Communicating with the Tidyverse with ISO data

Alexa Kelly

7/2/2020

In this RMarkdown, we will explore: 1. The relationship between weekly working hours and monetary compensation in European countries, according to the International Labour Organization (ILO).

2. Communicating with the tidyverse. They say that a picture is worth a thousand words. Successfully promoting your data analysis is not only a matter of accurate and effective graphics, but also of aesthetics and uniqueness.

More specifically, it will show how to leverage the power of ggplot2 themes for producing publication-quality graphics that stick out from the mass of boilerplate plots out there. And, how to tweak and get the most out of ggplot2 in order to produce unconventional plots that draw attention on social media.

In the end, combining this knowledge will produce a slick and custom-styled report with RMarkdown and CSS – all of that within the powerful tidyverse.

```r
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: tidyverse

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

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

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

```r
if(!require(forcats)) install.packages("forcats", repos = "http://cran.us.r-project.org")

if(!require(ggplot2)) install.packages("ggplot2", repos = "http://cran.us.r-project.org")

if(!require(dplyr)) install.packages("dplyr")


library(dplyr)
library(forcats)
library(ggplot2)
library(tidyverse)

load(url("http://s3.amazonaws.com/assets.datacamp.com/production/course_5807/datasets/ilo_data.RData"))
```

```r
# Count the resulting rows and examine
ilo_data  %>% count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   380
```

```
ilo_data
```

```
## # A tibble: 380 x 4
##    country        year  hourly_compensation working_hours
##    <fct>          <fct>               <dbl>         <dbl>
##  1 Finland        1980                 8.61          35.6
##  2 France         1980                 8.90          35.4
##  3 Italy          1980                 8.09          35.7
##  4 Norway         1980                11.8           30.4
##  5 Spain          1980                 5.86          36.8
##  6 Sweden         1980                12.4           29.2
##  7 Switzerland    1980                11.0           34.7
##  8 United Kingdom 1980                 7.52          34.1
##  9 Finland        1981                 8.33          35.7
## 10 France         1981                 7.98          34.8
## # ... with 370 more rows
```

The loaded data contains 380 rows.

Can visualizations reveal a correlation between the relationship between weekly working hours and monetary compensation in European countries within the datatset?

```
# Turn year and country into a factor
ilo_data_corrected <- ilo_data %>% mutate(year = as.factor(as.numeric(year)),country = as.factor(countr

ilo_data_corrected # See the results
```

```
## # A tibble: 380 x 4
##    country        year  hourly_compensation working_hours
##    <fct>          <fct>               <dbl>         <dbl>
##  1 Finland        1                    8.61          35.6
##  2 France         1                    8.90          35.4
##  3 Italy          1                    8.09          35.7
##  4 Norway         1                   11.8           30.4
##  5 Spain          1                    5.86          36.8
##  6 Sweden         1                   12.4           29.2
##  7 Switzerland    1                   11.0           34.7
##  8 United Kingdom 1                    7.52          34.1
##  9 Finland        2                    8.33          35.7
## 10 France         2                    7.98          34.8
## # ... with 370 more rows
```

```
#Filter for European countries, that also have sufficient data:
european_countries  <- c("Finland","France","Italy","Norway","Spain","Sweden","Switzerland","United Kin

european_countries  # Examine the European countries vector
```

```
##  [1] "Finland"        "France"         "Italy"          "Norway"
##  [5] "Spain"          "Sweden"         "Switzerland"    "United Kingdom"
##  [9] "Belgium"        "Ireland"        "Luxembourg"     "Portugal"
## [13] "Netherlands"    "Germany"        "Hungary"        "Austria"
## [17] "Czech Rep."
```

```
# Only retain European countries
ilo_data <- ilo_data %>% filter(country %in% european_countries)
str(ilo_data) # Examine the structure of ilo_data
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    380 obs. of  4 variables:
##  $ country             : Factor w/ 30 levels "Australia","Austria",..: 8 9 15 22 25 27 28 29 8 9 ...
##  $ year                : Factor w/ 27 levels "1980","1981",..: 1 1 1 1 1 1 1 1 2 2 ...
##  $ hourly_compensation: num  8.61 8.9 8.09 11.8 5.86 ...
##  $ working_hours      : num  35.6 35.4 35.7 30.4 36.8 ...
```

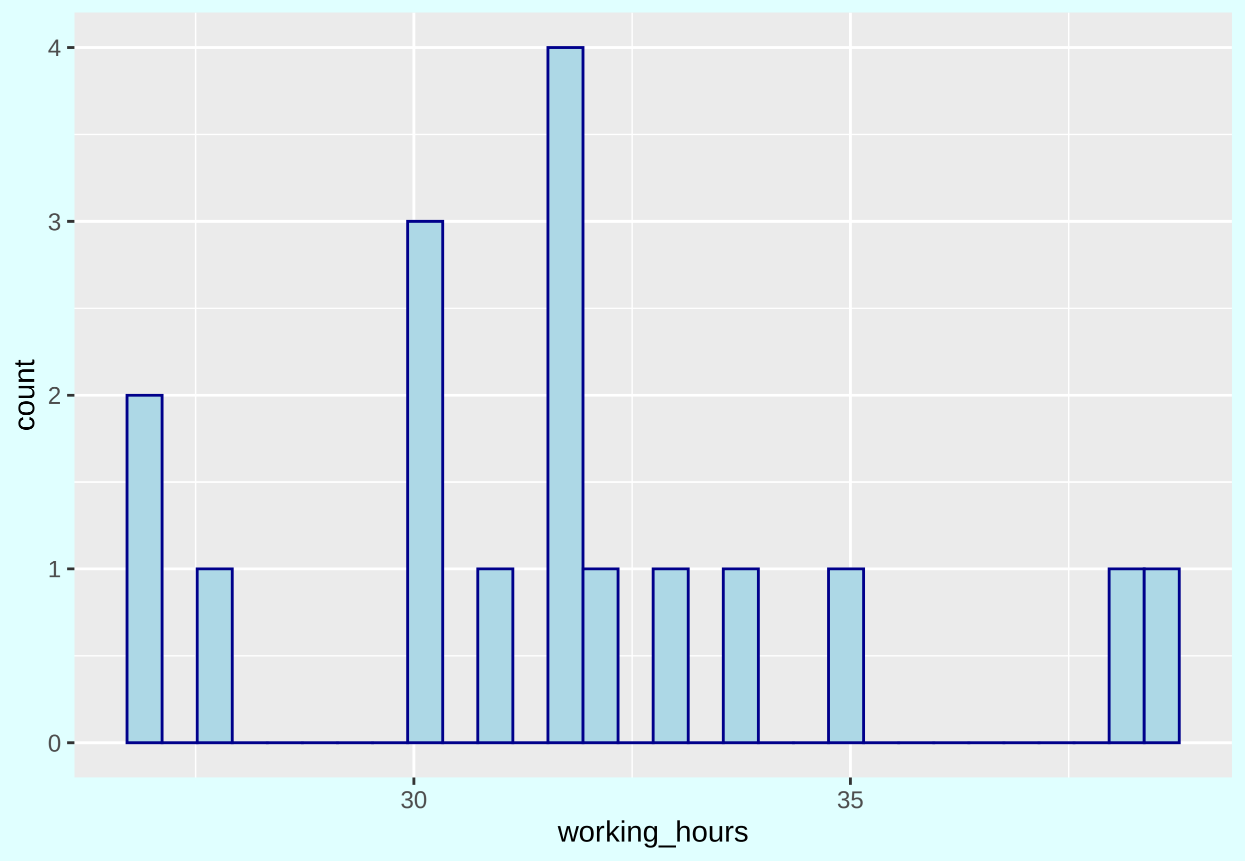
```
# Group and summarize the data
ilo_data %>%
  group_by(year) %>%
  summarize(mean_hourly_compensation = mean(hourly_compensation),
            mean_working_hours = mean(working_hours))
```

