R Notebook

This is an R Markdown (http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

Add tools to ease removing duplicates and tools for data cleaning,
manipulation, and visualization in R.

Add libraries that enable removing duplicates and manipulating texts, and other formatting tricks

```
options(repos = "http://cran.r-project.org")  # Example mirror URL

install.packages("tidyverse", repos="http://cran.r-project.org")
```

```
## Installing package into 'C:/Users/travi/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## package 'tidyverse' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\travi\AppData\Local\Temp\RtmpcN9eEw\downloaded_packages
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.3.3
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.3     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.3     ✓ tibble    3.2.1
## ✓ lubridate 1.9.2     ✓ tidyr     1.3.0
## ✓ purrr     1.0.2
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(dplyr) #Simplify manipulation tasks

#Confirm working directory
#getwd()
#Modify working directory if necessary
#setwd()

#Store current timestamp as 'now' to use in file name
now <- format(Sys.time(), format = "%m.%d.%Y.%Hh %Mm %Ss")
#print(now)
```

Connect to data sources Grab primary data source: 311 data. Considered connecting to API but each day results could change and make my analysis look inaccurate.

```
original_main_df <- read.csv("cosa_311_Service_Requests_March2024_mung.csv", na.strings=c("","  ","NA"), stringsAsFactors=FAL
SE)

#Grab police response data that I manually scraped and then attempted to add geocode information on
police_jan2019_df <- read.csv("geocded911calls.csv", na.strings=c("","  ","NA"), stringsAsFactors=FALSE)

#Grab 78205 Weather dataset that I paid $10 for
weather_df <- read.csv("openweathermap_78205_Jan1980_Feb2024.csv", na.strings=c("","  ","NA"), stringsAsFactors=FALSE)
```

