Define your user id

```
# your username is the xxx@syr.edu
username <- "yjian@syr.edu"

# make sure we have access to tidyverse
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.5
## ✔ forcats   1.0.0     ✔ stringr   1.5.1
## ✔ ggplot2   3.5.0     ✔ tibble    3.2.1
## ✔ lubridate 1.9.3     ✔ 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 conflic
ts to become errors
```

Run this code. DO NOT CHANGE THIS CODE.

```
if (username == "XXX") {
    print("update the username")
    df <- NULL
} else {
    df <- read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/ids_data.cs
v")
    idx = df$PaymentMode != "Online"
    sub1 <- df[idx, ]
    sub <- df[!idx, ]
    index <- sample(c(1:nrow(sub), nrow(sub)/(sample(c(1:5), 1) * 2), replace = FALS
E))
    sub$TotalSales[index] <- sub$TotalSales[index] + sub$TotalSales[index] * 0.5
    df <- rbind(sub, sub1)
    cn <- names(df)
    ncn = cn
    set.seed(sum(as.integer(sapply(strsplit(username, "")[[1]], charToRaw))))
    for (i in 1:length(cn)) {
        ncn[i] <- paste0(cn[i], "_", apply(matrix(sample(c(letters, LETTERS, 0:9),
            1, replace = TRUE), ncol = 1), 2, paste0, collapse = ""))
    }
    names(df) <- ncn
}
```

```
## Rows: 15000 Columns: 9
## ── Column specification ────────────────────────────────────────────────────
## Delimiter: ","
## chr (6): Date, Location, ProductCategory, CustomerAgeGroup, CustomerGender, ...
## dbl (3): StoreID, ItemsSold, TotalSales
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Use ‘df’ for this assignment

The dataset contains information about store sales, including:

- Date of the sale
- Store ID and location (state)
- Number of items sold
- Total sales amount (in USD)
- Product category
- Customer demographics (age group, gender)
- Mode of payment (cash, credit card, etc.)

```
str(df)
```

```
## tibble [15,000 × 9] (S3: tbl_df/tbl/data.frame)
##  $ Date_S            : chr [1:15000] "1/1/23" "1/1/23" "1/1/23" "1/1/23" ...
##  $ StoreID_I         : num [1:15000] 1 14 3 2 16 15 3 6 8 18 ...
##  $ Location_O        : chr [1:15000] "Illinois" "Texas" "Florida" "California" ...
##  $ ItemsSold_3       : num [1:15000] 12 15 19 7 9 14 6 8 2 14 ...
##  $ TotalSales_t      : num [1:15000] 157 125 176 115 157 ...
##  $ ProductCategory_s : chr [1:15000] "Groceries" "Electronics" "Toys" "Electronic
s" ...
##  $ CustomerAgeGroup_X: chr [1:15000] "31-40" "41-50" "41-50" "60+" ...
##  $ CustomerGender_Y  : chr [1:15000] "Male" "Male" "Female" "Female" ...
##  $ PaymentMode_v     : chr [1:15000] "Online" "Online" "Online" "Online" ...
```

Section 1: Data Visualizations

A. Generate a chloropleth map, showing total Items Sold per state. The outline of each state should be black. [2 points]