```
## # A tibble: 27 x 3
##    year  mean_hourly_compensation mean_working_hours
##    <fct>                    <dbl>              <dbl>
##  1 1980                      9.27               34.0
##  2 1981                      8.69               33.6
##  3 1982                      8.36               33.5
##  4 1983                      7.81               33.9
##  5 1984                      7.54               33.7
##  6 1985                      7.79               33.7
##  7 1986                      9.70               34.0
##  8 1987                     12.1                33.6
##  9 1988                     13.2                33.7
## 10 1989                     13.1                33.5
## # ... with 17 more rows
```

Let's take a look at what the distribution for working_hours looks like:

```
plot_data <- ilo_data %>%
filter(year == 2006)
ggplot(plot_data) + geom_histogram(
aes(x = working_hours), fill="lightblue", color="darkblue")  +
  theme(
    plot.background = element_rect(fill = "lightcyan1"))
```

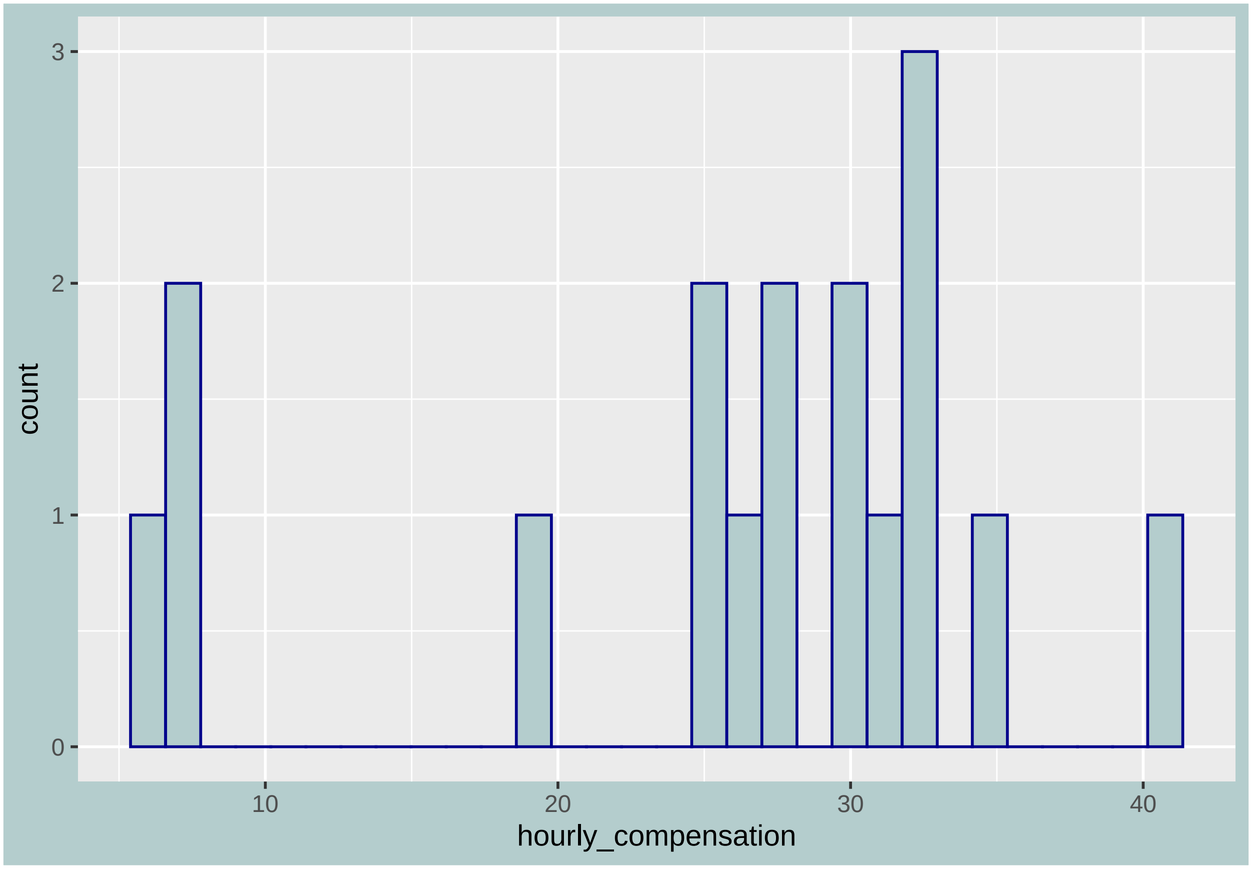
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



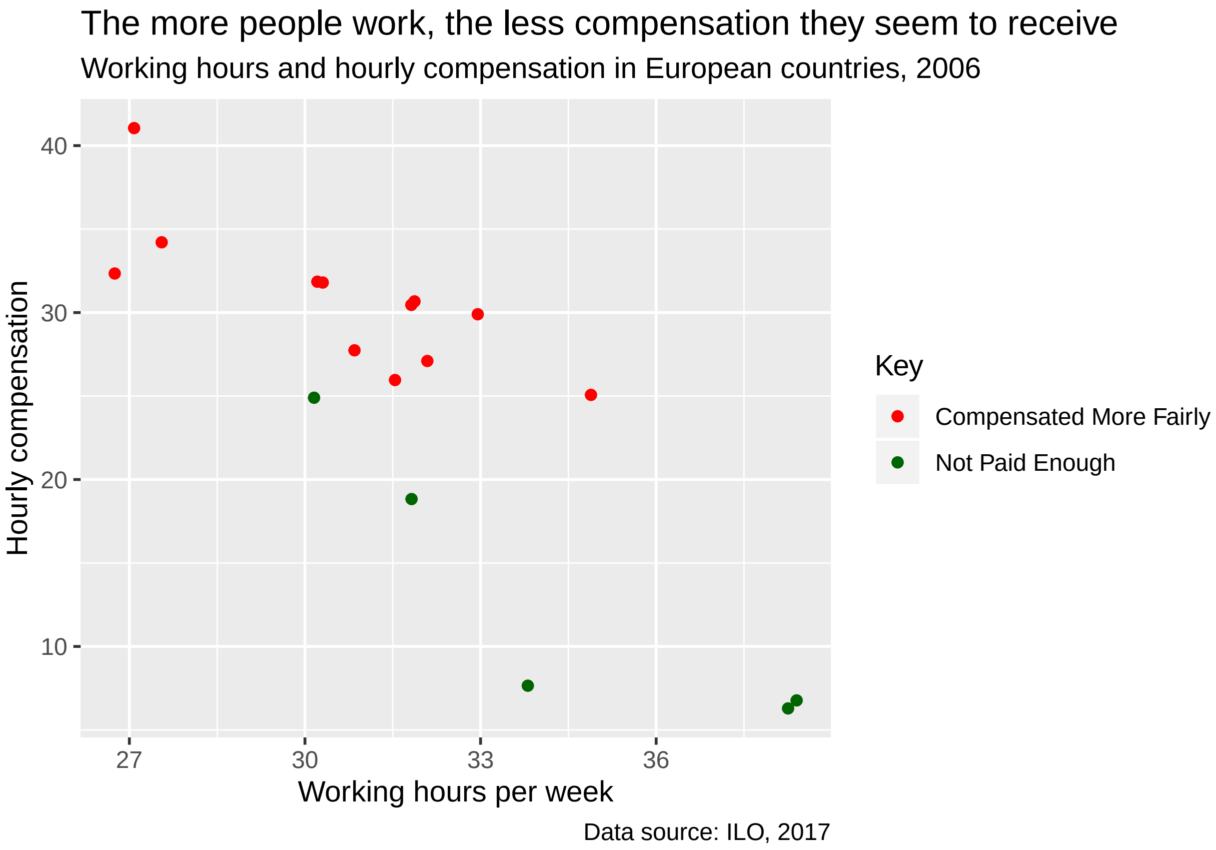
Let’s take a look at what the distribution for hourly_compensation looks like:

```
plot_data <- ilo_data %>%
filter(year == 2006)
ggplot(plot_data) + geom_histogram(
aes(x = hourly_compensation), fill="lightcyan3", color="darkblue") +
  theme(
    plot.background = element_rect(fill = "lightcyan3"))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
# Filter for 2006
plot_data <- ilo_data %>% filter(year %in% 2006)

# Add labels and save the scatter plot into a variable: ilo_plot
ilo_plot <- ggplot(plot_data) +
  geom_point(aes(x = working_hours, y = hourly_compensation, color = ifelse(hourly_compensation < 25, "
  labs(
    x = "Working hours per week",
    y = "Hourly compensation",
    title = "The more people work, the less compensation they seem to receive",
    subtitle = "Working hours and hourly compensation in European countries, 2006",
    caption = "Data source: ILO, 2017"
  ) +
  scale_color_manual(name="Key", values = c("red","darkgreen"))


ilo_plot
```

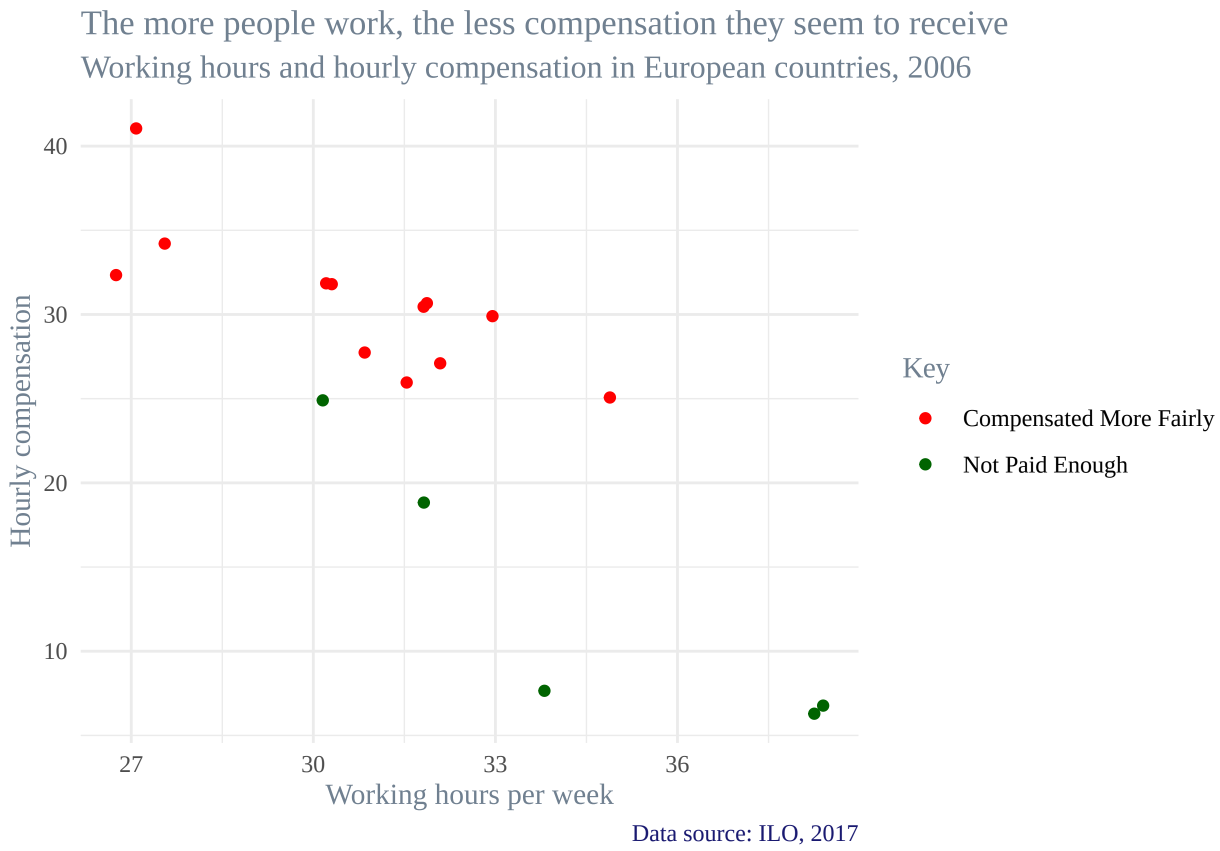


The more people work, the less compensation they seem to receive

Working hours and hourly compensation in European countries, 2006

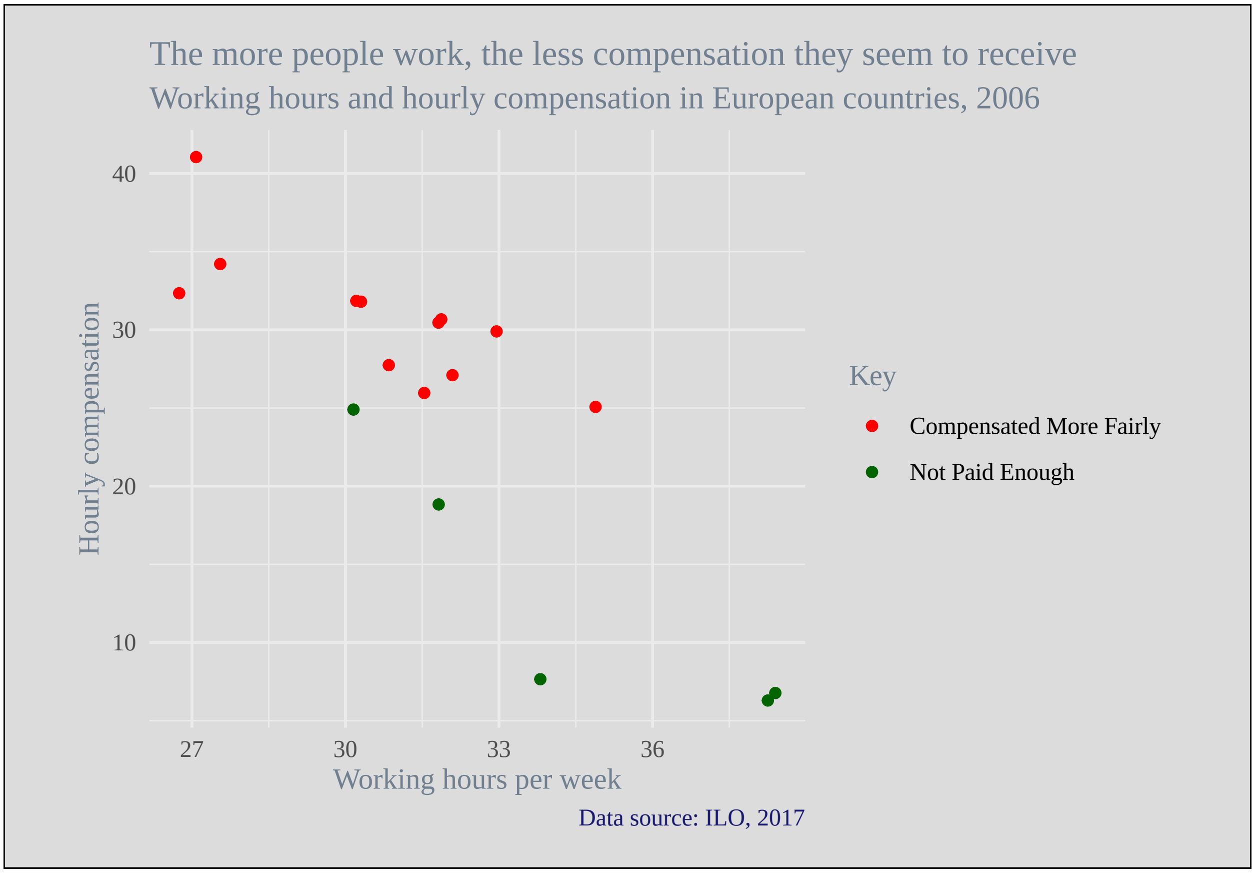
Data source: ILO, 2017

```
ilo_plot <- ilo_plot + theme_minimal() +
  # Customize the "minimal" theme with another custom "theme" call
  theme(
    text = element_text(family = "serif"),
    title = element_text(color = "slategrey"),
    plot.caption = element_text(color = "midnightblue"),
    plot.subtitle = element_text(size = 12)
        )

ilo_plot # Render the plot object
```



“theme” calls can be stacked upon each other, so this is already the third call of “theme”.

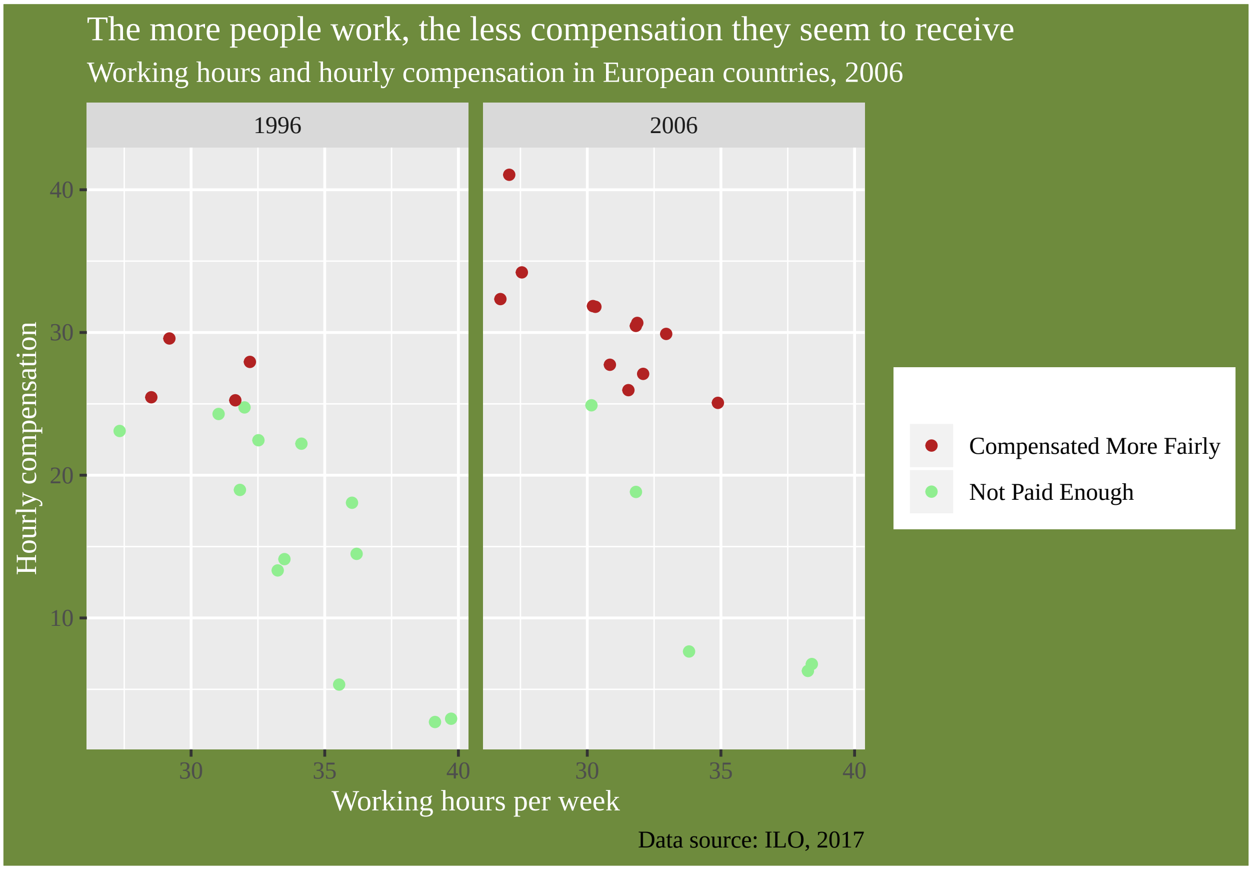
```
ilo_plot +
  theme(
    plot.background = element_rect(fill = "gainsboro"),
    plot.margin = unit(c(5, 10, 5, 10), units = "mm"))
```



