Homework 1

Andey Nunes, MS
Jordan Hilton
additional team member name(s) here

January 16, 2019

Document Setup

The first step for this week is to set up the R Markdown document options. Be sure that prior to executing code in this document that the following R packages are installed and updated in your R session:

- knitr
- pander
- readxl
- tidyverse

Tidyverse is an ecosystem of packages that work nicely together for data science tools. When the tidyverse package is installed, all the packages and their dependencies are automatically loaded into the R session. The packages included in the tidyverse package are listed here.

broom, cli, crayon, dplyr, dbplyr, forcats, ggplot2, haven, hms, httr, jsonlite, lubridate, magrittr, modelr, purrr, readr, readxl (>=, reprex, rlang, rstudioapi, rvest, stringr, tibble, tidyr, xml2, tidyverse

Next step, load the data sets for the homework. Summaries are included in the appendix.

```r
catalog <- read_excel("catalog.xls")
customers <- read_excel("customers.xls")
order_lines <- read_excel("order_lines.xlsx", skip = 2)
```

```
## New names:
## * `` -> `..2`
```

```r
order_lines_sheet3 <- read_excel("order_lines.xlsx", sheet = 3)
# reading this file in still poses problems...
orders <- read_excel("orders.xls")
```

```r
# inspect the head and tail of the data set
glimpse(order_lines)
```

```
## Observations: 1,356
## Variables: 2
## $ `Sum of Shipped Total` <chr> "Row Labels", "411", "Multi-PlierÂ® 800...
## $ `..2`                  <chr> "Total", "27507.100000000122", "27507.1...
```

```r
tail(order_lines)
```

```
## # A tibble: 6 x 2
##   `Sum of Shipped Total`                                 ..2
##   <chr>                                                  <chr>
## 1 Lariatâ„¢ 3.5                                           0
## 2 597                                                    0
## 3 Weapons Cleaning Kit - Law Enforcement, Pistol/Sub-Gun 0
## 4 548                                                    0
## 5 Hunter's Pruning Kit - Sport Saw & 1/2                 0
## 6 Grand Total                                            1113312.1600000011
```

```r
# notice that R has imported the first row as "Row Labels" and "Total"
# and the last row is the grand total at the end of the data set
# Lets move that first row into the names for order_lines
names(order_lines) <- as.character(order_lines[1,]) %>%
  str_replace_all(" ","_") %>% str_replace_all("`","") %>% str_to_lower()
# now remove that row
order_lines <- order_lines[-1,]
# now lets pull out that grand total and save it as its own number
order_lines_grand_total <- order_lines[length(order_lines$row_labels),2]
# now lets remove that row as well, so that all of our rows are just our actual data observations
order_lines <- order_lines[-length(order_lines$row_labels),]
# check out the head and tail again
glimpse(order_lines)
```

```
## Observations: 1,354
## Variables: 2
## $ row_labels <chr> "411", "Multi-PlierÂ® 800 - Legend", "757", "LMFâ„¢...
## $ total      <chr> "27507.100000000122", "27507.100000000122", "21591....
```

```r
tail(order_lines)
```

```
## # A tibble: 6 x 2
##   row_labels                                            total
##   <chr>                                                 <chr>
## 1 728                                                   0
## 2 Lariatâ„¢ 3.5                                          0
## 3 597                                                   0
## 4 Weapons Cleaning Kit - Law Enforcement, Pistol/Sub-Gun 0
## 5 548                                                   0
## 6 Hunter's Pruning Kit - Sport Saw & 1/2                0
```

```r
# when this .xlsx file is opened in Google Sheets there are 677 lines of data
# once the row labels and grand total lines are removed, glimpse shows
# 1354 observations, which is 2 lines for each observation
# I'm guessing there is a name behind each id number visible in the google sheet
# lets test this by creating 2 data frames from this table, one with the
# rows with only the id numbers the other with the id names
# then compare to check that their "Total" columns are the same
#id_numbers <- one_or_more(DGT) %R% optional(one_or_more(DGT))

#OL_id_numbers <- order_lines %>%
#  filter(str_length("row_labels") <= 4)

#OL_prod_name <- order_lines %>%
#  filter(str_detect("row_labels", "//w"))

#test_same_totals <- OL_id_numbers == OL_prod_name

# now we can separate the "Row Labels"
```