Re-Run Code from HERE

If need to reset the datasets based on unintended changes can start here instead of reconnecting to the source to reset.

```
main_df <- original_main_df
dim(main_df)
```

```
## [1] 606061     18
```

```
dim(police_jan2019_df)
```

```
## [1] 83040    16
```

```
#Remove duplicates rows. These are rows that have exact same values in all fields
main_df <- unique(main_df[,])
police_jan2019_df <- unique(police_jan2019_df[,])


#Deduplicate on a specifc column if necessary
#main_df <- main_df[!duplicated(main_df$INCIDENT_NUMBER), ]
dim(main_df)
```

```
## [1] 606061     18
```

```
dim(police_jan2019_df)
```

```
## [1] 83040    16
```

```
#view all column names in each dataset
names(main_df)
```

```
##  [1] "X_id"                 "Category"             "CASEID"
##  [4] "OPENEDDATETIME"       "SLA_Date"             "CLOSEDDATETIME"
##  [7] "Late..Yes.No."        "Dept"                 "REASONNAME"
## [10] "TYPENAME"             "CaseStatus"           "SourceID"
## [13] "OBJECTDESC"           "Council.District"     "XCOORD"
## [16] "YCOORD"               "Report.Starting.Date" "Report.Ending.Date"
```

```
names(police_jan2019_df)
```

```
##  [1] "id"                  "INCIDENT_NUMBER"    "CATEGORY"
##  [4] "PROBLEM_TYPE"        "RESPONSE_DATETIME"  "RESPONSE_DATE"
##  [7] "RESPONSE_TIMEOFDAY" "ADDRESS"            "HOA"
## [10] "SCHOOL_DISTRICT"     "COUNCIL_DISTRICT"   "ZIPCODE"
## [13] "friendly_address"    "lon"                "lat"
## [16] "geoAddress"
```

```
names(weather_df)
```

```
##  [1] "dt"                  "dt_iso_UTC"          "dt_iso_dmyh"
##  [4] "dt_iso_date"         "dt_iso_time"         "timezone"
##  [7] "zipcode"             "lat"                 "lon"
## [10] "temp"                "visibility"          "dew_point"
## [13] "feels_like"          "temp_min"            "temp_max"
## [16] "pressure"            "humidity"            "wind_speed"
## [19] "wind_deg"            "wind_gust"           "rain_1h"
## [22] "rain_3h"             "snow_1h"             "snow_3h"
## [25] "clouds_all"          "weather_id"          "weather_main"
## [28] "weather_description" "weather_icon"
```

```
#View structure and class of datasets
str(main_df)
```

```
## 'data.frame':    606061 obs. of  18 variables:
##  $ X_id                : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Category            : chr  "Traffic Signals and Signs" "Solid Waste Services" "Traffic Signals and Signs" "Traffic Sig
nals and Signs" ...
##  $ CASEID              : int  1014001765 1014303843 1014504120 1014549690 1014569491 1014633806 1014679731 1014680784 101
4702023 1014709576 ...
##  $ OPENEDDATETIME      : chr  "11/3/2017" "3/7/2018" "5/11/2018" "5/27/2018" ...
##  $ SLA_Date            : chr  "9/22/2021" "3/16/2018" "10/31/2018" "11/15/2018" ...
##  $ CLOSEDDATETIME      : chr  "3/16/2023" "3/8/2023" "3/16/2023" "3/16/2023" ...
##  $ Late..Yes.No.       : chr  "YES" "YES" "YES" "YES" ...
##  $ Dept                : chr  "Public Works" "Solid Waste Management" "Public Works" "Public Works" ...
##  $ REASONNAME          : chr  "Signals" "Waste Collection" "Traffic Engineering Design" "Traffic Engineering Design" ...
##  $ TYPENAME            : chr  "Signal Timing Modification By Engineer" "Additional Cart Request" "Traffic Signal New Requ
est" "Traffic Signal New Request" ...
##  $ CaseStatus          : chr  "Closed" "Closed" "Closed" "Closed" ...
##  $ SourceID            : chr  "Constituent Call" "Constituent Call" "Constituent Call" "Constituent Call" ...
##  $ OBJECTDESC          : chr  "PATRON and POTEET JDTN FY" "1819  POPLAR ST W, San Antonio, 78207" "HILDEBRAND E and NEW B
RNFLS N" "PERRIN BEITEL and SUNSHADOW ST" ...
##  $ Council.District    : int  4 1 2 10 1 7 10 9 2 9 ...
##  $ XCOORD              : int  2111686 2121442 2140051 2155827 2131743 2075934 2176790 2144102 2155408 2141851 ...
##  $ YCOORD              : int  13670865 13708006 13717304 13738349 13703551 13733809 13764615 13784362 13728538 13777050
...
##  $ Report.Starting.Date: chr  "3/2/2023" "3/2/2023" "3/2/2023" "3/2/2023" ...
##  $ Report.Ending.Date  : chr  "3/2/2024" "3/2/2024" "3/2/2024" "3/2/2024" ...
```

```
class(main_df)
```

```
## [1] "data.frame"
```

```
str(police_jan2019_df)
```

```
## 'data.frame':    83040 obs. of  16 variables:
##  $ id                : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ INCIDENT_NUMBER   : chr  "SAPD-2019-0000004" "SAPD-2019-0000006" "SAPD-2019-0000009" "SAPD-2019-0000011" ...
##  $ CATEGORY          : chr  "Other Calls" "Other Calls" "Other Calls" "Other Calls" ...
##  $ PROBLEM_TYPE      : chr  "Disturbance Fireworks" "Disturbance Fireworks" "SAPD Emergency Call" "Disturbance Fireworks"
...
##  $ RESPONSE_DATETIME : chr  "1/1/2019 0:00" "1/1/2019 0:00" "1/1/2019 0:00" "1/1/2019 0:01" ...
##  $ RESPONSE_DATE     : chr  "1/1/2019" "1/1/2019" "1/1/2019" "1/1/2019" ...
##  $ RESPONSE_TIMEOFDAY: chr  "0:00:02" "0:00:31" "0:00:48" "0:01:17" ...
##  $ ADDRESS           : chr  "3500 Stonehaven Dr" "3600 Callaghan Rd" "9500 Woodland Hills" "Fair Ave / Clark Ave" ...
##  $ HOA               : chr  "Vance Jackson" NA "Great Northwest" NA ...
##  $ SCHOOL_DISTRICT   : chr  "Northside ISD" "Northside ISD" "Northside ISD" "San Antonio ISD" ...
##  $ COUNCIL_DISTRICT  : int  8 6 6 3 5 1 6 10 3 3 ...
##  $ ZIPCODE           : chr  "78230" "78238" "78250" "78223" ...
##  $ friendly_address  : chr  "3500 Stonehaven Dr, SAN ANTONIO, TX, USA" "3600 Callaghan Rd, SAN ANTONIO, TX, USA" "9500 Wo
odland Hills, SAN ANTONIO, TX, USA" "Fair Ave and Clark Ave, SAN ANTONIO, TX, USA" ...
##  $ lon               : num  -98.6 -98.6 -98.7 -98.4 -98.5 ...
##  $ lat               : num  29.5 29.5 29.5 29.4 29.4 ...
##  $ geoAddress        : chr  "3500 stonehaven rd, san antonio, tx 78230, usa" "3600 callaghan rd, san antonio, tx 78228, u
sa" "9500 woodland hills, san antonio, tx 78250, usa" "clark ave & fair ave, san antonio, tx 78223, usa" ...
```

```
class(police_jan2019_df)
```

```
## [1] "data.frame"
```

```
str(weather_df)
```

```
## 'data.frame':    405413 obs. of  29 variables:
##  $ dt                : int  315532800 315536400 315540000 315543600 315547200 315550800 315554400 315558000 315561600 31
5565200 ...
##  $ dt_iso_UTC        : chr  "1980-01-01 00:00:00 +0000 UTC" "1980-01-01 01:00:00 +0000 UTC" "1980-01-01 02:00:00 +0000 U
TC" "1980-01-01 03:00:00 +0000 UTC" ...
##  $ dt_iso_dmyh       : chr  "1/1/1980 0:00" "1/1/1980 1:00" "1/1/1980 2:00" "1/1/1980 3:00" ...
##  $ dt_iso_date       : chr  "1/1/1980" "1/1/1980" "1/1/1980" "1/1/1980" ...
##  $ dt_iso_time       : chr  "0:00" "1:00" "2:00" "3:00" ...
##  $ timezone          : int  -21600 -21600 -21600 -21600 -21600 -21600 -21600 -21600 -21600 -21600 ...
##  $ zipcode           : int  78205 78205 78205 78205 78205 78205 78205 78205 78205 78205 ...
##  $ lat               : num  29.4 29.4 29.4 29.4 29.4 ...
##  $ lon               : num  -98.5 -98.5 -98.5 -98.5 -98.5 ...
##  $ temp              : num  56.1 53.6 46.7 42.2 39.8 ...
##  $ visibility        : int  10000 NA NA 10000 NA NA 10000 10000 NA 10000 ...
##  $ dew_point         : num  32.1 36.9 37.4 31.7 34.2 ...
##  $ feels_like        : num  53.3 51.2 44.9 38.5 36.7 ...
##  $ temp_min          : num  54.5 50.8 45.2 40.3 39 ...
##  $ temp_max          : num  57.3 56.6 48.2 44.2 40.7 ...
##  $ pressure          : int  1022 1022 1022 1023 1023 1024 1024 1023 1023 1022 ...
##  $ humidity          : int  40 53 70 66 80 83 78 75 82 77 ...
##  $ wind_speed        : num  5.82 3.29 4.14 5.82 4.45 4.25 4.7 3.36 4.52 5.82 ...
##  $ wind_deg          : int  340 30 37 330 41 39 360 360 85 310 ...
##  $ wind_gust         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ rain_1h           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ rain_3h           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ snow_1h           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ snow_3h           : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ clouds_all        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ weather_id        : int  800 800 800 800 800 800 800 800 800 800 ...
##  $ weather_main      : chr  "Clear" "Clear" "Clear" "Clear" ...
##  $ weather_description: chr  "sky is clear" "sky is clear" "sky is clear" "sky is clear" ...
##  $ weather_icon      : chr  "01n" "01n" "01n" "01n" ...
```

```
class(weather_df)
```

```
## [1] "data.frame"
```

