L01 Introduction

Data Visualization (STAT 302)

Shay Lebovitz

Contents

Overview

The goals of this lab are to (1) ensure that the major software for this course is properly installed and functional, (2) develop and follow a proper workflow, and (3) work together to construct a few plots to explore a dataset using `ggplot2` — demonstration of the utility and power of `ggplot2`.

Don't worry if you cannot do everything here by yourself. You are just getting started and the learning curve is steep, but remember that the instructional team and your classmates will be there to provide support. Persevere and put forth an honest effort and this course will payoff.

```
library(tidyverse)
library(skimr)
```

Dataset

We'll be using data from the **lego** package which is already in the `/data` subdirectory, along with many other processed datasets, as part of the zipped folder for this lab.

Tasks

Complete the following tasks. For many of these you'll need to simply indicate that you have completed the task. In others, you'll need to run some R code and/or supply a sentence or two.

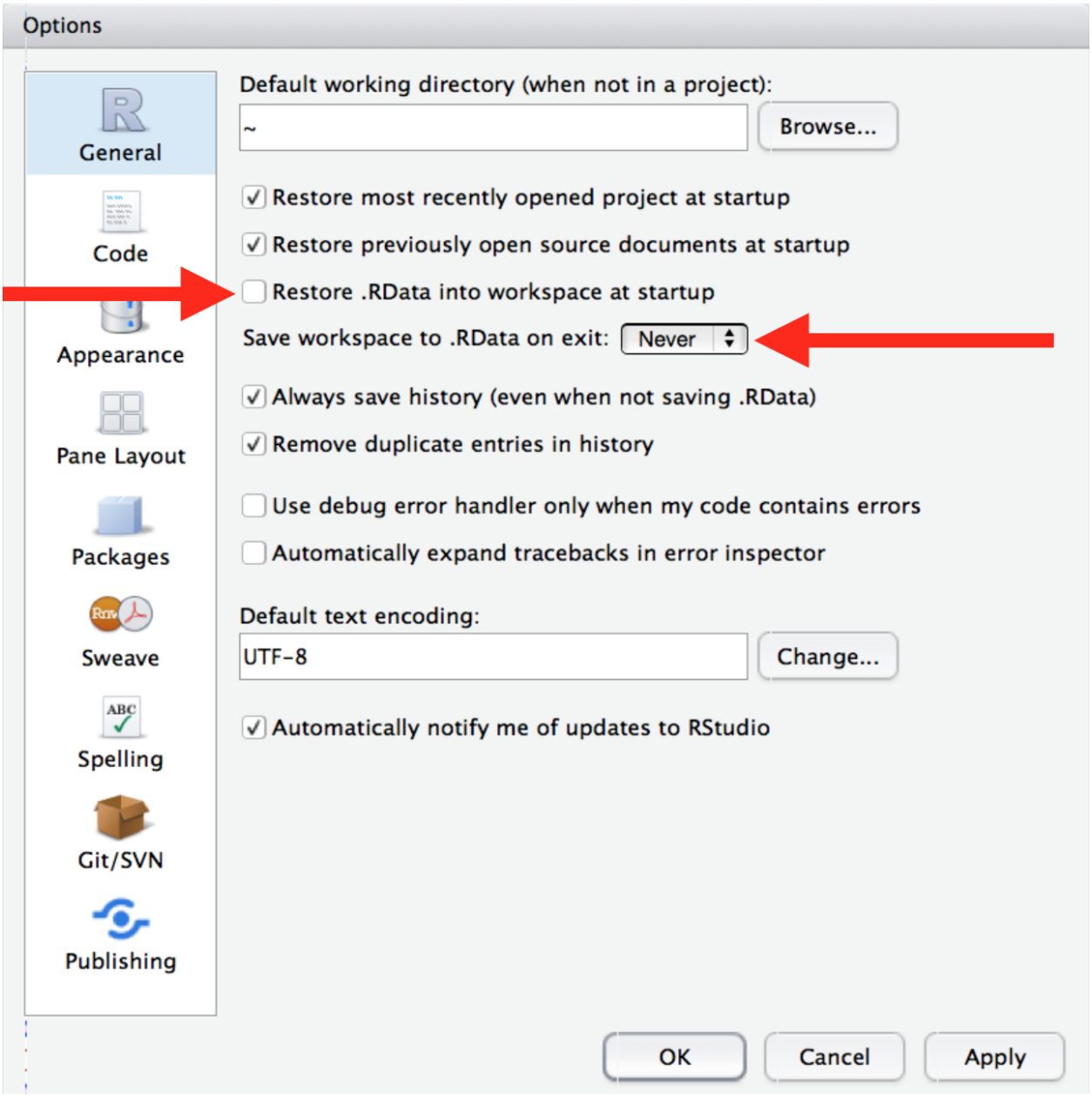
Task 1

Download and install R Software.

Done

Task 2

1. Download and install RStudio.
2. Open RStudio and ensure it and R have been properly installed.
3. Go to **Tools** > **Global Options** and change the two indicated settings below and click **Applly**.



Done

Task 3

Install the following packages:

- `tidyverse`
- `ggstance`

- `skimr`

Done

Task 4

1. Download `data_vis_labs.zip` from Canvas.
2. Unzip the file and place the unzipped `data_vis_labs` directory where you would like to keep all of your lab work for this course.
3. Open RStudio and create a project folder for this *exsisting directory*.

4. Appropriately rename `template_L01.Rmd` for submission (e.g. *Kuper_Arend_L01.Rmd*).
5. Compile the *_L01.Rmd file with `Cmd/Ctrl + Shift + K`.

Done

Task 5

Optional: It is always handy to have a versatile text editor and I would suggest downloading Sublime Text. It is free to use.

Exercise 1

Let's look at some interesting patterns in the history of LEGO! We'll be using data from the **lego** package located `data/legosets.rda`. We will work through this exercise together in class.

```
load(file = "data/legosets.rda")
```

Inspect the data

The **lego** package provides a helpful dataset some interesting variables. Let's take a quick look at the data.

```
