MACHINE LEARNING IN THE TIDYVERSE

Exploring coefficients across models

Dmitriy (Dima) Gorenshteyn
Lead Data Scientist,
Memorial Sloan Kettering Cancer Center

77 models

```
gap_nested <- gapminder %>%
              group_by(country) %>%
              nest()
gap_models <- gap_nested %>%
              mutate(model = map(data, ~lm(life_expectancy~year, data = .x)))
```

```
gap_models
# A tibble: 77 x 3
   country    data              model
   <fct>      <list>            <list>
 1 Algeria    <tibble [52 × 6]> <S3: lm>
 2 Argentina  <tibble [52 × 6]> <S3: lm>
 3 Australia  <tibble [52 × 6]> <S3: lm>
 4 Austria    <tibble [52 × 6]> <S3: lm>
 5 Bangladesh <tibble [52 × 6]> <S3: lm>
 6 Belgium    <tibble [52 × 6]> <S3: lm>
```

Regression coefficients

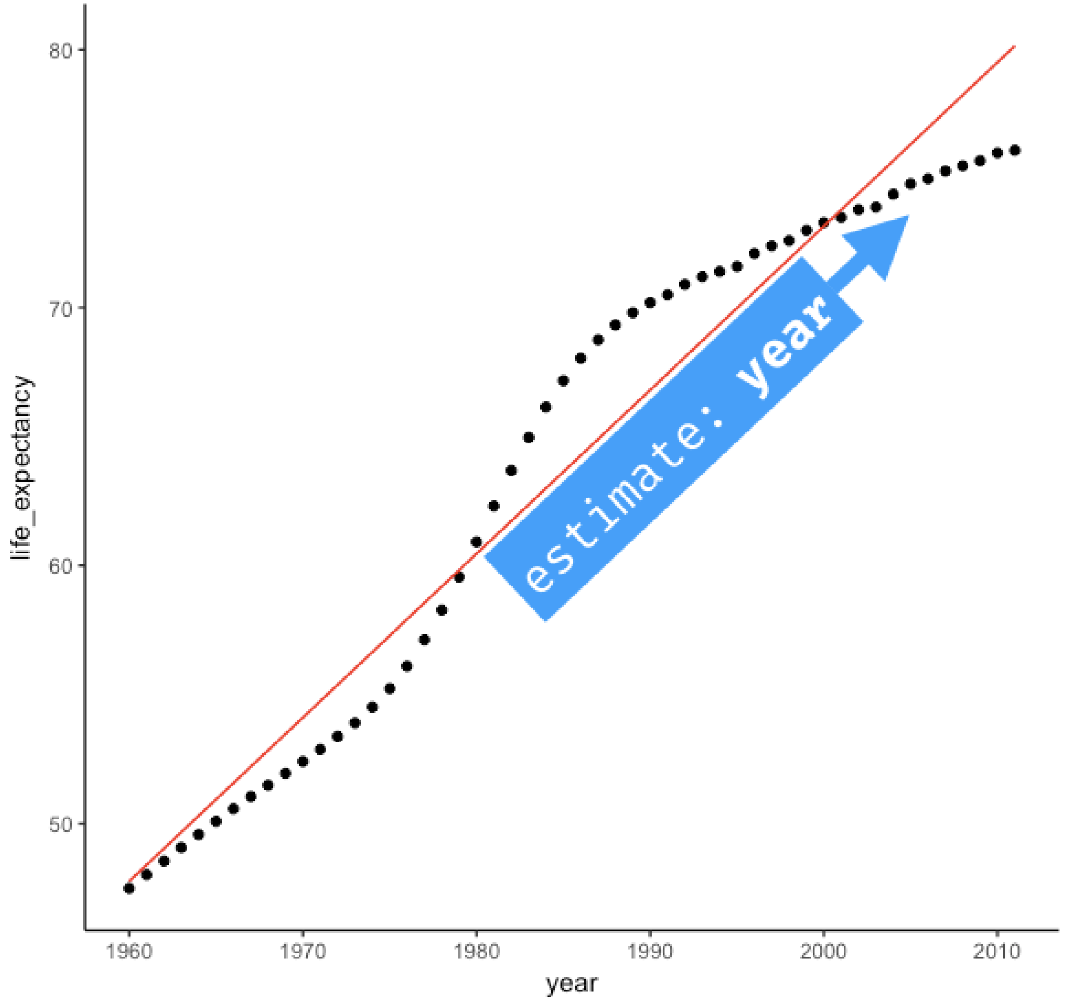
$$y = \alpha + \beta x$$

Regression coefficients

$$y = \alpha + \beta x$$

Life Expectancy	=	Term: **(intercept)**	+	Term: **year**	Year

```
tidy(gap_models$model[[1]])
         term      estimate      ...
1 (Intercept) -1196.5647772      ...
2        year     0.6348625      ...
```



