445_Assignment_4

Tucker Harris

2023-11-03

Question 1a

```
Survey <-
  read.csv('https://www.lock5stat.com/datasets3e/StudentSurvey.csv',
           na.strings=c('',' '))

head(Survey)
```

```
##        Year Sex Smoke   Award HigherSAT Exercise TV Height Weight Siblings
## 1    Senior   M    No Olympic      Math       10  1     71    180        4
## 2 Sophomore   F   Yes Academy      Math        4  7     66    120        2
## 3 FirstYear   M    No   Nobel      Math       14  5     72    208        2
## 4    Junior   M    No   Nobel      Math        3  1     63    110        1
## 5 Sophomore   F    No   Nobel    Verbal        3  3     65    150        1
## 6 Sophomore   F    No   Nobel    Verbal        5  4     65    114        2
##   BirthOrder VerbalSAT MathSAT  SAT  GPA Pulse Piercings
## 1          4       540     670 1210 3.13    54         0
## 2          2       520     630 1150 2.50    66         3
## 3          1       550     560 1110 2.55   130         0
## 4          1       490     630 1120 3.10    78         0
## 5          1       720     450 1170 2.70    40         6
## 6          2       600     550 1150 3.20    80         4
```

Question 1b)

```
SurveyA <- count( Survey, Year, Sex )

SurveyA <- SurveyA %>% drop_na()

SurveyA
```

```
##        Year Sex  n
## 1 FirstYear   F 43
## 2 FirstYear   M 51
## 3    Junior   F 18
## 4    Junior   M 17
## 5    Senior   F 10
## 6    Senior   M 26
## 7 Sophomore   F 96
## 8 Sophomore   M 99
```

```
# I could not figure out how to order the Years. I will hopefully have time to return to this later
```

Question 1c)

```
SurveyB <- SurveyA %>% pivot_wider(names_from = Year, values_from = n)


SurveyB
```

```
## # A tibble: 2 x 5
##   Sex   FirstYear Junior Senior Sophomore
##   <chr>     <int>  <int>  <int>     <int>
## 1 F            43     18     10        96
## 2 M            51     17     26        99
```

Question 2a)