##Convert all of the dates that came over as ‘factors’ to ‘dates’ ##

```
main_df$OPENEDDATETIME <- as.Date(main_df$OPENEDDATETIME, "%d/%m/%Y")
main_df$CLOSEDDATETIME <- as.Date(main_df$CLOSEDDATETIME, "%d/%m/%Y")
main_df$SLA_Date <- as.Date(main_df$SLA_Date, "%d/%m/%Y")
main_df$CASEID <- as.integer(main_df$CASEID) #Store CaseID as integer
main_df$Council.District <- as.factor(main_df$Council.District) #Treat council district as factor and not a number
main_df$Council.DistrictNum <- as.numeric(main_df$Council.District) #Separate field to treat district as number when needed

#Rename ugly names
colnames(main_df)[colnames(main_df) == "Late..Yes.No."] <- "Late"
```

Confirm conversions of field types

```
#confirm conversion
class(main_df$OPENEDDATETIME)
```

```
## [1] "Date"
```

```
class(main_df$CLOSEDDATETIME)
```

```
## [1] "Date"
```

```
class(main_df$CASEID)
```

```
## [1] "integer"
```

```
class(main_df$Council.District)
```

```
## [1] "factor"
```

```
class(main_df$Council.DistrictNum)
```

```
## [1] "numeric"
```

Create DAYSTOCLOSE, SLA Length, and PCTofSLAtime (aka percentage of allotted SLA time that it took to close the case)

```
main_df$DAYSTOCLOSE <- as.numeric((main_df$CLOSEDDATETIME - main_df$OPENEDDATETIME))


main_df$SLA_Length <- as.numeric((main_df$SLA_Date - main_df$OPENEDDATETIME))

main_df$PCTofSLAtime <- as.numeric((main_df$DAYSTOCLOSE / main_df$SLA_Length ))
```

Create a function to format field as % instead of decimal

```
percentage <- function(x, digits = 4) {
  paste0(formatC(100 * x, format = "f", digits = digits), "%")
}

main_df$PCT_SLAtime <- percentage(main_df$PCTofSLAtime)
```

Preview values

```
#Take a peak at the first/last several values for each field
head(main_df, n = 30)
```

