ETC1010: Introduction to Data Analysis

Week 7, part B

Week of introduction

Lecturer: *Nicholas Tierney*

Department of Econometrics and Business Statistics

nicholas.tierney@monash.edu

May 2020

Recap

- Models as functions
- Linear models

Overview

- Feedback from the tests
- What is R^2
  - (pull examples from exercise)
- short exercise in class to calculate correlation and r2 and answer questions
- augment?
- understanding residuals
- components of variation?

Other Admin

Project deadline (Next Week)

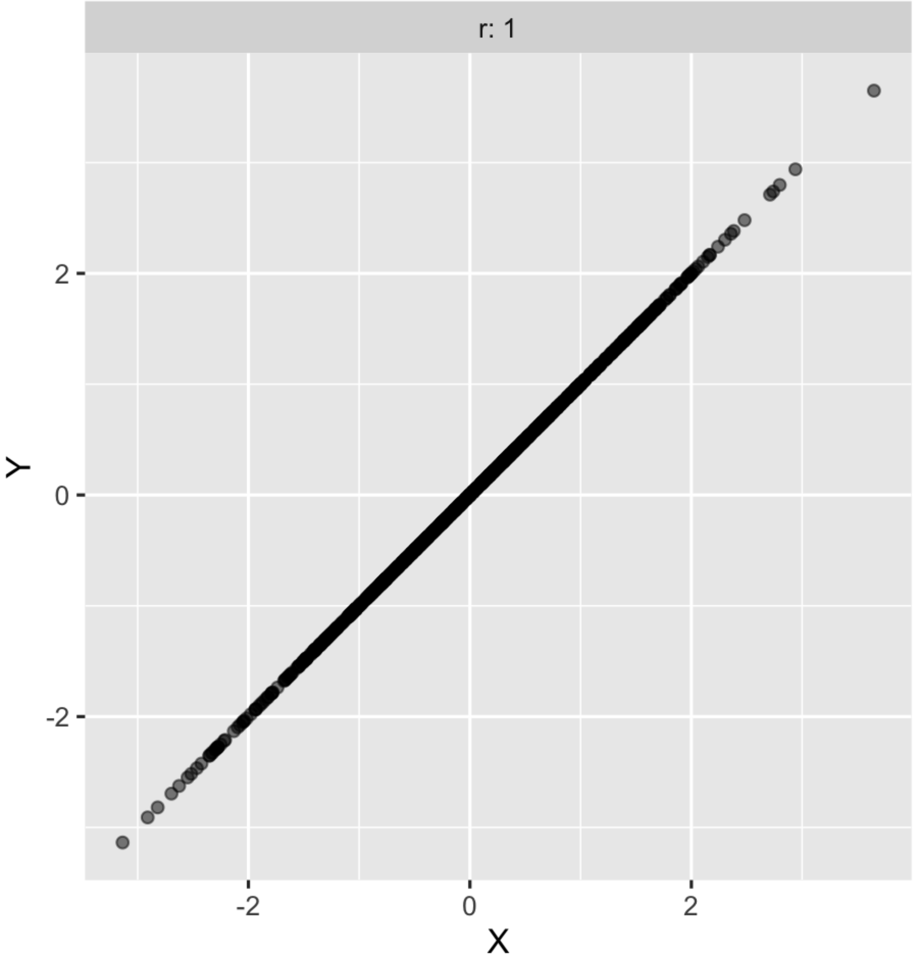
Find team members, and potential topics to study (ed quiz will be posted soon)

What is correlation?

- Linear association between two variables can be described by correlation
- Ranges from -1 to +1