```
library(ggplot2)
library(dplyr)
library(maps)
```

```
##
## Attaching package: 'maps'
```

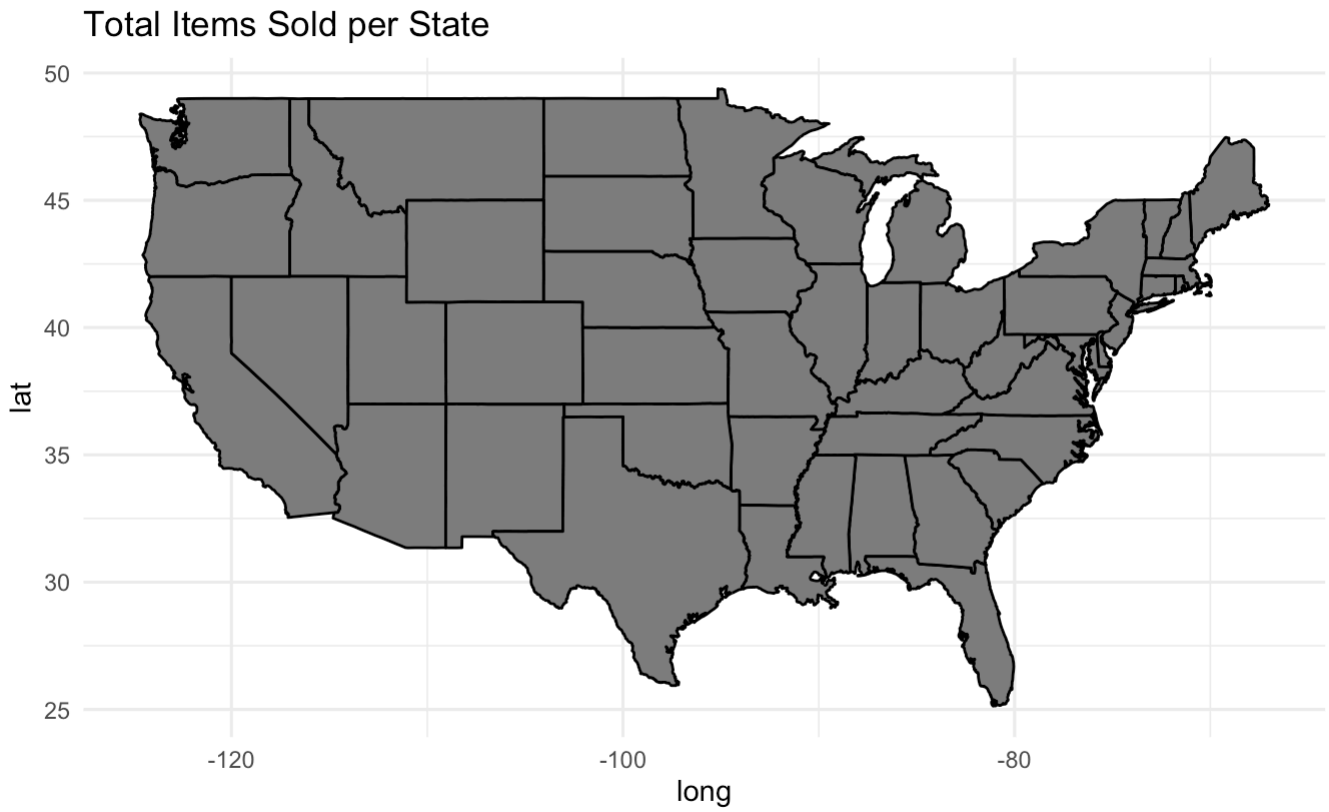
```
## The following object is masked from 'package:purrr':
##
##     map
```

```
items_sold_per_state <- df %>%
    group_by(Location_0) %>%
    summarise(TotalItemsSold = sum(ItemsSold_3))

states_map <- map_data("state")

map_data <- merge(states_map, items_sold_per_state, by.x = "region", by.y = "Location
_0",
    all.x = TRUE)

ggplot(data = map_data, aes(x = long, y = lat, group = group, fill = TotalItemsSold))
+
    geom_polygon(color = "black") + coord_fixed(1.3) + theme_minimal() + labs(fill =
"Items Sold",
    title = "Total Items Sold per State")
```



B) Explain (in a comment) if the map is useful, and if so, what is the key insight. If not, explain why you think it is not useful. [1 point]

```
# Maps can quickly show geographic patterns in sales distribution, and we can
# know which states are performing better and which states need more attention
# and resources.
```

Section 2: Predictive Modeling

A. Build a model to predict number of total sales, based on the available data (ignore the date column). [1 point]

```
sales_model <- lm(TotalSales_t ~ StoreID_I + Location_O + ItemsSold_3 + ProductCatego
ry_s +
    CustomerAgeGroup_X + CustomerGender_Y, data = df)
summary(sales_model)
```