Custom functions

This section is for building some custom functions that will come in handy later

```r
# count the number of missing data entries
countNA <- function(x) {sum(is.na(x)) }

# get the range of a numeric vector by taking the difference
# between the high and low values from the range output
# if the vector is not numeric, then provide NA
get_range <- function(x) {ifelse(is.numeric(x), diff(range(x)), NA)}

# This function creates the generic structure for the tables in Part B.
# The variable_class use of map_chr() will throw an error on the data-time
# object because that class has multiple assignments
# value_type is temporarily NA, reassign one of: "question", "answer", "link"

make_partBtable <- function(x){
   df <- tibble(variable_name = names(x),
                variable_type = NA,
                variable_class = map_chr(x, class),
                count_missing = map_int(x, countNA),
                count_unique = map_dbl(x, ~length(unique(.x)) ),
                variable_range = map_dbl(x, get_range))

   return(df)

}
```

Homework Questions

Part A: General Questions

1. Key business questions

- What is the company's revenue?
- What is the company's profit?
- How profitable is each product?
- How many orders are there for each product?
- How many active customers are there?

2. How does each table relate to answering those questions?

- The catalog table lists each product along with information about that product (such as price, manufacturer, and name).
- The customers table lists each of the company's customers, along with information about that customer (such as location and name).
- The orders table has one record for every order a customer made, with the total cost of that order and information about the number of items in the order and its shipping weight.
- The order_lines table has one record for each different item that was purchased in a single order, along with links to the order.

3. How do I have to link the tables in order to be able to answer those questions?

Part B: Specific Questions

For each data set, we include a table that gives the field (variable) names, whether they are a *link*, *answer* or *question* field, the data class, how many missing observations, and if numeric a range is given.

Catalog

This data set has 761 observations on 7 variables with details as follows:

```r
catalog_table <- make_partBtable(catalog)
catalog_table$variable_type <- c("link", "link", "answer", "question",
                                 "question", "question", "answer")



# pander(catalog_table, caption = "Catalog Data Table Details")
kable(catalog_table, caption = "Catalog Data Table Details")
```

Table 1: Catalog Data Table Details

variable_name	variable_type	variable_class	count_missing	count_unique	variable_range
id	link	numeric	0	761	818
product_code	link	character	1	761	NA
catalog_price	answer	numeric	0	134	654
category1	question	character	645	10	NA
manufact_id	question	numeric	0	5	8
vendor_id	question	numeric	0	5	8
name	answer	character	1	756	NA

Customers

Many of these fields are character string fields or identification fields. While the range values are given, they are not applicable to this data table.

This data set has 22070 observations on 10 variables with details as follows:

```r
customers_table <- make_partBtable(customers)

customers_table$variable_type <- c("link", "link", rep("question", 6), "question or answer", "link")
# id variables and customer code are "links"
# names and bt_* are questions of who and where

#pander(customers_table, caption = "Customers Data Table Details")
kable(customers_table, caption = "Customers Data Table Details")
```

Table 2: Customers Data Table Details

variable_name	variable_type	variable_class	count_missing	count_unique	variable_range
cust_id	link	numeric	0	22070	22482
merchant_id	link	numeric	0	2	1
firstName	question	character	12070	502	NA
lastName	question	character	12070	1001	NA
bt_city	question	character	1	9032	NA
bt_state	question	character	137	67	NA
bt_country	question	character	0	79	NA

variable_name	variable_type	variable_class	count_missing	count_unique	variable_range
bt_zip	question	character	0	12434	NA
cc_type	question or answer	character	0	4	NA
custcode	link	character	0	22069	NA

