Longitudinal Data Analysis

Adam J Sullivan, PhD

04/11/2018

Longitudinal Data Analysis

Cross-Sectional Studies

- Up until this point we have focused on what we call Cross-sectional data.
- Cross-sectional data is data that is contained at a single time point.
- This is basically a snapshot in time so there is no way to measure how things change over the course of time.
- These are affordable to run and can measure many things at the same time.

Longitudinal Studies

- Longitudinal studies observe a group of subjects over a certain period of time.
- This means that there are at least 2 time points of measuring variables.
- Longitudinal studies allow you to observe how a treatment impacts things.

Longitudinal vs Cross-sectional

- Cross-sectional are quicker and cheaper to run.
- Many times cross-sectional studies are run first to test for associations prior to a longitudinal.
- Longitudinal allows for causality to be explored.
- Longitudinal data is more difficult to analyze.

What makes Longitudinal Data Different?

- **Correlation**
  - In cross-sectional studies we assume that data points are independent of each other.
  - We cannot do this in longitudinal because each subjects data are completely dependent.
  - If a subject has high cholesterol at one occasion they are likely to be high at the next occasion.

What makes Longitudinal Data Different?

- **Variability**
  - In typical linear regression we assume that we have homoscedasticity.
  - In longitudinal data the variability at the beginning of the study is likely different than at the end.

Notation

- We must consider sum new notation as we now have multiple variable observations for each subject.
- Y_{ij} which is the outcome variable for the i^{th} subject $(i = 1, \ldots, N)$ at the j^{th} occasion $(j = 1, \ldots, n)$.
- We use this notation when we have measures that are equally separated over time.

How does this data look?

- Each individual has data that looks like:

$$Y_i = \begin{bmatrix} Y_{i1} \\ Yi2 \\ \vdots \\ Y_{in} \end{bmatrix}$$

How does this data look?

- That means the whole data looks like:

$$\begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & \dots & Y_{1n} \\ Y_{21} & Y_{22} & Y_{23} & \dots & Y_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ Y_{N1} & Y_{N2} & Y_{N3} & \dots & Y_{Nn} \end{bmatrix}$$

Exploring Longitudinal Data

- We will first consider graphs to explore longitudinal data.
- We will consider some data from Gapminder.
- This tracks data for all countries over the world.
- Our data will consider looking at life expectancy over the course of time.

Exploring Longitudinal Data

- The data

```
library(gapminder)
gapminder
```

Exploring Longitudinal Data

```
## # A tibble: 1,704 x 6
##    country     continent  year lifeExp      pop gdpPercap
##    <fctr>      <fctr>    <int>   <dbl>    <int>     <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333       779
##  2 Afghanistan Asia       1957    30.3  9240934       821
##  3 Afghanistan Asia       1962    32.0 10267083       853
##  4 Afghanistan Asia       1967    34.0 11537966       836
##  5 Afghanistan Asia       1972    36.1 13079460       740
##  6 Afghanistan Asia       1977    38.4 14880372       786
##  7 Afghanistan Asia       1982    39.9 12881816       978
##  8 Afghanistan Asia       1987    40.8 13867957       852
##  9 Afghanistan Asia       1992    41.7 16317921       649
## 10 Afghanistan Asia       1997    41.8 22227415       635
## # ... with 1,694 more rows
```

Exploring Longitudinal Data

- Spaghetti plot

```
library(gapminder)
library(tidyverse)
library(ggplot2)
gapminder %>%
  ggplot(aes(year, lifeExp, group = country)) +
    geom_line(alpha = 1/3)
```

Exploring Longitudinal Data

Exploring Longitudinal Data

- What do we see?

Exploring Data

- Consider basic Regression Models
- Lets do this for just one country.
- We will consider say the country of Kenya

```
kenya <- gapminder %>%
  filter(country=="Kenya")
kenya_mod <- lm(lifeExp ~year, data=kenya)
```

