IMT 573 Lab: Working with Data

MILONI DESAI

October 17th, 2019

Collaborators

Objectives

Before beginning this assignment, please ensure you have access to R and RStudio; this can be on your own personal computer or on the IMT 573 R Studio Server.

1. Download the `lab3_working_with_data.rmd` file from Canvas or save a copy to your local directory on RStudio Server. Open `lab3_working_with_data.rmd` in RStudio and supply your solutions to the assignment by editing `lab3_working_with_data.rmd`.
2. Replace the "Insert Your Name Here" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. Be sure to include well-documented (e.g. commented) code chucks, figures, and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text. Be sure that each visualization adds value to your written explanation; avoid redundancy – you do no need four different visualizations of the same pattern.
4. Collaboration on problem sets is fun and useful, and we encourage it, but each student must turn in an individual write-up in their own words as well as code/work that is their own. Regardless of whether you work with others, what you turn in must be your own work; this includes code and interpretation of results. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.
5. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment.
6. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit PDF`, rename the knitted PDF file to `lab3_YourLastName_YourFirstName.pdf`, and submit the PDF file on Canvas.

Setup

In this lab you will need, at minimum, the following R packages.

```
# Load standard libraries
library(tidyverse)
library(nycflights13)
#data(weather)
#view(weather)
```

Problem 1: Data Cleaning

In this problem we will use the `weather.txt` data. Import the data into **R** and answer the following questions.

(a) What are the variables in this dataset? Describe what each variable measures.

```
names(weather)
```

```
##  [1] "origin"     "year"       "month"      "day"        "hour"
##  [6] "temp"       "dewp"       "humid"      "wind_dir"   "wind_speed"
## [11] "wind_gust"  "precip"     "pressure"   "visib"      "time_hour"
```

```
#The variables of interest here are id, year, month, element (tmax and tmin) and the temperature on eac
```

Hint: There are five variables of interest here.

(b) Tidy up the weather data such that each observation forms a row and each variable forms a column. Comment on your process and results.

```
#In tidying up the data we will use the year, month, id,element and we will make a 5th variable using t


weather_clean<-gather(weather,-c(1:4),"days","values")
#For the first step we gather our temperatures columns into
#a single row of values having the temperatures for
#each of the days, while the days column maps that temperature to the day from 1 to 31


weather_df<-spread(weather_clean,element,values)
#In the second step of cleaning we now want to map the temperature
#values to Tmax and Tmin the element column. Thus we spread the element
#column in to tmax and tmin and we fill the tempeaure values in those
#columns for each row of the days.
#View(weather_df)
```

Problem 2: Data Integration

Flight delays are often linked to weather conditions. We utilize both the `flights` and `weather` datasets from the `nycflights13` package to explore the following question: how does weather impact flights from NYC? Conduct a brief exploratory analysis to address the question posed above. In your EDA you might want to consider which weather variables are associated with impact on flights. Explain your choices in how you are measuring or evaluating impact on flights. Integrate the flights and weather datasets aid your analysis.

```
data(weather)
names(weather)
```

```
##  [1] "origin"     "year"       "month"      "day"        "hour"
##  [6] "temp"       "dewp"       "humid"      "wind_dir"   "wind_speed"
## [11] "wind_gust"  "precip"     "pressure"   "visib"      "time_hour"
```

```
#View(weather)
data(flights)
names(flights)
```

```
##  [1] "year"           "month"          "day"            "dep_time"
##  [5] "sched_dep_time" "dep_delay"      "arr_time"       "sched_arr_time"
##  [9] "arr_delay"      "carrier"        "flight"         "tailnum"
## [13] "origin"         "dest"           "air_time"       "distance"
```

```
## [17] "hour"           "minute"         "time_hour"
```

```r
#View(flights)
#Looking at the variables, we will join the two tables on
#the common variables between the two tables
#To better understand the effects of weather on flight delays, we
#join on the two datasets on the time_hour coulmn
#?merge
flight_weather<- merge(flights, weather, by = "time_hour", all.x= TRUE)
drops <- c("hour.y","year.y","day.y","month.y")
#When we join the two datasets on the time_hour column we get multiple
#issues with the combined dataset. We could want to ideally join the datasets,
#in a way where both the time hour and the orgin is used to join them.
#we also want to get rid of the duplicate year, month, day and hour columns
#in the new dataset.
#Once we are successfully able to join those, we would look sort the columns by max delay
#and see if the times with the maximum delay align with the max windspeed or precipitation
# we can also carry out further analysis to understand where the max weather changes are and see #if th
```