Order_lines

There is still an issue with this table where the row labels are getting mangled.

```
str(order_lines)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    1354 obs. of  2 variables:
##  $ row_labels: chr  "411" "Multi-PlierÂ® 800 - Legend" "757" "LMFâ„¢ II Infantry - Black" ...
##  $ total     : chr  "27507.100000000122" "27507.100000000122" "21591.649999999994" "21591.64999999999
```

```
order_lines_table <- make_partBtable(order_lines)

#pander(order_lines_table, caption = "Order_lines Data Table Details")
```

Orders

This data set has 23256 observations on 18 variables with details as follows:

```
orders_table <- tibble(variable_name = names(orders),
                    variable_type = c(rep("link",2),
                                      "question", #when
                                      rep("link",2),
                                      rep("question", 2),# which
                                      rep("answer", 7),# how much |total
                                      rep("question",4)), # when
                    # assign one of: "question", "answer", "link"
                    variable_class = c("numeric", "numeric",
                                       "date-time", "character",
                                       "numeric", "character",
                                       "character","numeric",
                                       "character",rep("numeric", 5),
                                       "date-time", "numeric",
                                       "logical", "logical"),
                    count_missing = map_int(orders, countNA),
                    variable_range = map_dbl(orders, get_range))


#pander(orders_table, caption = "Orders Data Table Details")
kable(orders_table, caption = "Orders Data Table Details")
```

Table 3: Orders Data Table Details

variable_name	variable_type	variable_class	count_missing	variable_range
order_id	link	numeric	0	23575
merchant_id	link	numeric	0	1
order_date	question	date-time	0	NA
po_number	link	character	22742	NA
cust_id	link	numeric	0	32482
order_status	question	character	0	NA

variable_name	variable_type	variable_class	count_missing	variable_range
ship_method	question	character	186	NA
items_amount	answer	numeric	0	9590
amt_bracket	answer	character	0	NA
total_weight	answer	numeric	0	483
total_ship	answer	numeric	0	631
total_hand	answer	numeric	0	0
total_tax	answer	numeric	0	0
total_amount	answer	numeric	0	9584
order_status_date	question	date-time	0	NA
send_inv_to_bill	question	numeric	0	1
coupon_code	question	logical	23256	NA
spec_instr	question	logical	23256	NA

Part C. Filter/Select Operations

For all these answers indicate clearly what fields you used, and why you chose those particular fields. If there were other fields you could have considered, indicate why you did not choose those.

4. Top 10 states for orders by dollar volume

5. Top 10 countries for orders by dollar volume

6. Top 10 selling products by units; then by dollar volume

7. For each of the top two US states and each of the top two countries (excluding the US) in questions 1 and 2, what are the 5 top selling products by units? By dollar volume? (5%)

8. Provide the customer ID's, order dates, and order amounts for all customers who have ordered more than once. (5%)

Part D. Sales increasing strategies

References

Appendix

Summary tables

```r
# this whole code chunk can be updated to be "include = FALSE"
# the use of head() is redundant since glimpse() shows more of the same information
# but also tells you how many observations are in the data set
# and doesn't truncate the list of variables

pander(summary(catalog), caption = "catalog summary table")
```

Table 4: catalog summary table (continued below)