Exploring Data - Full Data Line

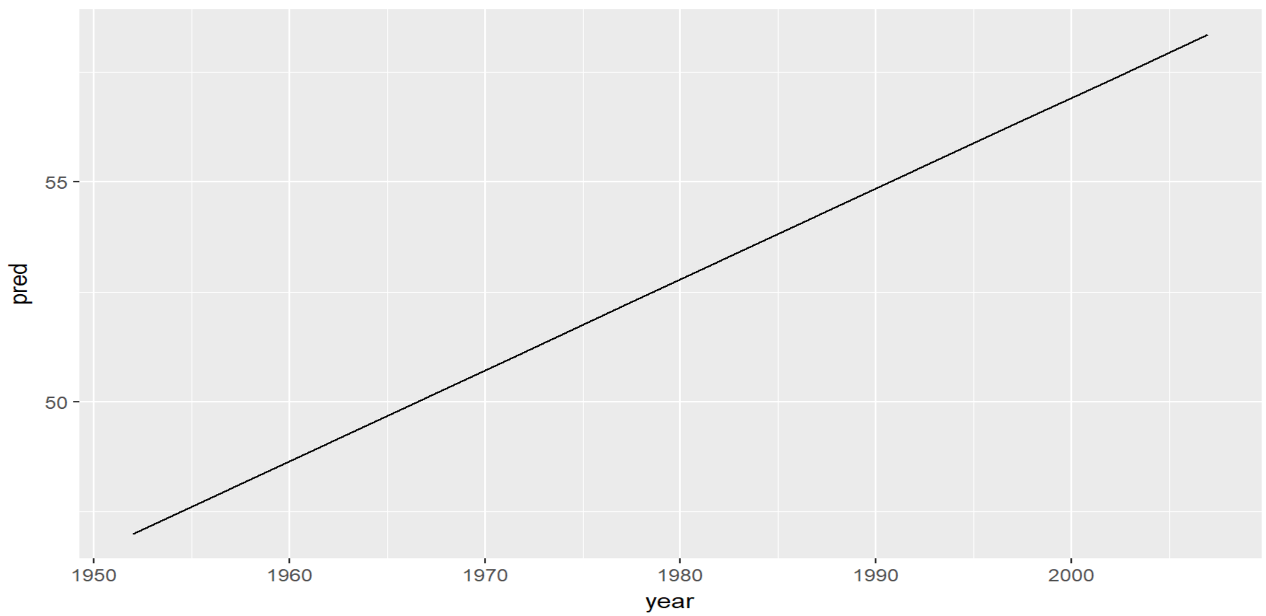
```
kenya %>%
  ggplot(aes(year, lifeExp)) +
  geom_line()
```

Exploring Data- Full Data Line

Linear Trend of Data - Linear Model

```
library(modelr)
kenya %>%
  add_predictions(kenya_mod) %>%
  ggplot(aes(year, pred)) +
  geom_line()
```

Linear Trend of Data - Linear Model



Linear Trend of Data - Left Over

```
library(modelr)
kenya %>%
  add_residuals(kenya_mod) %>%
  ggplot(aes(year, resid)) +
  geom_hline(yintercept = 0, colour = "white", size = 3) +
  geom_line()
```

Linear Trend of Data - Left Over

Exploring Data for All Countries

- Setting up the data

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

by_country
```

Exploring Data for All Countries

- Setting up the data

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

Modeling with Our Current Tools

- Let's just consider simple linear models for each country to explore the data.
- We are violating assumptions so we cannot interpret these models for effects but we can use them to explore the data in a simple manner.

Modeling with our Current Tools

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

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

Modeling with our Current Tools

```
## # A tibble: 142 x 4
##    country     continent data              model   
##    <fctr>      <fctr>    <list>            <list>  
##  1 Afghanistan Asia      <tibble [12 x 4]> <S3: lm>
##  2 Albania     Europe    <tibble [12 x 4]> <S3: lm>
##  3 Algeria     Africa    <tibble [12 x 4]> <S3: lm>
##  4 Angola      Africa    <tibble [12 x 4]> <S3: lm>
##  5 Argentina   Americas  <tibble [12 x 4]> <S3: lm>
##  6 Australia   Oceania   <tibble [12 x 4]> <S3: lm>
##  7 Austria     Europe    <tibble [12 x 4]> <S3: lm>
##  8 Bahrain     Asia      <tibble [12 x 4]> <S3: lm>
##  9 Bangladesh  Asia      <tibble [12 x 4]> <S3: lm>
## 10 Belgium     Europe    <tibble [12 x 4]> <S3: lm>
## # ... with 132 more rows
```

Our New Data

