Joining tables with `dplyr`

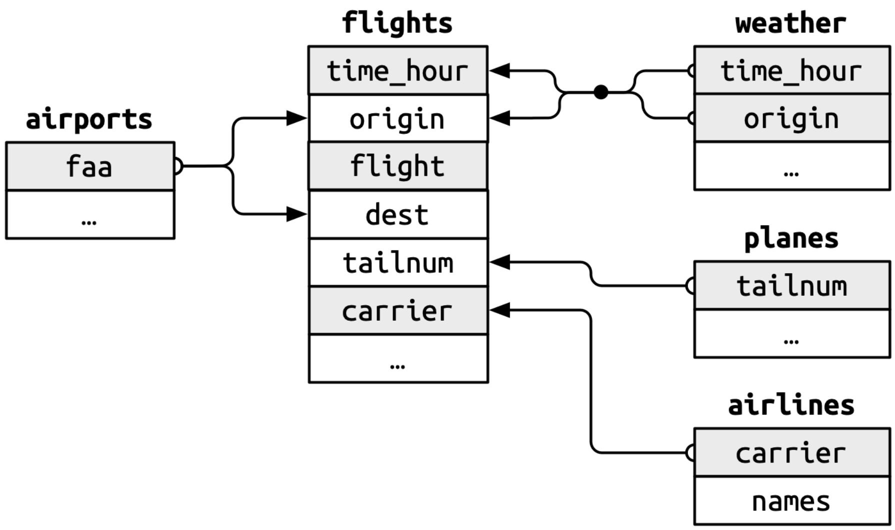
Emily Malcolm-White

`nycflights23` dataset

```r
#LOAD PACKAGES
library(tidyverse)

#LOAD DATA
library(nycflights23)
data("flights")
```

`nycflights23` contains information about all 435352 flights departing NYC in 2023.



Join dataframes

Matching key variable names

Some airline names might be easy to guess (ie. "UA" is United Airlines), but what airlines have the code "VX", "HA", and "B6"? Data on airline codes is provided in a dataset called `airlines`.

```
#data("airlines")
```

We want to have all this information in one data frame instead of two separate data frames.

The variable `carrier` in `flights` match the variable `carrier` in the `airlines` dataset – this is our *key variable*. In this case, they have the same name, but this doesn't necessarily have to be true.

```
flights_joined <- flights %>%
  inner_join(airlines, by="carrier")
```

Different key variable names

Say instead you are interested in the destinations of all domestic flights departing NYC in 2013, and you ask yourself questions like: "What cities are these airports in?", or "Is"ORD" Orlando?"

```
data("airports")
```

In `airports` the airport code is in `faa`, whereas in `flights` the airport codes are in `origin` and `dest`.

```
flights_with_airport_names <- flights %>%
  inner_join(airports, by = c("dest" = "faa"))
```

Let's construct the chain of pipe operators %>% that computes the number of flights from NYC to each destination, but also includes information about each destination airport:

```
named_dests <- flights %>%
  group_by(dest) %>%
  summarize(num_flights = n()) %>%
  arrange(desc(num_flights)) %>%
  inner_join(airports, by = c("dest" = "faa")) %>%
```

```r
  rename(airport_name = name)
named_dests
```

```
# A tibble: 114 x 9
   dest  num_flights airport_name               lat    lon   alt    tz dst   tzone
   <chr>       <int> <chr>                    <dbl>  <dbl> <dbl> <dbl> <chr> <chr>
 1 BOS         19036 General Edward Lawren~   42.4  -71.0    20    -5 A     Amer~
 2 ORD         18200 Chicago O'Hare Intern~   42.0  -87.9   672    -6 A     Amer~
 3 MCO         17756 Orlando International~   28.4  -81.3    96    -5 A     Amer~
 4 ATL         17570 Hartsfield Jackson At~   33.6  -84.4  1026    -5 A     Amer~
 5 MIA         16076 Miami International A~   25.8  -80.3     8    -5 A     Amer~
 6 LAX         15968 Los Angeles Internati~   33.9 -118.    125    -8 A     Amer~
 7 FLL         14239 Fort Lauderdale Holly~   26.1  -80.2     9    -5 A     Amer~
 8 CLT         12866 Charlotte Douglas Int~   35.2  -80.9   748    -5 A     Amer~
 9 DFW         11675 Dallas Fort Worth Int~   32.9  -97.0   607    -6 A     Amer~
10 SFO         11651 San Francisco Interna~   37.6 -122.     13    -8 A     Amer~
# i 104 more rows
```

