p8105_hw1_fz2377

Tara Zhan

Sep. 14th, 2023

Problem 1

Here's a short description of the dataset via R codes:

```
options(repos=c(CRAN="THE URL OF YOUR FAVORITE MIRROR"))
local({r <- getOption("repos")
       r["CRAN"] <- "http://cran.r-project.org"
       options(repos=r)})
install.packages("moderndive")
```

```
##
## The downloaded binary packages are in
##  /var/folders/d8/lq84dmfs4zzf2_02b4602_c00000gn/T//RtmpEgXelD/downloaded_packages
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(moderndive)
data("early_january_weather")
```

This "early_january_weather" dataset contains 15 variables: origin, year, month, day, hour, temp, dewp, humid, wind_dir, wind_speed, wind_gust, precip, pressure, visib, time_hour.

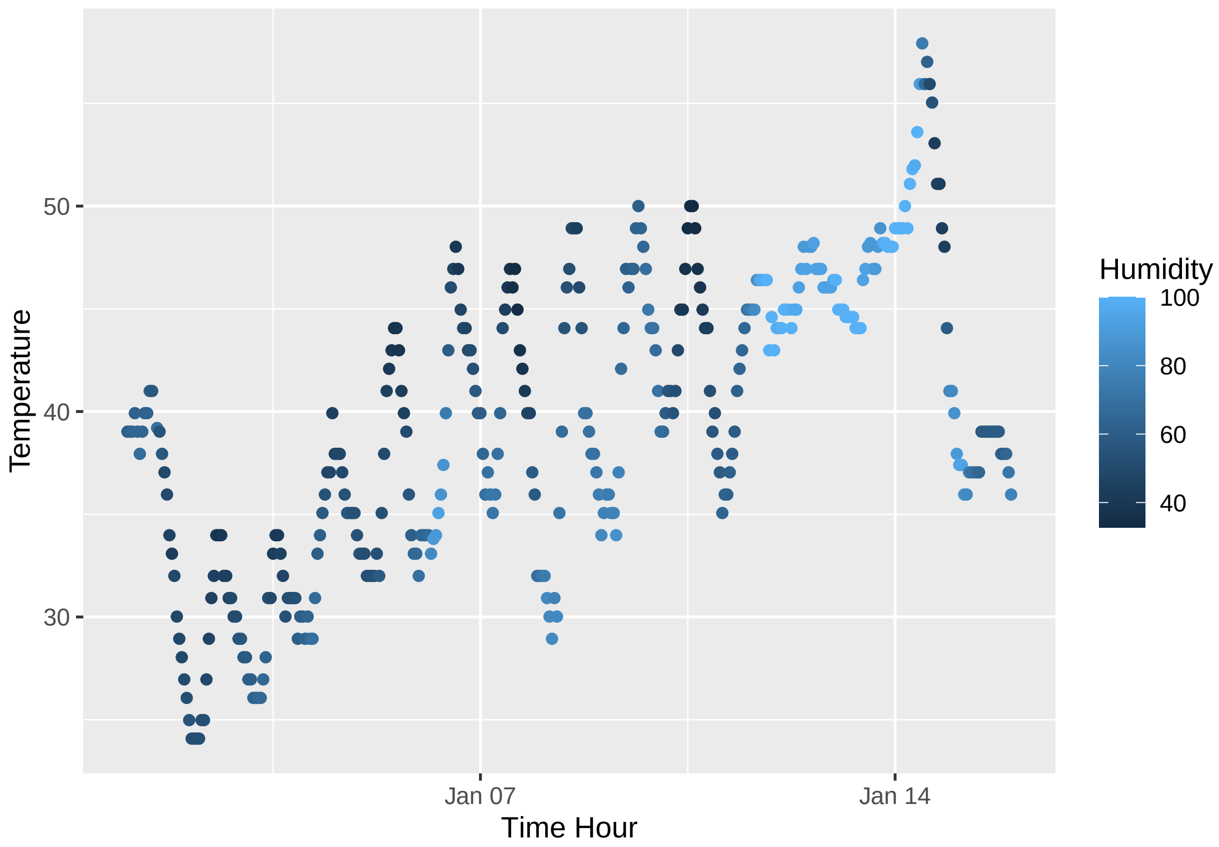
This dataset has a total of 358 and 15. The mean temperature is approximately 39.5821229 degrees.

A scatterplot was created to display the relationship between temperature and time hour, with humidity levels. As the time increases, the temperature increases, and the humidity level reached maximum around Jan. 14th. Then, the scatterplot was exported to the project directory as "scatterplot.png".

```
#Structure of the dataset
str(early_january_weather)
```

```
## tibble [358 x 15] (S3: tbl_df/tbl/data.frame)
##  $ origin    : chr [1:358] "EWR" "EWR" "EWR" "EWR" ...
##  $ year      : int [1:358] 2013 2013 2013 2013 2013 2013 2013 2013 2013 2013 ...
##  $ month     : int [1:358] 1 1 1 1 1 1 1 1 1 1 ...
##  $ day       : int [1:358] 1 1 1 1 1 1 1 1 1 1 ...
##  $ hour      : int [1:358] 1 2 3 4 5 6 7 8 9 10 ...
##  $ temp      : num [1:358] 39 39 39 39.9 39 ...
##  $ dewp      : num [1:358] 26.1 27 28 28 28 ...
##  $ humid     : num [1:358] 59.4 61.6 64.4 62.2 64.4 ...
##  $ wind_dir  : num [1:358] 270 250 240 250 260 240 240 250 260 260 ...
##  $ wind_speed: num [1:358] 10.36 8.06 11.51 12.66 12.66 ...
##  $ wind_gust : num [1:358] NA NA NA NA NA NA NA NA NA NA ...
##  $ precip    : num [1:358] 0 0 0 0 0 0 0 0 0 0 ...
##  $ pressure  : num [1:358] 1012 1012 1012 1012 1012 ...
##  $ visib     : num [1:358] 10 10 10 10 10 10 10 10 10 10 ...
##  $ time_hour : POSIXct[1:358], format: "2013-01-01 01:00:00" "2013-01-01 02:00:00" ...
```

```
library(ggplot2)
#Make a scatterplot
scatterplot <- ggplot(early_january_weather, aes(x = time_hour, y = temp, color = humid)) +
  geom_point() +
  labs(x = "Time Hour", y = "Temperature", color = "Humidity")
#Print the scatterplot
print(scatterplot)
```



```
#ggsave
ggsave("scatterplot.png", plot = scatterplot, width = 8, height = 6)
```

Problem 2

```
# Create a data frame
data <- data.frame(
  random_numeric = rnorm(10),
  logical_vector = rnorm(10) > 0,
  character_vector = sample(letters, 10, replace = TRUE),
  factor_vector = as.factor(sample(1:3, 10, replace = TRUE))
)
#The mean of each variable
mean_numeric <- mean(data$random_numeric)
mean_logical <- mean(data$logical_vector)
mean_character <- mean(data$character_vector)
```

```
## Warning in mean.default(data$character_vector): argument is not numeric or
## logical: returning NA
```

```
mean_factor <- mean(data$factor_vector)
```

```
## Warning in mean.default(data$factor_vector): argument is not numeric or
## logical: returning NA
```

```
#Convert variables to numeric
data$logical_vector <- as.numeric(data$logical_vector)
data$character_vector <- as.numeric(data$character_vector)
```

```
## Warning: NAs introduced by coercion
```

```
data$factor_vector <- as.numeric(data$factor_vector)
```