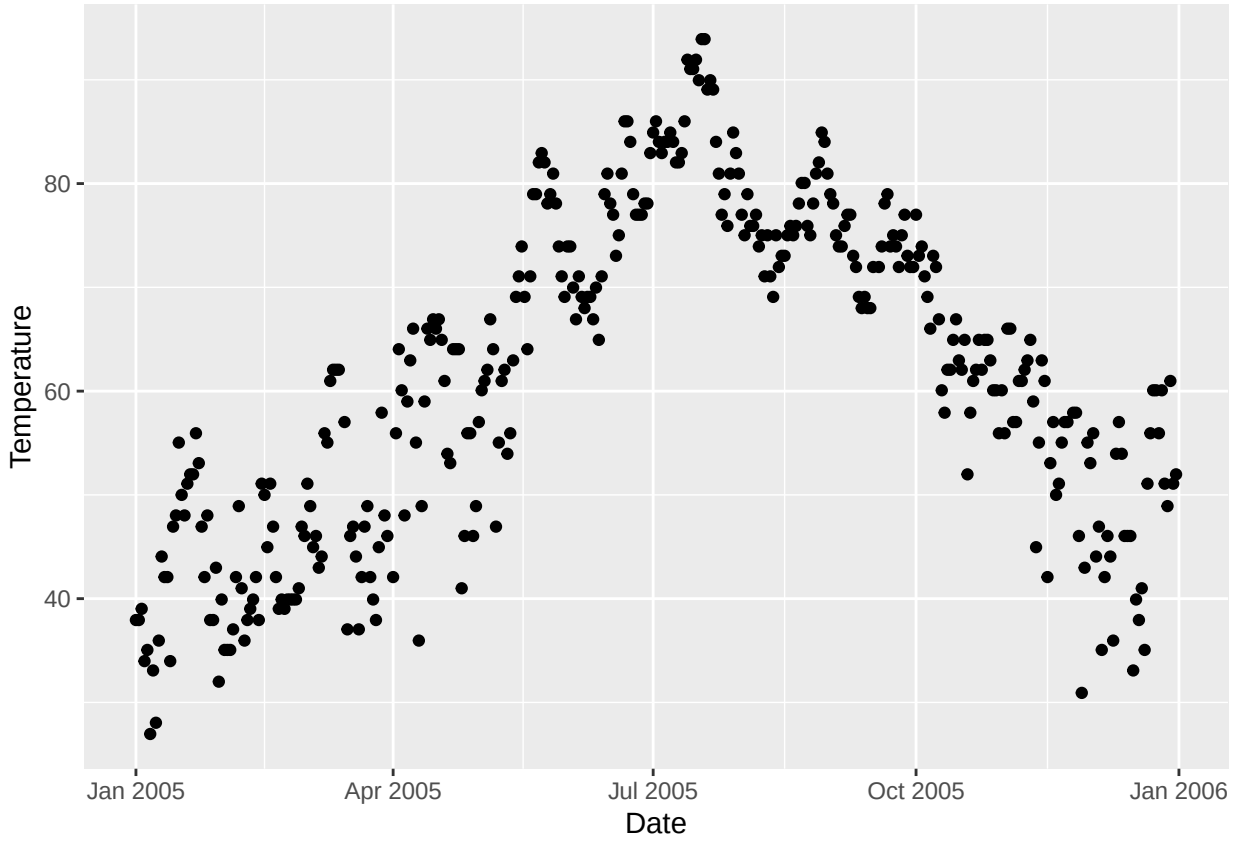
```
Temperature <-  read.csv('https://raw.githubusercontent.com/dereksonderegger/444/master/data-raw/FlagMa

head(Temperature)
```

```
##   X Year Month    X1    X2    X3    X4    X5    X6    X7    X8    X9   X10
## 1 1 1985     5 71.06 71.06 68.00 68.00 64.94 64.04 64.04 64.94 69.08 66.02
## 2 2 1985     6 62.96 62.96 64.94 60.08 69.08 75.92 82.04 86.00 84.92 84.02
## 3 3 1985     7 80.96 86.00 89.96 87.98 91.94 91.94 89.06 87.98 89.96 87.08
## 4 4 1985     8 77.00 68.00 78.08 80.06 82.04 80.96 82.94 82.94 80.06 80.06
## 5 5 1985     9 82.94 75.02 73.94 71.96 66.92 62.96 62.96 64.04 68.00 64.94
## 6 6 1985    10 64.04 60.08 64.04 71.06 71.06 75.02 69.08 53.96 51.08 55.04
##     X11   X12   X13   X14   X15   X16   X17   X18   X19   X20   X21   X22   X23
## 1 51.08 55.94 59.00 57.92 66.02 66.92 66.02 66.02 68.00 66.92 66.92 62.96    NA
## 2 82.04 82.94 84.92 82.94 82.94 86.00 84.92 87.08 87.08 84.02 80.96 82.04 84.92
## 3 84.02 84.02 84.92 87.08 84.02 84.92 78.98 80.96 75.92 73.94 64.94 71.96 73.04
## 4 80.06 75.92 78.98 78.98 80.96 80.96 78.98 75.02 82.04 82.04 69.98 80.06 84.02
## 5 68.00 66.02 66.92 75.02 73.94 73.04 73.04 69.98 51.98 59.00 55.04 68.00 68.00
## 6 50.00 55.04 57.02 53.96 51.08 55.94 57.02 60.08 62.96 62.96 62.06 55.04 53.96
##     X24   X25   X26   X27   X28   X29   X30   X31
## 1 69.98 73.94 71.06 71.06 69.08 73.04 69.08 62.06
## 2 80.96 73.94 71.96 73.04 80.96 84.02 82.04    NA
## 3 80.06 80.06 77.00 80.06 82.04 77.00 73.94 73.04
## 4 87.98 91.04 84.92 84.02 80.96 78.98 82.04 82.94
## 5 71.06 71.96 73.04 71.96 64.94 60.98 64.04    NA
## 6 68.00 66.92 64.94 66.92 68.00 64.04 64.94 62.06
```

```
Temperature05 <- Temperature %>% filter (Year == 2005)

Temperature1 <- Temperature05 %>% pivot_longer(X1:X31, names_to = "Day", values_to = "Temperature")


Temperature1 <- Temperature1 %>% mutate(Day = str_remove_all( Day, 'X'))

DayTemp <- Temperature1 %>% mutate(date = make_date(year = Year, month = Month, day = Day))

DayTemp1 <- DayTemp %>% select( Temperature, date )

DayTemp1 <- DayTemp1 %>% mutate(Temperature = as.numeric(Temperature))
```

```
ggplot( data = DayTemp1, aes(x=date, y=Temperature) ) + geom_point() +
  labs("Daily 2005 temperatures", x = "Date", y = "Temperature" )
```

```
## Warning: Removed 11 rows containing missing values (`geom_point()`).
```