Coefficients of multiple models

```
gap_models %>%
  mutate(coef = map(model, ~tidy(.x))) %>%
  unnest(coef)
```

```
# A tibble: 154 x 6
   country    term        estimate std.error statistic  p.value
   <fct>      <chr>          <dbl>     <dbl>     <dbl>    <dbl>
 1 Algeria    (Intercept) -1197     39.9         -30.0 1.32e-33
 2 Algeria    year            0.635  0.0201       31.6 1.11e-34
 3 Argentina  (Intercept) - 372      7.91        -47.0 4.66e-43
 4 Argentina  year            0.223  0.00398      56.0 8.78e-47
 5 Australia  (Intercept) - 429      9.37        -45.8 1.71e-42
 6 Australia  year            0.254  0.00472      53.9 5.83e-46
 7 Austria    (Intercept) - 415      8.04        -51.6 5.07e-45
 8 Austria    year            0.246  0.00405      60.8 1.48e-48
```

MACHINE LEARNING IN THE TIDYVERSE

Let's practice!

MACHINE LEARNING IN THE TIDYVERSE

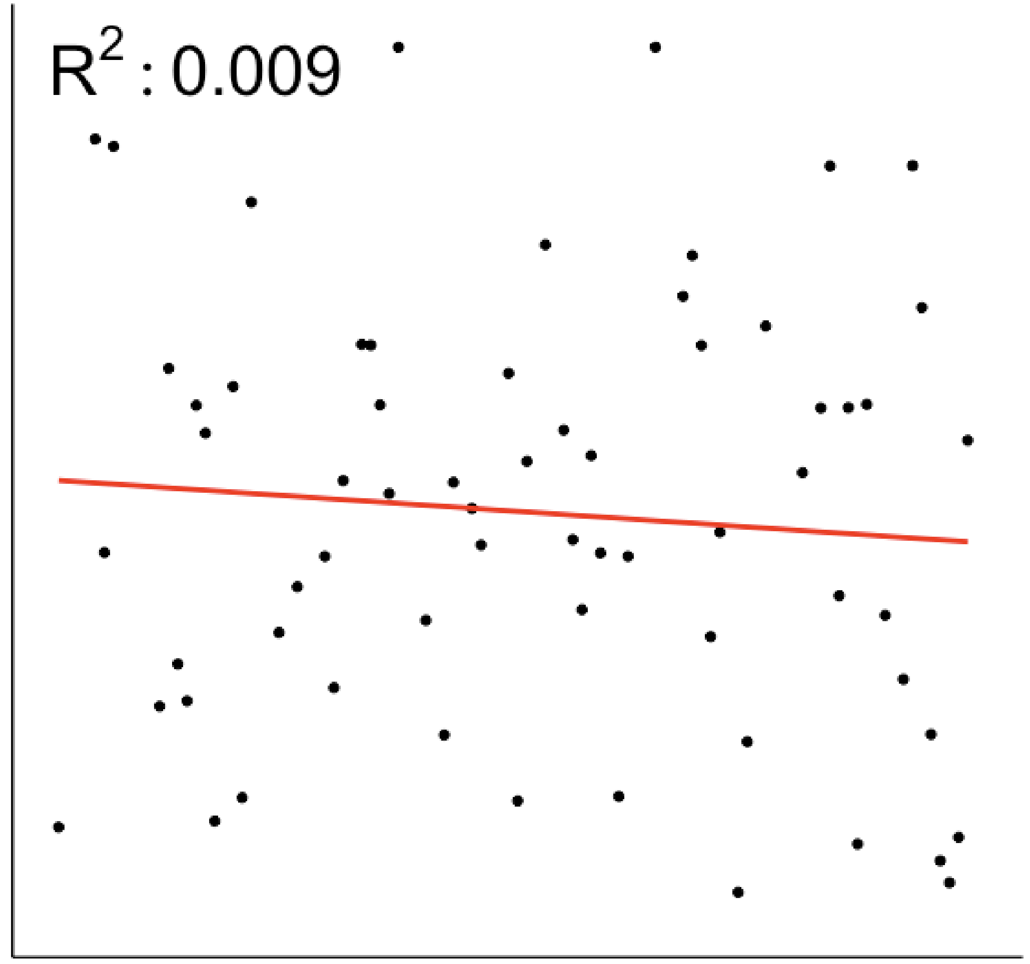
Evaluating the fit of many models

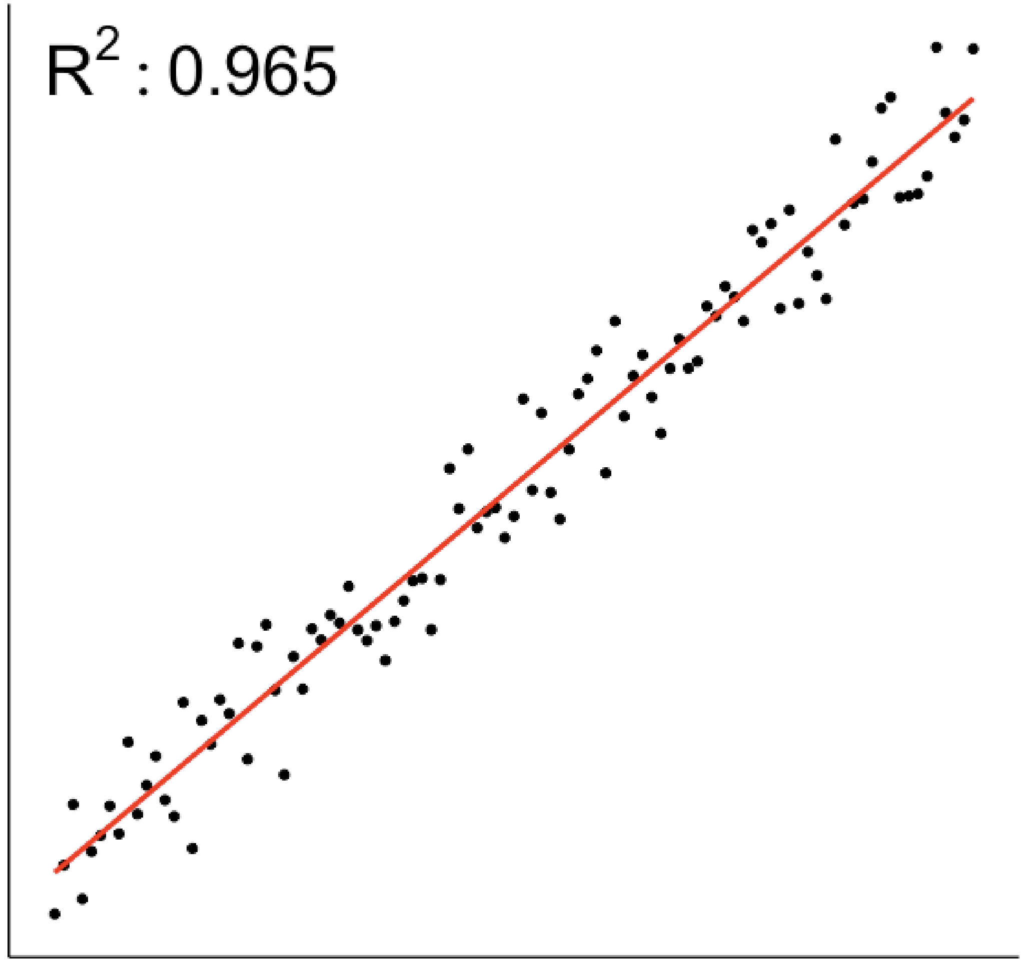
Dmitriy (Dima) Gorenshteyn
Lead Data Scientist,
Memorial Sloan Kettering Cancer Center

The fit of our models

$$R^2 = \frac{\%\ variation\ explained\ by\ the\ model}{\%\ total\ variation\ in\ the\ data}$$

