                    <fct>     <list>            <list>
##  1 Algeria                  Africa    <tibble [12 x 4]> <rpart>
##  2 Angola                   Africa    <tibble [12 x 4]> <rpart>
```

```
##  3 Benin                     Africa    <tibble [12 x 4]> <rpart>
##  4 Botswana                  Africa    <tibble [12 x 4]> <rpart>
##  5 Burkina Faso              Africa    <tibble [12 x 4]> <rpart>
##  6 Burundi                   Africa    <tibble [12 x 4]> <rpart>
##  7 Cameroon                  Africa    <tibble [12 x 4]> <rpart>
##  8 Central African Republic Africa    <tibble [12 x 4]> <rpart>
##  9 Chad                      Africa    <tibble [12 x 4]> <rpart>
## 10 Comoros                   Africa    <tibble [12 x 4]> <rpart>
## # ... with 132 more rows
```

```
by_country3 <- by_country3 %>%
  mutate(
    resids3 = map2(data, model, add_residuals)
  )

resids3 <- unnest(by_country3, resids3)

resids3 %>%
  ggplot(aes(gdpPercap, resid, group = country)) +
    geom_line(alpha = 1 / 3) +
    facet_wrap(~continent)
```



3. Discuss which family performs best. How do you determine the performance?

It looks like we've missed some mild patterns. There's also something interesting going on in Africa: we see some very large residuals which suggests our model isn't fitting so well there. Relatively, none of these families perform ideally. In general, all families had a lack of patterns among their residuals.This is due to the fact that the trend among life expectancy and gdpPercap is not linear but rather, another type such as exponential.