	X_id <int>	Category <chr>	CASEID <int>	OPENEDDATETIME <date>	SLA_Date <date>	CLOSEDDATETIME <date>	Late <chr>
1	1	Traffic Signals and Signs	1014001765	2017-03-11	<NA>	<NA>	YES
2	2	Solid Waste Services	1014303843	2018-07-03	<NA>	2023-08-03	YES
3	3	Traffic Signals and Signs	1014504120	2018-11-05	<NA>	<NA>	YES
4	4	Traffic Signals and Signs	1014549690	<NA>	<NA>	<NA>	YES
5	5	Solid Waste Services	1014569491	2018-03-06	2018-11-06	<NA>	YES
6	6	Graffiti	1014633806	<NA>	<NA>	<NA>	YES
7	7	Traffic Signals and Signs	1014679731	2018-11-07	<NA>	<NA>	YES
8	8	Traffic Signals and Signs	1014680784	2018-11-07	<NA>	<NA>	YES
9	9	Traffic Signals and Signs	1014702023	<NA>	2019-04-01	2023-07-08	YES
10	10	Traffic Signals and Signs	1014709576	<NA>	2019-08-01	2023-07-08	YES

1-10 of 30 rows | 1-8 of 24 columns

Previous **1** 2 3 Next

```
tail(main_df)
```

	X_id <int>	Category <chr>	CASEID <int>	OPENEDDATETIME <date>	SLA_Date <date>	CLOSEDDATETIME <date>	Late <chr>
606056	606056	Animals	1019442049	2024-01-03	2024-11-03	<NA>	NO
606057	606057	Animals	1019442050	2024-01-03	2024-11-03	<NA>	NO
606058	606058	Animals	1019442051	2024-01-03	2024-11-03	<NA>	NO
606059	606059	Animals	1019442052	2024-01-03	2024-02-03	2024-01-03	NO
606060	606060	Animals	1019442053	2024-01-03	2024-02-03	<NA>	NO

	X_id	Category	CASEID	OPENEDDATETIME	SLA_Date	CLOSEDDATETIME	Late
	<int>	<chr>	<int>	<date>	<date>	<date>	<chr>
606061	606061	Property Maintenance	1019442054	2024-01-03	2024-08-05	<NA>	NO

6 rows | 1-8 of 24 columns

Remove rows that have nonsensical values

```
#Sort data frame in descending order by the number of DAYSTOCLOSE
main_df <- main_df %>% arrange(desc(DAYSTOCLOSE))


#Remove observations where DAYSTOCLOSE is negative. Accounts for 6,951 rows
main_df <- main_df %>% filter(is.na(DAYSTOCLOSE) | DAYSTOCLOSE >= 0)

#Remove observations where DAYSTOCLOSE is negative. Accounts for 11,689 rows.
#Total of about 3.3% of the original dataset removed due to errors
main_df <- main_df %>% filter(is.na(SLA_Length) | SLA_Length >= 0)
```

```
#Calculate various numerical summaries related to distribution and center for each field in the data set.
summary(main_df)
```

```
##       X_id          Category             CASEID          OPENEDDATETIME
##  Min.   :     1   Length:587421      Min.   :1.014e+09   Min.   :2017-03-11
##  1st Qu.:152941   Class :character   1st Qu.:1.019e+09   1st Qu.:2023-03-11
##  Median :305199   Mode  :character   Median :1.019e+09   Median :2023-07-08
##  Mean   :304701                      Mean   :1.019e+09   Mean   :2023-07-17
##  3rd Qu.:457090                      3rd Qu.:1.019e+09   3rd Qu.:2023-11-08
##  Max.   :606061                      Max.   :1.019e+09   Max.   :2024-12-02
##                                                          NA's   :368042
##     SLA_Date          CLOSEDDATETIME           Late
##  Min.   :2018-11-06   Min.   :2023-01-04   Length:587421
##  1st Qu.:2023-05-04   1st Qu.:2023-04-12   Class :character
##  Median :2023-09-03   Median :2023-08-03   Mode  :character
##  Mean   :2023-09-09   Mean   :2023-08-22
##  3rd Qu.:2024-01-05   3rd Qu.:2023-11-11
##  Max.   :2025-12-06   Max.   :2024-12-02
##  NA's   :357058       NA's   :384732
##      Dept            REASONNAME          TYPENAME          CaseStatus
##  Length:587421      Length:587421      Length:587421      Length:587421
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    SourceID          OBJECTDESC        Council.District     XCOORD
##  Length:587421      Length:587421      5      : 90066   Min.   :2030256
##  Class :character   Class :character   1      : 81597   1st Qu.:2104412
##  Mode  :character   Mode  :character   2      : 79925   Median :2121513
##                                        3      : 78230   Mean   :2121220
##                                        4      : 61010   3rd Qu.:2139655
##                                        7      : 54099   Max.   :2242563
##                                        (Other):142494   NA's   :9
##      YCOORD         Report.Starting.Date Report.Ending.Date Council.DistrictNum
##  Min.   :13599140   Length:587421        Length:587421      Min.   : 1.000
##  1st Qu.:13691981   Class :character     Class :character   1st Qu.: 3.000
##  Median :13707052   Mode  :character     Mode  :character   Median : 5.000
##  Mean   :13710877                                          Mean   : 5.552
##  3rd Qu.:13728721                                          3rd Qu.: 7.000
##  Max.   :13838129                                          Max.   :11.000
##  NA's   :9
##   DAYSTOCLOSE       SLA_Length      PCTofSLAtime    PCT_SLAtime
##  Min.   :   0.0   Min.   :   0.0   Min.   :0.0      Length:587421
##  1st Qu.:   0.0   1st Qu.:  62.0   1st Qu.:0.0      Class :character
##  Median :  31.0   Median : 125.0   Median :0.3      Mode  :character
##  Mean   :  59.9   Mean   : 142.4   Mean   :Inf
##  3rd Qu.:  91.0   3rd Qu.: 185.0   3rd Qu.:0.6
##  Max.   :1857.0   Max.   :1489.0   Max.   :Inf
##  NA's   :450722   NA's   :468942   NA's   :506001
```

