Assignment 4

Veronika Palotai

2019 10 04

Lecture 3 Script

First choose a new team for next week

Follows Grolemund and Wickham, chapter 5

- Install the dataset if you don't have it
- install.packages("nycflights13")

```
#install.packages("nycflights13")
library(nycflights13)
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------ tidyverse 1.2.1 --
```

```
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     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()
```

Today, we'll cover

- filter()
- arrange()
- select()

Next week, we'll cover

- mutate()
- summarise()
- group_by(), which tells the other verbs to use the data by groups

All take as first argument a data frame (or tibble) and return a data frame (or tibble). Together they form the verbs of the tidyverse.

Class Exercise

For 2 minutes, think about why it is a nice property (and a conscious design choice) that all verbs take as a first argument a data frame and return a data frame. Talk with your neighbour about this.

Answer

We don't change the underlying dataset ths way, you can chain multiple functions together because the data type doesn't change -> it becomes a pipe

- Filtering (choosing) rows with filter() dplyr functions don't change the data frame that you give it. They return a new one.
- Save the filtered data
- Assign and print, use (varname <- ...)
- Check it really assigned

Some notes on comparisons

In short, you can't rely on "It works because it works for what I tried". For floating point comparisons, use near() to compare numbers#

Exercise: What counts as near? Find out. Can you change it?

Using near is safer than using '==' because it has a built in tolerance ('tol') which can be modified:

```
near(x, y, tol = .Machine$double.eps^0.5)
```

Multiple constraints

Class exercise: How do we know these actually worked?

Class Exercise: What does this do?

```
(mystery_filter <- filter(flights, !(arr_delay > 120 | dep_delay > 120)))
```

Vote:

1. All flights that started and landed 120 minutes late
2. All flights that started 120 minutes late or landed 120 minutes late
3. All flights that started less than 120 minutes late or landed less than 120 minutes late
4. All flights that started and landed less than 120 minutes late

Correct answer: 4.

Class Exercise: Get the filter command for number 3. above

Answer

```
(practice_filter <- filter(flights, (dep_delay < 120) | (arr_delay < 120)))
```

Class Exercise: get all flights that departed with less than 120 minutes delay, but arrived with more than 120 minutes delay.

```
dep_ok_arr_not <- filter(flights, dep_delay <= 120, arr_delay > 120)
```

Let's look at the data to see what the departure was for planes that arrived late but didn't start quite as late