The fit of our models





Glance across your models

```
model_perf <- gap_models %>%
  mutate(coef = map(model, ~glance(.x))) %>%
  unnest(coef)
```

```
model_perf

# A tibble: 77 x 14
   country  data   model r.squared adj.r.squared sigma statistic    ...
   <fct>    <lis> <lis>      <dbl>         <dbl> <dbl>     <dbl>    ...
 1 Algeria  <tib… <S3:…      0.952         0.951 2.18       996     ...
 2 Argenti… <tib… <S3:…      0.984         0.984 0.431     3137     ...
 3 Austral… <tib… <S3:…      0.983         0.983 0.511     2905     ...
 4 Austria  <tib… <S3:…      0.987         0.986 0.438     3702     ...
 5 Banglad… <tib… <S3:…      0.949         0.947 1.83       921     ...
 6 Belgium  <tib… <S3:…      0.990         0.990 0.331     5094     ...
 # ... with 71 more rows
```

Best & worst fitting models

```
model_perf %>%
  top_n(n = 2, wt = r.squared)

# A tibble: 2 x 14
  country data  model r.squared adj.r.squared sigma statistic
  <fct>   <lis> <lis>     <dbl>         <dbl> <dbl>     <dbl>
1 Canada  <tib… <S3:…     0.995         0.995 0.231     10117
2 Italy   <tib… <S3:…     0.997         0.997 0.226     15665
```

```
> model_perf %>%
    top_n(n = 2, wt = -r.squared)

# A tibble: 2 x 14
  country data  model r.squared adj.r.squared sigma statistic
  <fct>   <lis> <lis>     <dbl>         <dbl> <dbl>     <dbl>
1 Botswa~ <tib… <S3:…   0.0136       -0.00608  5.11     0.692
2 Lesotho <tib… <S3:…   0.00296      -0.0170   5.32     0.148
```

MACHINE LEARNING IN THE TIDYVERSE

Let's practice!

MACHINE LEARNING IN THE TIDYVERSE

Visually inspect the fit of your models

Dmitriy (Dima) Gorenshteyn
Lead Data Scientist,
Memorial Sloan Kettering Cancer Center

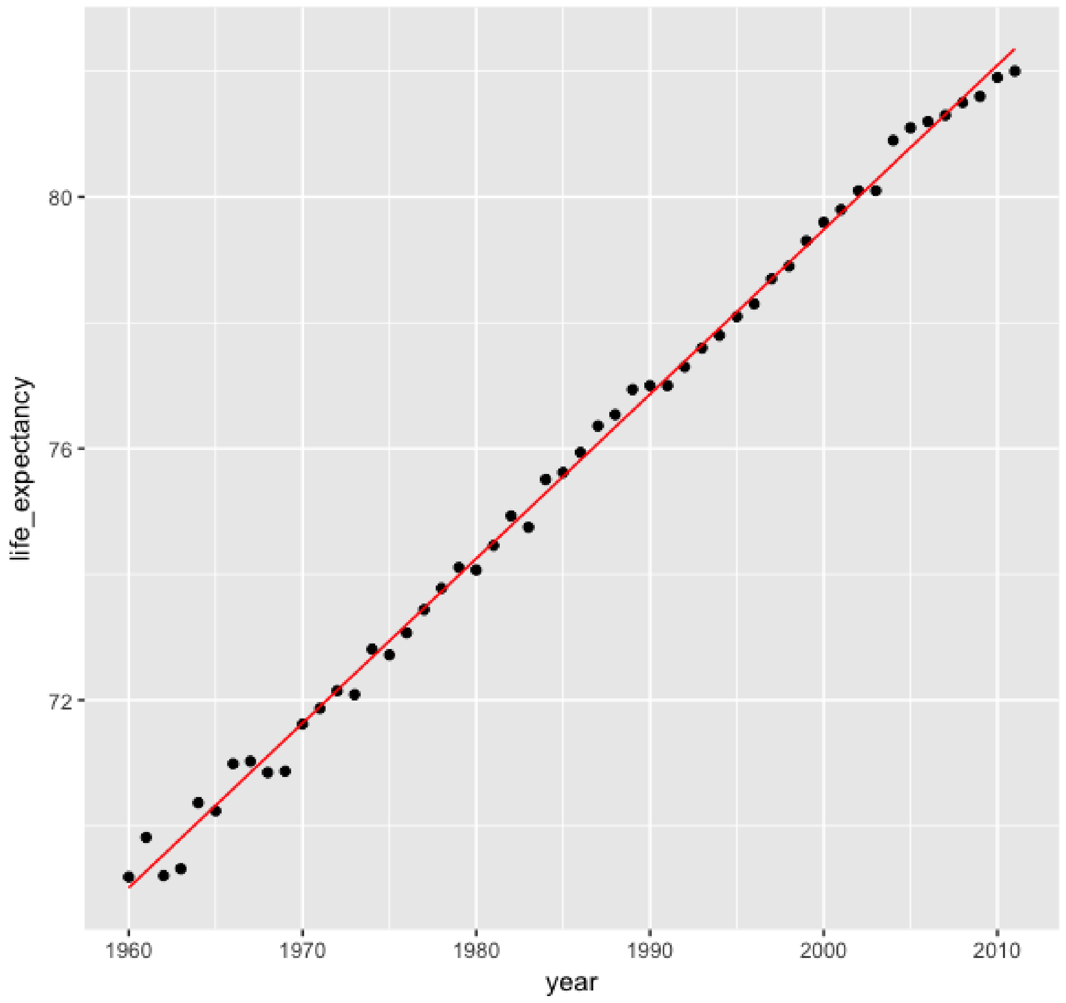
Building augmented datframes

```
augmented_models <- gap_models %>%
                    mutate(augmented = map(model, ~augment(.x))) %>%
                    unnest(augmented)
```

