Tidying Data

By Tony Fraser

2023-09-30

Assignment

Demonstrate tidy transformation capability using the airline dataset depicted below. As well, perform a s simple analysis on the same data.

	A	B	C	D	E	F	G
1			Los Angeles	Phoenix	San Diego	San Francisc	Seattle
2	ALASKA	on time	497	221	212	503	1841
3		delayed	62	12	20	102	305
4							
5	AM WEST	on time	694	4840	383	320	201
6		delayed	117	415	65	129	61

Figure 1: Wide Data Set

Load Data

```
library(tidyverse)

data <- read.table("airlines.csv", header = TRUE, sep = ",",
                   stringsAsFactors = FALSE, fill = TRUE,
                   na.strings = c("NA", "")) #<- empty records straight into NA
# > data
#         X     X.1 Los.Angeles Phoenix San.Diego San.Francisco Seattle
# 1  ALASKA on time         497     221       212           503    1841
# 2    <NA> delayed          62      12        20           102     305
# 3    <NA>    <NA>          NA      NA        NA            NA      NA
# 4 AM WEST on time         694    4840       383           320     201
# 5    <NA> delayed         117     415        65           129      61
```

Basic cleaning

Remove empty rows, and run vertical fills.

```
data_cleaned <- data %>%
  rename(Airline = X, Status = X.1) %>% # Rename columns
  fill(Airline, .direction = "down") %>%
  filter(!is.na(Status)) # Remove rows where Status is NA

#   Airline  Status Los.Angeles Phoenix San.Diego San.Francisco Seattle
# 1  ALASKA on time         497     221       212           503    1841
# 2  ALASKA delayed          62      12        20           102     305
# 3 AM WEST on time         694    4840       383           320     201
# 4 AM WEST delayed         117     415        65           129      61
```

tidy with pivot_wider() and pivot_longer()

"pivot_longer() and pivot_wider() are fundamental functions in the tidyverse for reshaping data in R. In essence, they either consolidate multiple columns into key-value pairs, or expand fields into distinct columns.

```
gathered <- data_cleaned %>%
  pivot_longer(cols = c(Los.Angeles, Phoenix, San.Diego,  San.Francisco, Seattle),
     names_to = "City",
     values_to = "Flights")
  # gather(key = "City", value = "Flights", -Airline, -Status)    #old way
# > gathered
#    Airline  Status          City Flights
# 1   ALASKA on time   Los.Angeles     497
# 2   ALASKA delayed   Los.Angeles      62
# 3  AM WEST on time   Los.Angeles     694
# 4  AM WEST delayed   Los.Angeles     117
# 5   ALASKA on time       Phoenix     221
# 6   ALASKA delayed       Phoenix      12
# 7  AM WEST on time       Phoenix    4840
# 8  AM WEST delayed       Phoenix     415
# 9   ALASKA on time     San.Diego     212
# 10  ALASKA delayed     San.Diego      20
# 10 other records omitted

spread <- gathered %>%
    pivot_wider(names_from = Status, values_from = Flights)
    # spread(key = Status, value = Flights) <- used to be this.    # old way
# > spread
#    Airline          City delayed on time
# 1   ALASKA   Los.Angeles      62     497
# 2   ALASKA       Phoenix      12     221
# 3   ALASKA     San.Diego      20     212
# 4   ALASKA San.Francisco     102     503
# 5   ALASKA       Seattle     305    1841
# 6  AM WEST   Los.Angeles     117     694
# 7  AM WEST       Phoenix     415    4840
# 8  AM WEST     San.Diego      65     383
# 9  AM WEST San.Francisco     129     320
# 10 AM WEST       Seattle      61     201
```

A basic view of airline performance

```
library(ggplot2)


ggplot(spread, aes(x = City, y = delayed, fill = Airline)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Delays by City and Airline",
       y = "Number of Delays",
       x = "City") +
  theme_minimal()
```