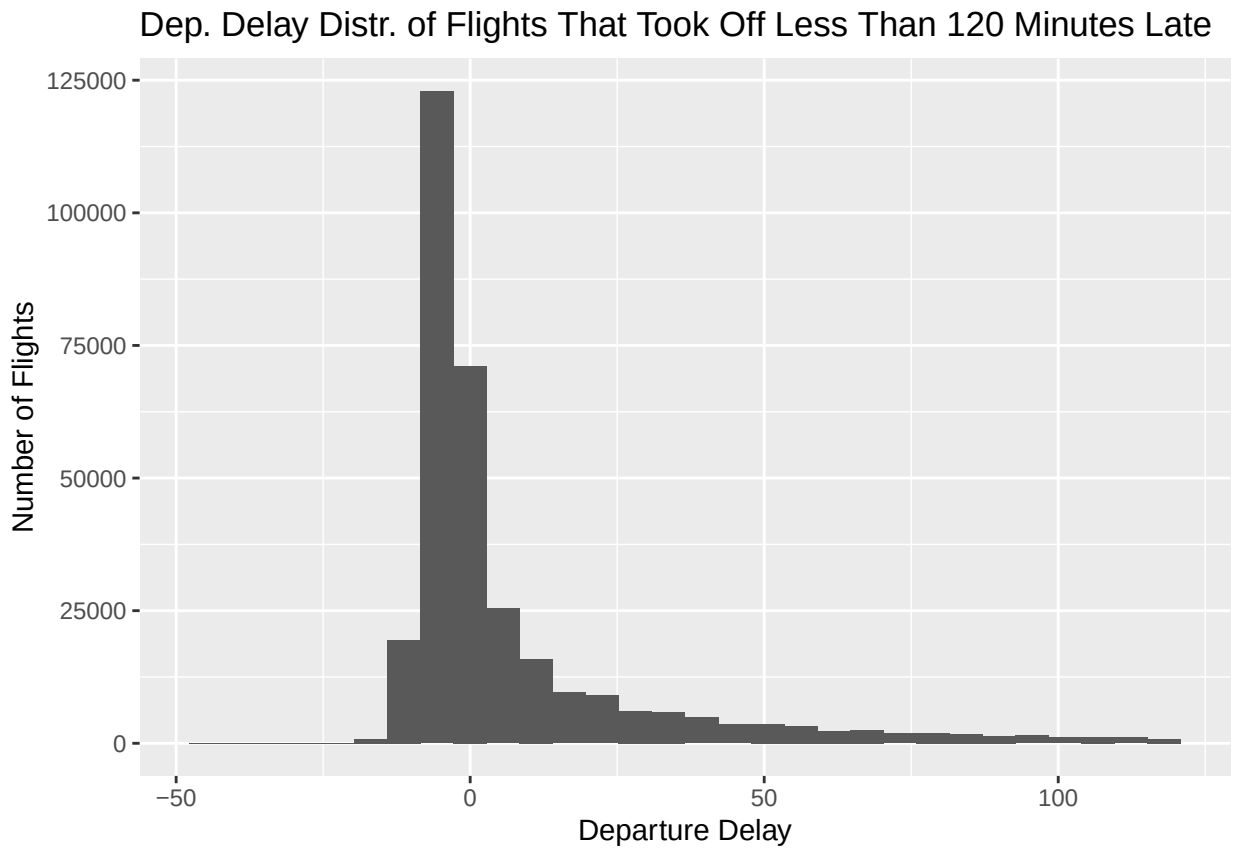
Filter flights by those that had dep_delay <= 120, then plot histogram

```
dep_delay_ok <- filter(flights, dep_delay <= 120)

ggplot(data = dep_delay_ok,
       mapping = aes(x = dep_delay)) +
  geom_histogram() +
```

```
  labs(x = "Departure Delay",
       y = "Number of Flights",
       title = "Dep. Delay Distr. of Flights That Took Off Less Than 120 Minutes Late")
```

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



NA: Not available

```
NA > 5
10 == NA
NA == NA
FALSE & NA # false and sth is false
TRUE & NA # output depends on what NA is
```

Nice example from G&W

Let x be Mary's age. We don't know how old she is. Let y be John's age. We don't know how old he is.

```
x <- NA
y <- NA
```

Are John and Mary the same age?

```
x == y
```

We don't know!

arrange()

Some examples:

```
arrange(flights, year, month, day)
arrange(flights, dep_delay)
arrange(flights, desc(dep_delay))
```

Class exercise: How can we get the missing values at the top?

Fastest flight

```
arrange(flights, air_time)
```

Better ways of getting some special columns

select()

```
select(flights, year, month, day)
select(flights, air_time)
select(arrange(flights, air_time), air_time, origin, dest)
```

That's tedious to write. Hence the pipe.

```
flights %>%
  arrange(air_time) %>%
  select(air_time, origin, dest)
```

Notice that the data doesn't have to be mentioned and the first argument should not have to be provided

Some helper functions

```
select(flights, year:day)
flights %>% select(year:day) # same as above

colnames(flights)
```

Dropping Columns

```
select(flights, -(year:day))
```

start_with, end_with, contains

```
select(flights, starts_with("arr"))
select(flights, -starts_with("arr"))
select(flights, ends_with("hour"))
select(flights, -contains("time"))
```

For More Use Help

```
?select
```

Renaming Columns

```
rename(flights, destination = dest)
```

If it's difficult to see

```
flights %>% rename(destination = dest) %>% select(year:day, destination)
```

Moving Columns to The Start

```
select(flights, origin, dest, everything())
# takes origin and dest to the beginning, before everything
```

Class Exercise: What happens if you include a variable multiple times?