Merge dataframes on dt_iso_date and OPENED_DATE (assuming they are in similar format)

```
# Group by dt_iso_date and filter for the row with highest temp_max
weather_df$dt_iso_date <- as.Date(weather_df$dt_iso_date, "%d/%m/%Y")

weather_df <- weather_df %>%
  group_by(dt_iso_date) %>%
  #filter(temp_max == max(temp_max) & dt_iso_time == max(dt_iso_time) ) %>%
  arrange(desc(temp_max)) %>%  # Arrange by temp_max (highest first)
  top_n(1)  # Keep only the top row (highest temp_max)
```

```
## Selecting by weather_icon
```

```
weather_df <- weather_df %>%
  group_by(dt_iso_date) %>%
  #filter(temp_max == max(temp_max) & dt_iso_time == max(dt_iso_time) ) %>%
  arrange(desc(dt_iso_UTC)) %>%  # Arrange by temp_max (highest first)
  top_n(1)  # Keep only the top row (highest temp_max)
```

```
## Selecting by weather_icon
```

```
deduplicated_df <- weather_df %>%
  distinct(dt_iso_date)

# Optionally, add remaining columns (if any)
if (ncol(weather_df) > 1) {
  # Group by dt_iso_date and keep the first row (similar to method 1)
  deduplicated_df <- weather_df %>%
    group_by(dt_iso_date) %>%
    slice_head(n=1)
}

deduplicated_df$dt_iso_date <- as.Date(deduplicated_df$dt_iso_date, "%d/%m/%Y")
merged_df <- merge(main_df, deduplicated_df, by.x = "OPENEDDATETIME", by.y = "dt_iso_date", all.x = TRUE)
main_df <- merged_df
tail(main_df)
```

	OPENEDDATETIME	X_id	Category	CASEID	SLA_Date	CLOSEDDATETIME	Late
	<date>	<int>	<chr>	<int>	<date>	<date>	<chr>
587416	<NA>	370669	Solid Waste Services	1018962992	<NA>	<NA>	NO
587417	<NA>	370626	Property Maintenance	1018962923	<NA>	<NA>	NO
587418	<NA>	369534	Property Maintenance	1018961246	<NA>	<NA>	YES
587419	<NA>	369896	Streets & Infrastructure	1018961794	<NA>	<NA>	NO
587420	<NA>	369897	Property Maintenance	1018961798	<NA>	<NA>	NO
587421	<NA>	369898	Property Maintenance	1018961797	<NA>	<NA>	NO

6 rows | 1-8 of 52 columns

Explore correlations

```
#Check for correlation in various numeric fields
cor(main_df$Council.DistrictNum, main_df$DAYSTOCLOSE, use = "complete.obs")
```

```
## [1] 0.004512774
```

```
cor(main_df$SLA_Length, main_df$DAYSTOCLOSE, use = "complete.obs")
```

```
## [1] 0.2693264
```

```
cor(main_df$CASEID, main_df$Council.DistrictNum, use = "complete.obs")
```

```
## [1] 0.008665759
```

```
cor(main_df$Council.DistrictNum, main_df$temp_max, use = "complete.obs")
```

```
## [1] 0.004393366
```

```
cor(main_df$DAYSTOCLOSE, main_df$temp_max, use = "complete.obs")
```

```
## [1] 0.007050427
```

```
cor(main_df$SLA_Length, main_df$temp_max, use = "complete.obs")
```

```
## [1] -0.08194322
```