# quick look
legosets
```

```
## # A tibble: 6,172 x 14
##    Item_Number Name   Year Theme Subtheme Pieces Minifigures Image_URL GBP_MSRP
##    <chr>       <chr> <int> <chr> <chr>     <int>       <int> <chr>        <dbl>
##  1 10246       Dete~  2015 Adva~ "Modula~   2262           6 http://i~   133.
##  2 10247       Ferr~  2015 Adva~ "Fairgr~   2464          10 http://i~   150.
##  3 10248       Ferr~  2015 Adva~ "Vehicl~   1158          NA http://i~    70.0
##  4 10249       Toy ~  2015 Adva~ "Winter~    898          NA http://i~    60.0
##  5 10581       Ducks  2015 Duplo "Forest~     13           1 http://i~     9.99
##  6 10582       Anim~  2015 Duplo "Forest~     39           2 http://i~    17.0
##  7 10583       Fish~  2015 Duplo "Forest~     32           2 http://i~    20.0
##  8 10584       Fore~  2015 Duplo "Forest~    105           3 http://i~    50.0
##  9 10585       Mom ~  2015 Duplo ""           13           2 http://i~     8.99
## 10 10586       Ice ~  2015 Duplo ""           11           2 http://i~    13.0
## # ... with 6,162 more rows, and 5 more variables: USD_MSRP <dbl>,
## #   CAD_MSRP <dbl>, EUR_MSRP <dbl>, Packaging <chr>, Availability <chr>
```

```
glimpse(legosets)
```

```
## Rows: 6,172
## Columns: 14
## $ Item_Number  <chr> "10246", "10247", "10248", "10249", "10581", "10582", ...
## $ Name         <chr> "Detective's Office", "Ferris Wheel", "Ferrari F40", "...
## $ Year         <int> 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, ...
## $ Theme        <chr> "Advanced Models", "Advanced Models", "Advanced Models...
## $ Subtheme     <chr> "Modular Buildings", "Fairground", "Vehicles", "Winter...
## $ Pieces       <int> 2262, 2464, 1158, 898, 13, 39, 32, 105, 13, 11, 52, 13...
## $ Minifigures  <int> 6, 10, NA, NA, 1, 2, 2, 3, 2, 2, 3, 1, NA, NA, NA, NA,...
## $ Image_URL    <chr> "http://images.brickset.com/sets/images/10246-1.jpg", ...
## $ GBP_MSRP     <dbl> 132.99, 149.99, 69.99, 59.99, 9.99, 16.99, 19.99, 49.9...
## $ USD_MSRP     <dbl> 159.99, 199.99, 99.99, 79.99, 9.99, 19.99, 24.99, 59.9...
## $ CAD_MSRP     <dbl> 199.99, 229.99, 119.99, NA, 12.99, 24.99, 29.99, 69.99...
## $ EUR_MSRP     <dbl> 149.99, 179.99, 89.99, 69.99, 9.99, 19.99, 24.99, 59.9...
## $ Packaging    <chr> "Box", "Box", "Box", "Box", "Box", "Box", "Box", "Box"...
## $ Availability <chr> "Retail - limited", "Retail - limited", "LEGO exclusiv...
```

```
skim_without_charts(legosets)
```

