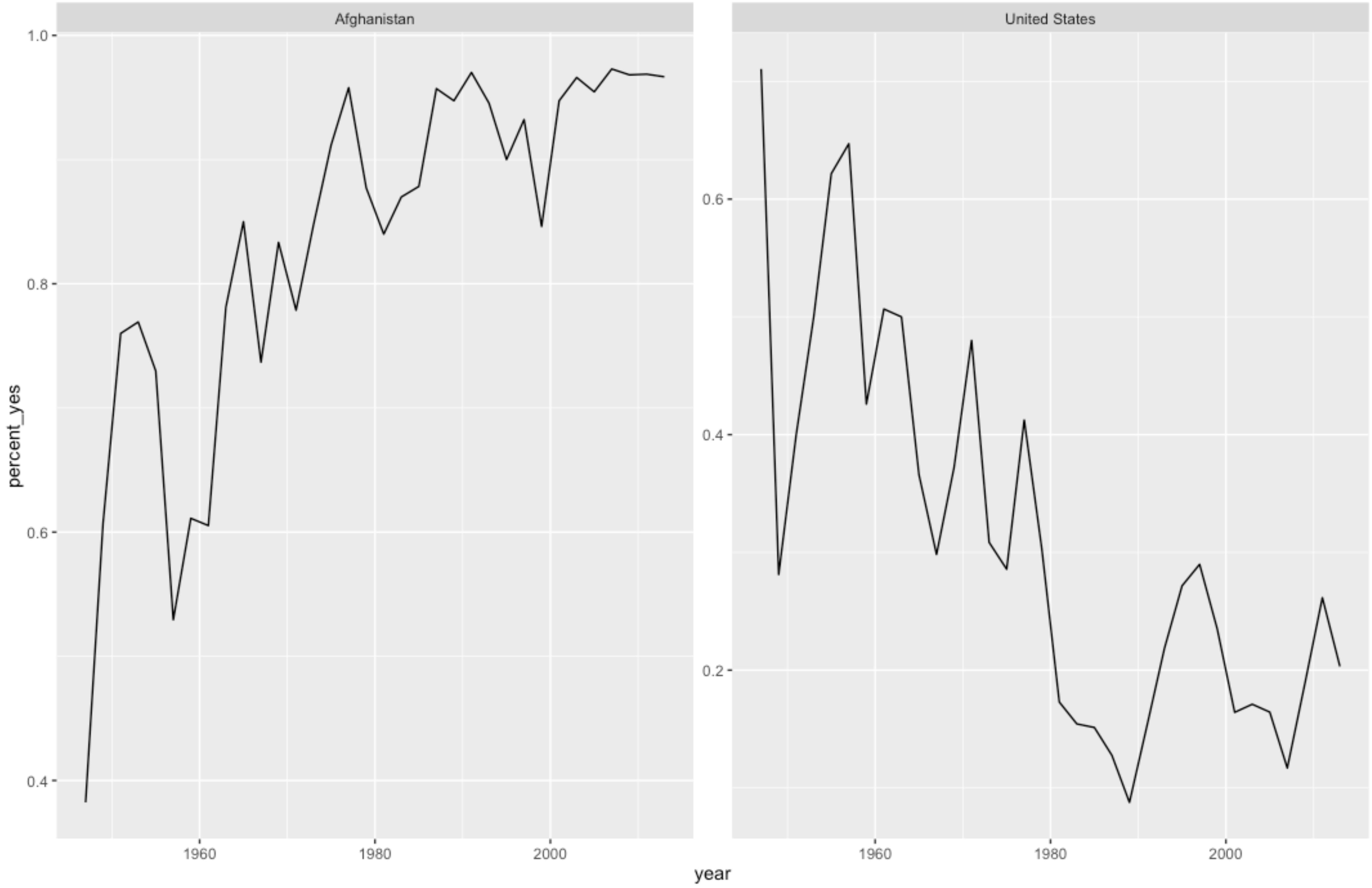
