Metro Interstate Traffic Volume Data Set

Exploratory Data Analysis (EDA) - Example

*Use this notebook to practice your exploratory data analysis and visualization skills.*

**Original dataset** can be found at: <https://archive.ics.uci.edu/ml/datasets/Metro+Interstate+Traffic+Volume>

Version of the data used in this notebook is also available in [this repo](https://github.com/reisanar/datasets/blob/master/trafficMN.csv).

## Load packages

# use data science tools from the tidyverse  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✓ ggplot2 3.3.4 ✓ purrr 0.3.4  
## ✓ tibble 3.1.2 ✓ dplyr 1.0.6  
## ✓ tidyr 1.1.3 ✓ stringr 1.4.0  
## ✓ readr 1.4.0 ✓ forcats 0.5.1

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

## Read the data

This datasets contains hourly Interstate 94 Westbound traffic volume for [Minnesota (MN) Department of Transportation (DoT)](https://www.dot.state.mn.us/traffic/data/coll-methods.html) Automatic Traffic Recorder (ATR) station 301, roughly midway between Minneapolis and St Paul, MN. Hourly weather features and holidays are also included for impacts on traffic volume.

The dataset used in this notebook is from the [UCI Machine Learning Repository](https://archive.ics.uci.edu/ml/index.php), and it contains 48204 observations with 9 different attributes 9. A data dictionary is included below:

| Variable Name | Type | Description |
| --- | --- | --- |
| holiday | Categorical | US National holidays plus regional holiday, Minnesota State Fair |
| temp | Numeric | Average temperature in Kelvin |
| rain\_1h | Numeric | Amount in mm of rain that occurred in the hour |
| snow\_1h | Numeric | Amount in mm of snow that occurred in the hour |
| clouds\_all | Numeric | Percentage of cloud cover |
| weather\_main | Categorical | Short textual description of the current weather |
| weather\_description | Categorical | Longer textual description of the current weather |
| date\_time | DateTime | Hour of the data collected in local CST time |
| traffic\_volume | Numeric | Hourly I-94 ATR 301 reported westbound traffic volume |

*Central Standard Time (CST) is 6 hours behind Coordinated Universal Time (UTC).*

Below we read the .csv file using readr::read\_csv() (the readr package is part of the tidyverse)

traffic <- read\_csv("https://raw.githubusercontent.com/reisanar/datasets/master/trafficMN.csv",  
 col\_types = cols())

We can use the summary() function to get basic summary statistics on the different attributes of this dataset.

summary(traffic)

## holiday temp rain\_1h snow\_1h   
## Length:48204 Min. : 0.0 Min. : 0.000 Min. :0.0000000   
## Class :character 1st Qu.:272.2 1st Qu.: 0.000 1st Qu.:0.0000000   
## Mode :character Median :282.4 Median : 0.000 Median :0.0000000   
## Mean :281.2 Mean : 0.334 Mean :0.0002224   
## 3rd Qu.:291.8 3rd Qu.: 0.000 3rd Qu.:0.0000000   
## Max. :310.1 Max. :9831.300 Max. :0.5100000   
## clouds\_all weather\_main weather\_description  
## Min. : 0.00 Length:48204 Length:48204   
## 1st Qu.: 1.00 Class :character Class :character   
## Median : 64.00 Mode :character Mode :character   
## Mean : 49.36   
## 3rd Qu.: 90.00   
## Max. :100.00   
## date\_time traffic\_volume  
## Min. :2012-10-02 09:00:00 Min. : 0   
## 1st Qu.:2014-02-06 11:45:00 1st Qu.:1193   
## Median :2016-06-11 03:30:00 Median :3380   
## Mean :2016-01-05 10:46:16 Mean :3260   
## 3rd Qu.:2017-08-11 06:00:00 3rd Qu.:4933   
## Max. :2018-09-30 23:00:00 Max. :7280

# find dimensions of the dataset  
dim(traffic)

## [1] 48204 9

# use the function glimpse() to see a different type of summary  
glimpse(traffic)

## Rows: 48,204  
## Columns: 9  
## $ holiday <chr> "None", "None", "None", "None", "None", "None", "N…  
## $ temp <dbl> 288.28, 289.36, 289.58, 290.13, 291.14, 291.72, 29…  
## $ rain\_1h <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…  
## $ snow\_1h <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…  
## $ clouds\_all <dbl> 40, 75, 90, 90, 75, 1, 1, 1, 20, 20, 20, 1, 1, 1, …  
## $ weather\_main <chr> "Clouds", "Clouds", "Clouds", "Clouds", "Clouds", …  
## $ weather\_description <chr> "scattered clouds", "broken clouds", "overcast clo…  
## $ date\_time <dttm> 2012-10-02 09:00:00, 2012-10-02 10:00:00, 2012-10…  
## $ traffic\_volume <dbl> 5545, 4516, 4767, 5026, 4918, 5181, 5584, 6015, 57…

Notice that the summary returned by using the glimpse() function, also includes the total number of observations (rows), total number of variables/attributes (columns), and the *type* of each column (as it was interpreted by the readr::read\_csv() function.