```
 # Filter ilo_data to retain the years 1996 and 2006
ilo_data <- ilo_data %>% filter(year=="1996" | year=="2006")
```

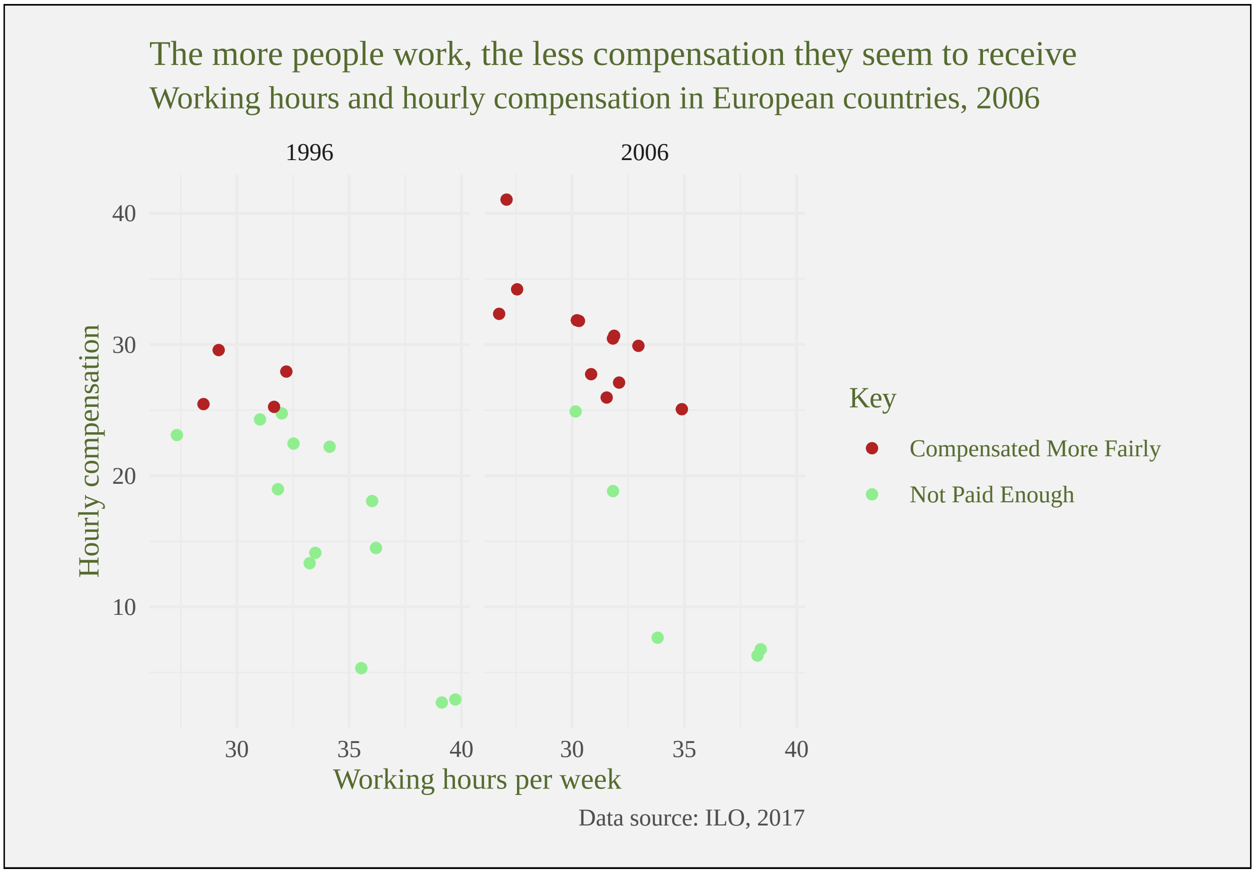
The data is now filtered so it only contains the years 1996 and 2006 – a good time range for comparison.

```
# Again, save the plot object into a variable so you can save typing later on
ilo_plot <- ggplot(ilo_data, aes(x = working_hours, y = hourly_compensation)) +
 geom_point(aes(color = ifelse(hourly_compensation < 25, "Not Paid Enough", "Compensated More Fairly")
 labs(
  x = "Working hours per week",
  y = "Hourly compensation",
  title = "The more people work, the less compensation they seem to receive",
  subtitle = "Working hours and hourly compensation in European countries, 2006",
  caption = "Data source: ILO, 2017"
 ) + theme(
    text = element_text(family = "serif"),
  title = element_text(color = "white"),
  plot.caption = element_text(color = "black"), plot.background = element_rect(fill = "darkolivegreen
      )+
 # Add facets here
facet_grid(facets = . ~ year) +
 scale_color_manual(name="Key", values = c("firebrick","lightgreen"))