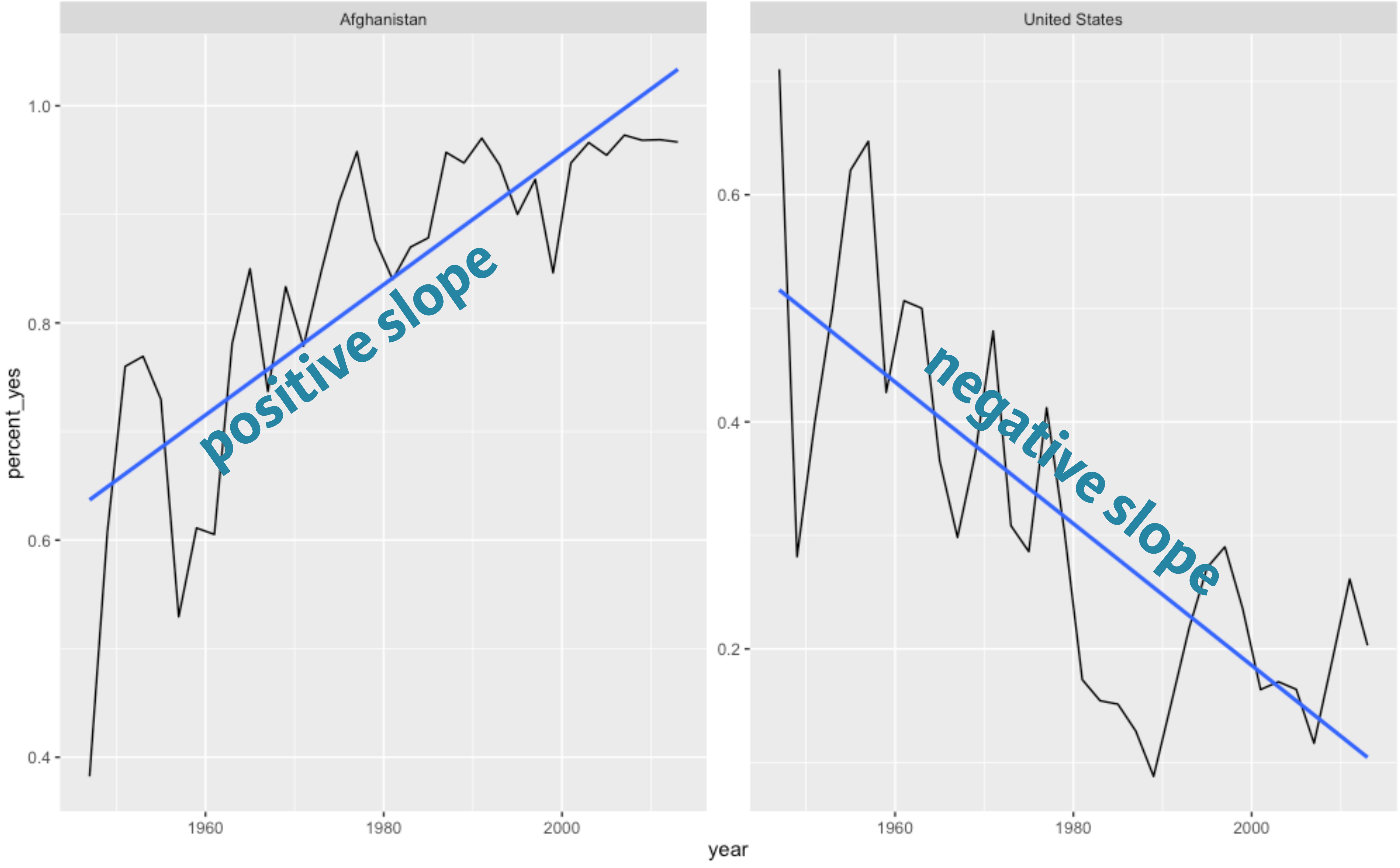
EXPLORATORY DATA ANALYSIS: CASE STUDY