It is also important to point out that the variable date\_time is of type ddtm, and R provides a great set of tools to [deal with date-time information](https://r4ds.had.co.nz/dates-and-times.html) thorugh the use of the lubridate package (part of the tidyverse!).

# load the lubridate package as well - not a core package, so it needs to be loaded  
suppressMessages(library(lubridate))

## Working with dates and times

The lubridate package allows us to esily handle date and time data, including extracting relevant information from a date-time variable (e.g., day of the week, month, hour, etc.) and even performing arithmetic operations that take into account time zones, daylight savings, and more (e.g., answering *how many days have passes since X date?*)

Let us explore some of this functionality by extracting the month, day, year, hour, and minute, from the column date\_time. Additionally, we will transform the temperature values from Kelvin to Fahrenheit (using the relation F = (0K − 273.15) × 9/5 + 32)

traffic\_MN<- traffic %>%   
 mutate(month = month(date\_time, label = T),   
 day = day(date\_time),   
 year = year(date\_time),  
 hour = hour(date\_time),   
 minute = minute(date\_time),   
 day\_of\_week = wday(date\_time, label = TRUE),   
 temp\_Fah = (temp - 273.15)\*9/5 + 32  
 )  
# check data   
traffic\_MN

## # A tibble: 48,204 x 16  
## holiday temp rain\_1h snow\_1h clouds\_all weather\_main weather\_description  
## <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr>   
## 1 None 288. 0 0 40 Clouds scattered clouds   
## 2 None 289. 0 0 75 Clouds broken clouds   
## 3 None 290. 0 0 90 Clouds overcast clouds   
## 4 None 290. 0 0 90 Clouds overcast clouds   
## 5 None 291. 0 0 75 Clouds broken clouds   
## 6 None 292. 0 0 1 Clear sky is clear   
## 7 None 293. 0 0 1 Clear sky is clear   
## 8 None 294. 0 0 1 Clear sky is clear   
## 9 None 294. 0 0 20 Clouds few clouds   
## 10 None 293. 0 0 20 Clouds few clouds   
## # … with 48,194 more rows, and 9 more variables: date\_time <dttm>,  
## # traffic\_volume <dbl>, month <ord>, day <int>, year <dbl>, hour <int>,  
## # minute <int>, day\_of\_week <ord>, temp\_Fah <dbl>

Just with the new columns above, we could do an analysis of traffic volume by day of the week, month, hour, and more.

## Aggregating data

Let us try to create a summary that tell us about the total traffic volume for each week of the different years for which we have data. A useful function for this is the [lubridate::week() function](https://lubridate.tidyverse.org/reference/week.html) that returns the number of complete seven day periods that have occurred between the date and January 1st, plus one.

traffic\_week <- traffic\_MN %>%  
 mutate(week\_of\_year = week(date\_time)) %>%  
 group\_by(week\_of\_year, year) %>%  
 summarize(total\_ATR = sum(traffic\_volume, na.rm = T), .groups = "drop")

We now use the weekly traffic data to create a trend line for the years 2013 to 2018 (i.e., ignore data from 2012 for this visualization)

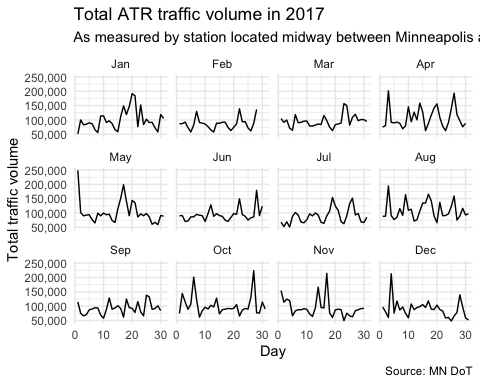
traffic\_week %>%  
 filter(year != 2012) %>%  
 ggplot() +   
 geom\_line(aes(x = week\_of\_year, y = total\_ATR)) +   
 facet\_wrap(vars(year)) +   
 labs(x = "Week of year", y = "Total traffic volume",   
 title = "Total ATR 301 reported westbound traffic volume",   
 subtitle = "Station located midway between Minneapolis and St Paul, MN",   
 caption = 'Source: MN DoT') +   
 scale\_y\_continuous(labels = scales::comma) +   
 theme\_minimal()

In the creation of the above plot we utilized the *faceting* (subplots) technique (using facet\_wrap()) to compare different years (this is also sometimes called *“small multiples”*). Notice also the use of the scales::comma function to improve the readability of the charts. Notice also the custom labels for the both axes, title, subtitle, and caption.