Table 1: Data summary

Name	legosets
Number of rows	6172
Number of columns	14
Column type frequency:	
character	7
numeric	7
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Item_Number	0	1	1	13	0	5854	0
Name	0	1	2	73	0	5519	0
Theme	0	1	4	28	0	115	0
Subtheme	0	1	0	32	2206	358	0
Image_URL	0	1	46	58	0	6172	0
Packaging	0	1	3	21	0	14	0
Availability	0	1	6	21	0	8	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Year	0	1.00	2004.71	8.91	1971.00	2000.00	2006.00	2012.00	2015.00
Pieces	112	0.98	215.17	356.20	0.00	30.00	82.00	256.25	5922.00
Minifigures	2672	0.57	2.85	2.72	1.00	1.00	2.00	4.00	32.00
GBP_MSRP	1980	0.68	23.45	31.93	0.00	5.99	12.99	29.99	509.99
USD_MSRP	355	0.94	27.90	39.32	0.00	6.00	14.99	34.99	789.99
CAD_MSRP	4190	0.32	46.34	58.46	2.99	12.99	24.99	54.99	789.99

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
EUR_MSRP	4399	0.29	35.98	46.61	0.00	9.99	19.99	39.99	699.99

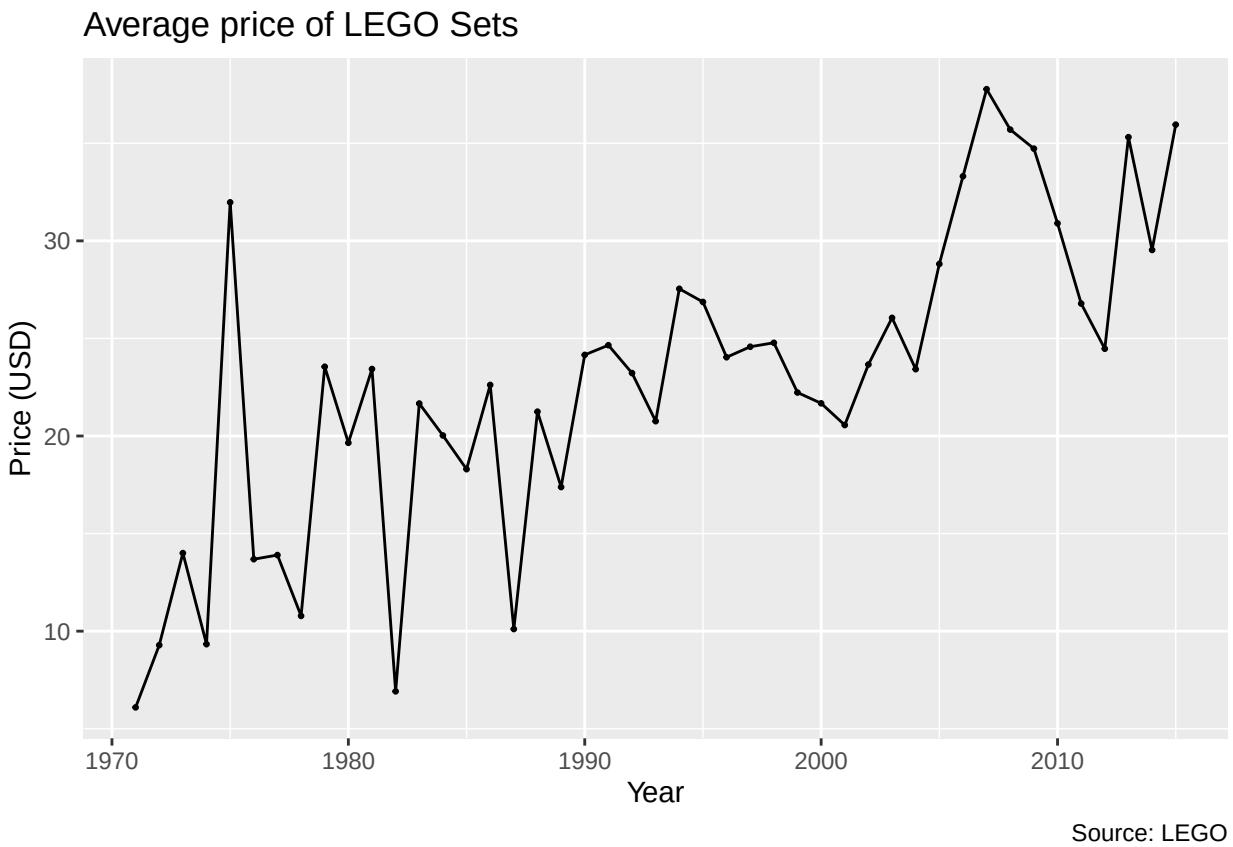
The `legosets` dataset contains information about legos over time from 1971 to 2015. It contains information such as number of pieces, item number, theme, subtheme, packaging, and price in various currencies.