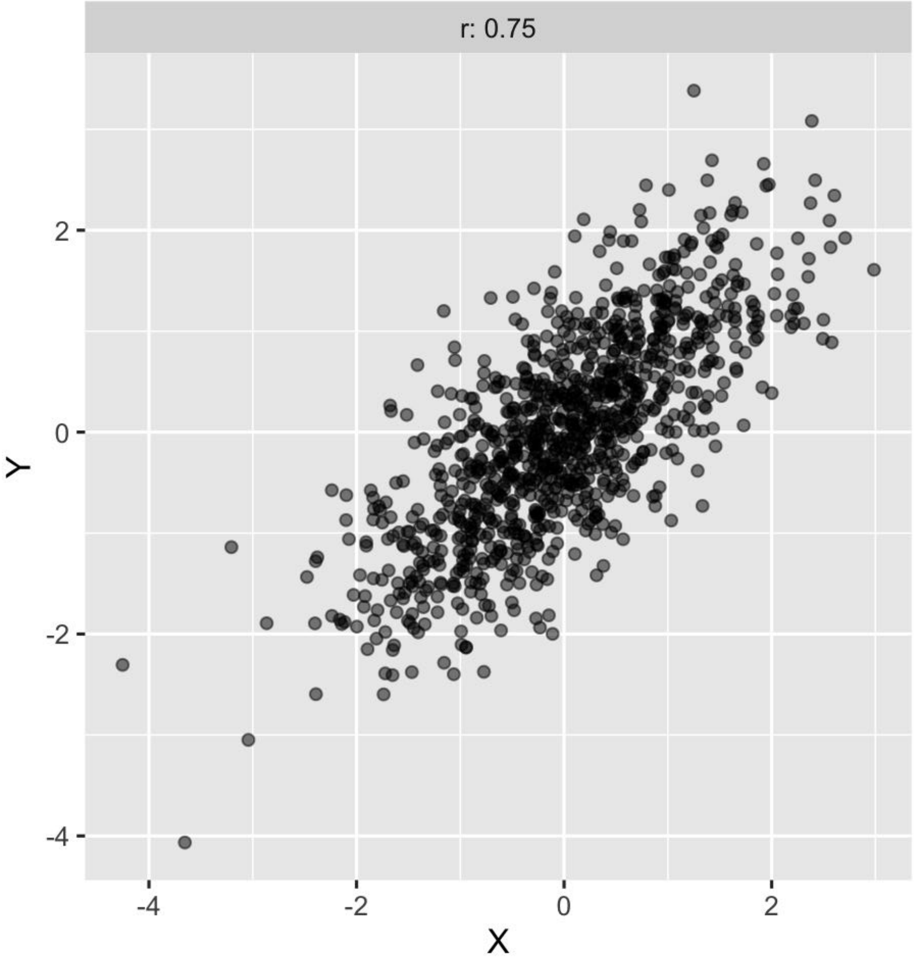
Strong Positive correlation

As one variable increases, so does another

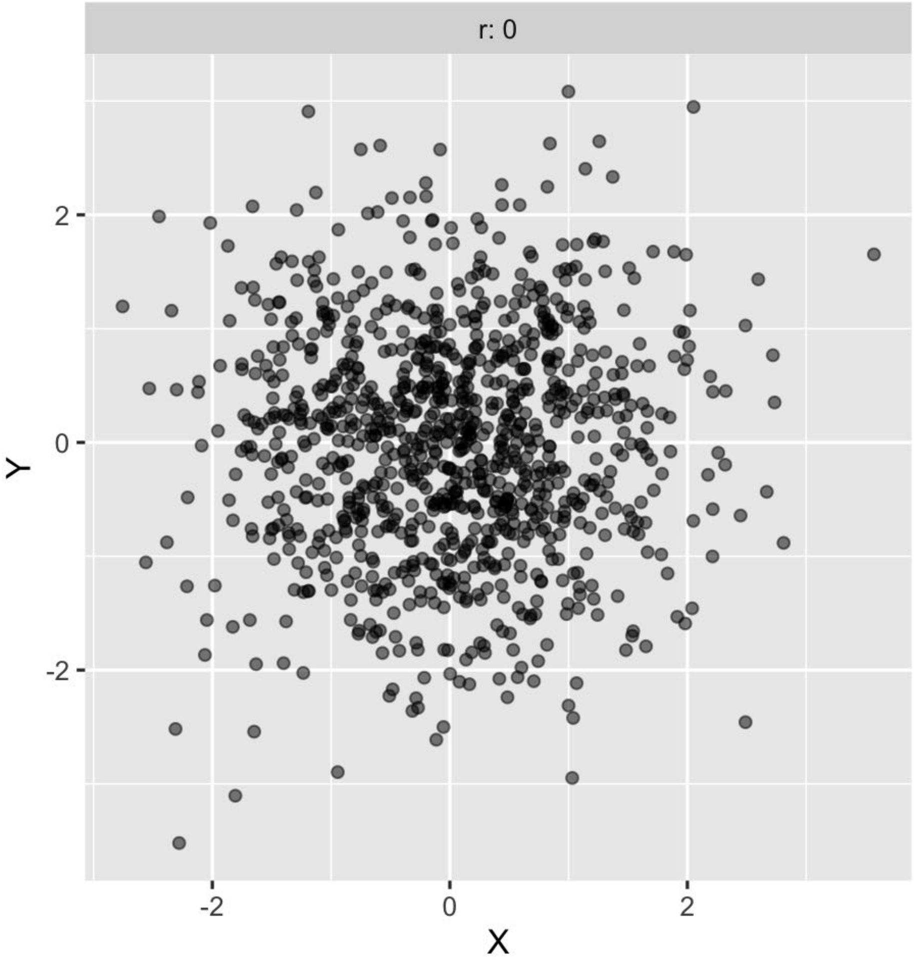


Strong Positive correlation

As one variable increases, so does another variable

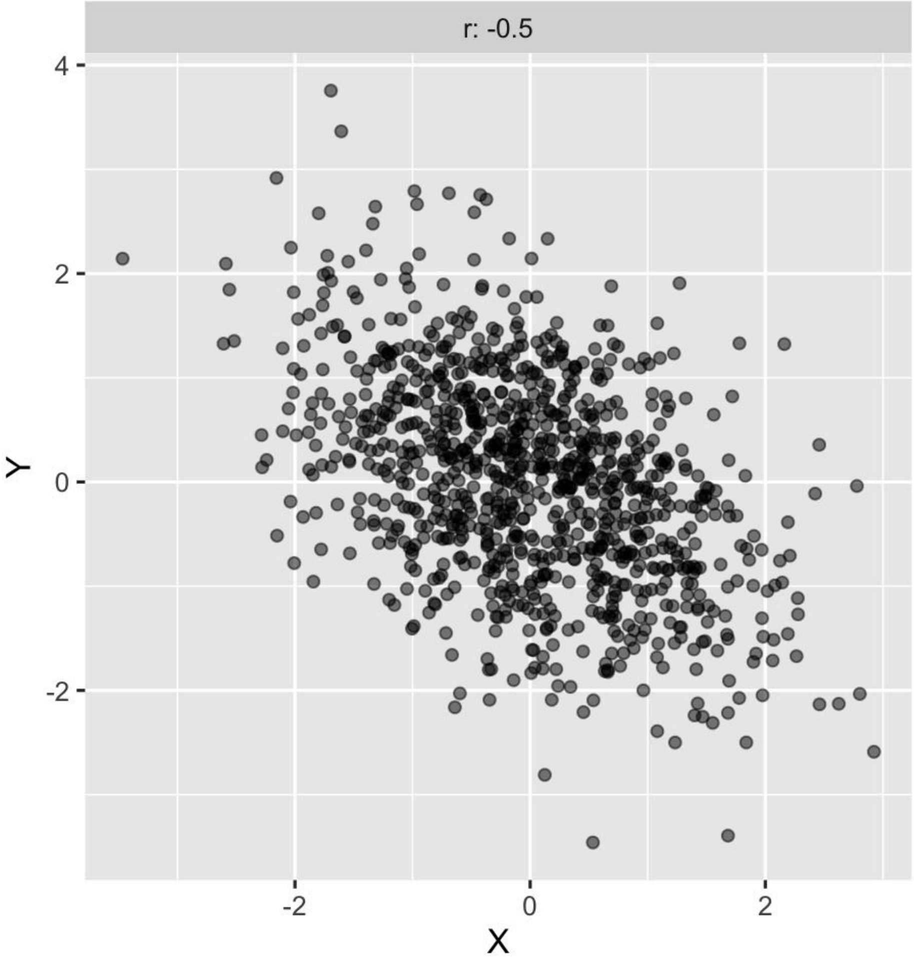


Zero correlation: neither variables are related



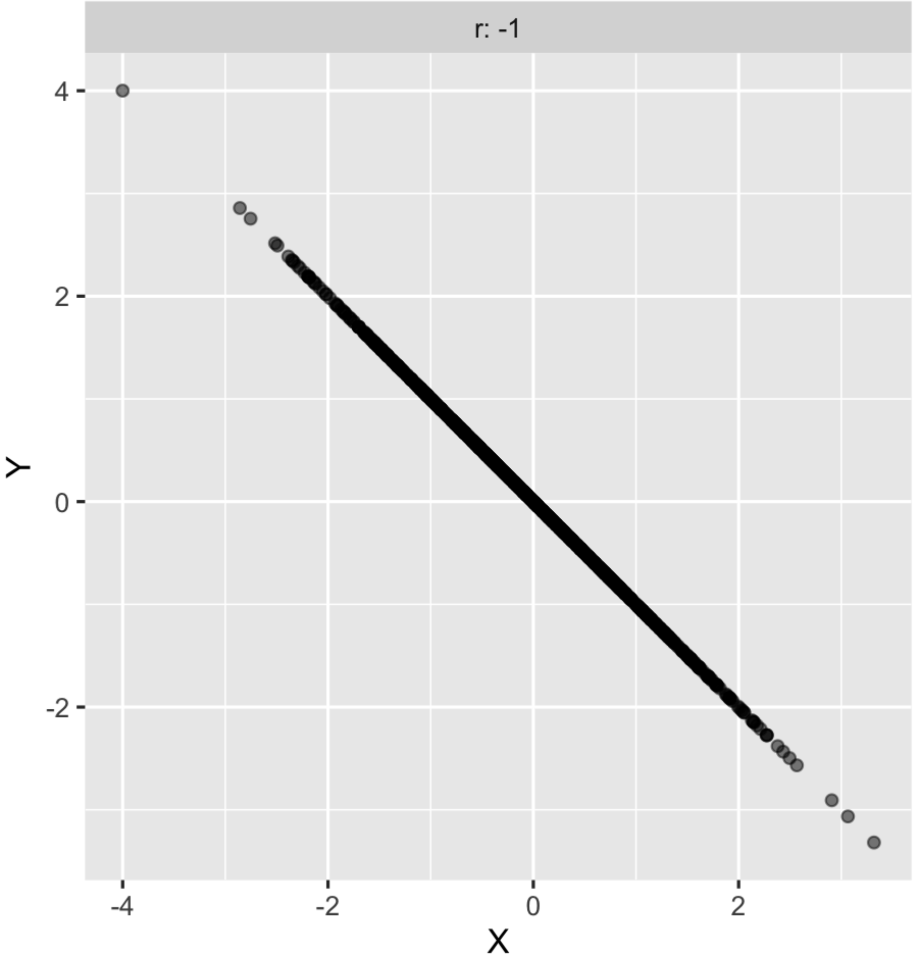
Strong negative correlation

As one variable increases, another decreases

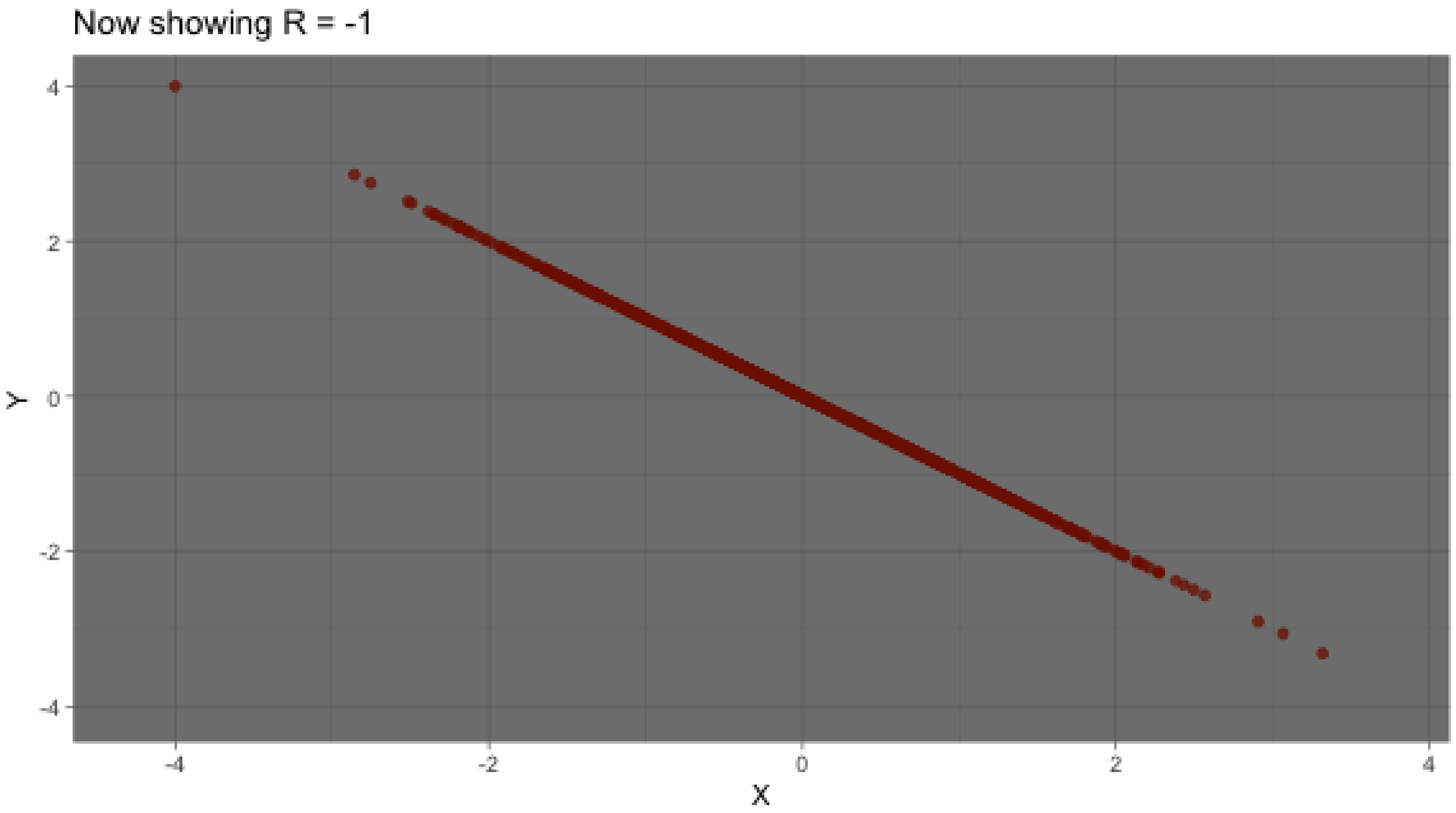


STRONG negative correlation

As one variable increases, another decreases



Correlation: The animation

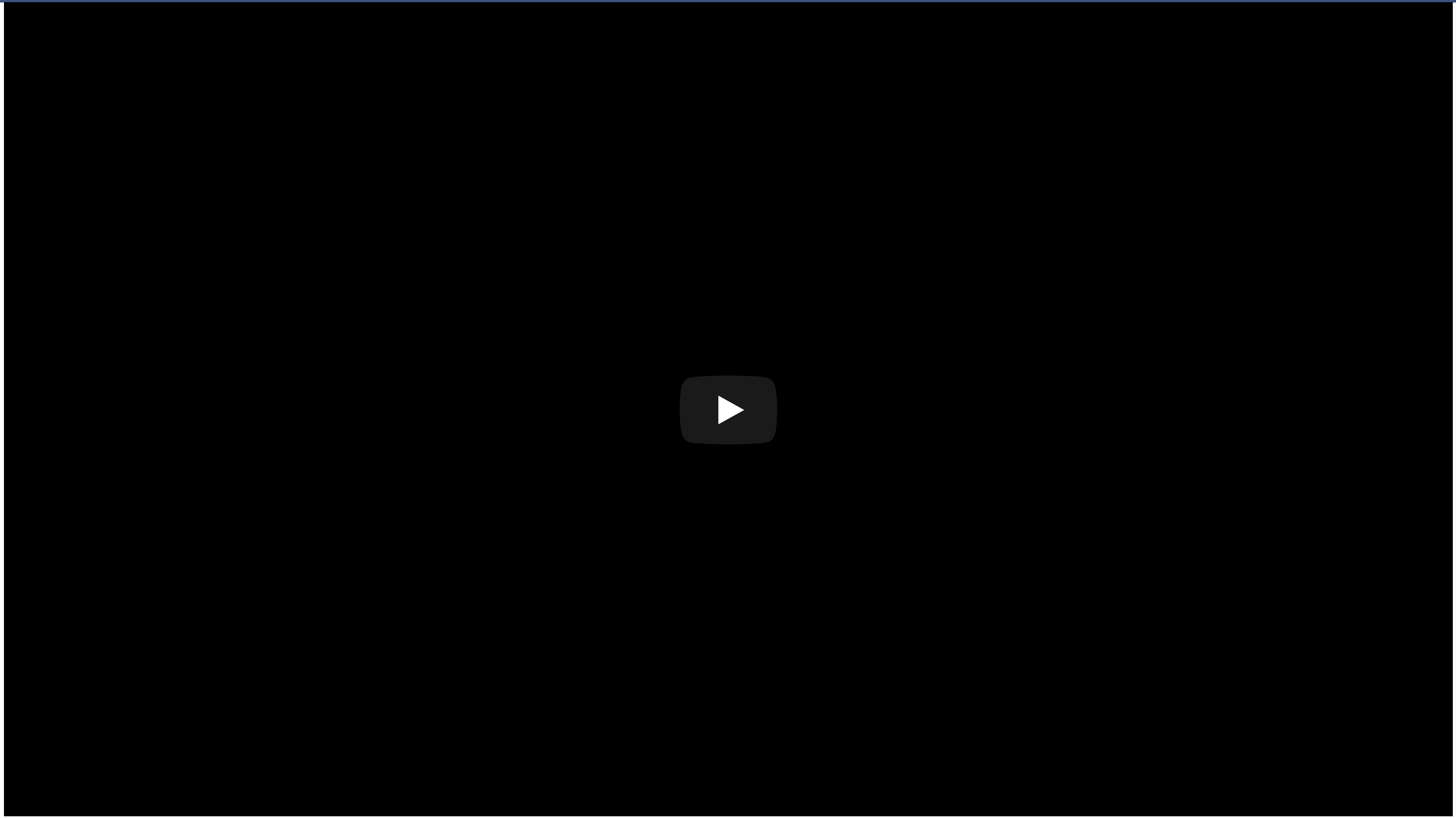


definition of correlation

For two variables X, Y, correlation is:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} = \frac{cov(X, Y)}{s_x s_y}$$

Dance of correlation



Remember! Correlation does not equal causation

What is R^2?

- (model variance)/(total variance), the amount of variance in response explained by the model.
- Always ranges between 0 and 1, with 1 indicating a perfect fit.
- Adding more variables to the model will always increase R^2, so what is important is how big an increase is gained. - Adjusted R^2 reduces this for every additional variable added.