Question 2b)

```
Temperature1315 <- Temperature %>% filter(Year >= 2013 & Year <= 2015)

Temp1 <- Temperature1315 %>% pivot_longer(X1:X31, names_to = "Day", values_to = "Temperature")

Temp1<- Temp1 %>% drop_na()

Temp1 <- Temp1 %>% mutate(Day = str_remove_all( Day, 'X'))

Temp1 <- Temp1 %>% mutate(date = make_date(year = Year, month = Month))

Temp1 <- Temp1 %>% group_by(Year, Month) %>% summarize(meanTemp = mean(Temperature))
```

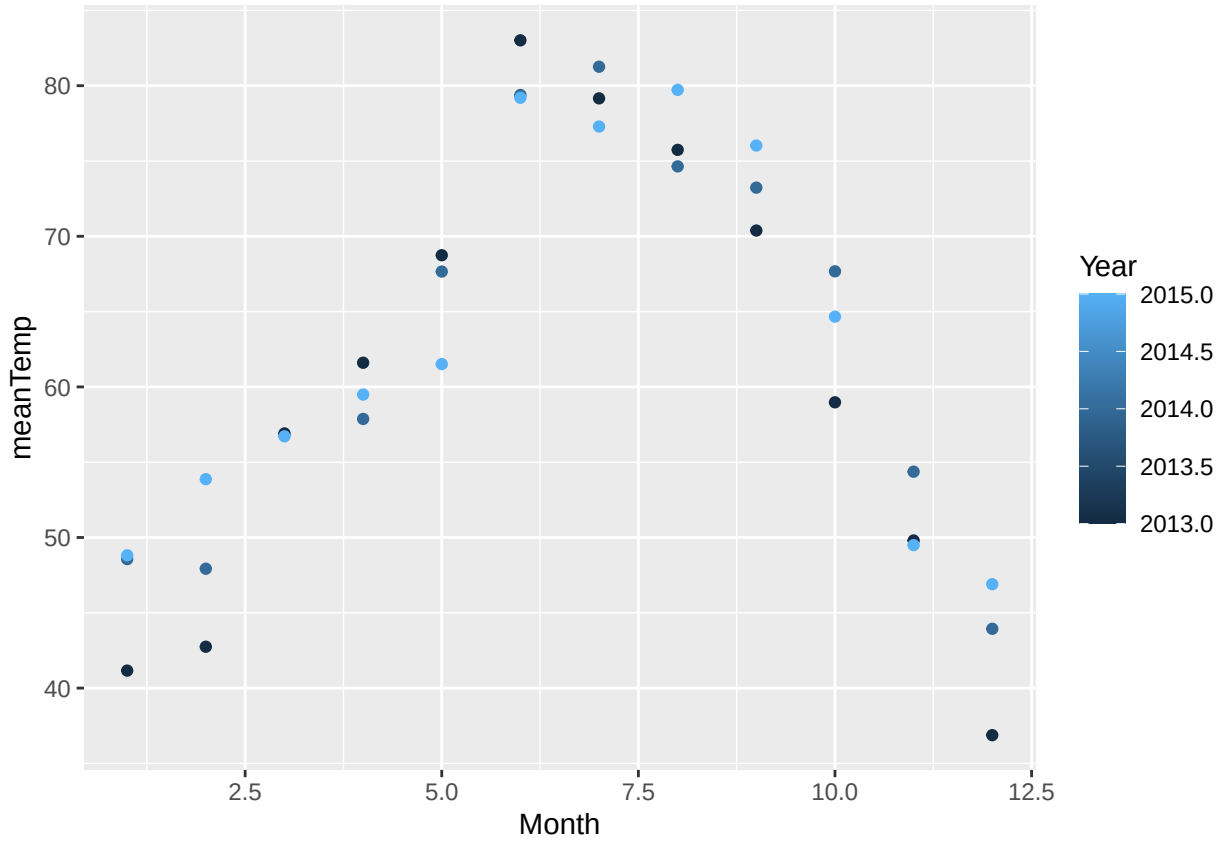
```
## `summarise()` has grouped output by 'Year'. You can override using the
## `.groups` argument.
```

```
Temp1
```

```
## # A tibble: 35 x 3
## # Groups:   Year [3]
##     Year Month meanTemp
##    <int> <int>    <dbl>
##  1  2013     1     41.2
```

```
##  2  2013     2     42.7
##  3  2013     3     56.9
##  4  2013     4     61.6
##  5  2013     5     68.7
##  6  2013     6     83.0
##  7  2013     7     79.2
##  8  2013     8     75.7
##  9  2013     9     70.4
## 10  2013    10     59.0
## # i 25 more rows
```

```
ggplot( data = Temp1, aes(x = Month, y = meanTemp, color = Year)) + geom_point()
```