Price per year

First, let's look at the average cost of LEGO sets over time. The main variable of interest here is `USD_MSRP`, or the manufacturer's suggested retail price in constant dollars (i.e. not adjusted for inflation).

```
#yearly data
legosets %>%
  filter(!is.na(USD_MSRP)) %>%
  group_by(Year) %>%
  summarize(Price = mean(USD_MSRP)) %>%
  ggplot(aes(Year, Price)) +
  geom_line() +
  geom_point(size = 0.5) +
  labs(y = "Price (USD)", title = "Average price of LEGO Sets", caption = "Source: LEGO")
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```



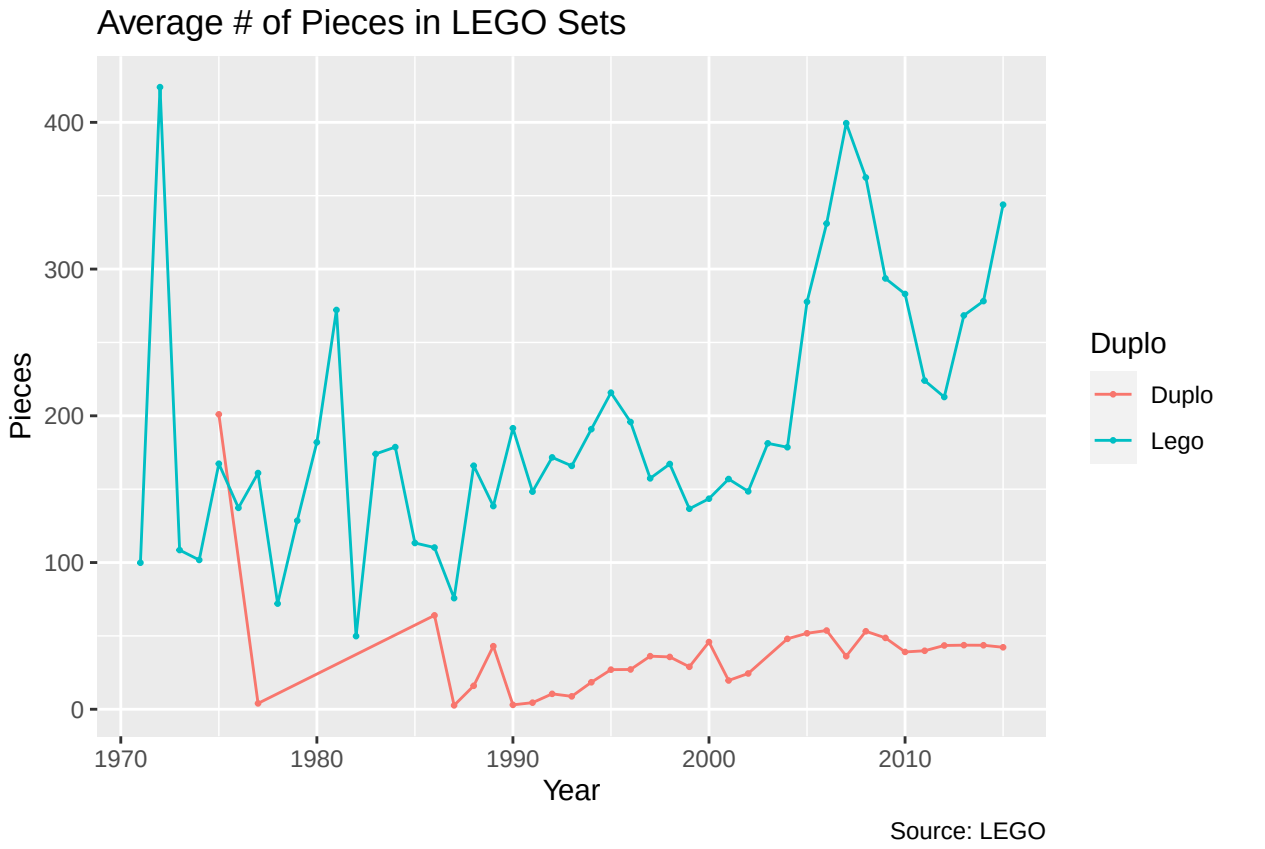
Pieces per year

Next, let's look at how the number of pieces per set has changed over time. Because Duplo sets are much smaller (since they're designed for toddlers), we'll make a special indicator variable for them.

```
#pieces per year
pieces_per_year <- legosets %>%
  filter(!is.na(Pieces)) %>%
  mutate(Duplo = if_else(Theme == "Duplo", "Duplo", "Lego")) %>%
  group_by(Year, Duplo) %>%
  summarize(avg_pieces = mean(Pieces))