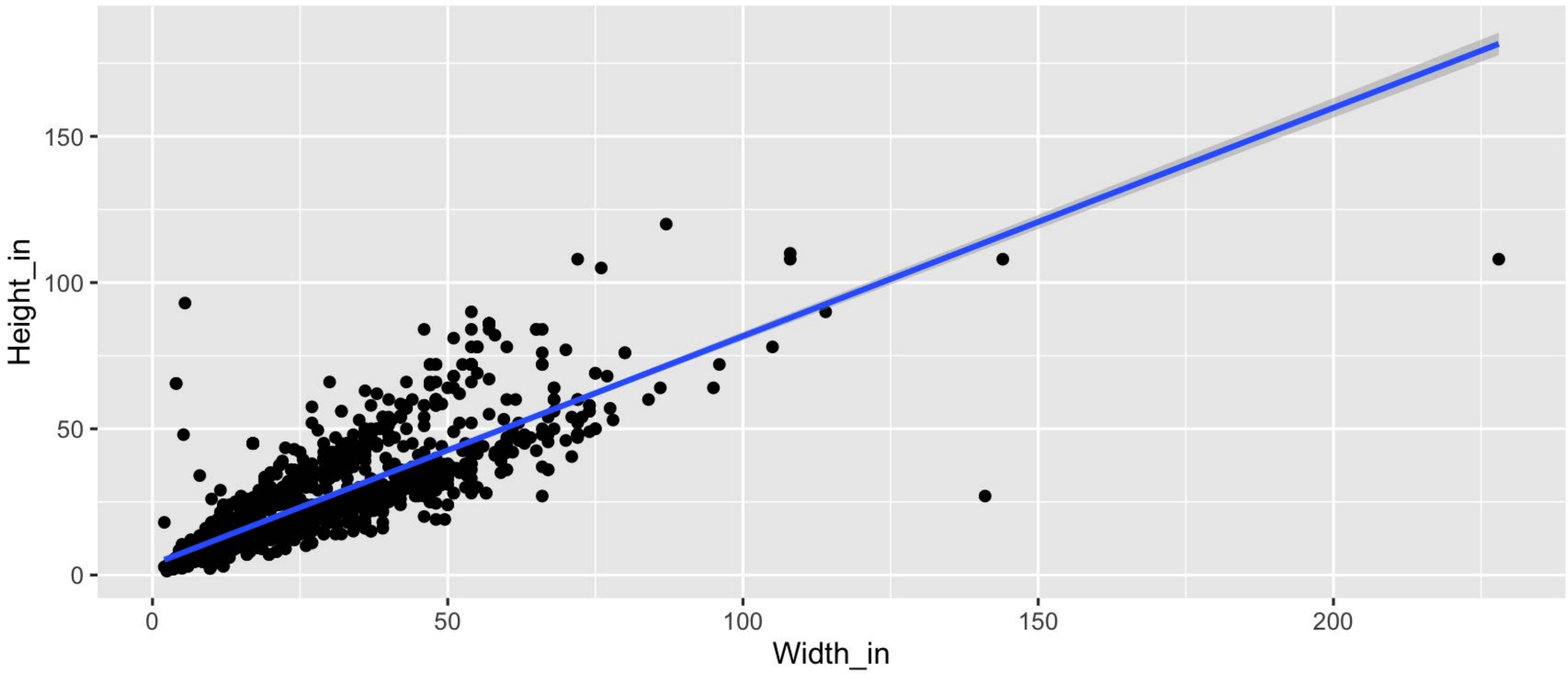
unpacking lm and model objects

```
pp <- read_csv("data/paris-paintings.csv", na = c("n/a", "", "NA"))
pp
```

```
## # A tibble: 3,393 x 61
##    name  sale  lot   position dealer  year origin_author origin_cat school_pntg
##    <chr> <chr> <chr>    <dbl> <chr>  <dbl> <chr>         <chr>      <chr>
##  1 L176… L1764 2       0.0328 L       1764 F             O          F
##  2 L176… L1764 3       0.0492 L       1764 I             O          I
##  3 L176… L1764 4       0.0656 L       1764 X             O          D/FL
##  4 L176… L1764 5       0.0820 L       1764 F             O          F
##  5 L176… L1764 5       0.0820 L       1764 F             O          F
##  6 L176… L1764 6       0.0984 L       1764 X             O          I
##  7 L176… L1764 7       0.115  L       1764 F             O          F
##  8 L176… L1764 7       0.115  L       1764 F             O          F
##  9 L176… L1764 8       0.131  L       1764 X             O          I
## 10 L176… L1764 9       0.148  L       1764 D/FL          O          D/FL
## # … with 3,383 more rows, and 52 more variables: diff_origin <dbl>, logprice <dbl>,
## #   price <dbl>, count <dbl>, subject <chr>, authorstandard <chr>, artistliving <db
## #   authorstyle <chr>, author <chr>, winningbidder <chr>, winningbiddertype <chr>,
## #   endbuyer <chr>, Interm <dbl>, type_intermed <chr>, Height_in <dbl>, Width_in <d
```

unpacking linear models

```r
ggplot(data = pp, aes(x = Width_in, y = Height_in)) +
  geom_point() +
  geom_smooth(method = "lm") # lm for linear model
```



template for linear model

```
lm(<FORMULA>, <DATA>)
```

```
<FORMULA>
```

```
RESPONSE ~ EXPLANATORY VARIABLES
```

Fitting a linear model

```r
m_ht_wt <- lm(Height_in ~ Width_in, data = pp)

m_ht_wt

##
## Call:
## lm(formula = Height_in ~ Width_in, data = pp)
##
## Coefficients:
## (Intercept)     Width_in
##      3.6214       0.7808
```

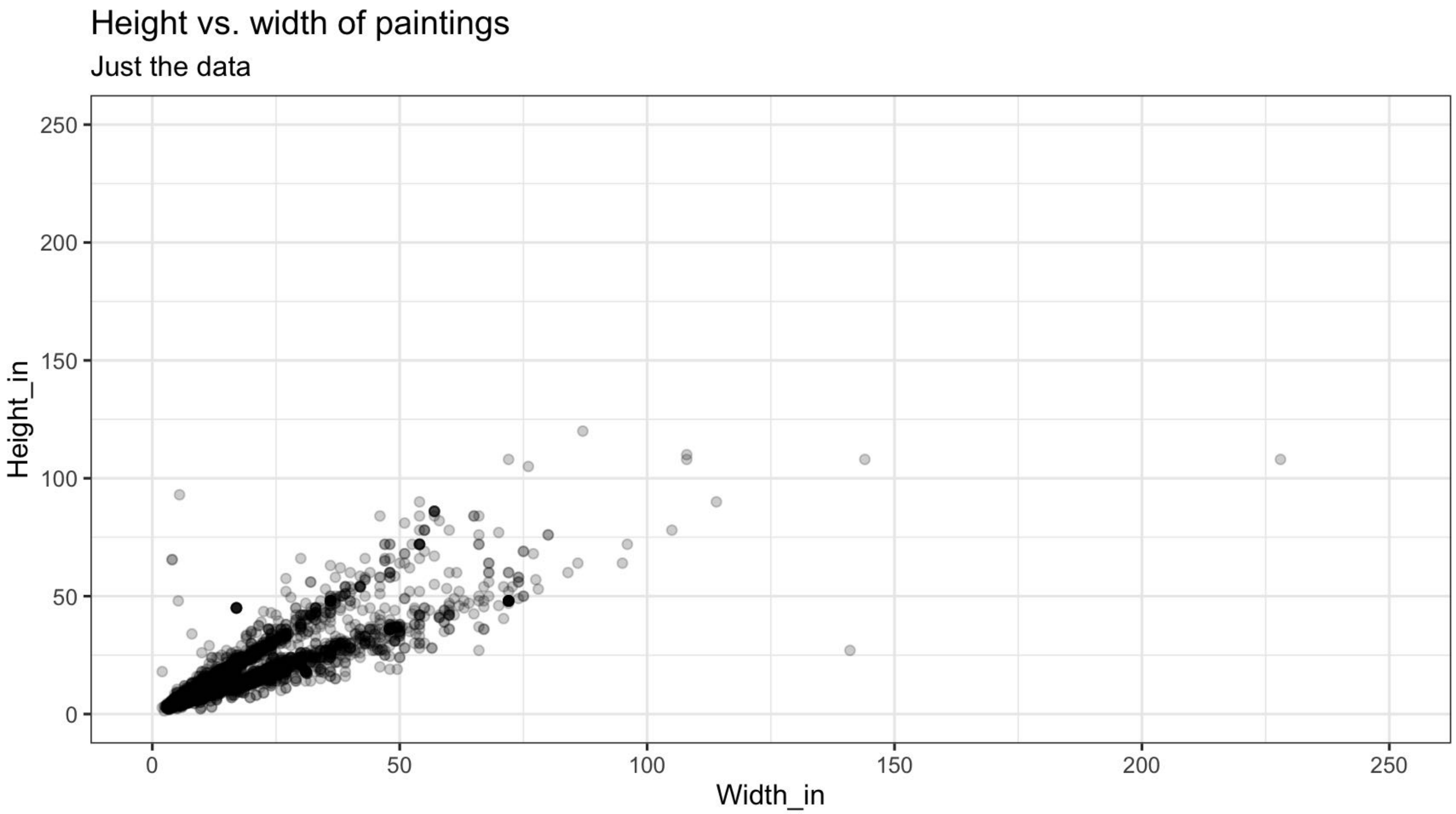
using tidy, augment, glance

tidy: return a tidy table of model information

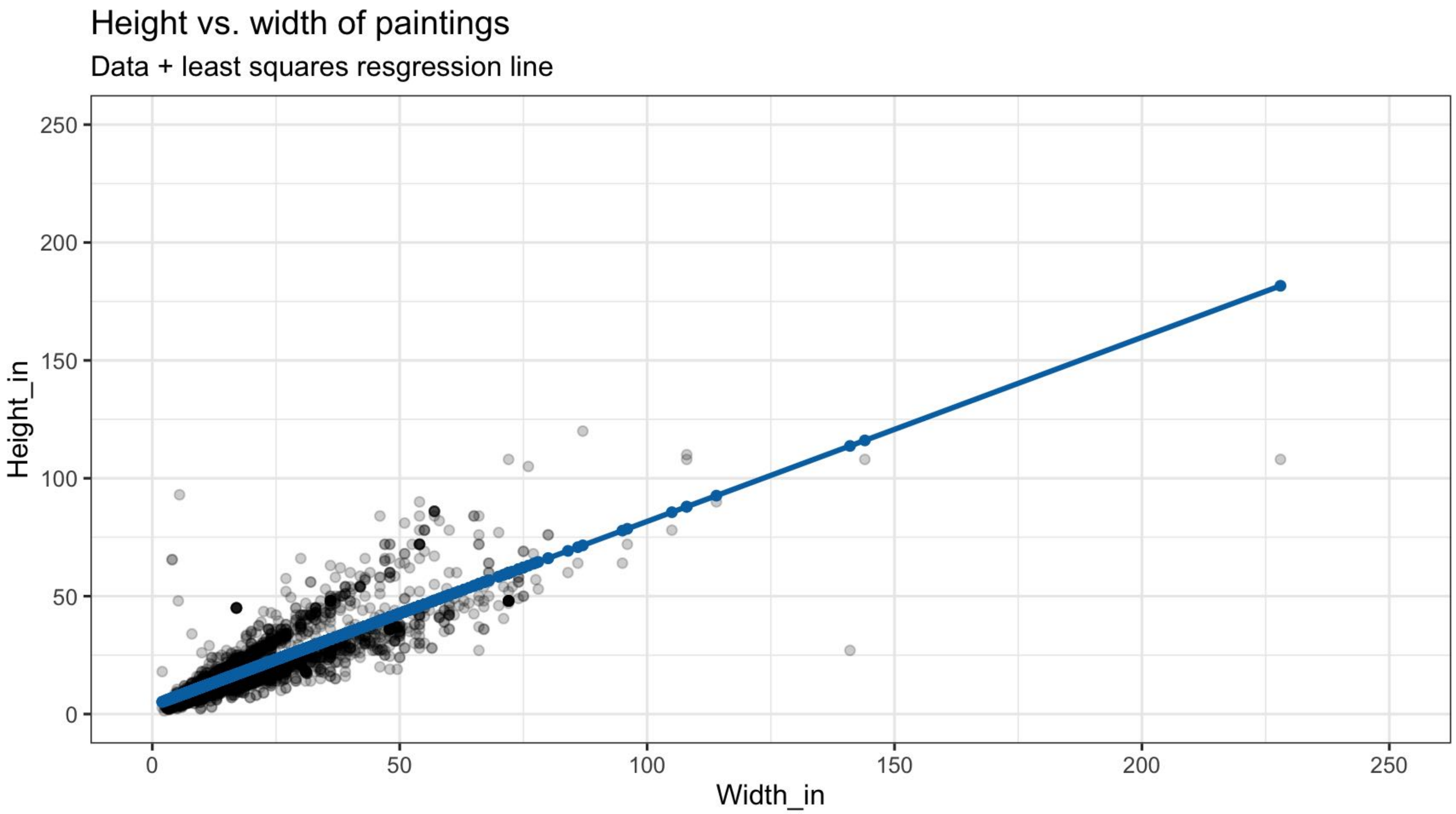
tidy(<MODEL OBJECT>)

```
tidy(m_ht_wt)

## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)    3.62    0.254        14.3 8.82e-45
## 2 Width_in       0.781   0.00950      82.1 0.
```