```
by_country %>%
  filter(continent == "Americas")
```

Our New Data

```
## # A tibble: 25 x 4
##    country            continent data              model
##    <fctr>             <fctr>    <list>            <list>
##  1 Argentina          Americas  <tibble [12 x 4]> <S3: lm>
##  2 Bolivia            Americas  <tibble [12 x 4]> <S3: lm>
##  3 Brazil             Americas  <tibble [12 x 4]> <S3: lm>
##  4 Canada             Americas  <tibble [12 x 4]> <S3: lm>
##  5 Chile              Americas  <tibble [12 x 4]> <S3: lm>
##  6 Colombia           Americas  <tibble [12 x 4]> <S3: lm>
##  7 Costa Rica         Americas  <tibble [12 x 4]> <S3: lm>
##  8 Cuba               Americas  <tibble [12 x 4]> <S3: lm>
##  9 Dominican Republic Americas  <tibble [12 x 4]> <S3: lm>
## 10 Ecuador            Americas  <tibble [12 x 4]> <S3: lm>
## # ... with 15 more rows
```

Adding in Residuals

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

Adding in Residuals

```
## # A tibble: 142 x 5
##    country     continent data              model    resids
##    <fctr>      <fctr>    <list>            <list>   <list>
##  1 Afghanistan Asia      <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
##  2 Albania     Europe    <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
##  3 Algeria     Africa    <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
##  4 Angola      Africa    <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
##  5 Argentina   Americas  <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
##  6 Australia   Oceania   <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
##  7 Austria     Europe    <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
##  8 Bahrain     Asia      <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
##  9 Bangladesh  Asia      <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
## 10 Belgium     Europe    <tibble [12 x 4]> <S3: lm> <tibble [12 x 5]>
## # ... with 132 more rows
```

Unnest Data

```
resids <- unnest(by_country, resids)
resids
```

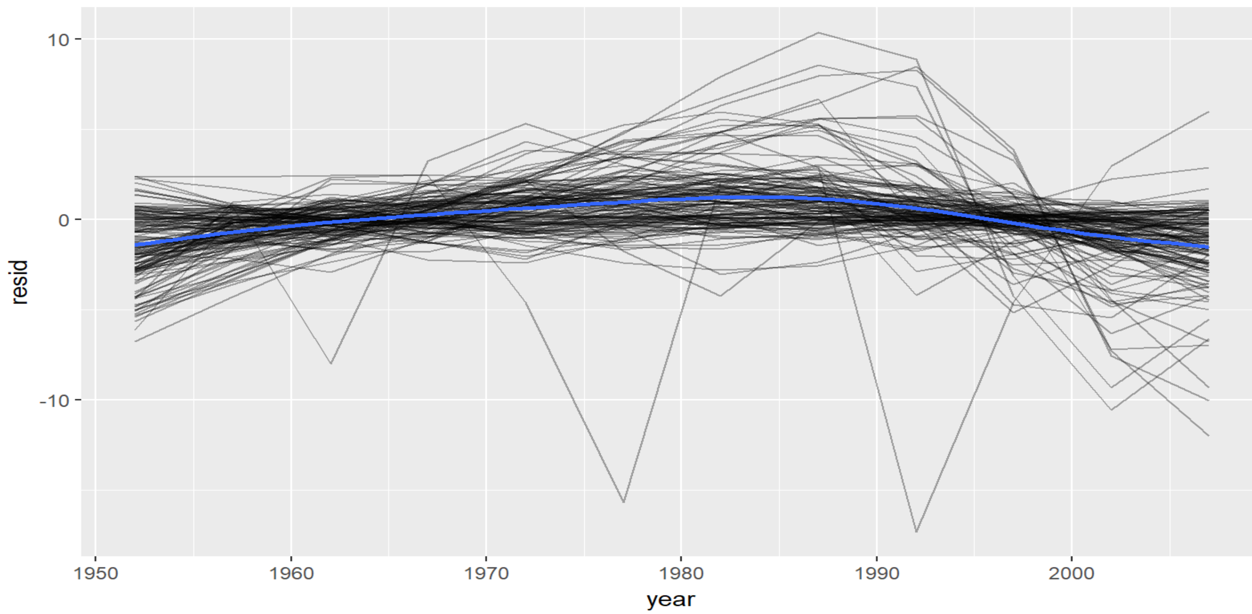
Unnest Data