Explore how data is distributed

```
#Average and distributions by district
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "1"])
```

```
## [1]    0    0   31   62 1857
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "2"])
```

```
## [1]    0    0   31   92 1428
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "3"])
```

```
## [1]    0    0   31   91 1334
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "4"])
```

```
## [1]    0    0   31   92 1607
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "5"])
```

```
## [1]    0    0   31   62 1639
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "6"])
```

```
## [1]    0    0   31   62 1308
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "7"])
```

```
## [1]   0   0  31  92 982
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "8"])
```

```
## [1]    0    0   31   92 1062
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "9"])
```

```
## [1]    0    0   31   91 1670
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "10"])
```

```
## [1]    0    0   31   92 1762
```

```
fivenum(main_df$DAYSTOCLOSE[main_df$Council.District == "0"])
```

```
## [1]    0    0    0   31 1434
```

```
mean(main_df$DAYSTOCLOSE, na.rm = TRUE)
```

```
## [1] 59.86102
```

```
#D10 AND D4 HAVE HIGHER 3RD QUARTILES
#D6 and D1 have lower
```

Hypthesis testings

```
# Test if there is a statistically significant difference between case closing times by council districts
#t.test(DAYSTOCLOSE ~ Late, data = main_df)
main_df %>%
  select(DAYSTOCLOSE, Council.District) %>%
  filter(Council.District %in% c("1","2")) %>%
  drop_na(DAYSTOCLOSE) %>%
  t.test(DAYSTOCLOSE ~ Council.District, data = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  DAYSTOCLOSE by Council.District
## t = -9.4747, df = 36543, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 1 and group 2 is not equal to 0
## 95 percent confidence interval:
##  -10.989351  -7.221967
## sample estimates:
## mean in group 1 mean in group 2
##        57.08079        66.18645
```

```
main_df %>%
  select(DAYSTOCLOSE, Council.District) %>%
  filter(Council.District %in% c("1","10")) %>%
  drop_na(DAYSTOCLOSE) %>%
  t.test(DAYSTOCLOSE ~ Council.District, data = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  DAYSTOCLOSE by Council.District
## t = -7.5685, df = 21058, p-value = 3.931e-14
## alternative hypothesis: true difference in means between group 1 and group 10 is not equal to 0
## 95 percent confidence interval:
##  -10.776102  -6.342691
## sample estimates:
##  mean in group 1 mean in group 10
##         57.08079         65.64019
```

```
main_df %>%
  select(DAYSTOCLOSE, Council.District) %>%
  filter(Council.District %in% c("5","10")) %>%
  drop_na(DAYSTOCLOSE) %>%
  t.test(DAYSTOCLOSE ~ Council.District, data = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  DAYSTOCLOSE by Council.District
## t = -9.7176, df = 17927, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 5 and group 10 is not equal to 0
## 95 percent confidence interval:
##  -12.399138  -8.236767
## sample estimates:
##  mean in group 5 mean in group 10
##         55.32223         65.64019
```

```
main_df %>%
  select(DAYSTOCLOSE, Council.District) %>%
  filter(Council.District %in% c("2","10")) %>%
  drop_na(DAYSTOCLOSE) %>%
  t.test(DAYSTOCLOSE ~ Council.District, data = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  DAYSTOCLOSE by Council.District
## t = 0.48551, df = 20586, p-value = 0.6273
## alternative hypothesis: true difference in means between group 2 and group 10 is not equal to 0
## 95 percent confidence interval:
##  -1.659080  2.751605
## sample estimates:
##  mean in group 2 mean in group 10
##         66.18645         65.64019
```

```
#Difference between D2 and D10 resolutions were the LEAST statistically significant result

main_df %>%
  select(DAYSTOCLOSE, Council.District) %>%
  filter(Council.District %in% c("2","5")) %>%
  drop_na(DAYSTOCLOSE) %>%
  t.test(DAYSTOCLOSE ~ Council.District, data = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  DAYSTOCLOSE by Council.District
## t = 12.365, df = 35987, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 2 and group 5 is not equal to 0
## 95 percent confidence interval:
##   9.142067 12.586362
## sample estimates:
## mean in group 2 mean in group 5
##        66.18645        55.32223
```

```
#Difference between D2 and D5 resolutions were the most statistically significant result

main_df %>%
  select(DAYSTOCLOSE, Council.District) %>%
  filter(Council.District %in% c("2","9")) %>%
  drop_na(DAYSTOCLOSE) %>%
  t.test(DAYSTOCLOSE ~ Council.District, data = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  DAYSTOCLOSE by Council.District
## t = 5.0694, df = 11451, p-value = 4.053e-07
## alternative hypothesis: true difference in means between group 2 and group 9 is not equal to 0
## 95 percent confidence interval:
##  3.986760 9.013555
## sample estimates:
## mean in group 2 mean in group 9
##        66.18645        59.68629
```