ilo_plot
```

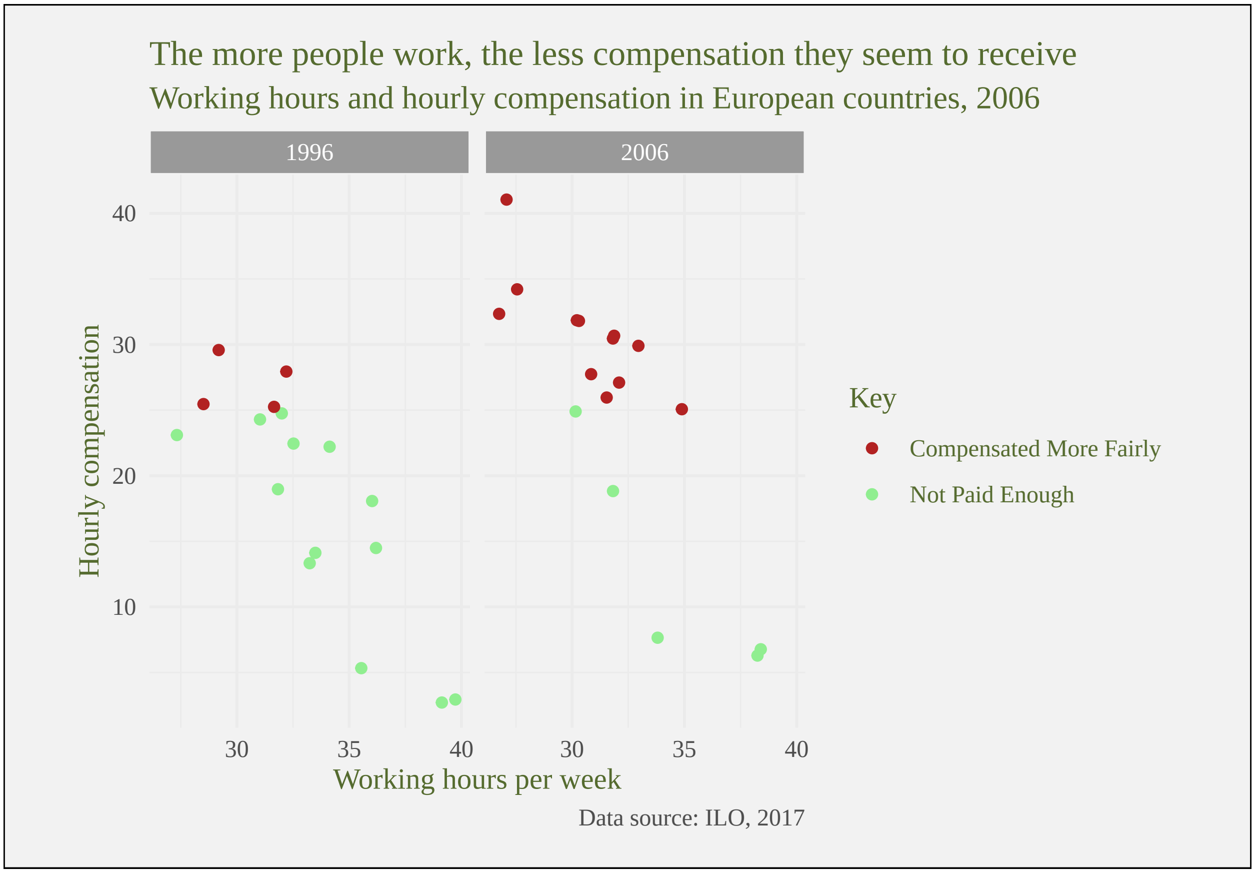


Another possible design, using a theme function:

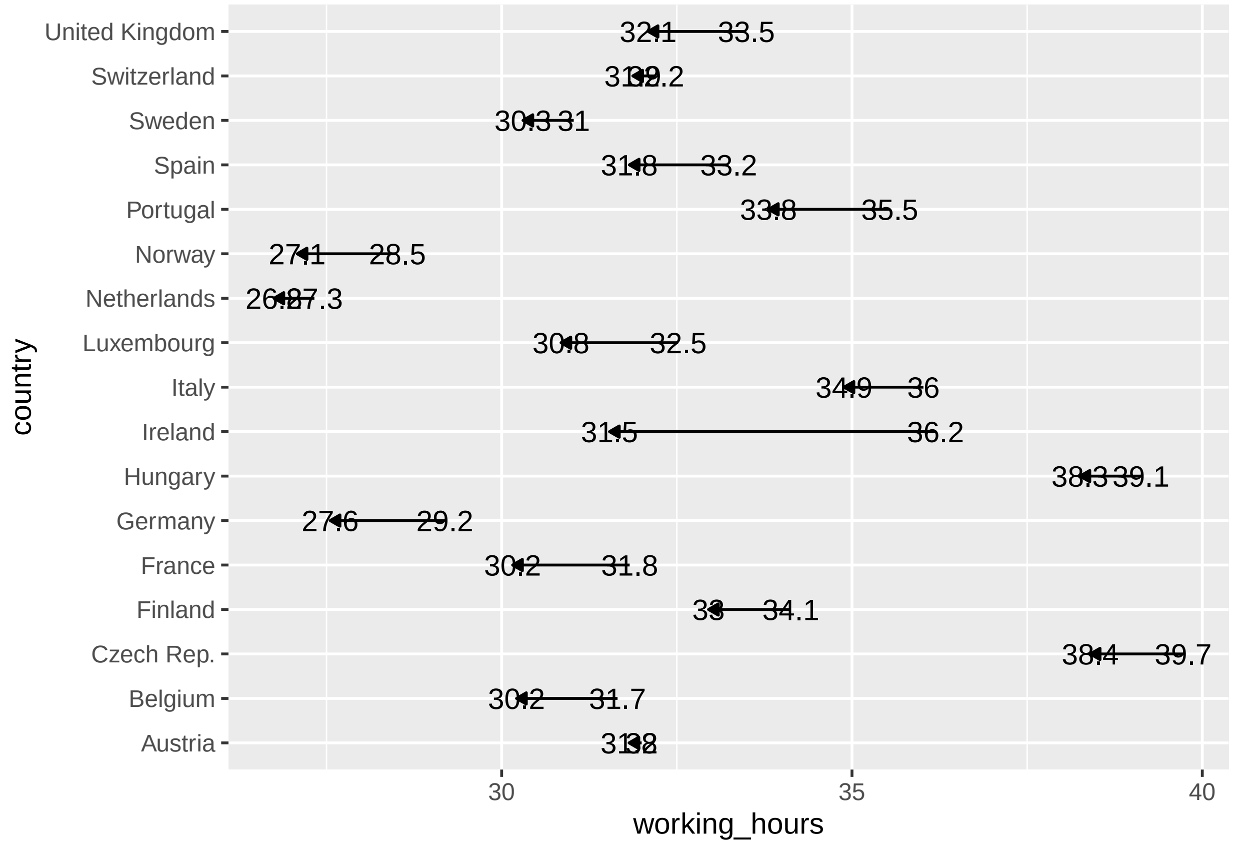
```
# Define theme function below
theme_ilo <- function() {
    theme_minimal() +
    theme(
    text = element_text(family = "serif", color = "darkolivegreen"),
    plot.subtitle = element_text(size = 12),
    plot.caption = element_text(color = "gray30"),
    plot.background = element_rect(fill = "gray95"),
    plot.margin = unit(c(5, 10, 5, 10), units = "mm")
  )
}