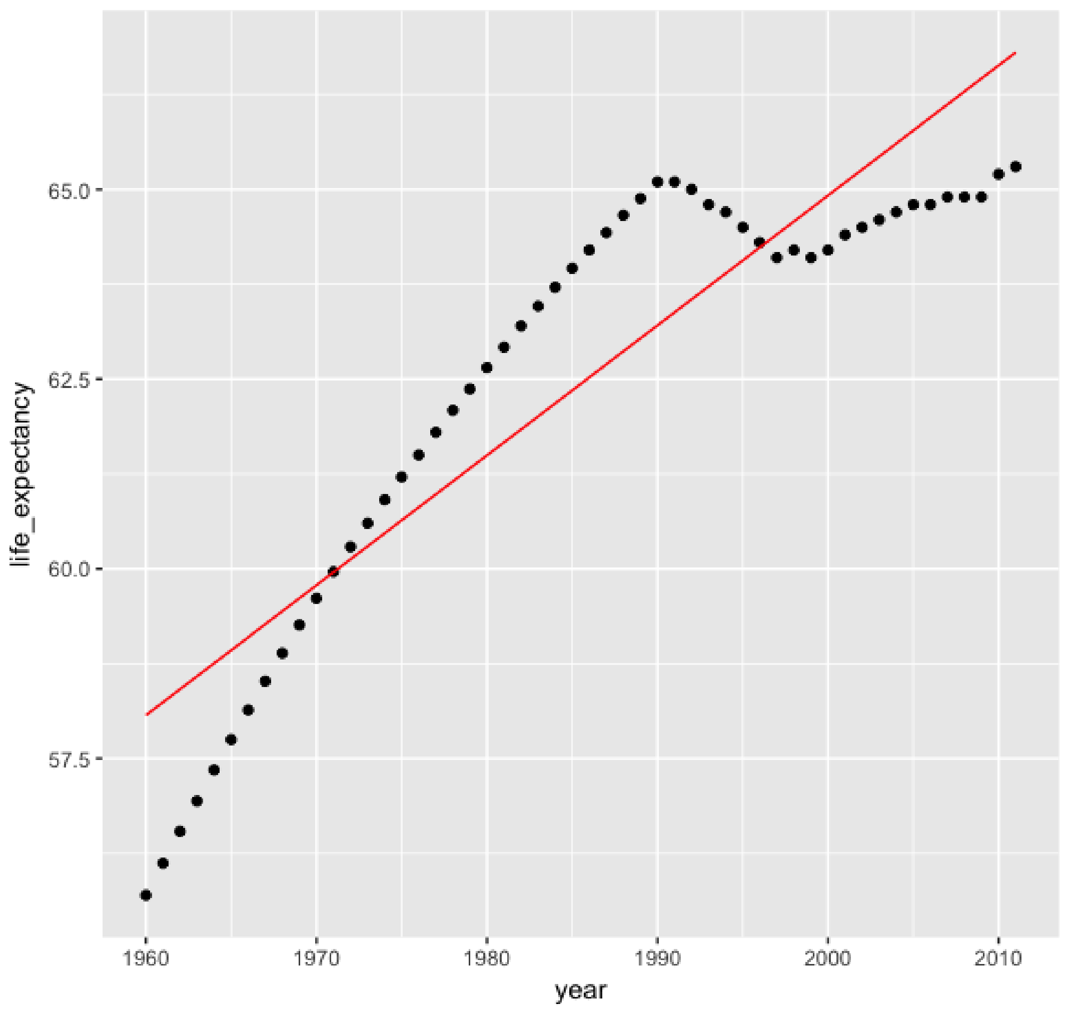
```
> augmented_models
# A tibble: 4,004 x 10
   country life_expectancy  year .fitted .se.fit .resid    .hat .sigma  ...
   <fct>             <dbl> <int>   <dbl>   <dbl>  <dbl>   <dbl>  <dbl>  ...
 1 Algeria            47.5  1960    47.8   0.595 -0.266 0.0747    2.20  ...
 2 Algeria            48.0  1961    48.4   0.578 -0.381 0.0705    2.20  ...
 3 Algeria            48.6  1962    49.0   0.561 -0.486 0.0664    2.20  ...
 4 Algeria            49.1  1963    49.7   0.544 -0.600 0.0625    2.20  ...
 5 Algeria            49.6  1964    50.3   0.527 -0.725 0.0587    2.20  ...
 6 Algeria            50.1  1965    50.9   0.511 -0.850 0.0551    2.20  ...
```