id	product_code	catalog_price	category1	manufact_id	vendor_id	name
Min. : 307	Length:761	Min. : 0	Length:761	Min. :0.0	Min. :0.0	Length:761
1st Qu.: 525	Class :character	1st Qu.: 18	Class :character	1st Qu.:1.0	1st Qu.:1.0	Class :character
Median : 728	Mode :character	Median : 34	Mode :character	Median :1.0	Median :1.0	Mode :character
Mean : 725	NA	Mean : 49	NA	Mean :1.2	Mean :1.2	NA
3rd Qu.: 930	NA	3rd Qu.: 57	NA	3rd Qu.:1.0	3rd Qu.:1.0	NA
Max. :1125	NA	Max. :654	NA	Max. :8.0	Max. :8.0	NA

```
head(catalog)
```

```
## # A tibble: 6 x 7
##      id product_code catalog_price category1  manufact_id vendor_id name
##   <dbl> <chr>                <dbl> <chr>            <dbl>     <dbl> <chr>
## 1   446 G79761                9.95 accessori~           1         1 Exchan~
## 2   455 plastic               0    <NA>                 1         1 Plasti~
## 3   445 G75329               12.0  fishing              1         1 Silver~
## 4   444 G75328               11.0  fillet               1         1 Silver~
## 5   443 G75231               13.0  fillet               1         1 "Gator~
## 6   442 G75230               12.0  fillet               1         1 "Gator~
```

```
glimpse(catalog)
```

```
## Observations: 761
## Variables: 7
## $ id            <dbl> 446, 455, 445, 444, 443, 442, 438, 439, 440, 441...
## $ product_code  <chr> "G79761", "plastic", "G75329", "G75328", "G75231...
## $ catalog_price <dbl> 9.9, 0.0, 11.9, 10.9, 12.9, 11.9, 9.5, 6.0, 6.0,...
## $ category1     <chr> "accessories", NA, "fishing", "fillet", "fillet"...
## $ manufact_id   <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ vendor_id     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ name          <chr> "Exchange-A-Blade Sheath for 7 inch saw", "Plast...
```

```
pander(summary(customers), caption = "customers summary table")
```

Table 6: customers summary table (continued below)

cust_id	merchant_id	firstName	lastName
Min. :10000	Min. :1.00	Length:22070	Length:22070
1st Qu.:15930	1st Qu.:1.00	Class :character	Class :character
Median :21448	Median :1.00	Mode :character	Mode :character
Mean :21408	Mean :1.05	NA	NA
3rd Qu.:26965	3rd Qu.:1.00	NA	NA

cust_id	merchant_id	firstName	lastName
Max. :32482	Max. :2.00	NA	NA

Table 7: Table continues below

bt_city	bt_state	bt_country	bt_zip
Length:22070	Length:22070	Length:22070	Length:22070
Class :character	Class :character	Class :character	Class :character
Mode :character	Mode :character	Mode :character	Mode :character
NA	NA	NA	NA
NA	NA	NA	NA
NA	NA	NA	NA

cc_type	custcode
Length:22070	Length:22070
Class :character	Class :character
Mode :character	Mode :character
NA	NA
NA	NA
NA	NA

```r
head(customers)
```

```
## # A tibble: 6 x 10
##   cust_id merchant_id firstName lastName bt_city bt_state bt_country bt_zip
##     <dbl>       <dbl> <chr>     <chr>    <chr>   <chr>    <chr>      <chr>
## 1   20696           2 Kristina  Chung    Piedmo~ OK       United St~ 73078
## 2   15465           1 Paige     Chen     Cincin~ OH       United St~ 45227
## 3   19830           2 Sherri    Melton   Shelby~ TN       United St~ 37160
## 4   25532           1 Gretchen  Hill     North ~ AZ       United St~ 86052
## 5   16044           1 Karen     Puckett  Petawa~ ON       Canada     K8H 2~
## 6   32394           1 Patrick   Song     Winche~ OR       United St~ 97495
## # ... with 2 more variables: cc_type <chr>, custcode <chr>
```

```r
glimpse(customers)
```

```
## Observations: 22,070
## Variables: 10
## $ cust_id     <dbl> 20696, 15465, 19830, 25532, 16044, 32394, 29572, 3...
## $ merchant_id <dbl> 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ firstName   <chr> "Kristina", "Paige", "Sherri", "Gretchen", "Karen"...
## $ lastName    <chr> "Chung", "Chen", "Melton", "Hill", "Puckett", "Son...
## $ bt_city     <chr> "Piedmont", "Cincinnati", "Shelbyville", "North ri...
## $ bt_state    <chr> "OK", "OH", "TN", "AZ", "ON", "OR", "GA", "VA", "K...
## $ bt_country  <chr> "United States", "United States", "United States",...
## $ bt_zip      <chr> "73078", "45227", "37160", "86052", "K8H 2X3", "97...
## $ cc_type     <chr> "Visa", "Visa", "Mastercard", "Visa", "Visa", "Mas...
## $ custcode    <chr> "P20696", "G15465", "P19830", "G25532", "G16044", ...
```

```r
pander(summary(order_lines), caption = "order_lines summary table")
```