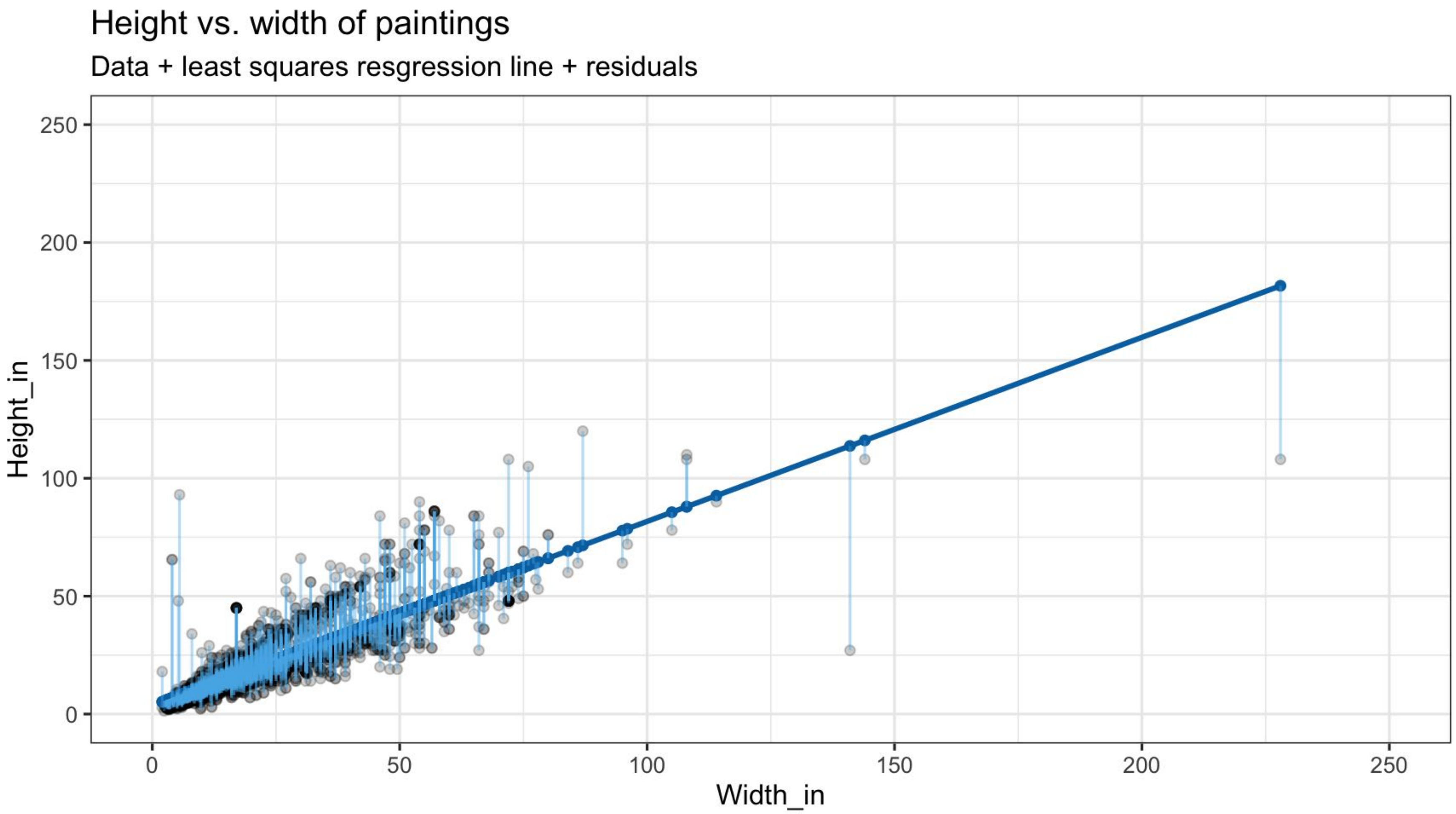
Visualizing residuals



Visualizing residuals (cont.)



Visualizing residuals (cont.)



glance: get a one-row summary out

glance(<MODEL OBJECT>)

```
glance(m_ht_wt)

## # A tibble: 1 x 11
##   r.squared adj.r.squared sigma statistic p.value    df  logLik    AIC    BIC devia
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <int>   <dbl>  <dbl>  <dbl>    <d
## 1     0.683         0.683  8.30     6749.       0     2 -11083. 22173. 22191.  2160
## # … with 1 more variable: df.residual <int>
```

AIC, BIC, Deviance

- **AIC**, **BIC**, and **Deviance** are evidence to make a decision
- Deviance is the residual variation, how much variation in response that IS NOT explained by the model. The close to 0 the better, but it is not on a standard scale. In comparing two models if one has substantially lower deviance, then it is a better model.
- Similarly BIC (Bayes Information Criterion) indicates how well the model fits, best used to compare two models. Lower is better.

augment: get the data

```
augment<MODEL>
```

or

```
augment(<MODEL>, <DATA>)
```

augment

```
augment(m_ht_wt)

## # A tibble: 3,135 x 10
##    .rownames Height_in Width_in .fitted .se.fit  .resid     .hat .sigma .cooksd .st
##    <chr>         <dbl>    <dbl>   <dbl>   <dbl>   <dbl>    <dbl>  <dbl>   <dbl>
##  1 1                37     29.5    26.7   0.166 10.3    0.000399   8.30 3.10e-4
##  2 2                18     14      14.6   0.165  3.45   0.000396   8.31 3.42e-5
##  3 3                13     16      16.1   0.158 -3.11   0.000361   8.31 2.54e-5
##  4 4                14     18      17.7   0.152 -3.68   0.000337   8.31 3.30e-5
##  5 5                14     18      17.7   0.152 -3.68   0.000337   8.31 3.30e-5
##  6 6                 7     10      11.4   0.185 -4.43   0.000498   8.31 7.09e-5
##  7 7                 6     13      13.8   0.170 -7.77   0.000418   8.30 1.83e-4
##  8 8                 6     13      13.8   0.170 -7.77   0.000418   8.30 1.83e-4
##  9 9                15     15      15.3   0.161 -0.333  0.000377   8.31 3.04e-7
## 10 10                9      7       9.09  0.204 -0.0870 0.000601   8.31 3.30e-8
## # … with 3,125 more rows
```

understanding residuals

- variation explained by the model
- residual variation: what's left over after fitting the model

Your turn: go to rstudio cloud and get started on exercise 7B

Going beyond a single model

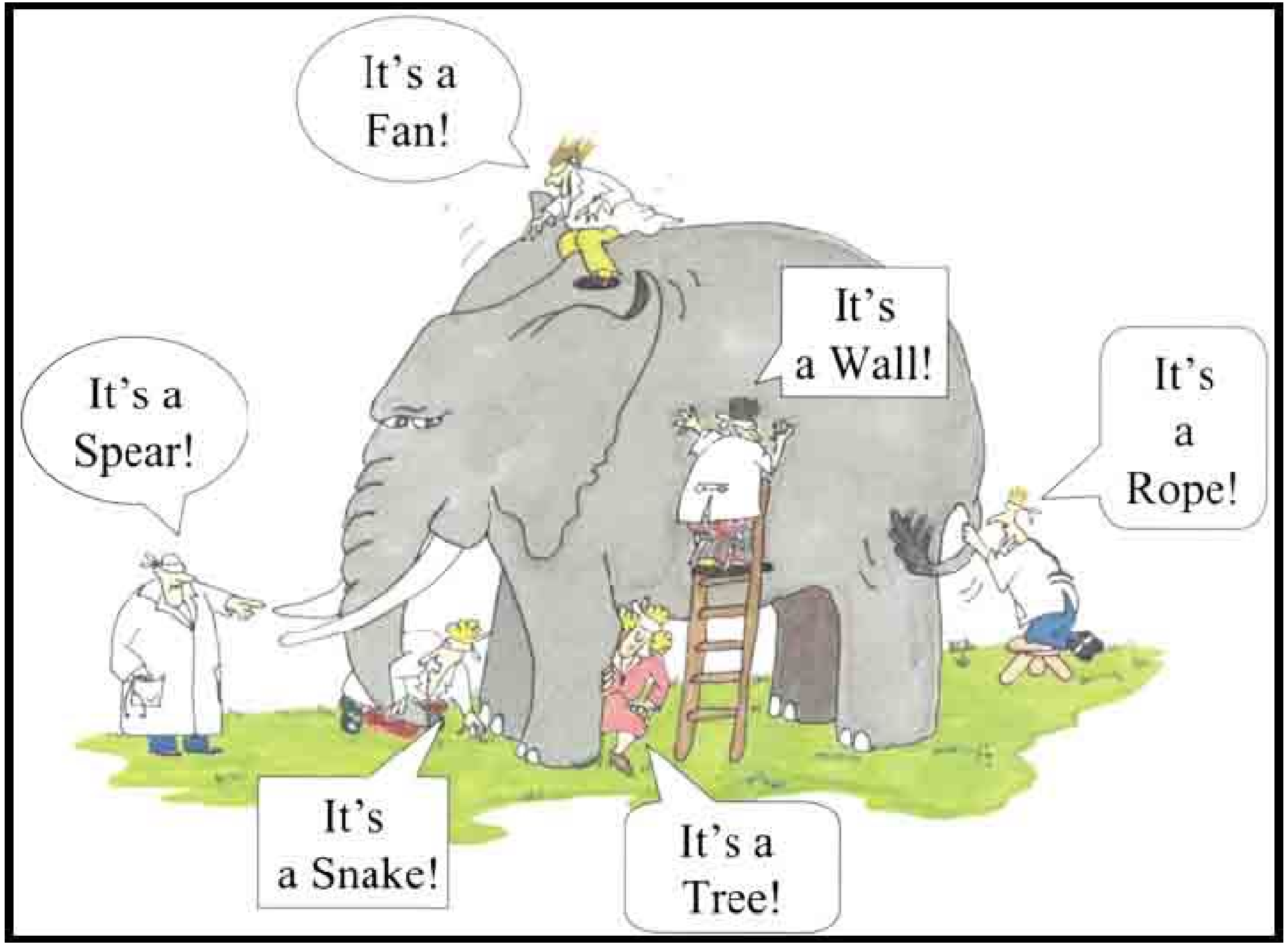


Image source: https://balajiviswanathan.quora.com/Lessons-from-the-Blind-men-and-the-elephant