Linear regression

Quantifying trends



Linear regression



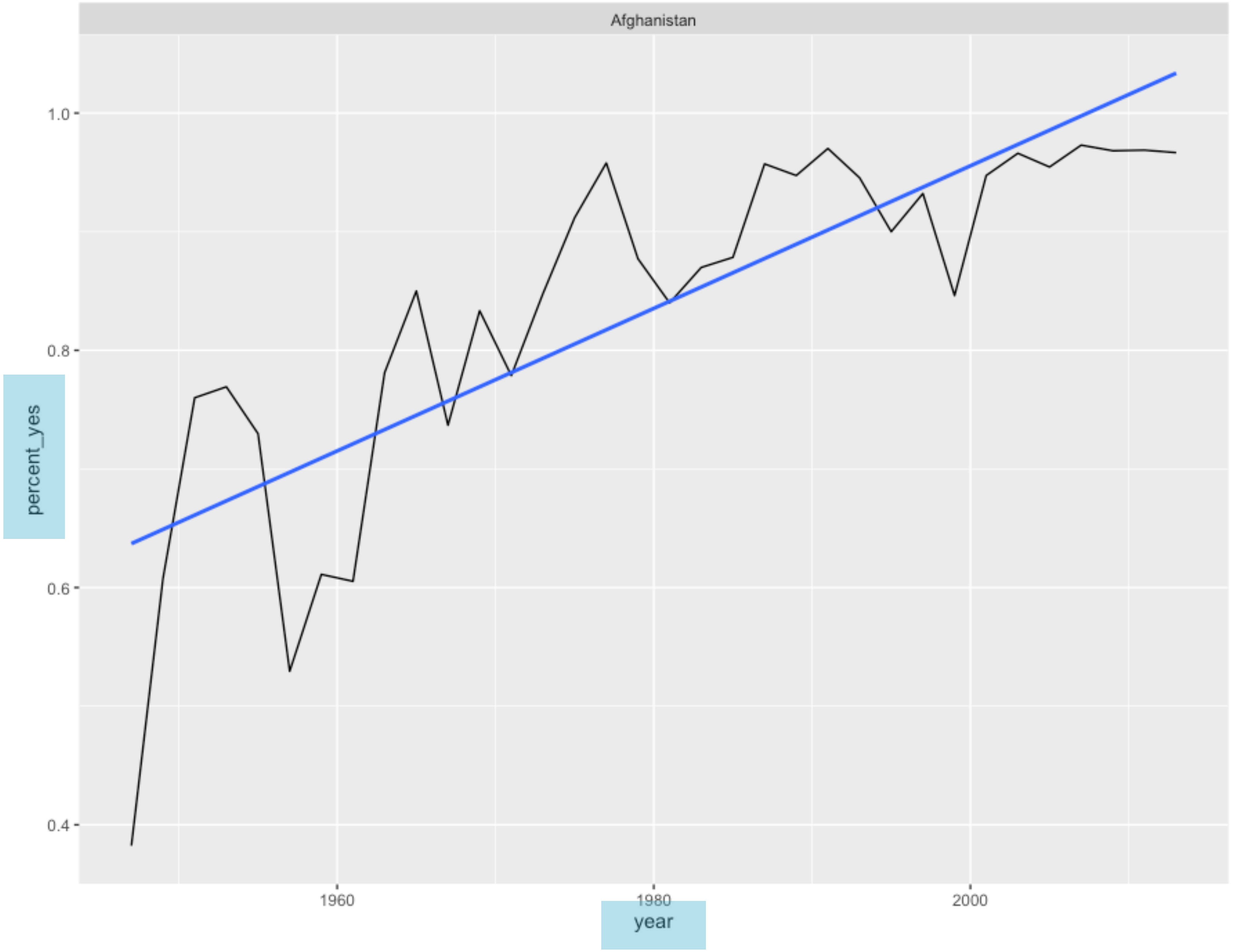
Fitting model to Afghanistan

```
> afghanistan <- by_year_country %>%
    filter(country == "Afghanistan")
> afghanistan
# A tibble: 34 × 4
    year     country total percent_yes
   <dbl>       <chr> <int>       <dbl>
1   1947 Afghanistan    34   0.3823529
2   1949 Afghanistan    51   0.6078431
3   1951 Afghanistan    25   0.7600000
4   1953 Afghanistan    26   0.7692308
5   1955 Afghanistan    37   0.7297297
6   1957 Afghanistan    34   0.5294118
7   1959 Afghanistan    54   0.6111111
8   1961 Afghanistan    76   0.6052632
9   1963 Afghanistan    32   0.7812500
10  1965 Afghanistan    40   0.8500000
# ... with 24 more rows
```

Fitting model to Afghanistan

explained by

```
> model <- lm(percent_yes ~ year, data = afghanistan)
```



dependent variable

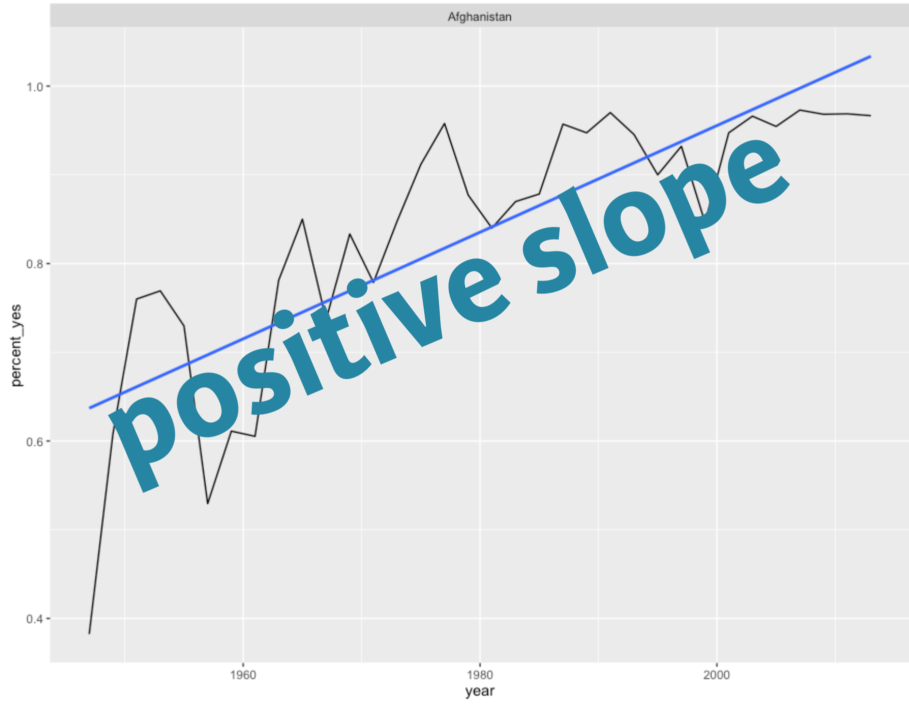
independent variable

Fitting model to Afghanistan

```
> summary(model)

Call:
lm(formula = percent_yes ~ year, data = afghanistan)

Residuals:
      Min        1Q    Median        3Q       Max
-0.254667 -0.038650 -0.001945  0.057110  0.140596

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.106e+01  1.471e+00  -7.523 1.44e-08 ***
year         6.009e-03  7.426e-04   8.092 3.06e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.08497 on 32 degrees of freedom
Multiple R-squared:  0.6717,  Adjusted R-squared:  0.6615
F-statistic: 65.48 on 1 and 32 DF,  p-value: 3.065e-09
```

6e-03 = .006

3e-09 = .000000003



positive slope

Visualization can surprise you, but it doesn’t scale well.
Modeling scales well, but it can’t surprise you.