Model for Italy $R^2 : 0.99$

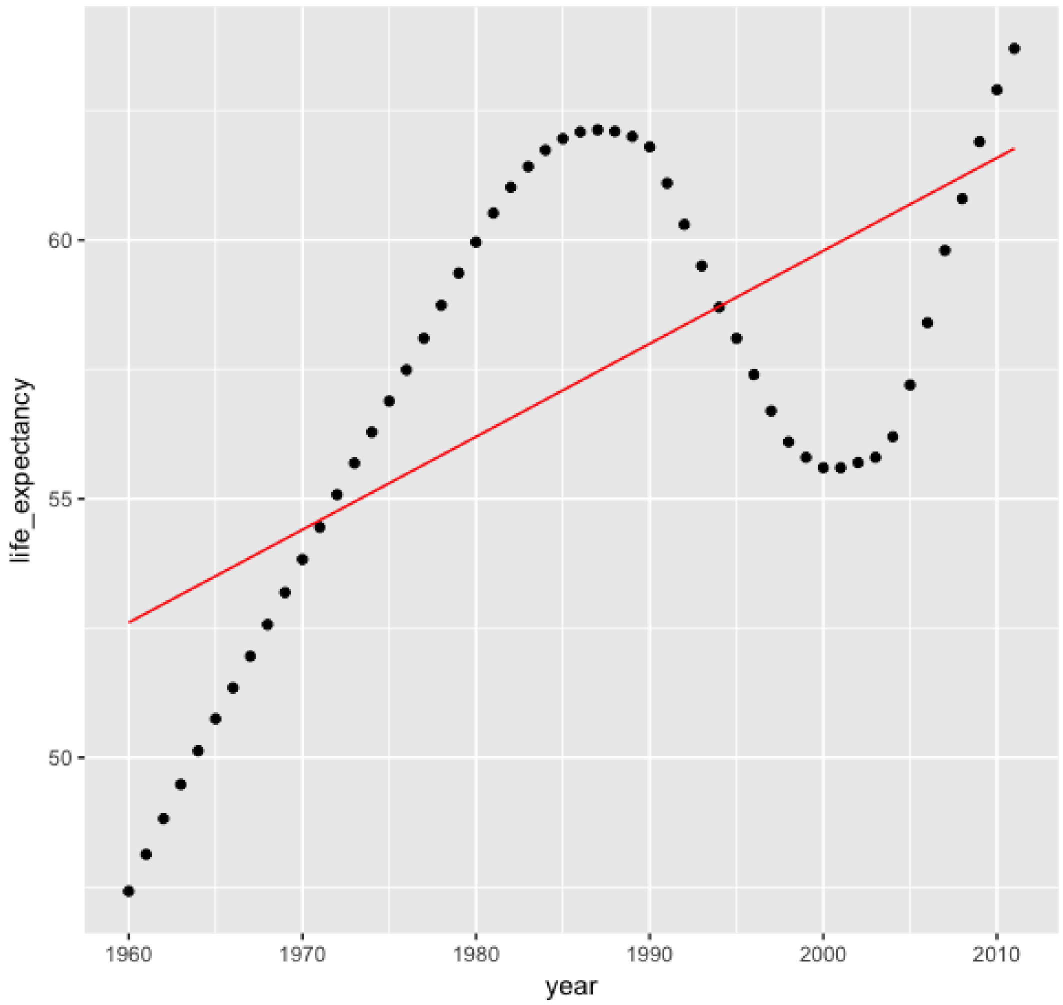
```
augmented_model %>% filter(country == "Italy") %>%
  ggplot(aes(x = year, y = life_expectancy)) +
  geom_point() +
  geom_line(aes(y = .fitted), color = "red")
```