```
##
## Call:
## lm(formula = TotalSales_t ~ StoreID_I + Location_O + ItemsSold_3 +
##     ProductCategory_s + CustomerAgeGroup_X + CustomerGender_Y,
##     data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -69.729 -10.137  -6.672   1.293  69.900
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  80.705838   0.911936  88.499   <2e-16 ***
## StoreID_I                    -0.001899   0.033927  -0.056   0.9554
## Location_OFlorida            -0.327097   0.615822  -0.531   0.5953
## Location_OIllinois           -1.144119   0.620504  -1.844   0.0652 .
## Location_ONew York           -0.856319   0.619799  -1.382   0.1671
## Location_OTexas              -0.758020   0.617621  -1.227   0.2197
## ItemsSold_3                  -0.002562   0.035771  -0.072   0.9429
## ProductCategory_sElectronics 10.316276   0.616554  16.732   <2e-16 ***
## ProductCategory_sFurniture   20.424815   0.616130  33.150   <2e-16 ***
## ProductCategory_sGroceries   30.475320   0.617415  49.360   <2e-16 ***
## ProductCategory_sToys        40.639911   0.614765  66.106   <2e-16 ***
## CustomerAgeGroup_X31-40       0.114734   0.619227   0.185   0.8530
## CustomerAgeGroup_X41-50      -0.522784   0.619123  -0.844   0.3985
## CustomerAgeGroup_X51-60      -0.094162   0.612629  -0.154   0.8778
## CustomerAgeGroup_X60+        -0.022253   0.620187  -0.036   0.9714
## CustomerGender_YMale         -0.234331   0.478828  -0.489   0.6246
## CustomerGender_YOther        -0.601315   0.479949  -1.253   0.2103
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.94 on 14983 degrees of freedom
## Multiple R-squared:  0.2661, Adjusted R-squared:  0.2653
## F-statistic: 339.5 on 16 and 14983 DF,  p-value: < 2.2e-16
```

B. Use a different model to predict if the total sales will be over $80 or lower than $80 (still ignore the date column). [2 points]

```
sales_over_80 <- ifelse(df$TotalSales_t > 80, 1, 0)
df$sales_over_80 <- as.factor(sales_over_80)

over_80_model <- glm(sales_over_80 ~ StoreID_I + Location_O + ItemsSold_3 + ProductCa
tegory_s +
    CustomerAgeGroup_X + CustomerGender_Y, family = binomial, data = df)
summary(over_80_model)
```

```
##
## Call:
## glm(formula = sales_over_80 ~ StoreID_I + Location_O + ItemsSold_3 +
##     ProductCategory_s + CustomerAgeGroup_X + CustomerGender_Y,
##     family = binomial, data = df)
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -0.692174   0.098801  -7.006 2.46e-12 ***
## StoreID_I                   -0.006427   0.003852  -1.669   0.0952 .
## Location_OFlorida           -0.064223   0.069976  -0.918   0.3587
## Location_OIllinois          -0.155353   0.070229  -2.212   0.0270 *
## Location_ONew York           0.091254   0.071264   1.281   0.2004
## Location_OTexas             -0.081950   0.070303  -1.166   0.2438
## ItemsSold_3                 -0.002199   0.004059  -0.542   0.5879
## ProductCategory_sElectronics  1.848976   0.057217  32.315  < 2e-16 ***
## ProductCategory_sFurniture    3.111654   0.073524  42.321  < 2e-16 ***
## ProductCategory_sGroceries    3.113418   0.073786  42.195  < 2e-16 ***
## ProductCategory_sToys         3.051701   0.072115  42.317  < 2e-16 ***
## CustomerAgeGroup_X31-40       0.066776   0.070829   0.943   0.3458
## CustomerAgeGroup_X41-50      -0.016558   0.070882  -0.234   0.8153
## CustomerAgeGroup_X51-60      -0.065354   0.069427  -0.941   0.3465
## CustomerAgeGroup_X60+        -0.082543   0.070014  -1.179   0.2384
## CustomerGender_YMale         -0.037266   0.054530  -0.683   0.4944
## CustomerGender_YOther        -0.070028   0.054617  -1.282   0.1998
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 17131  on 14999  degrees of freedom
## Residual deviance: 12983  on 14983  degrees of freedom
## AIC: 13017
##
## Number of Fisher Scoring iterations: 4
```

C. Explain your reason(s) for choosing the specific model for the two predictions. [1 point]

```
# A: I use a linear regression model to predict total sales because this is a
# continuous numerical prediction B: I chose logistic regression because it is
# suitable for binary classification problems
```

D. Discuss “how good” is each model - using non-technical terms. Is either (or both) good enough for the business owner to use? Explain your logic. [2 points]

```
# A is not good enough since its Adjusted R-squared is only 0.2653
```

E. Based on all your work, are there any ‘actionable insights’?
If so, explain the best actionable insight (pick one). Explain the insight, and how the business owner should use the insight. If there is no actionable insight, explain why each of the previous deliverables is not useful and/or not actionable. [1 point]

```
# Customer between 31-40 years old are more likely to purchase products over
# $80, we can target this group with special promotions
```