Going beyond a single model

- Beyond a single model
- Fitting many models

Gapminder

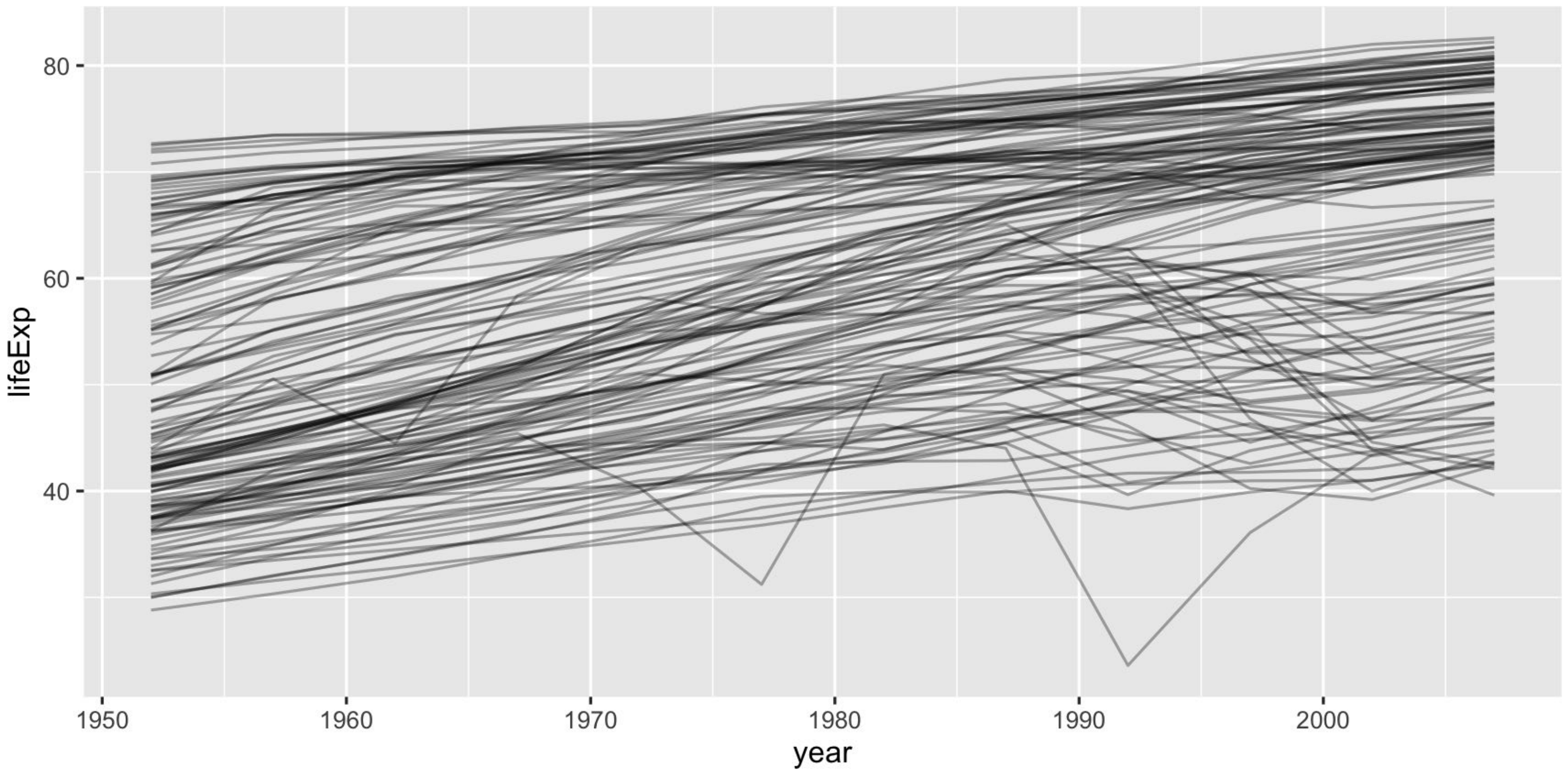
- Hans Rosling was a Swedish doctor, academic and statistician, Professor of International Health at Karolinska Institute. Sadley he passed away in 2017.
- He developed a keen interest in health and wealth across the globe, and the relationship with other factors like agriculture, education, energy.
- You can play with the gapminder data using animations at https://www.gapminder.org/tools/.

R package: gapminder

Contains subset of the data on five year intervals from 1952 to 2007.

```r
library(gapminder)
glimpse(gapminder)

## Observations: 1,704
## Variables: 6
## $ country   <fct> Afghanistan, Afghanistan, Afghanistan, Afghanistan, Afghanistan,
## $ continent <fct> Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia, Asia,
## $ year      <int> 1952, 1957, 1962, 1967, 1972, 1977, 1982, 1987, 1992, 1997, 2002,
## $ lifeExp   <dbl> 28.801, 30.332, 31.997, 34.020, 36.088, 38.438, 39.854, 40.822, 4
## $ pop       <int> 8425333, 9240934, 10267083, 11537966, 13079460, 14880372, 1288181
## $ gdpPercap <dbl> 779.4453, 820.8530, 853.1007, 836.1971, 739.9811, 786.1134, 978.0
```

"Change in life expectancy in countries over time?"



"Change in life expectancy in countries over time?"

- There generally appears to be an increase in life expectancy
- A number of countries have big dips from the 70s through 90s
- a cluster of countries starts off with low life expectancy but ends up close to the highest by the end of the period.

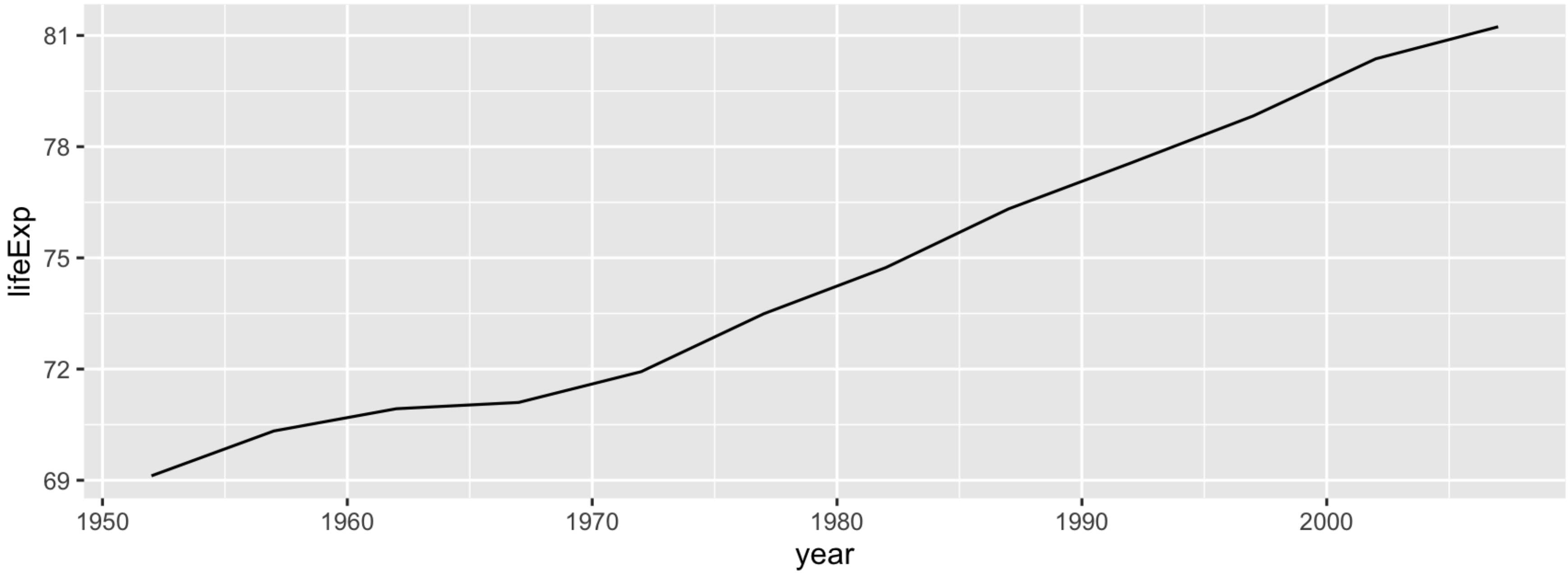
Gapminder: Australia

Australia was already had one of the top life expectancies in the 1950s.

```r
oz <- gapminder %>% filter(country == "Australia")

oz

## # A tibble: 12 x 6
##     country   continent  year lifeExp      pop gdpPercap
##     <fct>     <fct>     <int>   <dbl>    <int>     <dbl>
##   1 Australia Oceania    1952    69.1  8691212    10040.
##   2 Australia Oceania    1957    70.3  9712569    10950.
##   3 Australia Oceania    1962    70.9 10794968    12217.
##   4 Australia Oceania    1967    71.1 11872264    14526.
##   5 Australia Oceania    1972    71.9 13177000    16789.
##   6 Australia Oceania    1977    73.5 14074100    18334.
##   7 Australia Oceania    1982    74.7 15184200    19477.
##   8 Australia Oceania    1987    76.3 16257249    21889.
##   9 Australia Oceania    1992    77.6 17481977    23425.
##  10 Australia Oceania    1997    78.8 18565243    26998.
```

Gapminder: Australia

```
ggplot(data = oz,
       aes(x = year,
           y = lifeExp)) +
  geom_line()
```



Gapminder: Australia

```r
oz_lm <- lm(lifeExp ~ year, data = oz)

oz_lm

##
## Call:
## lm(formula = lifeExp ~ year, data = oz)
##
## Coefficients:
## (Intercept)         year
##   -376.1163       0.2277
```

Tidy Gapminder Australia

```
tidy(oz_lm)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -376.      20.5        -18.3 5.09e- 9
## 2 year           0.228    0.0104      21.9 8.67e-10
```

$$\widehat{lifeExp} = -376.1163 - 0.2277\ year$$

Center year

- Let us treat 1950 is the first year
- so for model fitting we are going to shift year to begin in 1950
- This improved interpretability.

```
gap <- gapminder %>% mutate(year1950 = year - 1950)
oz <- gap %>%  filter(country == "Australia")
```

Model for centered year

```r
oz_lm <- lm(lifeExp ~ year1950, data = oz)

oz_lm

##
## Call:
## lm(formula = lifeExp ~ year1950, data = oz)
##
## Coefficients:
## (Intercept)     year1950
##     67.9451       0.2277
```

Tidy the model

```
tidy(oz_lm)

## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   67.9      0.355      192.  3.70e-19
## 2 year1950       0.228    0.0104      21.9 8.67e-10
```

$$\widehat{lifeExp} = 67.9 + 0.2277\ year$$

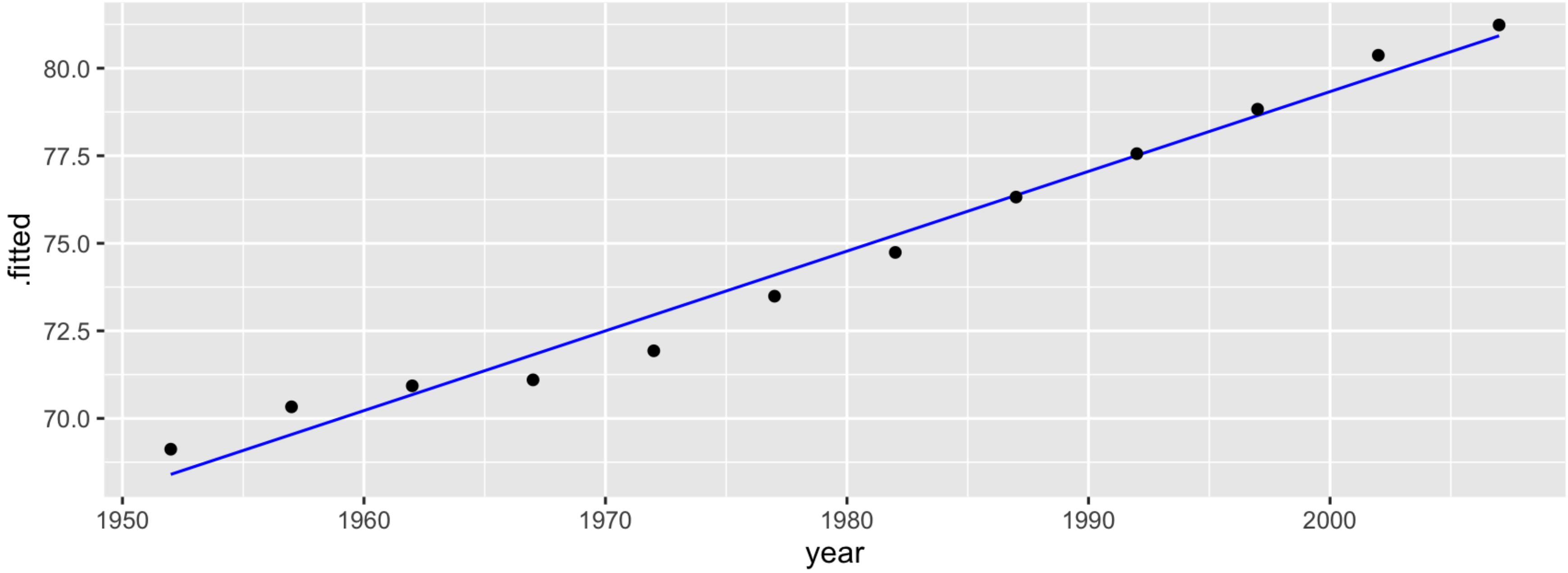
Augment

```
oz_aug <- augment(oz_lm, oz)

oz_aug
```

```
## # A tibble: 12 x 14
##    country continent  year lifeExp    pop gdpPercap year1950 .fitted .se.fit  .resi
##    <fct>   <fct>     <int>   <dbl>  <int>     <dbl>    <dbl>   <dbl>   <dbl>   <dbl
##  1 Austra… Oceania    1952    69.1 8.69e6    10040.        2    68.4   0.337  0.719
##  2 Austra… Oceania    1957    70.3 9.71e6    10950.        7    69.5   0.294  0.791
##  3 Austra… Oceania    1962    70.9 1.08e7    12217.       12    70.7   0.255  0.252
##  4 Austra… Oceania    1967    71.1 1.19e7    14526.       17    71.8   0.221 -0.716
##  5 Austra… Oceania    1972    71.9 1.32e7    16789.       22    73.0   0.195 -1.02
##  6 Austra… Oceania    1977    73.5 1.41e7    18334.       27    74.1   0.181 -0.604
##  7 Austra… Oceania    1982    74.7 1.52e7    19477.       32    75.2   0.181 -0.492
##  8 Austra… Oceania    1987    76.3 1.63e7    21889.       37    76.4   0.195 -0.050
##  9 Austra… Oceania    1992    77.6 1.75e7    23425.       42    77.5   0.221  0.050
## 10 Austra… Oceania    1997    78.8 1.86e7    26998.       47    78.6   0.255  0.182
## 11 Austra… Oceania    2002    80.4 1.95e7    30688.       52    79.8   0.294  0.583
## 12 Austra… Oceania    2007    81.2 2.04e7    34435.       57    80.9   0.337  0.310
## # … with 4 more variables: .hat <dbl>, .sigma <dbl>, .cooksd <dbl>, .std.resid <dbl
```