Model for Fiji $R^2 : 0.82$



Model for Kenya $R^2 : 0.42$



MACHINE LEARNING IN THE TIDYVERSE

Let's practice!

MACHINE LEARNING IN THE TIDYVERSE

Improve the fit of your models

Dmitriy (Dima) Gorenshteyn
Lead Data Scientist,
Memorial Sloan Kettering Cancer Center

Multiple Linear Regression model

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots$$

Life Expectancy	=	Term: **(intercept)**	+	Term: **year**	Year	+	Term: **population**	Population	+	Term: **...**	...

Available Features: year, population, infant_mortality, fertility, gdpPercap

Using all features

Simple Linear Model: **life_expectancy ~ year**

```
gap_models <- gap_nested %>%
  mutate(model = map(data, ~lm(formula = life_expectancy ~ year, data = .x)))
```

Multiple Linear Model: **life_expectancy ~ year + population + ...**

Multiple Linear Model: **life_expectancy ~ .**

```
gap_fullmodels <- gap_nested %>%
  mutate(model = map(data, ~lm(formula = life_expectancy ~ ., data = .x)))
```

Using broom with Multiple Linear Regression models

```
tidy(gap_fullmodels$model[[1]])
              term      estimate    std.error  statistic      p.value
1      (Intercept) -1.830195e+03 1.502271e+02 -12.182848 5.325478e-16
2             year  9.814091e-01 7.800580e-02  12.581232 1.693870e-16
3 infant_mortality -1.603504e-01 4.021732e-03 -39.870986 2.525847e-37
4        fertility -2.600935e-01 1.648652e-01  -1.577614 1.215074e-01
5       population -1.611437e-06 1.704374e-07  -9.454716 2.347590e-12
6        gdpPercap -1.797662e-03 4.878209e-04  -3.685086 6.008755e-04
```

```
augment(gap_fullmodels$model[[1]])
   life_expectancy year infant_mortality fertility population ...   .fitted
1            47.50 1960            148.2      7.65   11124892 ...  47.45394
2            48.02 1961            148.1      7.65   11404859 ...  48.35078
3            48.55 1962            148.2      7.65   11690152 ...  49.26449
...            ...  ...              ...       ...        ... ...  ...
```

```
glance(gap_fullmodels$model[[1]])
  r.squared adj.r.squared     sigma statistic      p.value df    logLik ...
1 0.9990732     0.9989724 0.3160595  9917.133 1.562325e-68  6 -10.70225 ...
```

Adjusted R^2

```
glance(gap_fullmodels$model[[1]])

  r.squared adj.r.squared     sigma statistic      p.value df    logLik ...
1 0.9990732     0.9989724 0.3160595  9917.133 1.562325e-68  6 -10.70225 ...
```

MACHINE LEARNING IN THE TIDYVERSE

Let's practice!