More hypothesis testing

```
#Hypothesis test on Close Time vs type of issue

main_df %>%
  select(DAYSTOCLOSE, Category) %>%
  filter(Category %in% c("Traffic Signals and Signs","Solid Waste Services")) %>%
  drop_na(DAYSTOCLOSE) %>%
  t.test(DAYSTOCLOSE ~ Category, data = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  DAYSTOCLOSE by Category
## t = 2.2407, df = 6211, p-value = 0.02508
## alternative hypothesis: true difference in means between group Solid Waste Services and group Traffic Signals and Signs i
s not equal to 0
## 95 percent confidence interval:
##  0.5393908 8.0834211
## sample estimates:
##      mean in group Solid Waste Services mean in group Traffic Signals and Signs
##                                60.53918                                56.22777
```

```
main_df %>%
  select(DAYSTOCLOSE, Category) %>%
  filter(Category %in% c("Animals","Property Maintenance")) %>%
  drop_na(DAYSTOCLOSE) %>%
  t.test(DAYSTOCLOSE ~ Category, data = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  DAYSTOCLOSE by Category
## t = -38.661, df = 42346, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group Animals and group Property Maintenance is not equal to 0
## 95 percent confidence interval:
##  -33.97234 -30.69391
## sample estimates:
##              mean in group Animals mean in group Property Maintenance
##                           37.47969                           69.81282
```

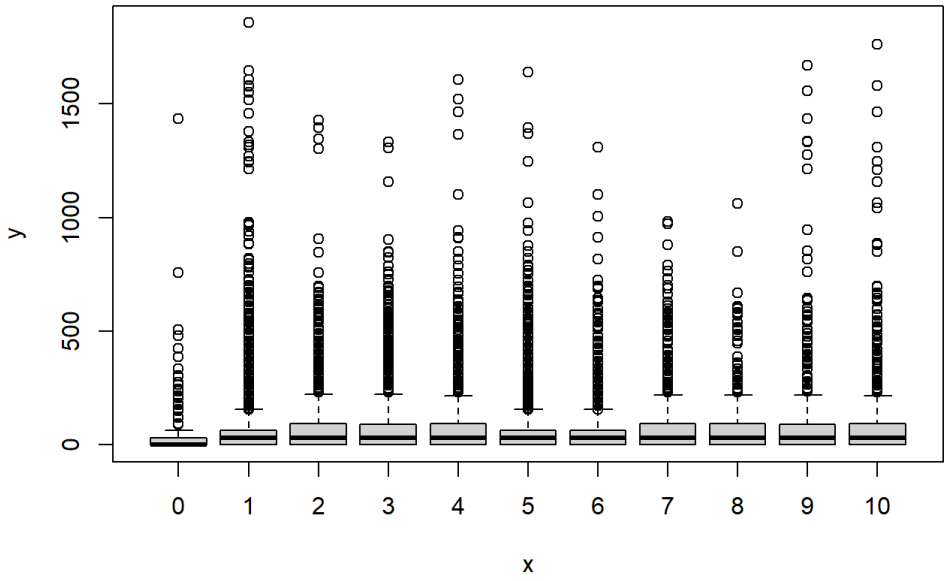
Correlations

```
#Check for correlation on ALL numeric field combinations
main_df_num <- data.frame(main_df$CASEID, !is.na(main_df$SLA_Length),
                          !is.na(main_df$PCTofSLA), !is.na(main_df$DAYSTOCLOSE),
                          main_df$Council.DistrictNum, !is.na(main_df$XCOORD),
                          !is.na(main_df$YCOORD), !is.na(main_df$temp_max),
                          !is.na(main_df$wind_speed), !is.na(main_df$feels_like),
                          !is.na(main_df$visibility), !is.na(main_df$humidity),
                          !is.na(main_df$wind_deg))
str(main_df_num)
```

```
## 'data.frame':    587421 obs. of  13 variables:
##  $ main_df.CASEID              : int  1014001765 1014569491 1014749704 1014303843 1014761061 1014954069 1015043717 101450
4120 1014679731 1014680784 ...
##  $ X.is.na.main_df.SLA_Length. : logi  FALSE TRUE FALSE FALSE FALSE TRUE ...
##  $ X.is.na.main_df.PCTofSLA.   : logi  FALSE FALSE FALSE FALSE FALSE TRUE ...
##  $ X.is.na.main_df.DAYSTOCLOSE.: logi  FALSE FALSE FALSE TRUE FALSE TRUE ...
##  $ main_df.Council.DistrictNum : num  5 2 3 2 11 11 10 3 11 10 ...
##  $ X.is.na.main_df.XCOORD.     : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ X.is.na.main_df.YCOORD.     : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ X.is.na.main_df.temp_max.   : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ X.is.na.main_df.wind_speed. : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ X.is.na.main_df.feels_like. : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ X.is.na.main_df.visibility. : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ X.is.na.main_df.humidity.   : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ X.is.na.main_df.wind_deg.   : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
```

```
numnames <- c('CaseID', 'SLAlength', 'PCTofSLA', 'DaystoClose',
              'CouncilDistrict', 'X-Coor', 'Y-Coor', 'temp_max',
              'wind_speed','feels_like','visibility','humidity',
              'wind_deg')
correlation_matrix <- matrix(cor(main_df_num), 13, 13, dimnames = list(numnames, numnames))
```