```
## # A tibble: 1,704 x 7
##    country     continent  year lifeExp      pop gdpPercap   resid
##    <fctr>      <fctr>    <int>   <dbl>    <int>     <dbl>   <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333       779 -1.11
##  2 Afghanistan Asia       1957    30.3  9240934       821 -0.952
##  3 Afghanistan Asia       1962    32.0 10267083       853 -0.664
##  4 Afghanistan Asia       1967    34.0 11537966       836 -0.0172
##  5 Afghanistan Asia       1972    36.1 13079460       740  0.674
##  6 Afghanistan Asia       1977    38.4 14880372       786  1.65
##  7 Afghanistan Asia       1982    39.9 12881816       978  1.69
##  8 Afghanistan Asia       1987    40.8 13867957       852  1.28
##  9 Afghanistan Asia       1992    41.7 16317921       649  0.754
## 10 Afghanistan Asia       1997    41.8 22227415       635 -0.534
## # ... with 1,694 more rows
```

Plotting all Residuals

```
resids %>%
  ggplot(aes(year, resid)) +
    geom_line(aes(group = country), alpha = 1 / 3) +
    geom_smooth(se = FALSE)
```

Plotting all Residuals



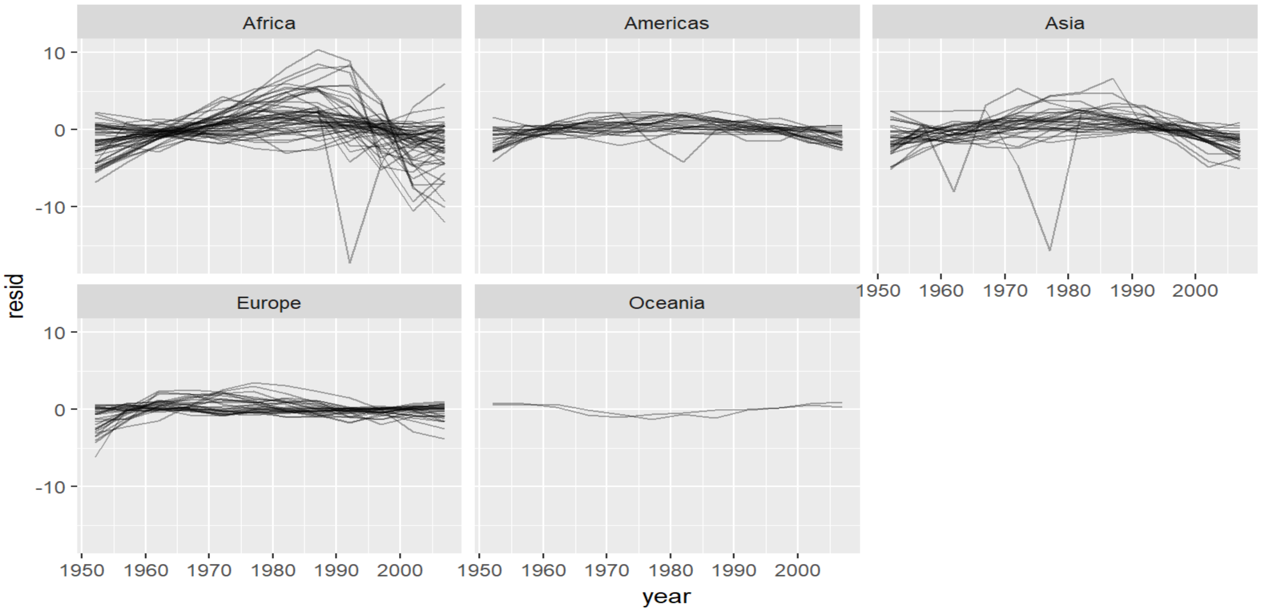
What do we see?

- Some of the residuals fit well.
- Others seem to have issues

Graph fit by Continent

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

Graph fit by Continent



What do we see?

Let's Check Fit Further

- We will look at our R^2 values.

```
glance <- by_country %>%
  mutate(glance = map(model, broom::glance)) %>%
  unnest(glance, .drop = TRUE)
glance
```

Let's Check Fit Further