Multiple Key variables

In order to join the flights and weather data frames, we need more than one key variable: `year`, `month`, `day`, `hour`, and `origin`. This is because the combination of these 5 variables act to uniquely identify each observational unit in the weather data frame: hourly weather recordings at each of the 3 NYC airports.

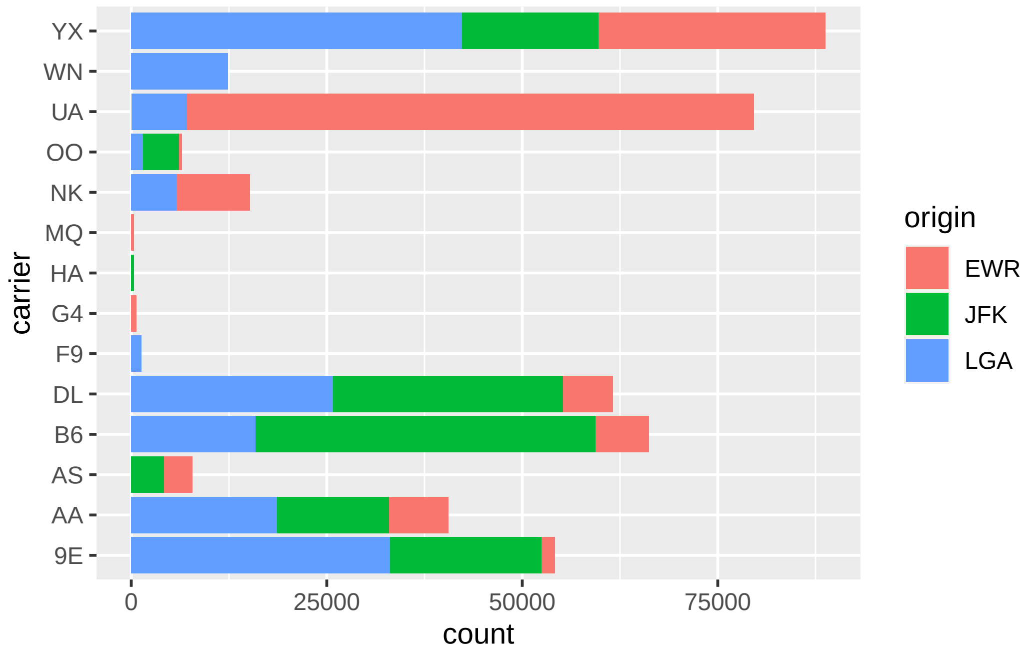
```r
data("weather")
```

```r
flights_weather_joined <- flights %>%
  inner_join(weather, by = c("year", "month", "day", "hour", "origin"))
```

Why is this useful?