-Hadley Wickham

Let’s practice!

EXPLORATORY DATA ANALYSIS: CASE STUDY

Tidying models with broom

A model fit is a “messy” object

```
> summary(model)

Call:
lm(formula = percent_yes ~ year, data = afghanistan)

Residuals:
      Min         1Q     Median         3Q        Max
-0.254667 -0.038650 -0.001945  0.057110  0.140596

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.106e+01  1.471e+00  -7.523 1.44e-08 ***
year         6.009e-03  7.426e-04   8.092 3.06e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.08497 on 32 degrees of freedom
Multiple R-squared:  0.6717,  Adjusted R-squared:  0.6615
F-statistic: 65.48 on 1 and 32 DF,  p-value: 3.065e-09
```

Models are difficult to combine

```
> model1 <- lm(percent_yes ~ year, data = afghanistan)
> model2 <- lm(percent_yes ~ year, data = united_states)
> model3 <- lm(percent_yes ~ year, data = canada)
```

broom turns a model into a data frame

```
> library(broom)
> tidy(model)
         term     estimate    std.error statistic      p.value
1 (Intercept) -11.063084650 1.4705189228 -7.523252 1.444892e-08
2        year   0.006009299 0.0007426499  8.091698 3.064797e-09
```

Tidy models can be combined

```
> model1 <- lm(percent_yes ~ year, data = afghanistan)
> model2 <- lm(percent_yes ~ year, data = united_states)

> tidy(model1)
         term      estimate    std.error statistic      p.value
1 (Intercept) -11.063084650 1.4705189228 -7.523252 1.444892e-08
2        year   0.006009299 0.0007426499  8.091698 3.064797e-09
> tidy(model2)
         term     estimate    std.error statistic      p.value
1 (Intercept) 12.664145512 1.8379742715  6.890274 8.477089e-08
2        year -0.006239305 0.0009282243 -6.721764 1.366904e-07

> bind_rows(tidy(model1), tidy(model2))
         term      estimate    std.error statistic      p.value
1 (Intercept) -11.063084650 1.4705189228 -7.523252 1.444892e-08
2        year   0.006009299 0.0007426499  8.091698 3.064797e-09
3 (Intercept)  12.664145512 1.8379742715  6.890274 8.477089e-08
4        year  -0.006239305 0.0009282243 -6.721764 1.366904e-07
```