```
## # A tibble: 142 x 13
##    count~ cont~ r.sq~ adj.~ sigma stati~  p.value    df logLik   AIC   BIC
##    <fctr> <fct> <dbl> <dbl> <dbl>  <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
##  1 Afgha~ Asia  0.948 0.942 1.22   181   9.84e- 8     2 -18.3  42.7  44.1
##  2 Alban~ Euro~ 0.911 0.902 1.98   102   1.46e- 6     2 -24.1  54.3  55.8
##  3 Alger~ Afri~ 0.985 0.984 1.32   662   1.81e-10     2 -19.3  44.6  46.0
##  4 Angola Afri~ 0.888 0.877 1.41    79.1 4.59e- 6     2 -20.0  46.1  47.5
##  5 Argen~ Amer~ 0.996 0.995 0.292 2246   4.22e-13     2 - 1.17  8.35  9.80
##  6 Austr~ Ocea~ 0.980 0.978 0.621  481   8.67e-10     2 -10.2  26.4  27.9
##  7 Austr~ Euro~ 0.992 0.991 0.407 1261   7.44e-12     2 - 5.16 16.3  17.8
##  8 Bahra~ Asia  0.967 0.963 1.64   291   1.02e- 8     2 -21.9  49.7  51.2
##  9 Bangl~ Asia  0.989 0.988 0.977  930   3.37e-11     2 -15.7  37.3  38.8
## 10 Belgi~ Euro~ 0.995 0.994 0.293 1822   1.20e-12     2 - 1.20  8.40  9.85
## # ... with 132 more rows, and 2 more variables: deviance <dbl>,
## #   df.residual <int>
```

How are the R^2

```
glance %>%
  arrange(r.squared)
```

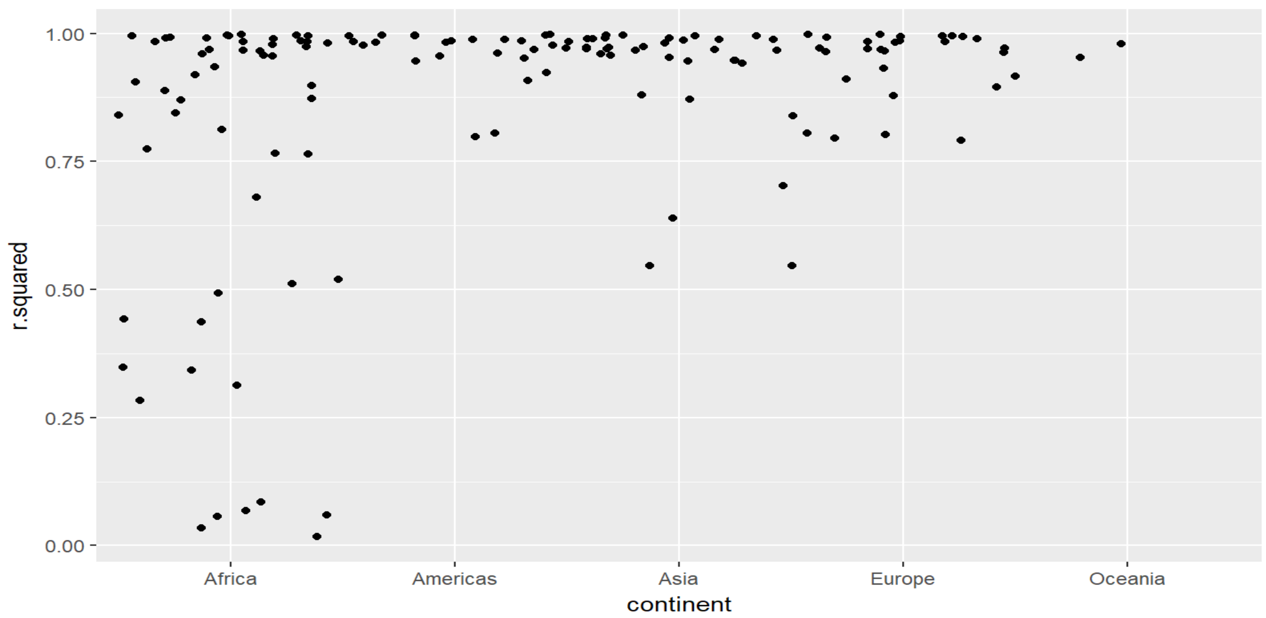
How are the R^2