Table 9: order_lines summary table

row_labels	total
Length:1354 Class :character Mode :character	Length:1354 Class :character Mode :character

```r
head(order_lines)
```

```
## # A tibble: 6 x 2
##   row_labels                       total
##   <chr>                            <chr>
## 1 411                              27507.100000000122
## 2 Multi-PlierÂ® 800 - Legend        27507.100000000122
## 3 757                              21591.649999999994
## 4 LMFâ„¢ II Infantry - Black        21591.649999999994
## 5 395                              20355.900000000009
## 6 Multi-PlierÂ® 600 Series - D.E.T. 20355.900000000009
```

```r
glimpse(order_lines)
```

```
## Observations: 1,354
## Variables: 2
## $ row_labels <chr> "411", "Multi-PlierÂ® 800 - Legend", "757", "LMFâ„¢...
## $ total      <chr> "27507.100000000122", "27507.100000000122", "21591....
```

```r
pander(summary(orders), caption = "orders summary table")
```

Table 10: orders summary table (continued below)

order_id	merchant_id	order_date	po_number
Min. :14000	Min. :1.00	Min. :2003-10-10 00:00:00	Length:23256
1st Qu.:20134	1st Qu.:1.00	1st Qu.:2006-04-28 00:00:00	Class :character
Median :25948	Median :1.00	Median :2007-07-02 00:00:00	Mode :character
Mean :25918	Mean :1.05	Mean :2007-08-11 16:51:42	NA
3rd Qu.:31761	3rd Qu.:1.00	3rd Qu.:2008-12-19 00:00:00	NA
Max. :37575	Max. :2.00	Max. :2011-01-21 00:00:00	NA

Table 11: Table continues below

cust_id	order_status	ship_method	items_amount
Min. : 0	Length:23256	Length:23256	Min. : 0
1st Qu.:15778	Class :character	Class :character	1st Qu.: 28
Median :21302	Mode :character	Mode :character	Median : 48
Mean :21295	NA	NA	Mean : 73
3rd Qu.:26849	NA	NA	3rd Qu.: 80
Max. :32482	NA	NA	Max. :9590

Table 12: Table continues below

amt_bracket	total_weight	total_ship	total_hand	total_tax
Length:23256	Min. : 0	Min. : 0	Min. :0	Min. :0
Class :character	1st Qu.: 1	1st Qu.: 7	1st Qu.:0	1st Qu.:0
Mode :character	Median : 2	Median : 8	Median :0	Median :0
NA	Mean : 3	Mean : 11	Mean :0	Mean :0
NA	3rd Qu.: 3	3rd Qu.: 10	3rd Qu.:0	3rd Qu.:0
NA	Max. :483	Max. :631	Max. :0	Max. :0

Table 13: Table continues below

total_amount	order_status_date	send_inv_to_bill	coupon_code
Min. : 6	Min. :2003-10-10 00:00:00	Min. :0.00	Mode:logical
1st Qu.: 36	1st Qu.:2006-05-30 18:00:00	1st Qu.:0.00	NA's:23256
Median : 57	Median :2007-07-12 00:00:00	Median :0.00	NA
Mean : 84	Mean :2007-08-21 21:51:27	Mean :0.05	NA
3rd Qu.: 94	3rd Qu.:2008-12-26 00:00:00	3rd Qu.:0.00	NA
Max. :9590	Max. :2011-01-21 00:00:00	Max. :1.00	NA