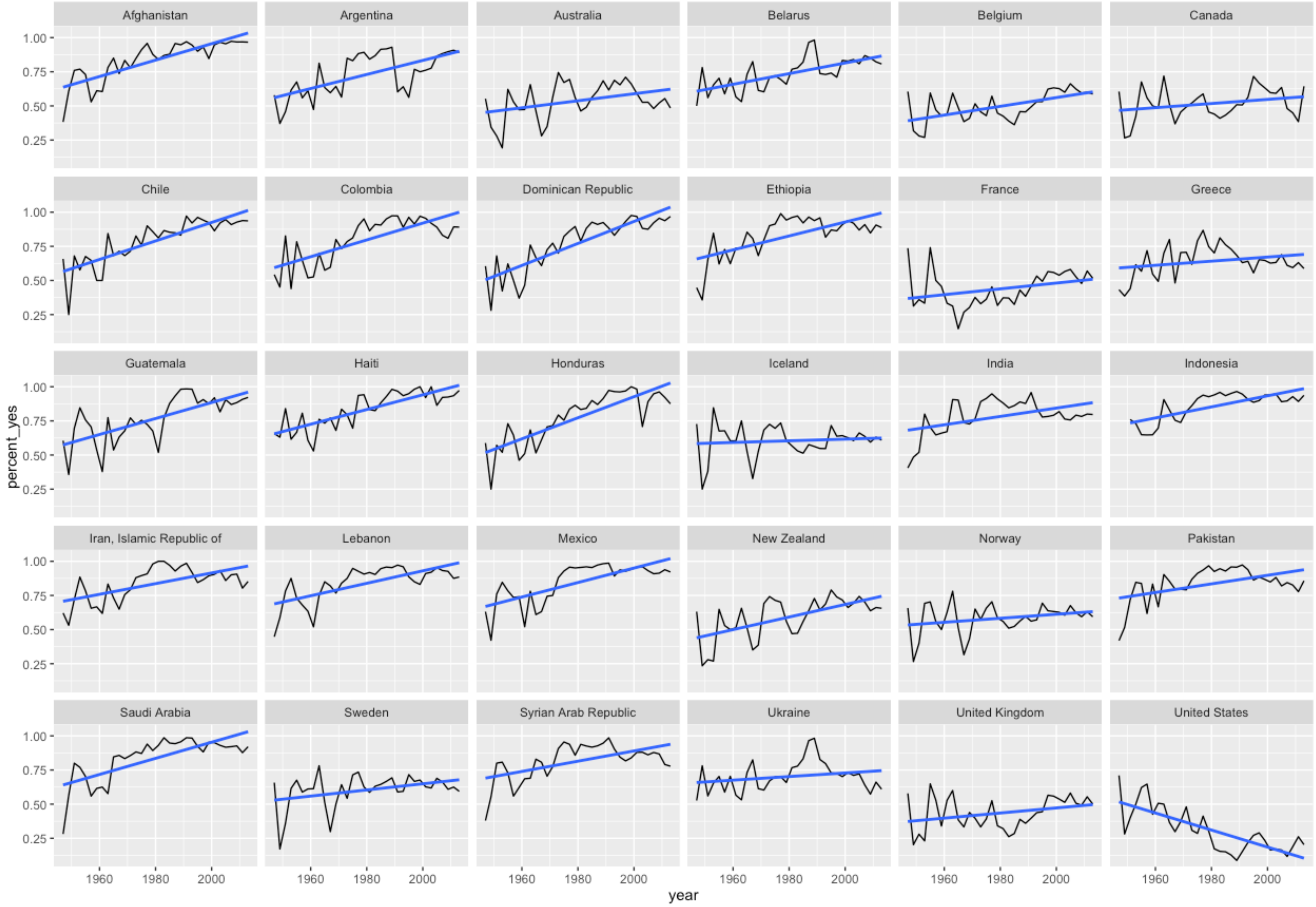
EXPLORATORY DATA ANALYSIS: CASE STUDY

Let's practice!

EXPLORATORY DATA ANALYSIS: CASE STUDY

Nesting for multiple models

One model for each country



Start with one row per country

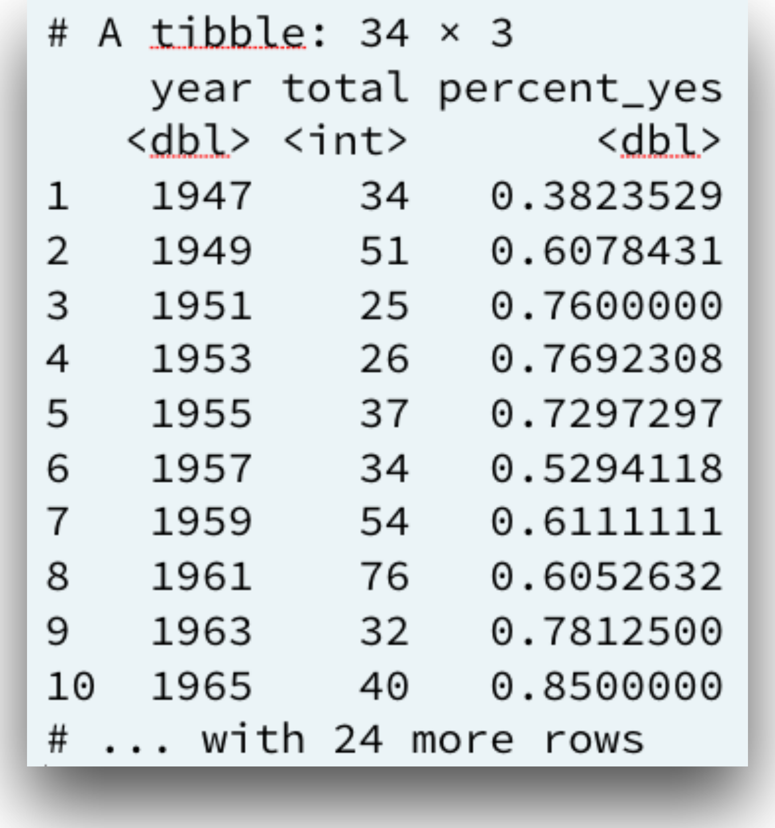
```
> by_year_country
# A tibble: 4,744 × 4
    year                       country total percent_yes
   <dbl>                         <chr> <int>       <dbl>
1   1947                   Afghanistan    34   0.3823529
2   1947                     Argentina    38   0.5789474
3   1947                     Australia    38   0.5526316
4   1947                       Belarus    38   0.5000000
5   1947                       Belgium    38   0.6052632
6   1947 Bolivia, Plurinational State of    37   0.5945946
7   1947                        Brazil    38   0.6578947
8   1947                        Canada    38   0.6052632
9   1947                         Chile    38   0.6578947
10  1947                      Colombia    35   0.5428571
# ... with 4,734 more rows
```

nest() turns it into one row per country

```
> library(tidyr)
> by_year_country %>%
    nest(-country)
# A tibble: 200 × 2
                                country               data
                                  <chr>             <list>
1                           Afghanistan <tibble [34 × 3]>
2                             Argentina <tibble [34 × 3]>
3                             Australia <tibble [34 × 3]>
4                               Belarus <tibble [34 × 3]>
5                               Belgium <tibble [34 × 3]>
6   Bolivia, Plurinational State of <tibble [34 × 3]>
7                                Brazil <tibble [34 × 3]>
8                                Canada <tibble [34 × 3]>
9                                 Chile <tibble [34 × 3]>
10                             Colombia <tibble [34 × 3]>
# ... with 190 more rows
```