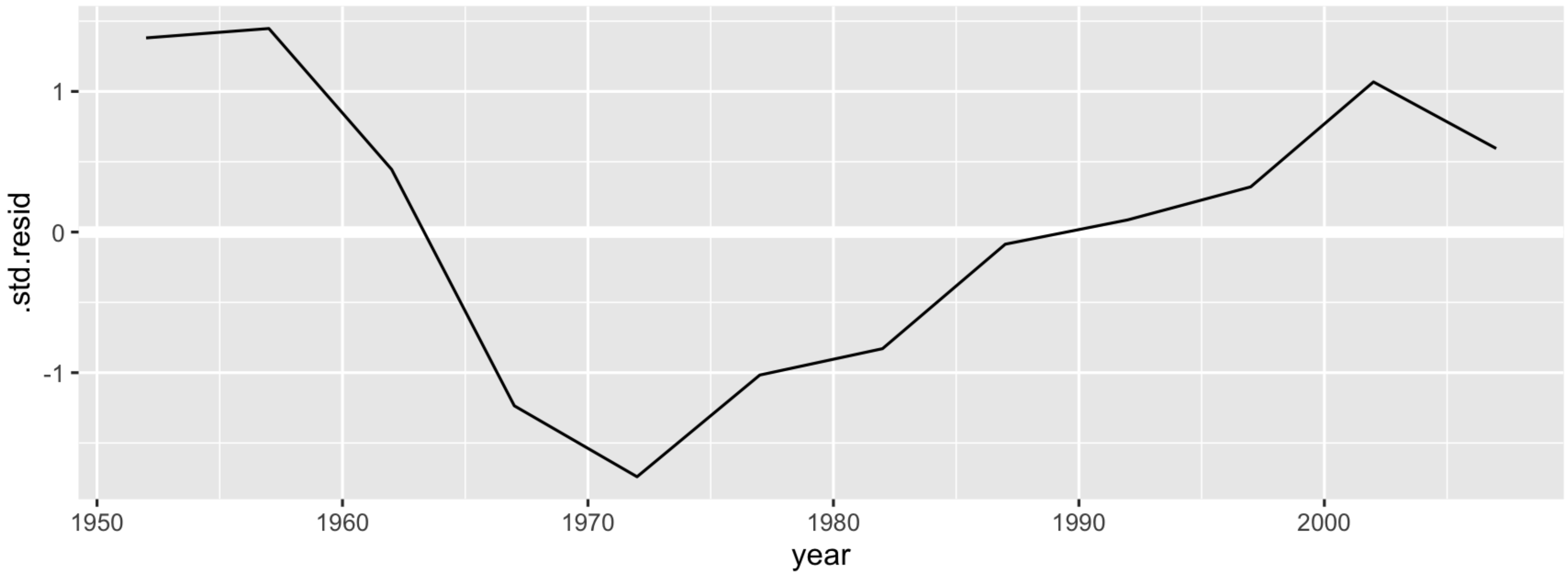
Plot fitted against values

```
ggplot(data = oz_aug,
       aes(x = year,
           y = .fitted)) +
  geom_line(colour = "blue") +
  geom_point(aes(x = year,
                 y = lifeExp))
```



Plot standardised residuals against year

```r
ggplot(data = oz_aug,
             aes(x = year,
                 y = .std.resid)) +
  geom_hline(yintercept = 0,
             colour = "white",
             size = 2) +
  geom_line()
```



Making inferences from this

- Life expectancy has increased 2.3 years every decade, on average.
- There was a slow period from 1960 through to 1972, probably related to mortality during the Vietnam war.

Can we fit for New Zealand?

```
nz <- gap %>%  filter(country == "New Zealand")
nz_lm <- lm(lifeExp ~ year1950, data = nz)
nz_lm

##
## Call:
## lm(formula = lifeExp ~ year1950, data = nz)
##
## Coefficients:
## (Intercept)     year1950
##     68.3013       0.1928
```

Can we fit for Japan?

```
japan <- gap %>%  filter(country == "Japan")
japan_lm <- lm(lifeExp ~ year1950, data = japan)
japan_lm
```

```
##
## Call:
## lm(formula = lifeExp ~ year1950, data = japan)
##
## Coefficients:
## (Intercept)     year1950
##     64.4162       0.3529
```

Can we fit for Italy?

```r
italy <- gap %>%  filter(country == "Italy")
italy_lm <- lm(lifeExp ~ year1950, data = italy)
italy_lm
```

```
## 
## Call:
## lm(formula = lifeExp ~ year1950, data = italy)
## 
## Coefficients:
## (Intercept)     year1950  
##     66.0574       0.2697
```

Is there a better way?

Like, what if we wanted to fit a model for ALL countries?

Let's tinker with the data.

nest() country level data (one row = one country)

```r
by_country <- gap %>%
  select(country, year1950, lifeExp, continent) %>%
  group_by(country, continent) %>%
  nest()

by_country
```

```
## # A tibble: 142 x 3
## # Groups:   country, continent [710]
##    country     continent data
##    <fct>       <fct>     <list>
##  1 Afghanistan Asia      <tibble [12 × 2]>
##  2 Albania     Europe    <tibble [12 × 2]>
##  3 Algeria     Africa    <tibble [12 × 2]>
##  4 Angola      Africa    <tibble [12 × 2]>
##  5 Argentina   Americas  <tibble [12 × 2]>
##  6 Australia   Oceania   <tibble [12 × 2]>
##  7 Austria     Europe    <tibble [12 × 2]>
##  8 Bahrain     Asia      <tibble [12 × 2]>
##  9 Bangladesh  Asia      <tibble [12 × 2]>
```

What is in data?

```
by_country$data[[1]]
```

```
## # A tibble: 12 x 2
##    year1950 lifeExp
##       <dbl>   <dbl>
##  1        2    28.8
##  2        7    30.3
##  3       12    32.0
##  4       17    34.0
##  5       22    36.1
##  6       27    38.4
##  7       32    39.9
##  8       37    40.8
##  9       42    41.7
## 10       47    41.8
## 11       52    42.1
## 12       57    43.8
```

It's a list!

fit a linear model to each one?

```
lm_afganistan <- lm(lifeExp ~ year1950, data = by_country$data[[1]])
lm_albania <- lm(lifeExp ~ year1950, data = by_country$data[[2]])
lm_algeria <- lm(lifeExp ~ year1950, data = by_country$data[[3]])
```

But we are copying and pasting this code **more than twice**...is there a better way?

A case for our friend, map ... ???

```
map(<data object>, <function>)
```

A case for map ???

```r
mapped_lm <- map(.x = by_country$data,
                 .f = function(x){
                   lm(lifeExp ~ year1950, data = x)
                   })

mapped_lm
```

```
## [[1]]
##
## Call:
## lm(formula = lifeExp ~ year1950, data = x)
##
## Coefficients:
## (Intercept)     year1950
##     29.3566       0.2753
##
##
## [[2]]
##
## Call:
```

Map inside the data?

```r
country_model <- by_country %>%
  mutate(model = map(.x = data,
                     .f = function(x){
                       lm(lifeExp ~ year1950, data = x)
    }))

country_model
```

```
## # A tibble: 142 x 4
## # Groups:   country, continent [710]
##    country     continent data              model
##    <fct>       <fct>     <list>            <list>
##  1 Afghanistan Asia      <tibble [12 × 2]> <lm>
##  2 Albania     Europe    <tibble [12 × 2]> <lm>
##  3 Algeria     Africa    <tibble [12 × 2]> <lm>
##  4 Angola      Africa    <tibble [12 × 2]> <lm>
##  5 Argentina   Americas  <tibble [12 × 2]> <lm>
##  6 Australia   Oceania   <tibble [12 × 2]> <lm>
##  7 Austria     Europe    <tibble [12 × 2]> <lm>
##  8 Bahrain     Asia      <tibble [12 × 2]> <lm>
```

A case for map (shorthand function)

```
country_model <- by_country %>%
  mutate(model = map(.x = data,
                     .f = ~lm(lifeExp ~ year1950, data = .)))

country_model

## # A tibble: 142 x 4
## # Groups:   country, continent [710]
##    country     continent data              model
##    <fct>       <fct>     <list>            <list>
##  1 Afghanistan Asia      <tibble [12 × 2]> <lm>
##  2 Albania     Europe    <tibble [12 × 2]> <lm>
##  3 Algeria     Africa    <tibble [12 × 2]> <lm>
##  4 Angola      Africa    <tibble [12 × 2]> <lm>
##  5 Argentina   Americas  <tibble [12 × 2]> <lm>
##  6 Australia   Oceania   <tibble [12 × 2]> <lm>
##  7 Austria     Europe    <tibble [12 × 2]> <lm>
##  8 Bahrain     Asia      <tibble [12 × 2]> <lm>
##  9 Bangladesh  Asia      <tibble [12 × 2]> <lm>
## 10 Belgium     Europe    <tibble [12 × 2]> <lm>
```

Where's the model?

```
country_model$model[[1]]

##
## Call:
## lm(formula = lifeExp ~ year1950, data = .)
##
## Coefficients:
## (Intercept)     year1950
##     29.3566       0.2753
```

We need to summarise this content

```
tidy(country_model$model[[1]])

## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   29.4      0.699       42.0 1.40e-12
## 2 year1950       0.275    0.0205      13.5 9.84e- 8
```

So should we repeat it for each one?

```
tidy(country_model$model[[1]])

## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   29.4      0.699       42.0 1.40e-12
## 2 year1950       0.275    0.0205      13.5 9.84e- 8

tidy(country_model$model[[2]])

## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   58.6      1.13        51.7 1.79e-13
## 2 year1950       0.335    0.0332      10.1 1.46e- 6

tidy(country_model$model[[3]])

## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
```

Use map

```
country_model %>%
  mutate(tidy = map(model, tidy))
```

```
## # A tibble: 142 x 5
## # Groups:   country, continent [710]
##    country     continent data              model  tidy
##    <fct>       <fct>     <list>            <list> <list>
##  1 Afghanistan Asia      <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
##  2 Albania     Europe    <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
##  3 Algeria     Africa    <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
##  4 Angola      Africa    <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
##  5 Argentina   Americas  <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
##  6 Australia   Oceania   <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
##  7 Austria     Europe    <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
##  8 Bahrain     Asia      <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
##  9 Bangladesh  Asia      <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
## 10 Belgium     Europe    <tibble [12 × 2]> <lm>   <tibble [2 × 5]>
## # … with 132 more rows
```

unnest

```
country_coefs <- country_model %>%
  mutate(tidy = map(model, tidy)) %>%
  unnest(tidy) %>%
  select(country, continent, term, estimate)

country_coefs
```

```
## # A tibble: 284 x 4
## # Groups:   country, continent [710]
##    country     continent term        estimate
##    <fct>       <fct>     <chr>          <dbl>
##  1 Afghanistan Asia      (Intercept)   29.4
##  2 Afghanistan Asia      year1950       0.275
##  3 Albania     Europe    (Intercept)   58.6
##  4 Albania     Europe    year1950       0.335
##  5 Algeria     Africa    (Intercept)   42.2
##  6 Algeria     Africa    year1950       0.569
##  7 Angola      Africa    (Intercept)   31.7
##  8 Angola      Africa    year1950       0.209
##  9 Argentina   Americas  (Intercept)   62.2
```

Pivot the term

```
tidy_country_coefs <- country_coefs %>%
  pivot_wider(id_cols = c(term, country, continent),
              names_from =  term,
              values_from = estimate) %>%
  rename(intercept = `(Intercept)`)

tidy_country_coefs
```

```
## # A tibble: 142 x 4
## # Groups:   country, continent [710]
##    country     continent intercept year1950
##    <fct>       <fct>         <dbl>    <dbl>
##  1 Afghanistan Asia           29.4    0.275
##  2 Albania     Europe         58.6    0.335
##  3 Algeria     Africa         42.2    0.569
##  4 Angola      Africa         31.7    0.209
##  5 Argentina   Americas       62.2    0.232
##  6 Australia   Oceania        67.9    0.228
##  7 Austria     Europe         66.0    0.242
##  8 Bahrain     Asia           51.8    0.468
```

Filter to only Australia

```
tidy_country_coefs %>%
  filter(country == "Australia")
```

```
## # A tibble: 1 x 4
## # Groups:   country, continent [710]
##   country   continent intercept year1950
##   <fct>     <fct>         <dbl>    <dbl>
## 1 Australia Oceania        67.9    0.228
```

Your turn: Five minute challenge

- Fit the models to all countries
- Pick your favourite country (not Australia), print the coefficients, and make a hand sketch of the the model fit.