Assignment 4

Resources

If you have no experience coding, this may be helpful: https://rstudio-education.github.io/hopr/

Assignment 4

1. Read Chapter 5 of Grolemund and Wickham parts 1 through 3 (until select) of Grolemund and Wickham for anything we did not cover. We will cover the remaining parts next week.
2. Turn the script (.R file) from class into a markdown file which displays the graphs and tables. Add any comments that might benefit you later on, such as reminders of things you found confusing, etc. Make sure that you comment the graphs where appropriate, either through captions or in the accompanying text.
3. Repeat the steps from chapter 5 in parts 1 through 3, but using hotels data instead of the nycflights data. Since the two datasets don't have the same columns, either pick some variable you'd like to filter on and see results on, or use the following suggested mapping:

- When filtering (etc) on month for flights, use stars in the hotels data
- Instead of flight duration, use hotel price
- For travel times, use distance (you can reuse distance for different types of time)

Example: Instead of doing filter(flights, month == 1) you should do filter(hotels, stars ==).

Create similar output to Grolemund and Wickham, i.e. show what the output is of various commands.

Part 3 of Assignment 4

5.1 Introduction

5.1.1 Prerequisites

CLEAR MEMORY

```
rm(list=ls())
```

Import Libraries

```
# install.packages("scales")
library(ggplot2)
library(tidyverse)
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

Setting the Path

```
dir <-  "D:/Egyetem/CEU/Coding_1/R-Coding/"
```

Location Folders

```
data_in <- paste0(dir,"da_data_repo/hotels-vienna/clean/")
data_out <-  paste0(dir,"da_case_studies/ch03-hotels-vienna-explore/")
output <- paste0(dir,"da_case_studies/ch03-hotels-vienna-explore/output/")
func <- paste0(dir, "da_case_studies/ch00-tech-prep/")
```

5.1.2 Hotels Vienna

Loading Dataset

```
vienna <- read_csv(paste0(data_in,"hotels-vienna.csv"))
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   country = col_character(),
##   city_actual = col_character(),
##   center1label = col_character(),
##   center2label = col_character(),
##   neighbourhood = col_character(),
##   city = col_character(),
##   offer_cat = col_character(),
##   accommodation_type = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

View Dataset

```
View(vienna)
```

5.1.3 dplyr basics

```
library(dplyr)
```

5.2 Filter Rows with filter()

```
filter(vienna, stars == 4.0)
four_star_hotels <- filter(vienna, stars == 4.0) # saving result
(four_star_hotels <- filter(vienna, stars == 4.0)) # printing result
```

5.2.1 Comparisons

```
filter(vienna, stars = 4.0)
#Error: `stars` (`stars = 4`) must not be named, do you need `==`?
```

We get an informative error that tells us to use '==' instead of '='.