-country means "nest all except country"

Contains the "nested" year, total, percent_yes data for just Afghanistan



```
# A tibble: 34 × 3
    year total percent_yes
   <dbl> <int>       <dbl>
1   1947    34   0.3823529
2   1949    51   0.6078431
3   1951    25   0.7600000
4   1953    26   0.7692308
5   1955    37   0.7297297
6   1957    34   0.5294118
7   1959    54   0.6111111
8   1961    76   0.6052632
9   1963    32   0.7812500
10  1965    40   0.8500000
# ... with 24 more rows
```

unnest() does the opposite

```
> by_year_country %>%
    nest(country) %>%
    unnest(data)
# A tibble: 4,744 × 4
    year total percent_yes        country
   <dbl> <int>       <dbl>          <chr>
1   1947    34   0.3823529    Afghanistan
2   1947    38   0.5789474      Argentina
3   1947    38   0.5789474 United Kingdom
4   1947    38   0.5526316      Australia
5   1947    38   0.5000000        Belarus
6   1947    38   0.5000000          Egypt
7   1947    38   0.5000000   South Africa
8   1947    38   0.5000000     Yugoslavia
9   1947    38   0.6052632        Belgium
10  1947    38   0.6052632         Canada
```

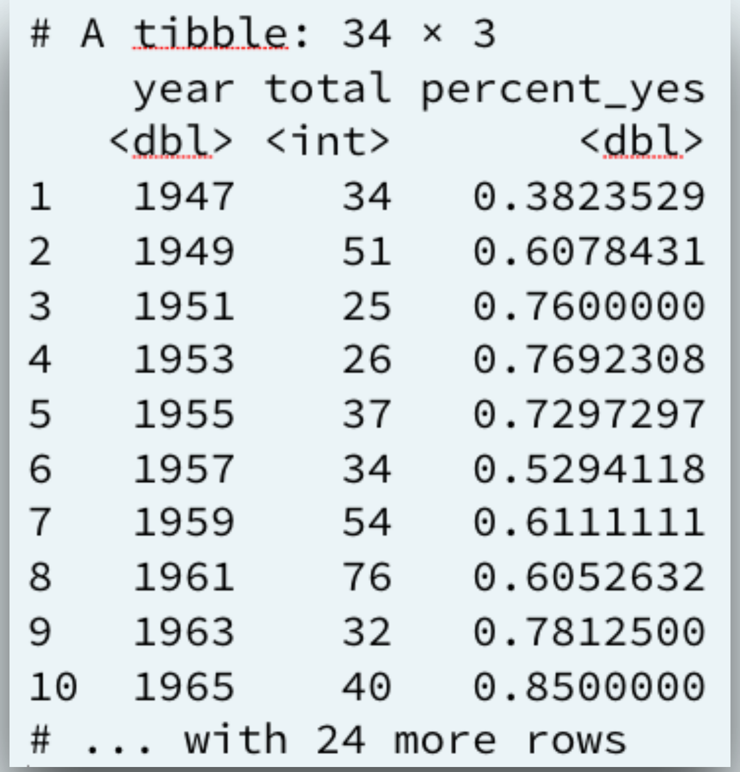
EXPLORATORY DATA ANALYSIS: CASE STUDY

Let’s practice!

Fitting multiple models

nest() turns data into one row per country

```
> library(tidyr)
> by_year_country %>%
    nest(-country)
# A tibble: 200 × 2
                            country                  data
                              <chr>                <list>
1                       Afghanistan <tibble [34 × 3]>
2                         Argentina <tibble [34 × 3]>
3                         Australia <tibble [34 × 3]>
4                           Belarus <tibble [34 × 3]>
5                           Belgium <tibble [34 × 3]>
6  Bolivia, Plurinational State of <tibble [34 × 3]>
7                            Brazil <tibble [34 × 3]>
8                            Canada <tibble [34 × 3]>
9                             Chile <tibble [34 × 3]>
10                         Colombia <tibble [34 × 3]>
# ... with 190 more rows
```



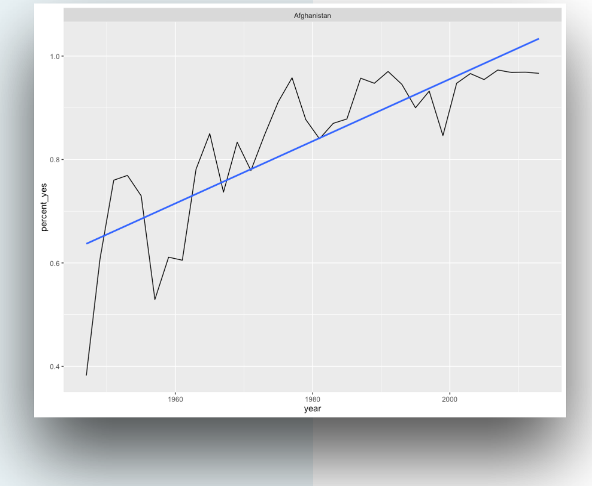
```
# A tibble: 34 × 3
    year total percent_yes
   <dbl> <int>       <dbl>
1   1947    34   0.3823529
2   1949    51   0.6078431
3   1951    25   0.7600000
4   1953    26   0.7692308
5   1955    37   0.7297297
6   1957    34   0.5294118
7   1959    54   0.6111111
8   1961    76   0.6052632
9   1963    32   0.7812500
10  1965    40   0.8500000
# ... with 24 more rows
```

map() applies an operation to each item in a list

```
> v <- list(1, 2, 3)
> map(v, ~ . * 10)
[[1]]
[1] 10

[[2]]
[1] 20

[[3]]
[1] 30
```