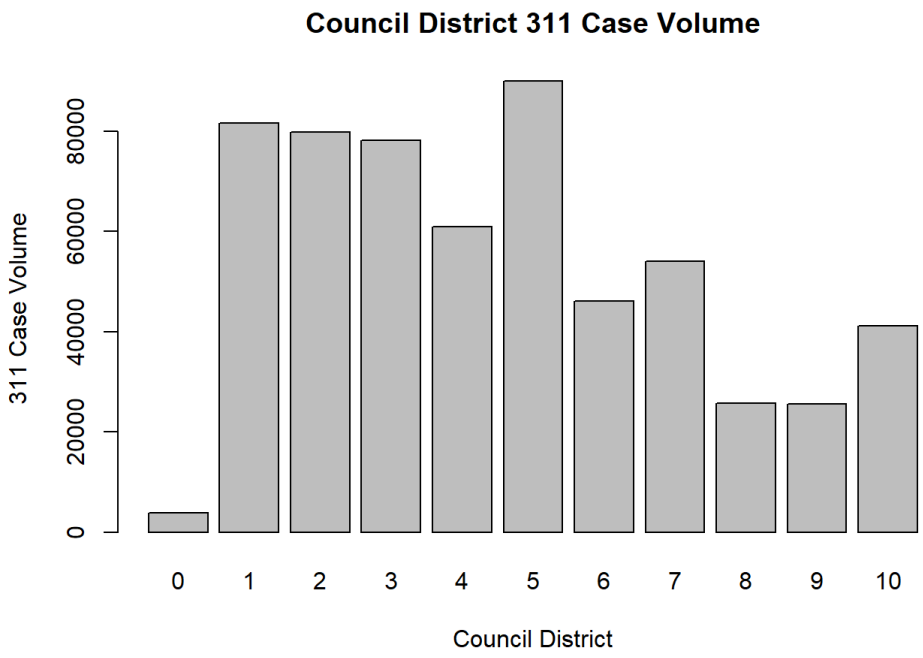
```
#There do not appear to be any meaningful correlations

#Make a dot chart that shows the number of days it took to close cases by district.
plot(main_df$Council.District, main_df$DAYSTOCLOSE)
```




```
#Calculate metrics for center and distribution values for each of the numerical fields.
summary(main_df_num)
```

```
##  main_df.CASEID      X.is.na.main_df.SLA_Length. X.is.na.main_df.PCTofSLA.
##  Min.   :1.014e+09   Mode :logical               Mode :logical
##  1st Qu.:1.019e+09   FALSE:468942                FALSE:506001
##  Median :1.019e+09   TRUE :118479                TRUE :81420
##  Mean   :1.019e+09
##  3rd Qu.:1.019e+09
##  Max.   :1.019e+09
##  X.is.na.main_df.DAYSTOCLOSE. main_df.Council.DistrictNum
##  Mode :logical                Min.   : 1.000
##  FALSE:450722                 1st Qu.: 3.000
##  TRUE :136699                 Median : 5.000
##                               Mean   : 5.552
##                               3rd Qu.: 7.000
##                               Max.   :11.000
##  X.is.na.main_df.XCOORD. X.is.na.main_df.YCOORD. X.is.na.main_df.temp_max.
##  Mode :logical           Mode :logical           Mode :logical
##  FALSE:9                 FALSE:9                 FALSE:1644
##  TRUE :587412            TRUE :587412            TRUE :585777
##
##
##
##  X.is.na.main_df.wind_speed. X.is.na.main_df.feels_like.
##  Mode :logical               Mode :logical
##  FALSE:1644                  FALSE:1644
##  TRUE :585777                TRUE :585777
##
##
##
##  X.is.na.main_df.visibility. X.is.na.main_df.humidity.
##  Mode :logical               Mode :logical
##  FALSE:1670                  FALSE:1644
##  TRUE :585751                TRUE :585777
##
##
##
##  X.is.na.main_df.wind_deg.
##  Mode :logical
##  FALSE:1644
##  TRUE :585777
##
##
##
```

```
#Create a bar chart of total cases per district.
councicasevol <- plot(main_df$Council.District)
title(main = "Council District 311 Case Volume", xlab = "Council District", ylab = "311 Case Volume",
      font = 2, col = "blue")
```



District case case loads

```
#Store district case totals in a vector
district_volume <- as.numeric(c(summary(main_df$Council.District)))

#Analyze the list of district case totals
summary(district_volume)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3854   33496   54099   53402   79078   90066
```

```
#Save list of 11 district names and verify structure.
Council.DistrictNumNames <- as.numeric(c(0:10))

str(Council.DistrictNumNames)
```

```
##  num [1:11] 0 1 2 3 4 5 6 7 8 9 ...
```

```
#Verifey district_volume structure
str(district_volume)
```

```
##  num [1:11] 3854 81597 79925 78230 61010 ...
```

```
#Calculate the correlation between district number and quantity of cases
cor(Council.DistrictNumNames, district_volume)
```

```
## [1] -0.2856547
```

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.