spec_instr
Mode:logical
NA's:23256
NA
NA
NA
NA

```
head(orders)
```

```
## # A tibble: 6 x 18
##   order_id merchant_id order_date          po_number cust_id order_status
##      <dbl>       <dbl> <dttm>              <chr>       <dbl> <chr>
## 1    14035           1 2003-10-17 00:00:00 <NA>        10034 S
## 2    14034           1 2003-10-16 00:00:00 <NA>        10033 S
## 3    14033           1 2003-10-16 00:00:00 <NA>        10032 S
## 4    14032           1 2003-10-16 00:00:00 <NA>        10031 S
## 5    14031           1 2003-10-16 00:00:00 <NA>        10030 S
## 6    14030           1 2003-10-16 00:00:00 <NA>        10029 S
## # ... with 12 more variables: ship_method <chr>, items_amount <dbl>,
## #   amt_bracket <chr>, total_weight <dbl>, total_ship <dbl>,
## #   total_hand <dbl>, total_tax <dbl>, total_amount <dbl>,
## #   order_status_date <dttm>, send_inv_to_bill <dbl>, coupon_code <lgl>,
## #   spec_instr <lgl>
```

```
glimpse(orders)
```

```
## Observations: 23,256
## Variables: 18
## $ order_id          <dbl> 14035, 14034, 14033, 14032, 14031, 14030, 14...
## $ merchant_id       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ order_date        <dttm> 2003-10-17, 2003-10-16, 2003-10-16, 2003-10...
## $ po_number         <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
## $ cust_id           <dbl> 10034, 10033, 10032, 10031, 10030, 10029, 10...
## $ order_status      <chr> "S", "S", "S", "S", "S", "S", "S", "S", "S",...
## $ ship_method       <chr> "GND", "3DS", "GND", "GND", "3DS", "1DA", "G...
## $ items_amount      <dbl> 58.9, 8.9, 50.0, 11.9, 9.9, 109.9, 23.9, 40....
## $ amt_bracket       <chr> "C", "A", "B", "B", "A", "D", "B", "B", "A",...
## $ total_weight      <dbl> 2.3, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.2, 1.0,...
## $ total_ship        <dbl> 5.5, 9.0, 5.2, 5.4, 9.0, 27.3, 5.3, 6.1, 5.4...
## $ total_hand        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ total_tax         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ total_amount      <dbl> 64, 18, 55, 17, 19, 137, 29, 46, 15, 23, 29,...
## $ order_status_date <dttm> 2003-10-17, 2003-10-17, 2003-10-17, 2003-10...
## $ send_inv_to_bill  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ coupon_code       <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
## $ spec_instr        <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
```

```r
unique_cat <- map_dbl(catalog, ~length(unique(.x)))
kable(unique_cat, caption = "Catalog Data: unique entry counts by data field")
```

Table 15: Catalog Data: unique entry counts by data field

	x
id	761
product_code	761
catalog_price	134
category1	10
manufact_id	5
vendor_id	5
name	756

```r
unique_cust <- map_dbl(customers, ~length(unique(.x)))
kable(unique_cust, caption = "Customers Data: unique entry counts by data field")
```

Table 16: Customers Data: unique entry counts by data field

	x
cust_id	22070
merchant_id	2
firstName	502
lastName	1001
bt_city	9032
bt_state	67
bt_country	79
bt_zip	12434
cc_type	4
custcode	22069

```r
unique_OL <- map_dbl(order_lines_sheet3, ~length(unique(.x)))
kable(unique_OL, caption = "Order Lines Data: unique entry counts by data field")
```

Table 17: Order Lines Data: unique entry counts by data field

	x
order_id	23266
order_line	22
line_status	5
line_status_date	1843
order_qty	43
shipped_qty	35
bo_exp_date	186
internal_note	1
spec_proc_note	1
spec_proc_id	1
order_line_id	31232
list_price	272
gift_note	1
distrib_id	1
product_id	678
Shipped Total	757
Ordered Total	912
format_id	7
options	1

```r
unique_orders <- map_dbl(orders, ~length(unique(.x)))
kable(unique_orders, caption = "Orders Data Table: unique entry counts by data field")
```

Table 18: Orders Data Table: unique entry counts by data field

	x
order_id	23256
merchant_id	2
order_date	2641
po_number	442
cust_id	22034
order_status	4
ship_method	16
items_amount	2105
amt_bracket	4
total_weight	444
total_ship	2298
total_hand	1
total_tax	1
total_amount	10444
order_status_date	1801
send_inv_to_bill	2
coupon_code	1
spec_instr	1