```

```
## `summarise()` regrouping output by 'Year' (override with `.groups` argument)
```

```
ggplot(pieces_per_year, aes(Year, avg_pieces, color = Duplo)) +
  geom_line() +
  geom_point(size = 0.5) +
  labs(y = "Pieces", title = "Average # of Pieces in LEGO Sets", caption = "Source: LEGO")
```



LEGO set themes

In the 1990s, LEGO began partnering with famous brands and franchises to boost its own sales. First, let's see how many different "themes" LEGO now offers:

```
legosets %>%
  distinct(Theme)
```

```
## # A tibble: 115 x 1
```

```
##    Theme
##    <chr>
##  1 Advanced Models
##  2 Duplo
##  3 Juniors
##  4 Classic
##  5 Architecture
##  6 Minecraft
##  7 Ideas
##  8 Friends
##  9 Legends of Chima
## 10 Star Wars
## # ... with 105 more rows
```

```
theme_counts <- legosets %>%
  count(Theme, sort = TRUE) %>%
  mutate(Theme = fct_rev(fct_inorder(Theme, ordered = TRUE)))

theme_counts %>%
  filter(n > 150) %>%
  ggplot(aes(Theme, n)) +
  geom_col() +
  labs(y = "Number of Sets") +
  coord_flip()
```