5.2.2 Logical Operators

```
filter(vienna, stars == 4.0 | stars == 3.0)
```

```
## # A tibble: 283 x 24
##    country city_actual rating_count center1label center2label neighbourhood
##    <chr>   <chr>              <dbl> <chr>        <chr>        <chr>
##  1 Austria Vienna                36 City centre  Donauturm    17. Hernals
##  2 Austria Vienna               189 City centre  Donauturm    17. Hernals
##  3 Austria Vienna                53 City centre  Donauturm    Alsergrund
##  4 Austria Vienna                55 City centre  Donauturm    Alsergrund
##  5 Austria Vienna                33 City centre  Donauturm    Alsergrund
##  6 Austria Vienna                57 City centre  Donauturm    Alsergrund
##  7 Austria Vienna               161 City centre  Donauturm    Alsergrund
##  8 Austria Vienna                NA City centre  Donauturm    Alsergrund
##  9 Austria Vienna               203 City centre  Donauturm    Alsergrund
## 10 Austria Vienna               251 City centre  Donauturm    Alsergrund
## # ... with 273 more rows, and 18 more variables: price <dbl>, city <chr>,
## #   stars <dbl>, ratingta <dbl>, ratingta_count <dbl>, scarce_room <dbl>,
## #   hotel_id <dbl>, offer <dbl>, offer_cat <chr>, year <dbl>, month <dbl>,
## #   weekend <dbl>, holiday <dbl>, distance <dbl>, distance_alter <dbl>,
## #   accommodation_type <chr>, nnights <dbl>, rating <dbl>
```

```
four_or_three_star_hotels <- filter(vienna, stars %in% c(4.0, 3.0))
filter(vienna, !(stars > 4.0 | distance > 5.0))
```

```
## # A tibble: 376 x 24
##    country city_actual rating_count center1label center2label neighbourhood
##    <chr>   <chr>              <dbl> <chr>        <chr>        <chr>
##  1 Austria Vienna                36 City centre  Donauturm    17. Hernals
##  2 Austria Vienna               189 City centre  Donauturm    17. Hernals
##  3 Austria Vienna                53 City centre  Donauturm    Alsergrund
##  4 Austria Vienna                55 City centre  Donauturm    Alsergrund
##  5 Austria Vienna                33 City centre  Donauturm    Alsergrund
##  6 Austria Vienna                57 City centre  Donauturm    Alsergrund
##  7 Austria Vienna               161 City centre  Donauturm    Alsergrund
##  8 Austria Vienna                50 City centre  Donauturm    Alsergrund
##  9 Austria Vienna                NA City centre  Donauturm    Alsergrund
## 10 Austria Vienna               203 City centre  Donauturm    Alsergrund
## # ... with 366 more rows, and 18 more variables: price <dbl>, city <chr>,
## #   stars <dbl>, ratingta <dbl>, ratingta_count <dbl>, scarce_room <dbl>,
## #   hotel_id <dbl>, offer <dbl>, offer_cat <chr>, year <dbl>, month <dbl>,
## #   weekend <dbl>, holiday <dbl>, distance <dbl>, distance_alter <dbl>,
## #   accommodation_type <chr>, nnights <dbl>, rating <dbl>
```

```
filter(vienna, stars <= 4.0, distance <= 5.0) # means the same as the one above
```

```
## # A tibble: 376 x 24
##    country city_actual rating_count center1label center2label neighbourhood
```

```
##    <chr>   <chr>              <dbl> <chr>        <chr>        <chr>
##  1 Austria Vienna                36 City centre  Donauturm    17. Hernals
##  2 Austria Vienna               189 City centre  Donauturm    17. Hernals
##  3 Austria Vienna                53 City centre  Donauturm    Alsergrund
##  4 Austria Vienna                55 City centre  Donauturm    Alsergrund
##  5 Austria Vienna                33 City centre  Donauturm    Alsergrund
##  6 Austria Vienna                57 City centre  Donauturm    Alsergrund
##  7 Austria Vienna               161 City centre  Donauturm    Alsergrund
##  8 Austria Vienna                50 City centre  Donauturm    Alsergrund
##  9 Austria Vienna                NA City centre  Donauturm    Alsergrund
## 10 Austria Vienna               203 City centre  Donauturm    Alsergrund
## # ... with 366 more rows, and 18 more variables: price <dbl>, city <chr>,
## #   stars <dbl>, ratingta <dbl>, ratingta_count <dbl>, scarce_room <dbl>,
## #   hotel_id <dbl>, offer <dbl>, offer_cat <chr>, year <dbl>, month <dbl>,
## #   weekend <dbl>, holiday <dbl>, distance <dbl>, distance_alter <dbl>,
## #   accommodation_type <chr>, nnights <dbl>, rating <dbl>
```