Question 4 a

```
A <- tribble(
  ~Name, ~Car,
  'Alice', 'Ford F150',
  'Bob',   'Tesla Model III',
  'Charlie', 'VW Bug')

B <- tribble(
  ~First.Name, ~Pet,
  'Bob',  'Cat',
  'Charlie', 'Dog',
  'Alice', 'Rabbit')
```

```
B <- B %>% arrange(First.Name)


AB <- cbind(A,B) %>% select(Name, Car, Pet)
AB
```

```
##      Name             Car    Pet
## 1   Alice       Ford F150 Rabbit
## 2     Bob Tesla Model III    Cat
## 3 Charlie          VW Bug    Dog
```

```
B <- B %>% rename("Name" = "First.Name")

full_join(A, B)
```

```
## Joining with `by = join_by(Name)`
```

```
## # A tibble: 3 x 3
##   Name    Car             Pet
##   <chr>   <chr>           <chr>
## 1 Alice   Ford F150       Rabbit
## 2 Bob     Tesla Model III Cat
## 3 Charlie VW Bug          Dog
```

Question 3b)

```
C <- tibble( Name = 'Alice', Pet = 'Guinea Pig')
B <- rbind(B, C)

B
```

```
## # A tibble: 4 x 2
##   Name    Pet
##   <chr>   <chr>
## 1 Alice   Rabbit
## 2 Bob     Cat
## 3 Charlie Dog
## 4 Alice   Guinea Pig
```

Question 3c)

```
full_join(A,B)
```

```
## Joining with `by = join_by(Name)`
```

```
## # A tibble: 4 x 3
##   Name    Car             Pet
##   <chr>   <chr>           <chr>
## 1 Alice   Ford F150       Rabbit
## 2 Alice   Ford F150       Guinea Pig
## 3 Bob     Tesla Model III Cat
## 4 Charlie VW Bug          Dog
```

Using Join is so much easier, as I could not figure out how to use cbind in this case. The number of rows being different gave me an error, so I was only able to use a join.