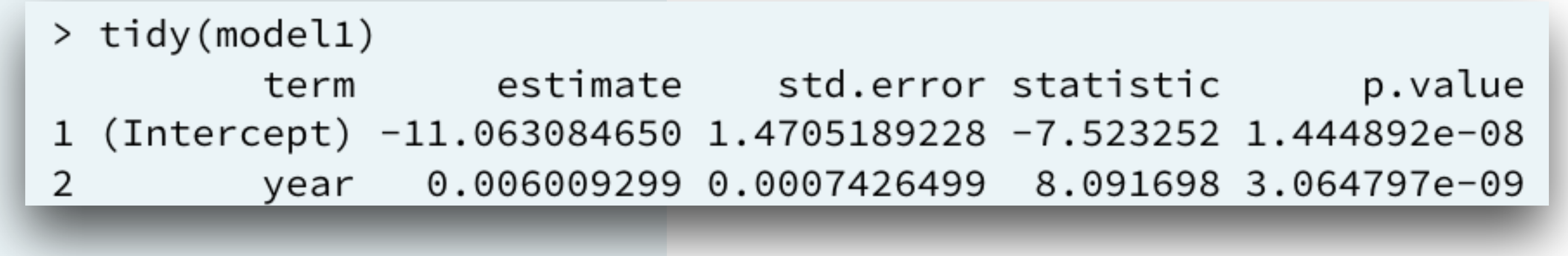
map() fits a model to each dataset

```
> library(purrr)
> by_year_country %>%
    nest(-country) %>%
    mutate(models = map(data, ~ lm(percent_yes ~ year, .)))
# A tibble: 200 × 3
                          country                 data   models
                            <chr>               <list>   <list>
1                     Afghanistan <tibble [34 × 3]> <S3: lm>
2                       Argentina <tibble [34 × 3]> <S3: lm>
3                       Australia <tibble [34 × 3]> <S3: lm>
4                         Belarus <tibble [34 × 3]> <S3: lm>
5                         Belgium <tibble [34 × 3]> <S3: lm>
6  Bolivia, Plurinational State of <tibble [34 × 3]> <S3: lm>
7                          Brazil <tibble [34 × 3]> <S3: lm>
8                          Canada <tibble [34 × 3]> <S3: lm>
9                           Chile <tibble [34 × 3]> <S3: lm>
10                       Colombia <tibble [34 × 3]> <S3: lm>
# ... with 190 more rows
```



tidy turns each model into a data frame

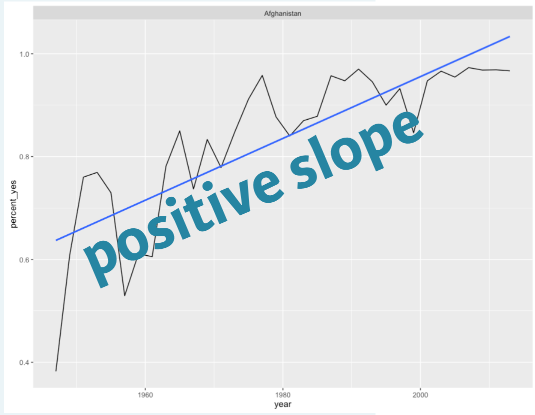
```
> by_year_country %>%
    nest(-country) %>%
    mutate(models = map(data, ~ lm(percent_yes ~ year, .))) %>%
    mutate(tidied = map(models, tidy))
# A tibble: 200 × 4
                          country              data    models                  tidied
                            <chr>            <list>    <list>                  <list>
1                     Afghanistan <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
2                       Argentina <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
3                       Australia <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
4                         Belarus <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
5                         Belgium <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
6  Bolivia, Plurinational State of <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
7                          Brazil <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
8                          Canada <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
9                           Chile <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
10                       Colombia <tibble [34 × 3]> <S3: lm> <data.frame [2 × 5]>
# ... with 190 more rows
```



```
> tidy(model1)
         term      estimate    std.error statistic      p.value
1 (Intercept) -11.063084650 1.4705189228 -7.523252 1.444892e-08
2        year   0.006009299 0.0007426499  8.091698 3.064797e-09
```