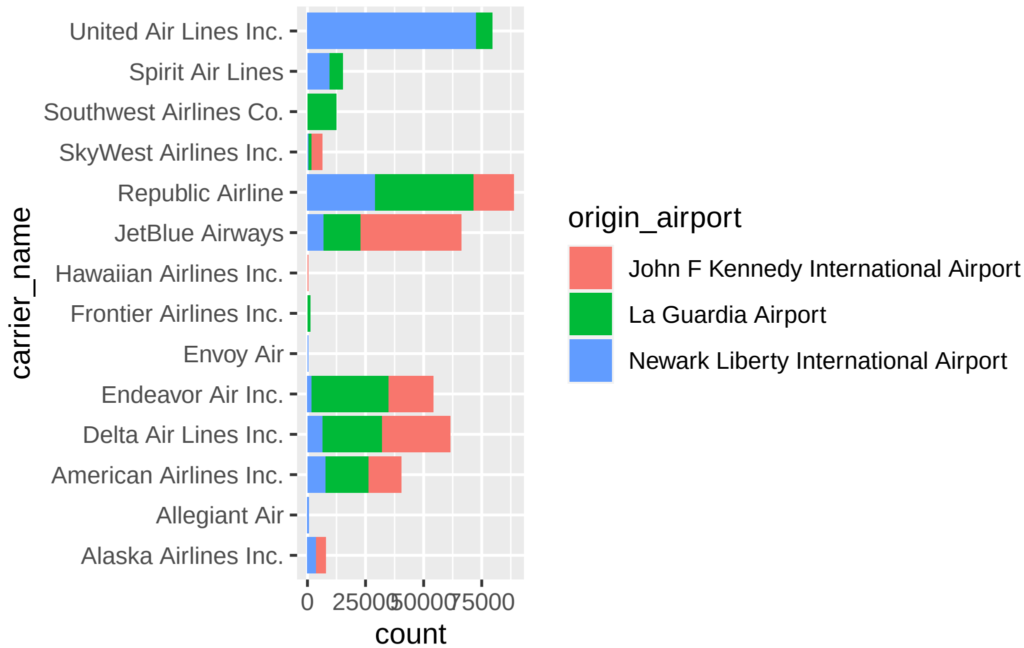
Updating labels:

```r
flights %>%
ggplot(aes(x = carrier, fill = origin)) +
  geom_bar() +
  coord_flip()
```



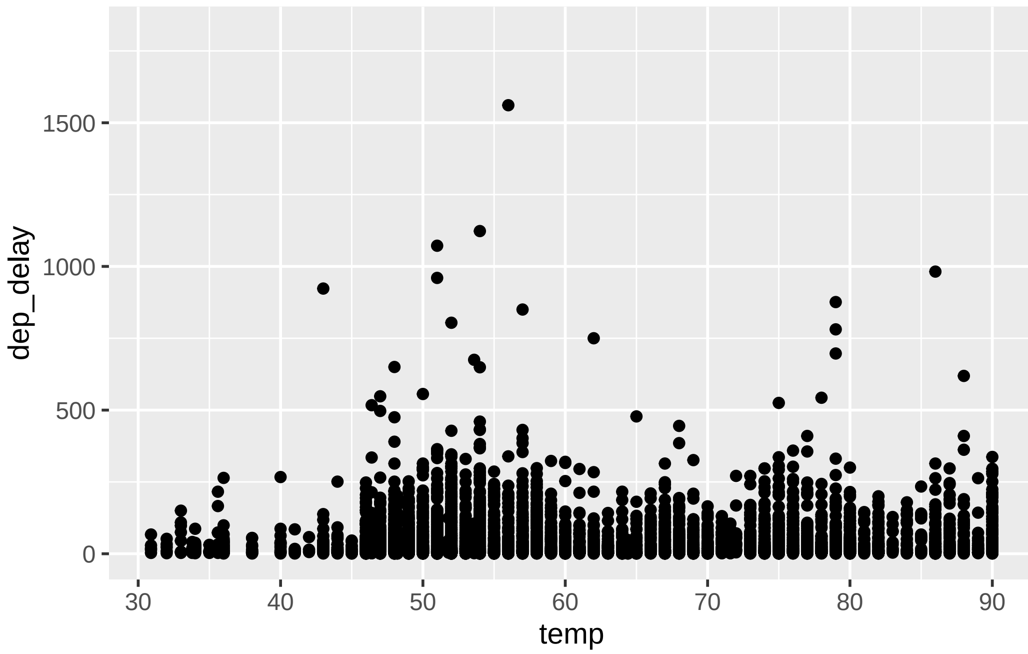

```
#VS

flights %>%
  inner_join(airports, by = c("origin" = "faa")) %>%
  rename(origin_airport = name) %>%
  inner_join(airlines, by = c("carrier")) %>%
  rename(carrier_name= name) %>%
ggplot(mapping = aes(x = carrier_name, fill = origin_airport)) +
  geom_bar() +
  coord_flip()
```



Exploring relationships between variables in separate tables:

```
flights_weather_joined %>%
  filter(dep_delay >0) %>%
  ggplot(aes(x=temp, y=dep_delay)) +
  geom_point()
```



Different Types of Joins

Common Issues with Joining

- duplicate keys
- lowercase/uppercase
- symbols or whitespace
- Make sure the join fields are the same format.