Question 4

```
Customers <- tribble(
  ~PersonID, ~Name, ~Street, ~City, ~State,
  1, 'Derek Sonderegger',  '231 River Run', 'Flagstaff', 'AZ',
  2, 'Aubrey Sonderegger', '231 River Run', 'Flagstaff', 'AZ',
  3, 'Robert Buscaglia', '754 Forest Heights', 'Flagstaff', 'AZ',
  4, 'Roy St Laurent', '845 Elk View', 'Flagstaff', 'AZ')

Retailers <- tribble(
  ~RetailID, ~Name, ~Street, ~City, ~State,
  1, 'Kickstand Kafe', '719 N Humphreys St', 'Flagstaff', 'AZ',
  2, 'MartAnnes', '112 E Route 66', 'Flagstaff', 'AZ',
  3, 'REI', '323 S Windsor Ln', 'Flagstaff', 'AZ' )

Cards <- tribble(
  ~CardID, ~PersonID, ~Issue_DateTime, ~Exp_DateTime,
  '9876768717278723',  1,  '2019-9-20 0:00:00', '2022-9-20 0:00:00',
  '5628927579821287',  2,  '2019-9-20 0:00:00', '2022-9-20 0:00:00',
  '7295825498122734',  3,  '2019-9-28 0:00:00', '2022-9-28 0:00:00',
  '8723768965231926',  4,  '2019-9-30 0:00:00', '2022-9-30 0:00:00' )

Transactions <- tribble(
  ~CardID, ~RetailID, ~DateTime, ~Amount,
  '9876768717278723', 1, '2019-10-1 8:31:23',    5.68,
  '7295825498122734', 2, '2019-10-1 12:45:45',  25.67,
  '9876768717278723', 1, '2019-10-2 8:26:31',    5.68,
  '9876768717278723', 1, '2019-10-2 8:30:09',    9.23,
  '5628927579821287', 3, '2019-10-5 18:58:57',  68.54,
  '7295825498122734', 2, '2019-10-5 12:39:26',  31.84,
  '8723768965231926', 2, '2019-10-10 19:02:20', 42.83)

Cards <- Cards %>%
  mutate( Issue_DateTime = lubridate::ymd_hms(Issue_DateTime),
          Exp_DateTime   = lubridate::ymd_hms(Exp_DateTime) )
Transactions <- Transactions %>%
  mutate( DateTime = lubridate::ymd_hms(DateTime))
```

A

```
# Filter Transactions for Derek
derek_transactions <- Customers %>%
 filter(Name == 'Derek Sonderegger')

derek_transactions <- derek_transactions %>%
 left_join(Cards, by = "PersonID") %>%
 left_join(Transactions, by = "CardID")

derek_transactions <- derek_transactions %>%
 left_join(Retailers, by = "RetailID") %>%
 select(DateTime, Amount, Name = Name.x)

derek_transactions
```

```
## # A tibble: 3 x 3
##   DateTime            Amount Name
##   <dttm>               <dbl> <chr>
## 1 2019-10-01 08:31:23   5.68 Derek Sonderegger
## 2 2019-10-02 08:26:31   5.68 Derek Sonderegger
## 3 2019-10-02 08:30:09   9.23 Derek Sonderegger
```

B

```
# Find Aubrey's card details
aubrey_card_info <- Customers %>%
  filter(Name == 'Aubrey Sonderegger') %>%
  left_join(Cards, by = "PersonID")

aubrey_card_info
```

```
## # A tibble: 1 x 8
##   PersonID Name               Street      City  State CardID Issue_DateTime
##      <dbl> <chr>              <chr>       <chr> <chr> <chr>  <dttm>
## 1        2 Aubrey Sonderegger 231 River ~ Flag~ AZ    56289~ 2019-09-20 00:00:00
## # i 1 more variable: Exp_DateTime <dttm>
```

```
# Get CardID and PersonID for Aubrey
cardID <- aubrey_card_info$CardID

cardID
```

```
## [1] "5628927579821287"
```

```
personID <- aubrey_card_info$PersonID

personID
```

```
## [1] 2
```

```
# Close the existing card (set Exp_DateTime to the time it's closed)
close_time <- ymd_hms('2019-10-15 16:28:21')

close_time
```

```
## [1] "2019-10-15 16:28:21 UTC"
```

```
Cards <- Cards %>%
  mutate(Exp_DateTime = ifelse(CardID == cardID, close_time, Exp_DateTime))

Cards
```

```
## # A tibble: 4 x 4
##   CardID           PersonID Issue_DateTime      Exp_DateTime
##   <chr>               <dbl> <dttm>                     <dbl>
## 1 9876768717278723        1 2019-09-20 00:00:00   1663632000
## 2 5628927579821287        2 2019-09-20 00:00:00   1571156901
## 3 7295825498122734        3 2019-09-28 00:00:00   1664323200
## 4 8723768965231926        4 2019-09-30 00:00:00   1664496000
```

```
# Generate a new CardID for Aubrey's new card (you can make up a new number)
new_card_id <- '0321200302132003'
```

```
# Insert the new card into the Cards table
new_card <- tibble(CardID = new_card_id, PersonID = personID,
                   Issue_DateTime = close_time,
                   Exp_DateTime = close_time + years(3))  # Set a new expiration date

new_card
```

```
## # A tibble: 1 x 4
##   CardID           PersonID Issue_DateTime      Exp_DateTime
##   <chr>               <dbl> <dttm>              <dttm>
## 1 0321200302132003        2 2019-10-15 16:28:21 2022-10-15 16:28:21
```

```
Cards <- rbind(Cards, new_card)

Cards
```

```
## # A tibble: 5 x 4
##   CardID           PersonID Issue_DateTime      Exp_DateTime
##   <chr>               <dbl> <dttm>                     <dbl>
## 1 9876768717278723        1 2019-09-20 00:00:00   1663632000
## 2 5628927579821287        2 2019-09-20 00:00:00   1571156901
## 3 7295825498122734        3 2019-09-28 00:00:00   1664323200
## 4 8723768965231926        4 2019-09-30 00:00:00   1664496000
## 5 0321200302132003        2 2019-10-15 16:28:21   1665851301
```