5.3 Arrange Rows with arrange()

```
arrange(vienna, stars, distance, price)
```

```
## # A tibble: 428 x 24
##    country city_actual rating_count center1label center2label neighbourhood
##    <chr>   <chr>              <dbl> <chr>        <chr>        <chr>
##  1 Austria Vienna                57 City centre  Donauturm    Leopoldstadt
##  2 Austria Vienna                81 City centre  Donauturm    Leopoldstadt
##  3 Austria Vienna                16 City centre  Donauturm    Innere Stadt
##  4 Austria Vienna                50 City centre  Donauturm    Alsergrund
##  5 Austria Vienna               170 City centre  Donauturm    Mariahilf
##  6 Austria Vienna               307 City centre  Donauturm    Leopoldstadt
##  7 Austria Vienna                30 City centre  Donauturm    Josefstadt
##  8 Austria Vienna                84 City centre  Donauturm    Josefstadt
##  9 Austria Vienna                 3 City centre  Donauturm    Mariahilf
## 10 Austria Vienna                 5 City centre  Donauturm    Leopoldstadt
## # ... with 418 more rows, and 18 more variables: price <dbl>, city <chr>,
## #   stars <dbl>, ratingta <dbl>, ratingta_count <dbl>, scarce_room <dbl>,
## #   hotel_id <dbl>, offer <dbl>, offer_cat <chr>, year <dbl>, month <dbl>,
## #   weekend <dbl>, holiday <dbl>, distance <dbl>, distance_alter <dbl>,
## #   accommodation_type <chr>, nnights <dbl>, rating <dbl>
```

```
arrange(vienna, desc(stars))
```

```
## # A tibble: 428 x 24
##    country city_actual rating_count center1label center2label neighbourhood
##    <chr>   <chr>              <dbl> <chr>        <chr>        <chr>
##  1 Austria Vienna                25 City centre  Donauturm    Alsergrund
##  2 Austria Vienna               144 City centre  Donauturm    Graben
##  3 Austria Vienna                78 City centre  Donauturm    Innere Stadt
##  4 Austria Vienna                 5 City centre  Donauturm    Innere Stadt
##  5 Austria Vienna               193 City centre  Donauturm    Innere Stadt
##  6 Austria Vienna               334 City centre  Donauturm    Innere Stadt
##  7 Austria Vienna               150 City centre  Donauturm    Innere Stadt
##  8 Austria Vienna               531 City centre  Donauturm    Innere Stadt
##  9 Austria Vienna               253 City centre  Donauturm    Innere Stadt
## 10 Austria Vienna               564 City centre  Donauturm    Innere Stadt
```

```
## # ... with 418 more rows, and 18 more variables: price <dbl>, city <chr>,
## #   stars <dbl>, ratingta <dbl>, ratingta_count <dbl>, scarce_room <dbl>,
## #   hotel_id <dbl>, offer <dbl>, offer_cat <chr>, year <dbl>, month <dbl>,
## #   weekend <dbl>, holiday <dbl>, distance <dbl>, distance_alter <dbl>,
## #   accommodation_type <chr>, nnights <dbl>, rating <dbl>
```