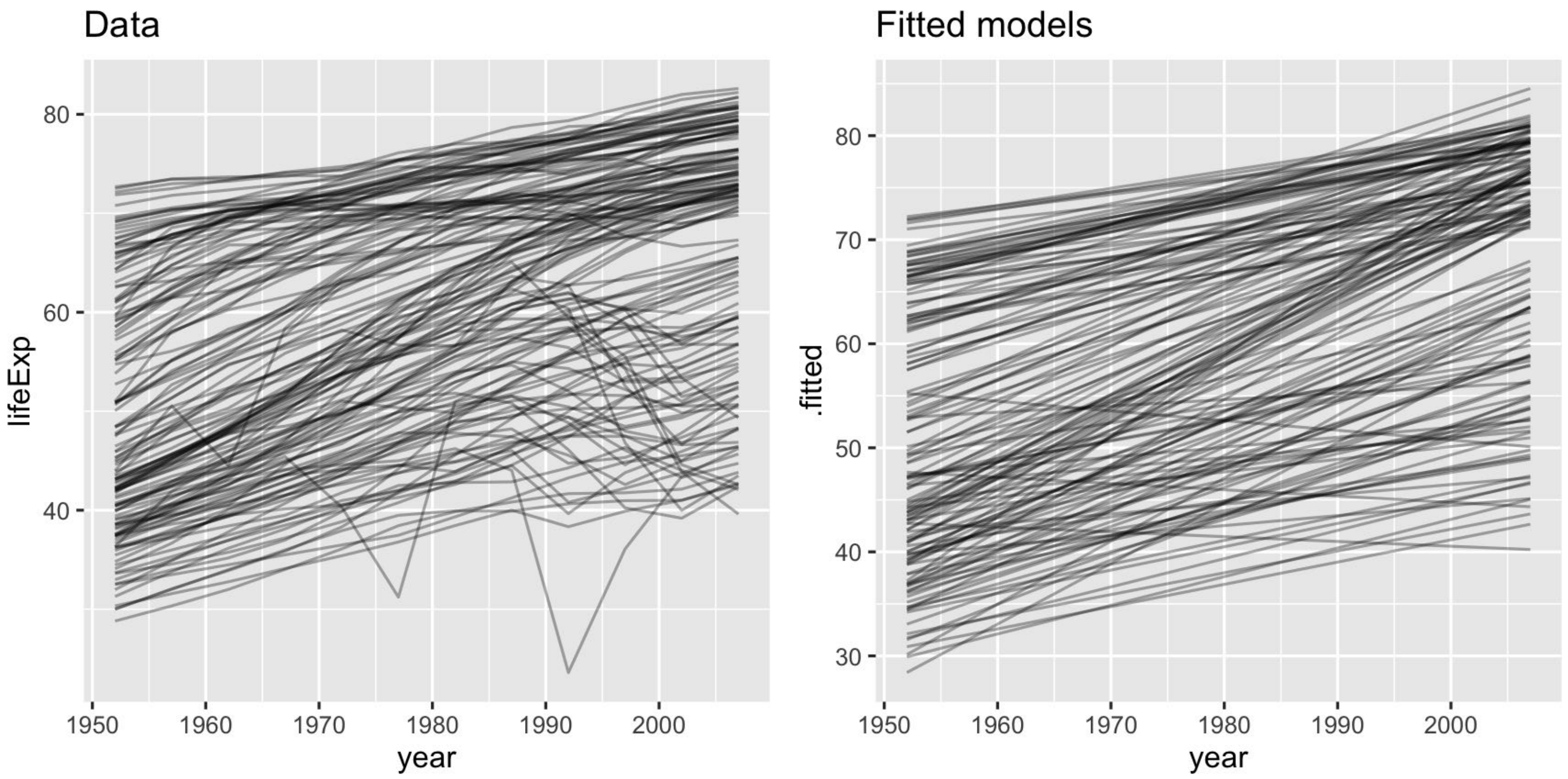
Plot all the models

```
country_aug <- country_model %>%
  mutate(augmented = map(model, augment)) %>%
  unnest(augmented)

country_aug

## # A tibble: 1,704 x 13
## # Groups:   country, continent [710]
##    country continent data   model lifeExp year1950 .fitted .se.fit  .resid   .hat .s
##    <fct>   <fct>     <lis>  <lis>   <dbl>    <dbl>   <dbl>   <dbl>   <dbl>  <dbl>  <
##  1 Afghan… Asia      <tib… <lm>     28.8        2    29.9   0.664 -1.11   0.295
##  2 Afghan… Asia      <tib… <lm>     30.3        7    31.3   0.580 -0.952  0.225
##  3 Afghan… Asia      <tib… <lm>     32.0       12    32.7   0.503 -0.664  0.169
##  4 Afghan… Asia      <tib… <lm>     34.0       17    34.0   0.436 -0.0172 0.127
##  5 Afghan… Asia      <tib… <lm>     36.1       22    35.4   0.385  0.674  0.0991
##  6 Afghan… Asia      <tib… <lm>     38.4       27    36.8   0.357  1.65   0.0851
##  7 Afghan… Asia      <tib… <lm>     39.9       32    38.2   0.357  1.69   0.0851
##  8 Afghan… Asia      <tib… <lm>     40.8       37    39.5   0.385  1.28   0.0991
##  9 Afghan… Asia      <tib… <lm>     41.7       42    40.9   0.436  0.754  0.127
## 10 Afghan… Asia      <tib… <lm>     41.8       47    42.3   0.503 -0.534  0.169
```

Plot all the models

```r
p1 <- gapminder %>%
  ggplot(aes(year, lifeExp, group = country)) +
    geom_line(alpha = 1/3) + labs(title = "Data")

p2 <- ggplot(country_aug) +
  geom_line(aes(x = year1950 + 1950,
                y = .fitted,
                group = country),
            alpha = 1/3) +
  labs(title = "Fitted models",
       x = "year")
```

Plot all the models

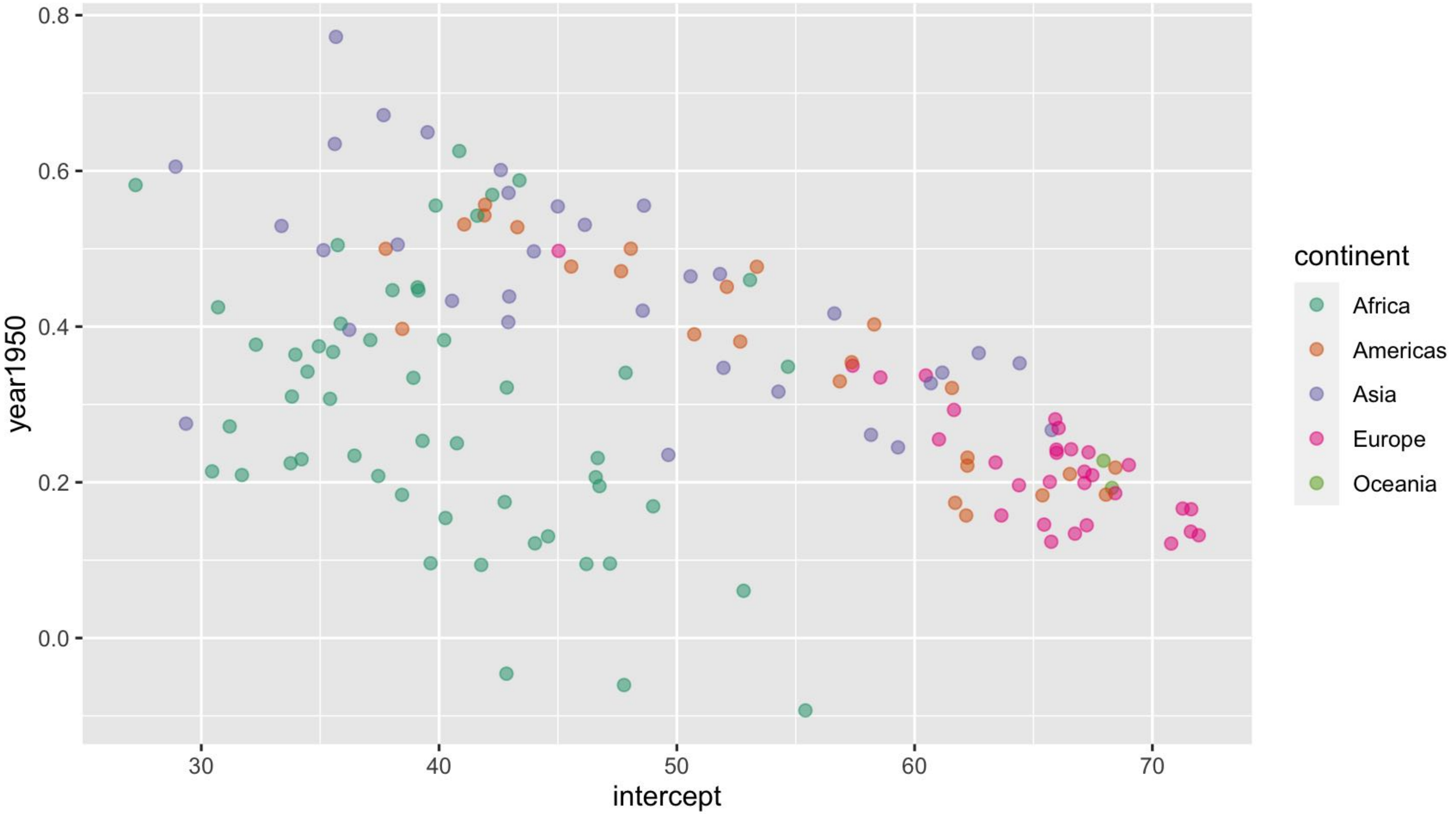


Plot all the model coefficients

```
p <- ggplot(tidy_country_coefs,
            aes(x = intercept,
                y = year1950,
                colour = continent,
                label = country)) +
  geom_point(alpha = 0.5,
             size = 2) +
  scale_color_brewer(palette = "Dark2")
```

