Untitled

Jawaid Hakim

2022-09-28

Contents

1 Assignment

The chart above describes arrival delays for two airlines across five destinations. Your task is to: 1. Create a .CSV file (or optionally, a MySQL database!) that includes all of the information above. You're encouraged to use a "wide" structure similar to how the information appears above, so that you can practice tidying and transformations as described below. 1. Read the information from your .CSV file into R, and use tidyr and dplyr as needed to tidy and transform your data. 1. Perform analysis to compare the arrival delays for the two airlines. 1. Your code should be in an R Markdown file, posted to rpubs.com, and should include narrative descriptions of your data cleanup work, analysis, and conclusions.

2 Solution

2.1 Data Preparation

Let's load a CSV file containing the data and format described above.

```
df <- read.csv("https://raw.githubusercontent.com/himalayahall/DATA607/main/Assignment%2006%20-%20Worki
```

A quick look at the data frame structure shows that the CSV contains two unnamed columns, which have been labeled X and X.1 in the data frame.

```
str(df)
```

```
## 'data.frame':    5 obs. of  7 variables:
##  $ X           : chr  "Alaska" "" "" "AM WEST" ...
##  $ X.1         : chr  "on time" "delayed" "" "on time" ...
##  $ Los.Angeles : int  497 62 NA 694 117
```

```
##  $ Phoenix      : int  221 12 NA 4840 415
##  $ San.Diego    : int  212 20 NA 383 65
##  $ San.Francisco: int  503 102 NA 320 129
##  $ Seattle      : int  1841 305 NA 201 61
```

Let's rename these columns as *airline* and *arrival_status*, respectively.

```
df <- df %>% rename('airline' = 1, 'arrival_status' = 2)
str(df)
```

```
## 'data.frame':    5 obs. of  7 variables:
##  $ airline       : chr  "Alaska" "" "" "AM WEST" ...
##  $ arrival_status: chr  "on time" "delayed" "" "on time" ...
##  $ Los.Angeles   : int  497 62 NA 694 117
##  $ Phoenix       : int  221 12 NA 4840 415
##  $ San.Diego     : int  212 20 NA 383 65
##  $ San.Francisco : int  503 102 NA 320 129
##  $ Seattle       : int  1841 305 NA 201 61
```

Now, let's look at the data. We notice the following: (a) airline name is missing on **delayed** rows, and (b) there is an empty row between the two airlines.

```
head(df)
```

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

Based on above data observations, let's remove rows with NA or empty **arrival_status**.

```
df <- df %>%
            filter(! is.na(arrival_status) & str_length(arrival_status) > 0) # remove rows with NA stat
head(df)
```

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

And finally, let's fill in the airline name on **delayed** rows: first replace empty **airline** values with NA, then fill in missing values **down** the airline column.

```
df <- df %>%
        mutate(airline = ifelse(airline == "", NA, airline)) %>%
        fill(airline, .direction = "down") # fill in missing values
head(df)
```

```
##   airline arrival_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
```

We notice that this data frame in the **wide** format. Specifically, destination cities are given as columns. Let's convert this to a **longer** format.

```
df <- df %>%
            pivot_longer(!c("airline", "arrival_status"), # pivot all cols EXCEPT airline and arrival_s
                         names_to = "dest",               # destination names
                         values_to = "count")             # destination late flight values
head(df)
```

```
## # A tibble: 6 x 4
##   airline arrival_status dest          count
##   <chr>   <chr>          <chr>         <int>
## 1 Alaska  on time        Los.Angeles     497
## 2 Alaska  on time        Phoenix         221
## 3 Alaska  on time        San.Diego       212
## 4 Alaska  on time        San.Francisco   503
## 5 Alaska  on time        Seattle        1841
## 6 Alaska  delayed        Los.Angeles      62
```

2.2 Data Analysis

Let's extract delayed flight data.

```
delayed_flights <- df %>%
                    filter(df$arrival_status == "delayed")
delayed_flights
```

```
## # A tibble: 10 x 4
##    airline arrival_status dest          count
##    <chr>   <chr>          <chr>         <int>
##  1 Alaska  delayed        Los.Angeles      62
##  2 Alaska  delayed        Phoenix          12
##  3 Alaska  delayed        San.Diego        20
##  4 Alaska  delayed        San.Francisco   102
##  5 Alaska  delayed        Seattle         305
##  6 AM WEST delayed        Los.Angeles     117
##  7 AM WEST delayed        Phoenix         415
##  8 AM WEST delayed        San.Diego        65
##  9 AM WEST delayed        San.Francisco   129
## 10 AM WEST delayed        Seattle          61
```

Plot delayed flight data as a histogram. Visual inspection shows that, in general, AM WEST has greater number of delayed flights to more destinations compared to Alaska. The one exception destination is Seattle.

```
    ggp <- ggplot(data=delayed_flights, aes(x=dest, y=count, fill=airline))
    ggp <- ggp +  ggtitle('Delayed Flights') + theme(plot.title = element_text(hjust = 0.5))
    ggp <- ggp + geom_text(aes(label=count), vjust=-0.2,
                           position = position_dodge(0.9), size=3.5) +
                           scale_fill_brewer(palette="Paired") +
           geom_bar(stat="identity", position=position_dodge())
ggp
```