**Practice**: what other type of comparison can you create? For example, if we focus on a particular year, could you analyze what occurs in the different months of the year in terms of the total traffic volume?

You could compare the traffic during different months of a given year using the code below:

traffic\_MN %>%  
 group\_by(day, month, year) %>%  
 summarize(total\_ATR = sum(traffic\_volume, na.rm = T), .groups = "drop") %>%  
 filter(year == 2017) %>%  
 ggplot() +   
 geom\_line(aes(x = day, y = total\_ATR)) +   
 facet\_wrap(vars(month)) +   
 labs(x = "Day", y = "Total traffic volume",   
 title = "Total ATR traffic volume in 2017",   
 subtitle = "As measured by station located midway between Minneapolis and St Paul, MN",   
 caption = 'Source: MN DoT') +   
 scale\_y\_continuous(labels = scales::comma) +   
 theme\_minimal()



**Practice**: how would you incorporate weather information in this type of analysis?

Some other things that one could consider when analyzing this type of data include:

* Seasonality (Yearly, Weekly, Daily, Holiday Related, Moving Holidays!)
* Change in behavior
* Change in population
* Construction seasons and improvements to roads
* Financial markets trends
* Big events (e.g., the Super Bowl - 2018/02/04 @ the U.S. Bank Stadium, Minneapolis)

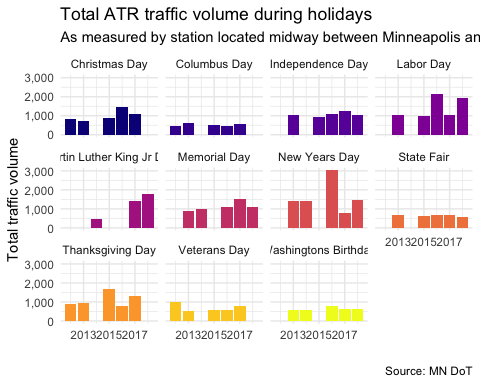
## Holidays

Analyze the traffic during holidays in multiple years.

traffic\_MN %>%  
 group\_by(holiday, year) %>%  
 summarize(total\_ATR = sum(traffic\_volume, na.rm = T), .groups = "drop")

## # A tibble: 60 x 3  
## holiday year total\_ATR  
## <chr> <dbl> <dbl>  
## 1 Christmas Day 2012 803  
## 2 Christmas Day 2013 712  
## 3 Christmas Day 2015 894  
## 4 Christmas Day 2016 1464  
## 5 Christmas Day 2017 1092  
## 6 Columbus Day 2012 455  
## 7 Columbus Day 2013 615  
## 8 Columbus Day 2015 494  
## 9 Columbus Day 2016 484  
## 10 Columbus Day 2017 549  
## # … with 50 more rows

traffic\_MN %>%  
 group\_by(holiday, year) %>%  
 summarize(total\_ATR = sum(traffic\_volume, na.rm = T), .groups = "drop") %>%  
 filter(holiday != "None") %>%  
 ggplot() +   
 geom\_col(aes(x = year, y = total\_ATR, fill = holiday)) +   
 facet\_wrap(vars(holiday)) +   
 labs(x = "", y = "Total traffic volume",   
 title = "Total ATR traffic volume during holidays",   
 subtitle = "As measured by station located midway between Minneapolis and St Paul, MN",   
 caption = 'Source: MN DoT') +   
 scale\_y\_continuous(labels = scales::comma) +   
 scale\_fill\_viridis\_d(option = "plasma") +   
 theme\_minimal() +   
 theme(legend.position = "none")



**Practice**: explore the relationship between other variables. Can you characterize the trends you observe? Do they make sense? What type of other data summaries would be useful in this case?

### Citation

*(as described in UCI Machine Learning repository page)*

* Traffic data from MN Department of Transportation
* Weather data from OpenWeatherMap
* John Hogue, Social Data Science & General Mills