# Apply the theme function
ilo_plot <- ilo_plot + theme_ilo()
ilo_plot # Examine ilo_plot
```



```
ilo_plot +
  # Add another theme call
  theme(
    # Change the background fill and color
    strip.background = element_rect(fill= "gray60", color="gray95"),
    # Change the color of the text
    strip.text = element_text(color="white")
  )
```

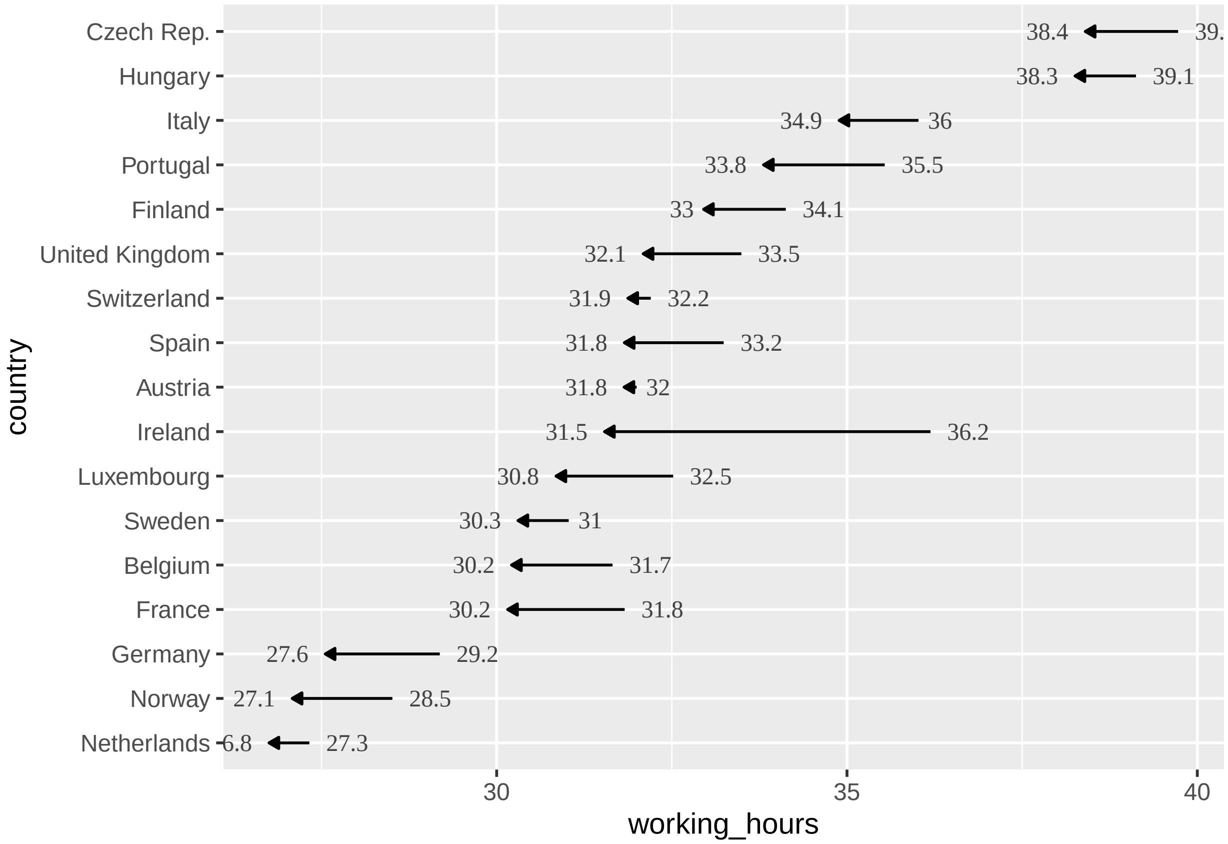


```
ggplot(ilo_data) +
  geom_path(aes(x = working_hours, y = country),
            # Add an arrow to each path
            arrow = arrow(length = unit(1.5, "mm"), type = "closed")) +
  # Add a geom_text() geometry
  geom_text(
          aes(x = working_hours,
              y = country,
              label = round(working_hours, 1)))
```

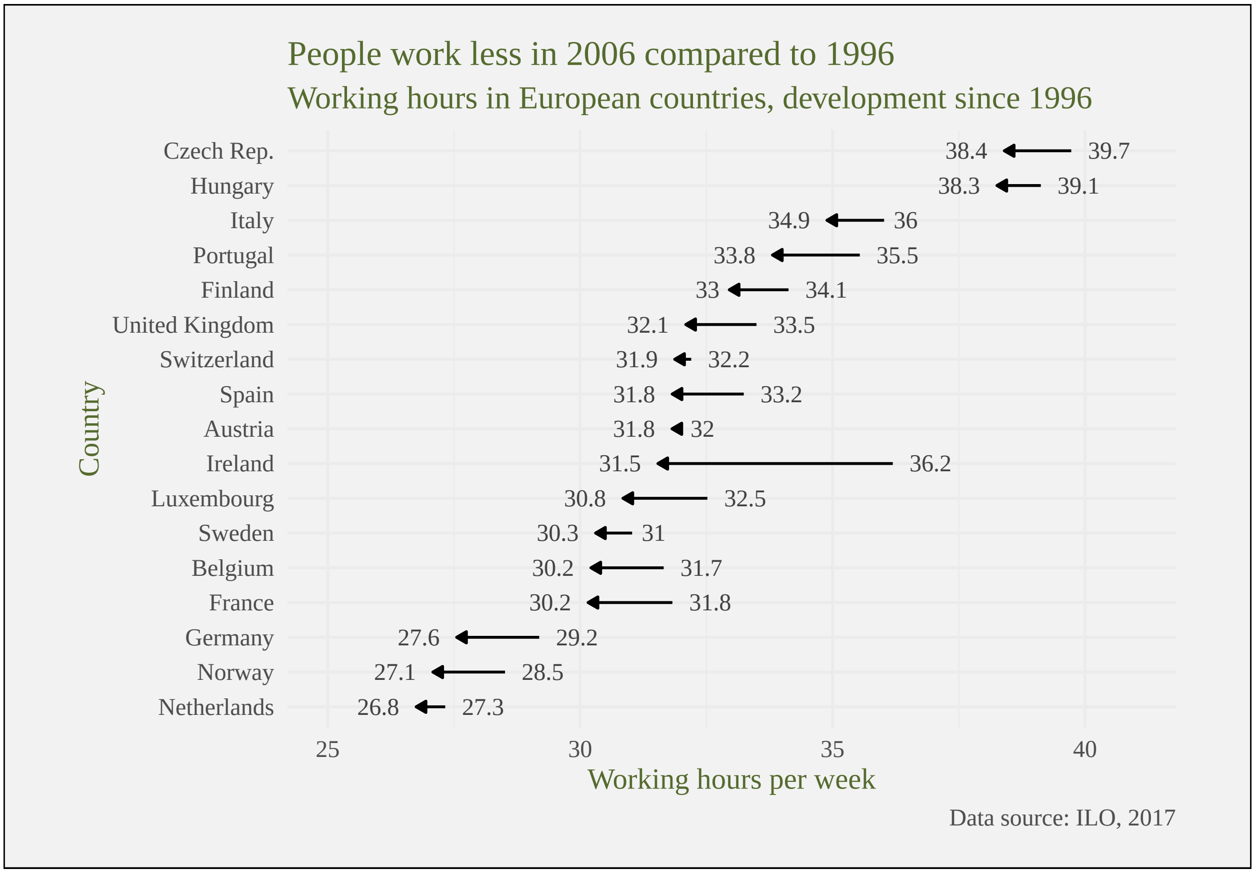