5.4 Select Columns with select()

```
select(vienna, stars, distance, price, accommodation_type)
```

```
## # A tibble: 428 x 4
##    stars distance price accommodation_type
##    <dbl>    <dbl> <dbl> <chr>
##  1     4      2.7    81 Apartment
##  2     4      1.7    81 Hotel
##  3     4      1.4    85 Hotel
##  4     3      1.7    83 Hotel
##  5     4      1.2    82 Hotel
##  6     5      0.9   229 Apartment
##  7     4      0.9   103 Hotel
##  8     4      1     150 Hotel
##  9     2      0.7    80 Hotel
## 10     3      1.5   153 Apartment
## # ... with 418 more rows
```

```
select(vienna, stars:distance)
```

```
## # A tibble: 428 x 12
##    stars ratingta ratingta_count scarce_room hotel_id offer offer_cat  year
##    <dbl>    <dbl>          <dbl>       <dbl>    <dbl> <dbl> <chr>     <dbl>
##  1     4      4.5            216           1    21894     1 15-50% o~  2017
##  2     4      3.5            708           0    21897     1 1-15% of~  2017
##  3     4      3.5            629           0    21901     1 15-50% o~  2017
##  4     3      4               52           0    21902     1 15-50% o~  2017
##  5     4      3.5            219           1    21903     1 15-50% o~  2017
##  6     5      4.5             27           1    21904     1 1-15% of~  2017
##  7     4      3.5            251           1    21906     0 0% no of~  2017
##  8     4      4.5            617           0    21907     0 0% no of~  2017
##  9     2      3.5            146           1    21908     1 1-15% of~  2017
## 10     3     NA               NA           1    21910     1 15-50% o~  2017
## # ... with 418 more rows, and 4 more variables: month <dbl>,
## #   weekend <dbl>, holiday <dbl>, distance <dbl>
```

```
select(vienna, -(stars:distance))
```

```
## # A tibble: 428 x 12
##    country city_actual rating_count center1label center2label neighbourhood
##    <chr>   <chr>              <dbl> <chr>        <chr>        <chr>
##  1 Austria Vienna                36 City centre  Donauturm    17. Hernals
##  2 Austria Vienna               189 City centre  Donauturm    17. Hernals
##  3 Austria Vienna                53 City centre  Donauturm    Alsergrund
##  4 Austria Vienna                55 City centre  Donauturm    Alsergrund
##  5 Austria Vienna                33 City centre  Donauturm    Alsergrund
##  6 Austria Vienna                25 City centre  Donauturm    Alsergrund
##  7 Austria Vienna                57 City centre  Donauturm    Alsergrund
```

```
##  8 Austria Vienna               161 City centre  Donauturm    Alsergrund
##  9 Austria Vienna                50 City centre  Donauturm    Alsergrund
## 10 Austria Vienna                NA City centre  Donauturm    Alsergrund
## # ... with 418 more rows, and 6 more variables: price <dbl>, city <chr>,
## #   distance_alter <dbl>, accommodation_type <chr>, nnights <dbl>,
## #   rating <dbl>
```

```
rename(vienna, nr_of_nights = nnights)
```

```
## # A tibble: 428 x 24
##    country city_actual rating_count center1label center2label neighbourhood
##    <chr>   <chr>              <dbl> <chr>        <chr>        <chr>
##  1 Austria Vienna                36 City centre  Donauturm    17. Hernals
##  2 Austria Vienna               189 City centre  Donauturm    17. Hernals
##  3 Austria Vienna                53 City centre  Donauturm    Alsergrund
##  4 Austria Vienna                55 City centre  Donauturm    Alsergrund
##  5 Austria Vienna                33 City centre  Donauturm    Alsergrund
##  6 Austria Vienna                25 City centre  Donauturm    Alsergrund
##  7 Austria Vienna                57 City centre  Donauturm    Alsergrund
##  8 Austria Vienna               161 City centre  Donauturm    Alsergrund
##  9 Austria Vienna                50 City centre  Donauturm    Alsergrund
## 10 Austria Vienna                NA City centre  Donauturm    Alsergrund
## # ... with 418 more rows, and 18 more variables: price <dbl>, city <chr>,
## #   stars <dbl>, ratingta <dbl>, ratingta_count <dbl>, scarce_room <dbl>,
## #   hotel_id <dbl>, offer <dbl>, offer_cat <chr>, year <dbl>, month <dbl>,
## #   weekend <dbl>, holiday <dbl>, distance <dbl>, distance_alter <dbl>,
## #   accommodation_type <chr>, nr_of_nights <dbl>, rating <dbl>
```

```
select(vienna, stars, price, distance, everything())
```

```
## # A tibble: 428 x 24
##    stars price distance country city_actual rating_count center1label
##    <dbl> <dbl>    <dbl> <chr>   <chr>              <dbl> <chr>
##  1     4    81      2.7 Austria Vienna                36 City centre
##  2     4    81      1.7 Austria Vienna               189 City centre
##  3     4    85      1.4 Austria Vienna                53 City centre
##  4     3    83      1.7 Austria Vienna                55 City centre
##  5     4    82      1.2 Austria Vienna                33 City centre
##  6     5   229      0.9 Austria Vienna                25 City centre
##  7     4   103      0.9 Austria Vienna                57 City centre
##  8     4   150      1   Austria Vienna               161 City centre
##  9     2    80      0.7 Austria Vienna                50 City centre
## 10     3   153      1.5 Austria Vienna                NA City centre
## # ... with 418 more rows, and 17 more variables: center2label <chr>,
## #   neighbourhood <chr>, city <chr>, ratingta <dbl>, ratingta_count <dbl>,
## #   scarce_room <dbl>, hotel_id <dbl>, offer <dbl>, offer_cat <chr>,
## #   year <dbl>, month <dbl>, weekend <dbl>, holiday <dbl>,
## #   distance_alter <dbl>, accommodation_type <chr>, nnights <dbl>,
## #   rating <dbl>
```