```
## # A tibble: 142 x 13
##    count~ cont~ r.squ~ adj.r.s~ sigma stat~ p.val~    df logL~   AIC   BIC
##    <fctr> <fct>  <dbl>    <dbl> <dbl> <dbl>  <dbl> <int> <dbl> <dbl> <dbl>
##  1 Rwanda Afri~ 0.0172 -0.0811   6.56 0.175 0.685      2 -38.5  83.0  84.5
##  2 Botsw~ Afri~ 0.0340 -0.0626   6.11 0.352 0.566      2 -37.7  81.3  82.8
##  3 Zimba~ Afri~ 0.0562 -0.0381   7.21 0.596 0.458      2 -39.6  85.3  86.7
##  4 Zambia Afri~ 0.0598 -0.0342   4.53 0.636 0.444      2 -34.1  74.1  75.6
##  5 Swazi~ Afri~ 0.0682 -0.0250   6.64 0.732 0.412      2 -38.7  83.3  84.8
##  6 Lesot~ Afri~ 0.0849 -0.00666  5.93 0.927 0.358      2 -37.3  80.6  82.1
##  7 Cote ~ Afri~ 0.283   0.212    3.93 3.95  0.0748     2 -32.3  70.7  72.1
##  8 South~ Afri~ 0.312   0.244    4.74 4.54  0.0588     2 -34.6  75.2  76.7
##  9 Uganda Afri~ 0.342   0.276    3.19 5.20  0.0457     2 -29.8  65.7  67.1
## 10 Congo~ Afri~ 0.348   0.283    2.43 5.34  0.0434     2 -26.6  59.2  60.6
## # ... with 132 more rows, and 2 more variables: deviance <dbl>,
## #   df.residual <int>
```

Graph R^2

```
glance %>%
  ggplot(aes(continent, r.squared)) +
    geom_jitter(width = 0.5)
```

Graph R^2

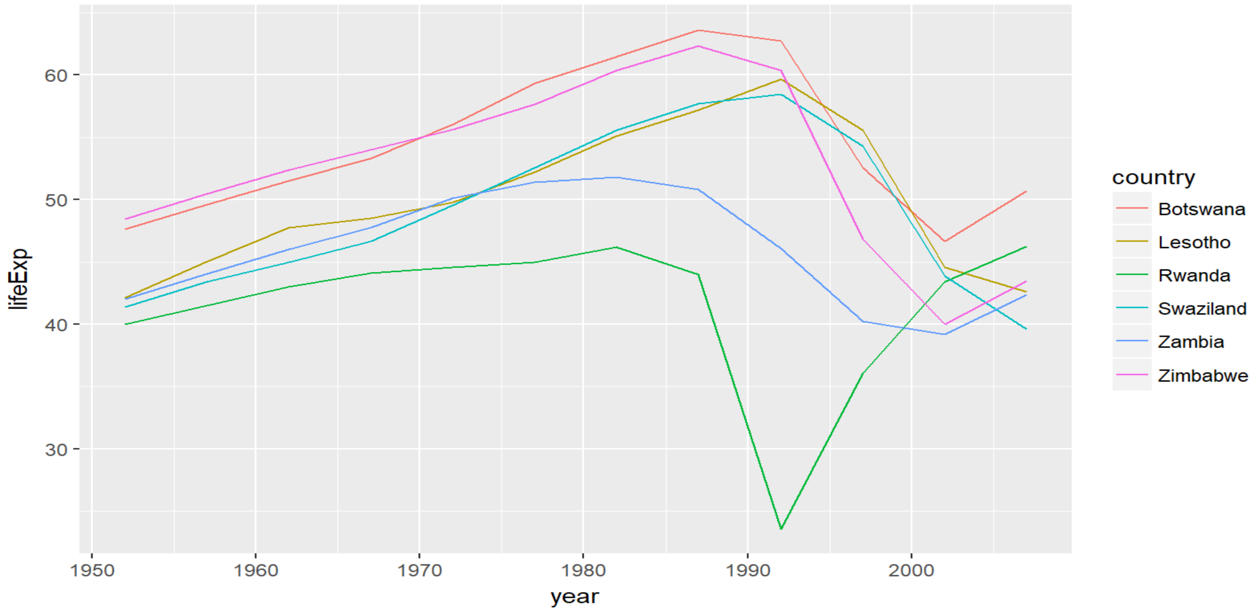

1.00
0.75
0.50
0.25
0.00
r.squared
Africa
Americas
Asia
Europe
Oceania
continent

Examining the worst Fits

```r
bad_fit <- filter(glance, r.squared < 0.25)

gapminder %>%
  semi_join(bad_fit, by = "country") %>%
  ggplot(aes(year, lifeExp, colour = country)) +
    geom_line()
```

Examining the worst Fits



What Do We See?

Issues with this?

- Our models do not all fit that well.
- We cannot interpret these models as the years are correlated with each other in each country.
- These models are also by country so we cannot get a specific average for the world from this.

Next Steps

- Begin to understand the correlation.
- Learn models that can fit this data all at once.
- Begin analyzing more complex data than just one variable by year .