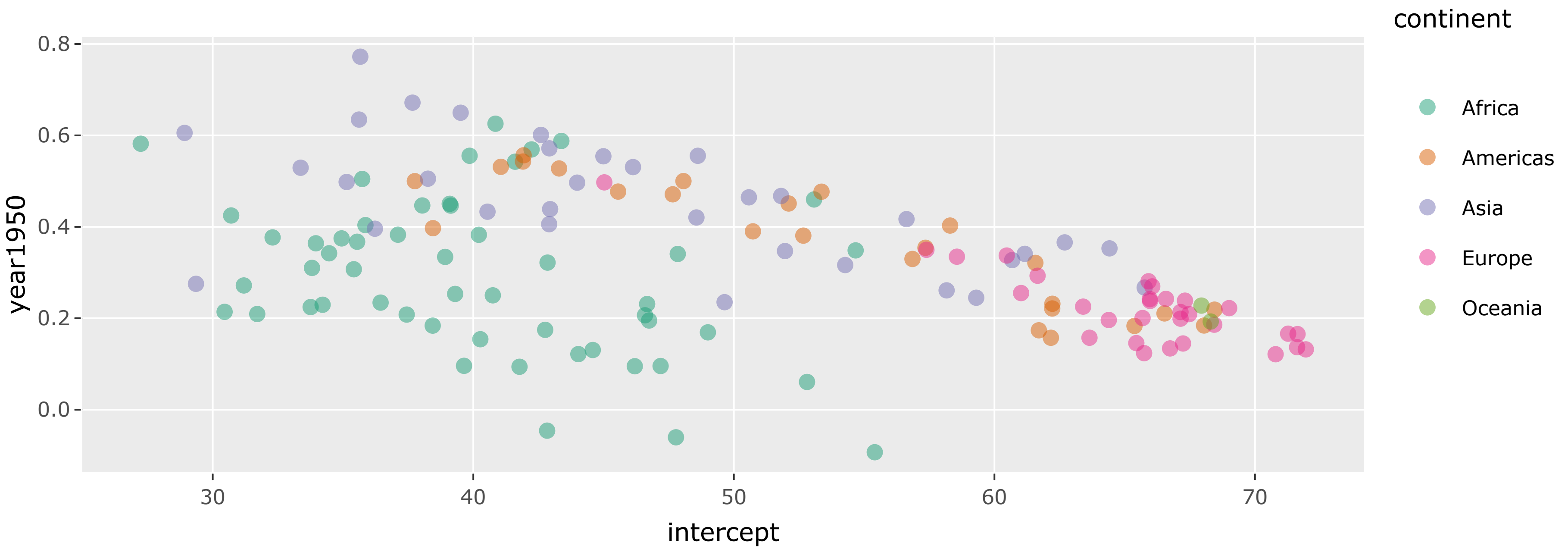
Plot all the model coefficients

```
p
```



Make it interactive!

```r
library(plotly)
ggplotly(p)
```



Let's summarise the information learned from the model coefficients.

- Generally the relationship is negative: this means that if a country started with a high intercept tends to have lower rate of increase.
- There is a difference across the continents: Countries in Europe and Oceania tended to start with higher life expectancy and increased; countries in Asia and America tended to start lower but have high rates of improvement; Africa tends to start lower and have a huge range in rate of change.
- Three countries had negative growth in life expectancy: Rwand, Zimbabwe, Zambia

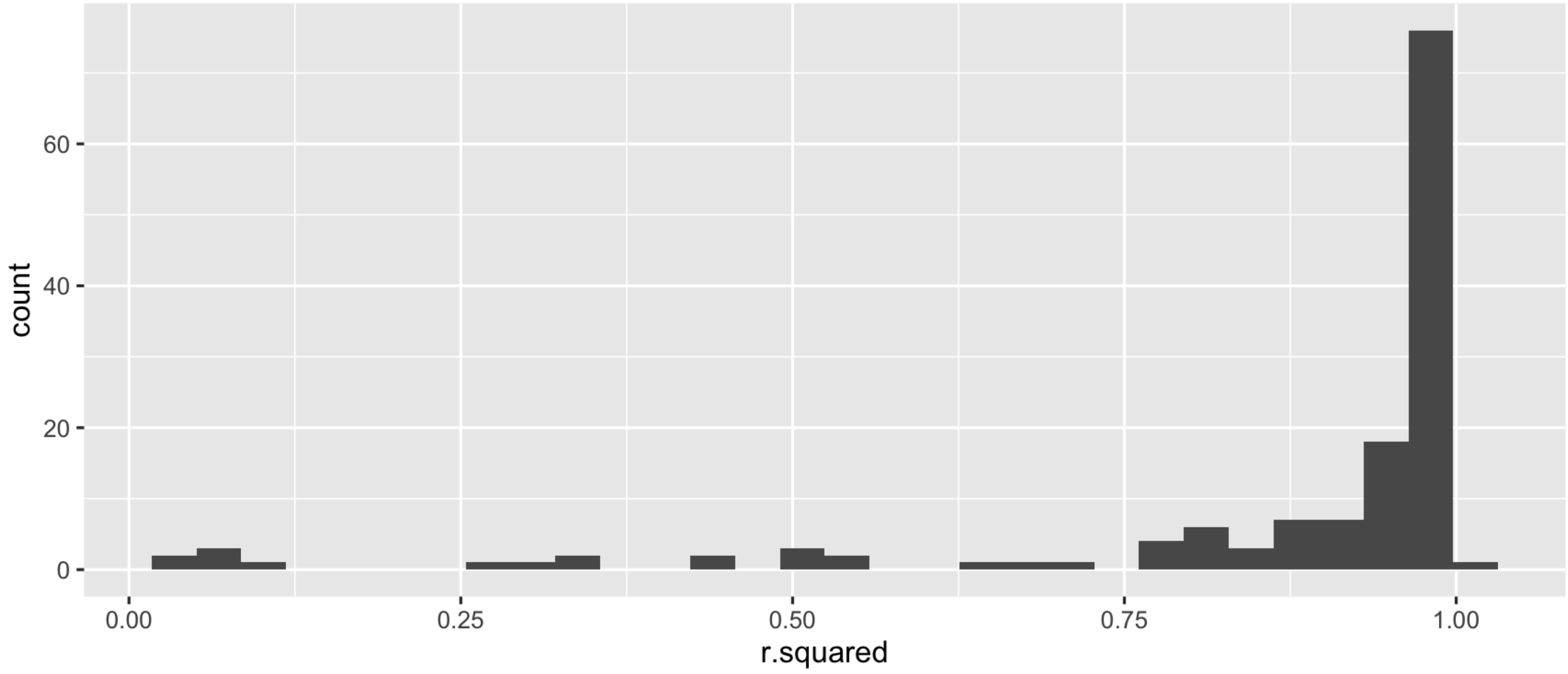
Model diagnostics by country

```
country_glance <- country_model %>%
  mutate(glance = map(model, glance)) %>%
  unnest(glance)

country_glance
```

```
## # A tibble: 142 x 15
## # Groups:   country, continent [710]
##    country continent data  model r.squared adj.r.squared sigma statistic  p.value
##    <fct>   <fct>     <lis> <lis>     <dbl>         <dbl> <dbl>     <dbl>    <dbl> <
##  1 Afghan… Asia      <tib… <lm>      0.948         0.942 1.22      181.   9.84e- 8
##  2 Albania Europe    <tib… <lm>      0.911         0.902 1.98      102.   1.46e- 6
##  3 Algeria Africa    <tib… <lm>      0.985         0.984 1.32      662.   1.81e-10
##  4 Angola  Africa    <tib… <lm>      0.888         0.877 1.41       79.1 4.59e- 6
##  5 Argent… Americas  <tib… <lm>      0.996         0.995 0.292    2246.   4.22e-13
##  6 Austra… Oceania   <tib… <lm>      0.980         0.978 0.621     481.   8.67e-10
##  7 Austria Europe    <tib… <lm>      0.992         0.991 0.407    1261.   7.44e-12
##  8 Bahrain Asia      <tib… <lm>      0.967         0.963 1.64      291.   1.02e- 8
##  9 Bangla… Asia      <tib… <lm>      0.989         0.988 0.977     930.   3.37e-11
## 10 Belgium Europe    <tib… <lm>      0.995         0.994 0.293    1822.   1.20e-12
```

Plot the R^2 values as a histogram.

```
ggplot(country_glance,
       aes(x = r.squared)) +
  geom_histogram()
```



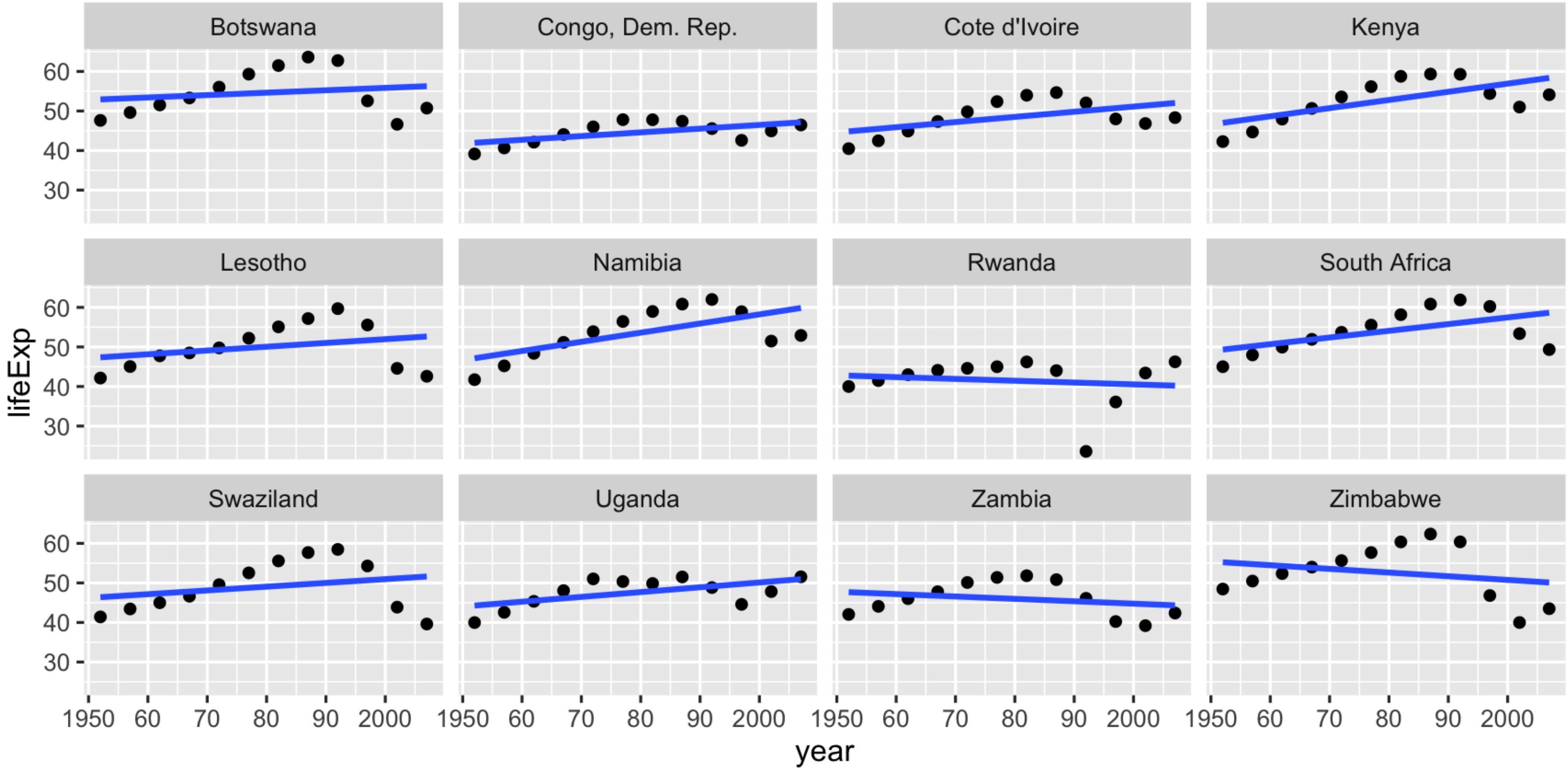
Countries with worst fit

Examine the countries with the worst fit, countries with $R^2 < 0.45$, by making scatterplots of the data, with the linear model overlaid.

```
badfit <- country_glance %>% filter(r.squared <= 0.45)

gap_bad <- gap %>% filter(country %in% badfit$country)

gg_bad_fit <-
ggplot(data = gap_bad,
       aes(x = year,
           y = lifeExp)) +
         geom_point() +
  facet_wrap(~country) +
  scale_x_continuous(breaks = seq(1950,2000,10),
                     labels = c("1950", "60","70", "80","90","2000")) +
  geom_smooth(method = "lm",
              se = FALSE)
```

Countries with worst fit

Each of these countries had been moving on a nice trajectory of increasing life expectancy, and then suffered a big dip during the time period.



Your Turn:

- Use google to explain these dips using world history and current affairs information.
- finish the lab exercise (with new data)
- once you are done, you can collect mid semester exam
- remember the project deadline: **Find team members, and potential topics to study (List of groups will be posted here)**