unnest() combines the tidied models

```
> by_year_country %>%
 1   nest(-country) %>%
 2   mutate(models = map(data, ~ lm(percent_yes ~ year, .))) %>%
 3   mutate(tidied = map(models, tidy)) %>%
 4   unnest(tidied)
# A tibble: 399 × 6
       country        term      estimate    std.error statistic      p.value
         <chr>       <chr>         <dbl>        <dbl>     <dbl>        <dbl>
1  Afghanistan (Intercept) -11.063084650 1.4705189228 -7.523252 1.444892e-08
2  Afghanistan        year   0.006009299 0.0007426499  8.091698 3.064797e-09
3    Argentina (Intercept)  -9.464512565 2.1008982371 -4.504984 8.322481e-05
4    Argentina        year   0.005148829 0.0010610076  4.852773 3.047078e-05
5    Australia (Intercept)  -4.545492536 2.1479916283 -2.116159 4.220387e-02
6    Australia        year   0.002567161 0.0010847910  2.366503 2.417617e-02
7      Belarus (Intercept)  -7.000692717 1.5024232546 -4.659601 5.329950e-05
8      Belarus        year   0.003907557 0.0007587624  5.149908 1.284924e-05
9      Belgium (Intercept)  -5.845534016 1.5153390521 -3.857575 5.216573e-04
10     Belgium        year   0.003203234 0.0007652852  4.185673 2.072981e-04
# ... with 389 more rows
```



Let’s practice!

EXPLORATORY DATA ANALYSIS: CASE STUDY

Working with many tidy models

We have a model for each country

```
> country_coefficients
# A tibble: 399 × 6
       country        term      estimate    std.error statistic      p.value
         <chr>       <chr>         <dbl>        <dbl>     <dbl>        <dbl>
1  Afghanistan (Intercept) -11.063084650 1.4705189228 -7.523252 1.444892e-08
2  Afghanistan        year   0.006009299 0.0007426499  8.091698 3.064797e-09
3    Argentina (Intercept)  -9.464512565 2.1008982371 -4.504984 8.322481e-05
4    Argentina        year   0.005148829 0.0010610076  4.852773 3.047078e-05
5    Australia (Intercept)  -4.545492536 2.1479916283 -2.116159 4.220387e-02
6    Australia        year   0.002567161 0.0010847910  2.366503 2.417617e-02
7      Belarus (Intercept)  -7.000692717 1.5024232546 -4.659601 5.329950e-05
8      Belarus        year   0.003907557 0.0007587624  5.149908 1.284924e-05
9      Belgium (Intercept)  -5.845534016 1.5153390521 -3.857575 5.216573e-04
10     Belgium        year   0.003203234 0.0007652852  4.185673 2.072981e-04
# ... with 389 more rows
```

Filter for the year term (slope)

```
> country_coefficients %>%
    filter(term == "year")
# A tibble: 199 × 6
                       country  term    estimate    std.error statistic      p.value
                         <chr> <chr>       <dbl>        <dbl>     <dbl>        <dbl>
1                  Afghanistan  year 0.006009299 0.0007426499  8.091698 3.064797e-09
2                    Argentina  year 0.005148829 0.0010610076  4.852773 3.047078e-05
3                    Australia  year 0.002567161 0.0010847910  2.366503 2.417617e-02
4                      Belarus  year 0.003907557 0.0007587624  5.149908 1.284924e-05
5                      Belgium  year 0.003203234 0.0007652852  4.185673 2.072981e-04
6  Bolivia, Plurinational State of  year 0.005802864 0.0009657515  6.008651 1.058595e-06
7                       Brazil  year 0.006107151 0.0008167736  7.477164 1.641169e-08
8                       Canada  year 0.001515867 0.0009552118  1.586943 1.223590e-01
9                        Chile  year 0.006775560 0.0008220463  8.242310 2.045608e-09
10                    Colombia  year 0.006157755 0.0009645084  6.384346 3.584226e-07
# ... with 189 more rows
```

Multiple hypothesis correction because some p-values will be less than .05 by chance

Filtered by adjusted p-value

```
> country_coefficients %>%
    filter(term == "year") %>%
    filter(p.adjust(p.value) < .05)
# A tibble: 61 × 6
                          country  term    estimate    std.error statistic      p.value
                            <chr> <chr>       <dbl>        <dbl>     <dbl>        <dbl>
1                     Afghanistan  year 0.006009299 0.0007426499  8.091698 3.064797e-09
2                       Argentina  year 0.005148829 0.0010610076  4.852773 3.047078e-05
3                         Belarus  year 0.003907557 0.0007587624  5.149908 1.284924e-05
4                         Belgium  year 0.003203234 0.0007652852  4.185673 2.072981e-04
5   Bolivia, Plurinational State of  year 0.005802864 0.0009657515  6.008651 1.058595e-06
6                          Brazil  year 0.006107151 0.0008167736  7.477164 1.641169e-08
7                           Chile  year 0.006775560 0.0008220463  8.242310 2.045608e-09
8                        Colombia  year 0.006157755 0.0009645084  6.384346 3.584226e-07
9                      Costa Rica  year 0.006539273 0.0008119113  8.054171 3.391094e-09
10                           Cuba  year 0.004610867 0.0007205029  6.399512 3.431579e-07
```

EXPLORATORY DATA ANALYSIS: CASE STUDY

Let’s practice!