```
# Reorder country factor levels
ilo_data <- ilo_data %>%
  # Arrange data frame
  arrange(year) %>%
  # Reorder countries by working hours in 2006
  mutate(country = fct_reorder(country, working_hours, last))
```

```
# plot again, and save plot into an object for reuse
ilo_dot_plot <- ggplot(ilo_data) +
  geom_path(aes(x = working_hours, y = country),
            arrow = arrow(length = unit(1.5, "mm"), type = "closed")) +
    # Specify the hjust aesthetic with a conditional value
    geom_text(
          aes(x = working_hours,
              y = country,
              label = round(working_hours, 1),
              hjust = ifelse(year == "2006", 1.4, -0.4)
            ),
          # Change the appearance of the text
          size = 3,
          family = "serif",
          color = "gray25"
          )

ilo_dot_plot
```



```
# Reuse ilo_dot_plot
ilo_dot_plot <- ilo_dot_plot +
  # Add labels to the plot
  labs(
    x = "Working hours per week",
    y = "Country",
    title = "People work less in 2006 compared to 1996",
    subtitle = "Working hours in European countries, development since 1996",
    caption = "Data source: ILO, 2017"
  ) +
  # Apply your theme
  theme_ilo() +
  # Change the viewport
  coord_cartesian(xlim = c(25, 41))

# View the plot
ilo_dot_plot
```



Data source: ILO, 2017

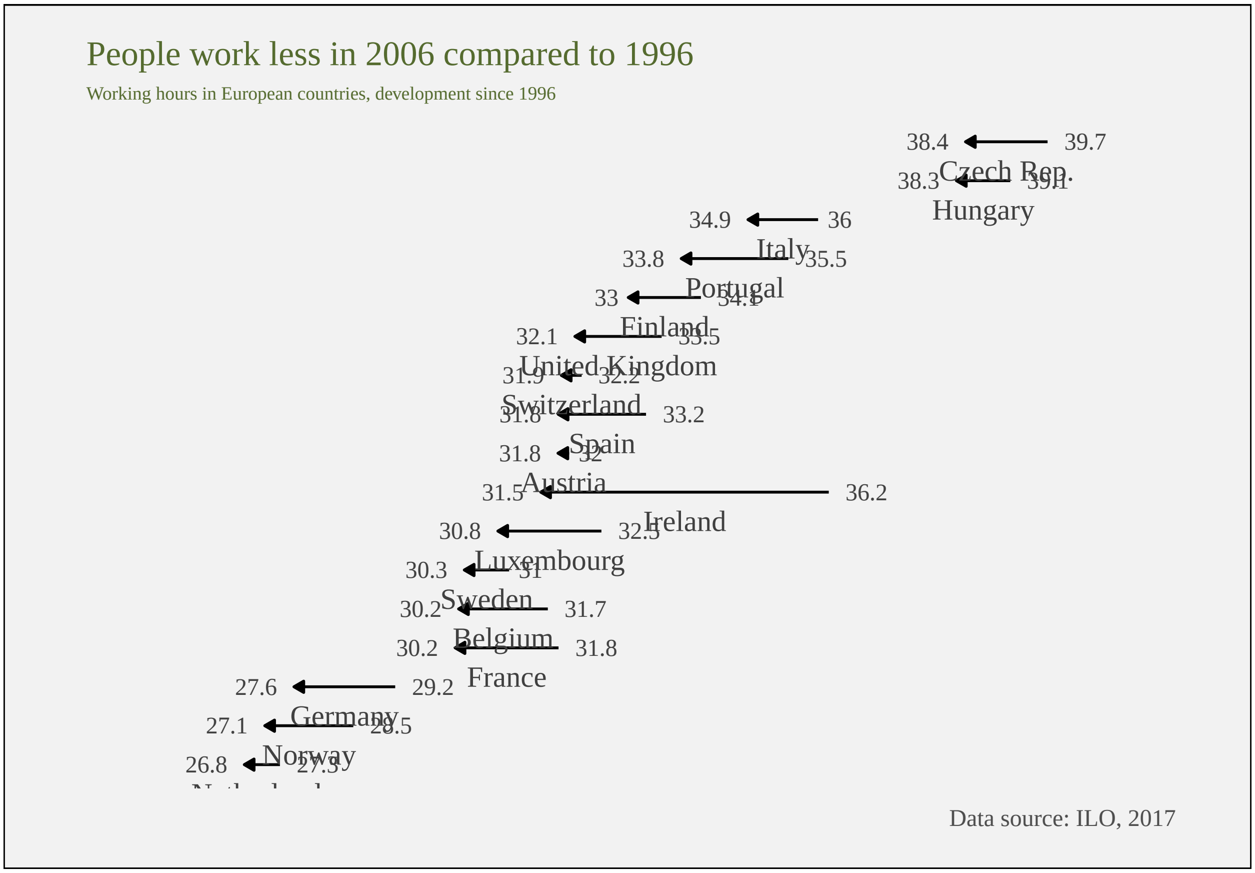
The Reduction in Weekly Working Hours in Europe.

Results… In the following, a plot that shows the reduction of weekly working hours from 1996 to 2006 in each country is produced.

```
# Compute temporary data set for optimal label placement
median_working_hours <- ilo_data %>%
  group_by(country) %>%
  summarize(median_working_hours_per_country = median(working_hours)) %>%
  ungroup()

# Have a look at the structure of this data set
str(median_working_hours)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    17 obs. of  2 variables:
##  $ country                         : Factor w/ 30 levels "Netherlands",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ median_working_hours_per_country: num  27 27.8 28.4 31 30.9 ...
```

```
ilo_dot_plot +
  # Add label for country
  geom_text(data = median_working_hours,
            aes(y = country, x = median_working_hours_per_country, label =country),
            vjust = 2,
            family = "Times",
            color = "gray25") +
  # Remove axes and grids
  theme(
    axis.ticks = element_blank(),
    axis.title = element_blank(),
    axis.text = element_blank(),
    panel.grid = element_blank(),
    # Also, let's reduce the font size of the subtitle
    plot.subtitle = element_text(size = 7)
  )
```



People work less in 2006 compared to 1996

Working hours in European countries, development since 1996

Data source: ILO, 2017