I added the new card in three times. I am not exactly sure how I did that or how to fix it.

C

```
# Define the transaction details
card <- new_card_id  # Use Aubrey's new card ID
retailid <- 1  # Retailer ID for Kickstand Kafe
datetime <- ymd_hms('2019-10-16 14:30:21')
amount <- 4.98

# Create a new transaction
new_transaction <- tibble(CardID = card, RetailID = retailid, DateTime = datetime, Amount = amount)

# Append the new transaction to the Transactions table
Transactions <- rbind(Transactions, new_transaction)



Transactions
```

```
## # A tibble: 8 x 4
##   CardID           RetailID DateTime            Amount
##   <chr>               <dbl> <dttm>               <dbl>
## 1 9876768717278723        1 2019-10-01 08:31:23   5.68
## 2 7295825498122734        2 2019-10-01 12:45:45  25.7
## 3 9876768717278723        1 2019-10-02 08:26:31   5.68
## 4 9876768717278723        1 2019-10-02 08:30:09   9.23
## 5 5628927579821287        3 2019-10-05 18:58:57  68.5
## 6 7295825498122734        2 2019-10-05 12:39:26  31.8
## 7 8723768965231926        2 2019-10-10 19:02:20  42.8
```

```
## 8 0321200302132003        1 2019-10-16 14:30:21   4.98
```

D

```r
# Define the transaction details
card <- '9876768717278723'  # OLD credit card
retailid <- 2  # Retailer ID for REI
datetime <- ymd_hms('2019-10-17 12:00:00')
amount <- 100.00  # Example amount

# Check if the card is currently valid
Valid_Cards <- Cards %>%
  filter(CardID == card, Issue_DateTime <= datetime, datetime <= Exp_DateTime)

# If the card is valid, insert the transaction into the table
if (nrow(Valid_Cards) == 1) {
  new_transaction <- tibble(CardID = card, RetailID = retailid, DateTime = datetime, Amount = amount)
  Transactions <- rbind(Transactions, new_transaction)
} else {
  print('Card Denied')
}
```

This does not print anything :(. I am not sure where my code is wrong.

E

```r
# Find Aubrey's card details
aubrey_card_info <- Customers %>%
  filter(Name == 'Aubrey Sonderegger') %>%
  left_join(Cards, by = "PersonID")

# Get Aubrey's CardIDs
aubrey_card_ids <- aubrey_card_info$CardID

# Filter and join transactions and retailer names for Aubrey's cards
aubrey_statement <- Transactions %>%
  filter(CardID %in% aubrey_card_ids) %>%
  left_join(Retailers, by = "RetailID") %>%
  select(DateTime, Amount, Name)

aubrey_statement
```

```
## # A tibble: 2 x 3
##   DateTime            Amount Name
##   <dttm>               <dbl> <chr>
## 1 2019-10-05 18:58:57  68.5  REI
## 2 2019-10-16 14:30:21   4.98 Kickstand Kafe
```