External Resources

- R for Data Science, Joins

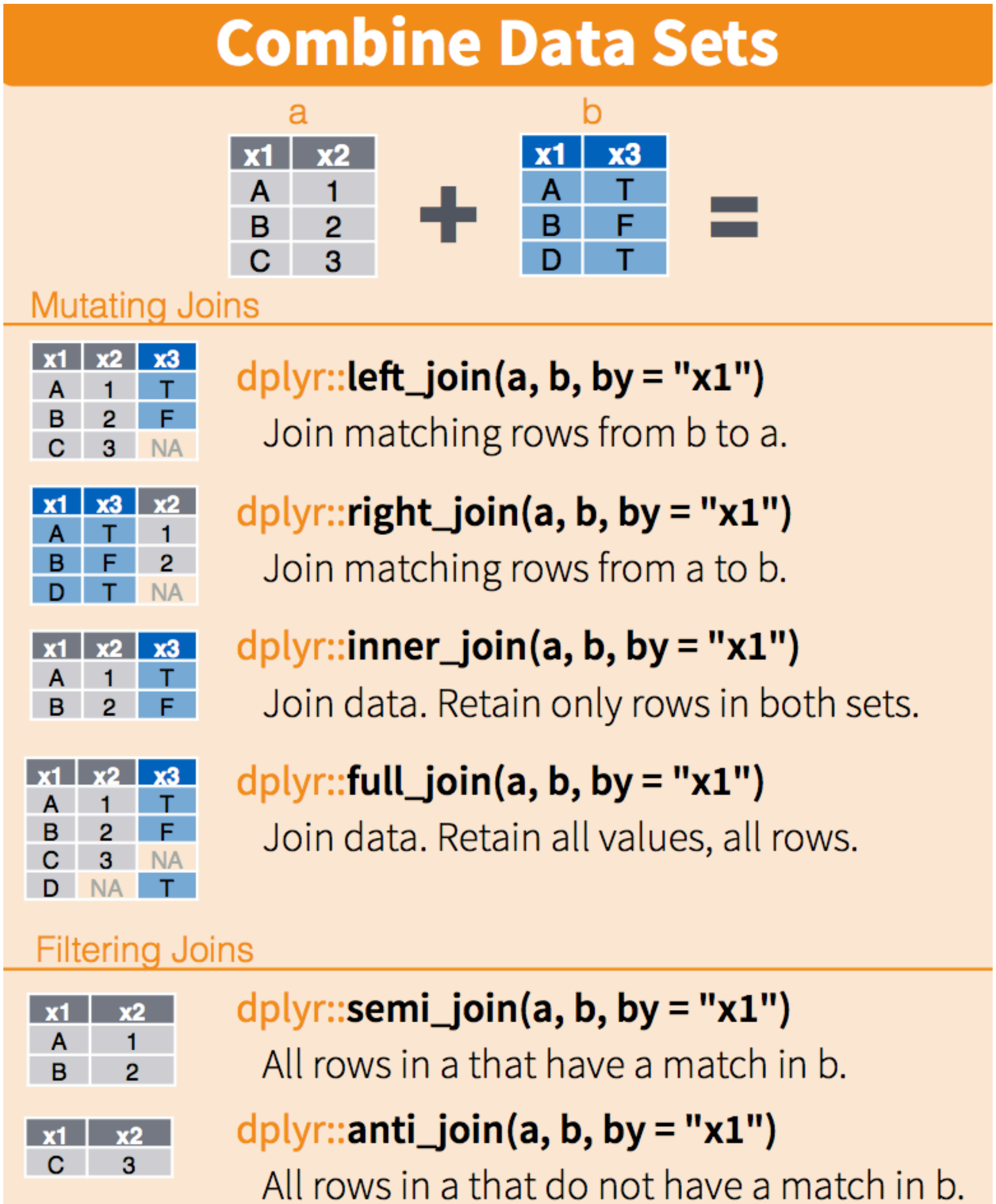


Figure 1: Credit: RStudio