

Geo-med Pt 2

Sean Dudley

2025-01-29

Beginning

This will be redoing and recreating the project that I did in Excel for Geo-med I may not do all. This will be a recreation of sorts though to show coding ability. As a quick aside when I used the function `head()` that gives just the first few rows showing the whole table is too much.

Loading Packages needed for this

```
library(readxl)
library(openxlsx)
```

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

```
library(dplyr)
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
## filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## intersect, setdiff, setequal, union
```

These are the libraries we will use our functions from to recreate work that is already done.

Loading sheets

```
Sales <- read_excel("2.2 Sales Data.xlsx")
Cost <- read_excel("2.1 Cost.xlsx")
head(Cost)
```

```
## # A tibble: 6 x 3
##   'Item Number' MSRP Cost
##   <chr>         <dbl> <dbl>
## 1 00002         2     1.27
## 2 00003        2.75  1.81
## 3 000038-000   10     6.89
## 4 00008       15.8  11.4
## 5 00009       41    29.8
## 6 00012       88.2  57.3
```

```
head(Sales)
```

```
## # A tibble: 6 x 6
##   Year Month 'Item No' 'Price List Name' 'UOM Each Qty' 'Sales Amt'
##   <dbl> <dbl> <chr>      <chr>                <dbl>      <dbl>
## 1  2017     1 1207      FSS PRICE LIST             1      401.
## 2  2017     9 00014    2016 DOMESTIC PRICE LIST     1      60.2
## 3  2017     6 00015    2016 DOMESTIC PRICE LIST     1      22.0
## 4  2017     5 00019    2016 DOMESTIC PRICE LIST     1       2.68
## 5  2017    12 00019    2018 DOMESTIC PRICE LIST     3       8.61
## 6  2017    10 00024    2018 DOMESTIC PRICE LIST     1      67.5
```

I decided to refer to each of them as Sales and Cost now to Profit Margin as that will give us the most data we can have from this table. As you can see by just the head of the table these are the correct sheets so then to merge.

```
Cost <- Cost %>%
  mutate(Profit_Margin = ((MSRP - Cost) / MSRP) * 100)
head(Cost)
```

```
## # A tibble: 6 x 4
##   'Item Number' MSRP Cost Profit_Margin
##   <chr>         <dbl> <dbl>      <dbl>
## 1 00002         2     1.27      36.4
## 2 00003        2.75  1.81      34.0
## 3 000038-000   10     6.89      31.1
## 4 00008       15.8  11.4      27.3
## 5 00009       41    29.8      27.3
## 6 00012       88.2  57.3      35.0
```

That gives us the profit margin which I will use to create 2 more data sets with the negatives (zero and below) and positives.

```
Positive_PM <- Cost[Cost$Profit_Margin > 0, ]
Negative_PM <- Cost[Cost$Profit_Margin <= 0, ]
sort_NPM <- Negative_PM[order(-Negative_PM$Profit_Margin), ]
omit_npm <- na.omit(sort_NPM)
head(Positive_PM)
```

```
## # A tibble: 6 x 4
##   'Item Number' MSRP Cost Profit_Margin
##   <chr>         <dbl> <dbl>      <dbl>
```

```
## 1 00002      2      1.27      36.4
## 2 00003      2.75  1.81      34.0
## 3 000038-000  10      6.89      31.1
## 4 00008     15.8  11.4      27.3
## 5 00009     41    29.8      27.3
## 6 00012     88.2  57.3      35.0
```

```
head(omit_npm)
```

```
## # A tibble: 5 x 4
##   'Item Number' MSRP Cost Profit_Margin
##   <chr>         <dbl> <dbl>         <dbl>
## 1 AB007002     25    25             0
## 2 AB007006     90    90             0
## 3 AB008009     90    90             0
## 4 10230        29.8  35.5        -19.1
## 5 10220        29.8  43.2        -45.2
```

This give us both the positive and negative (zero and below in this case) Profit margins in 2 separate data sets to draw conclusions from. The negatvie had to be cleaned up as it was mostly NA when positives omitted.

Merging sheets

```
colnames(Cost)[colnames(Cost) == "Item Number"] <- "Item No"
merged <- merge(x = Sales, y = Cost, by = "Item No", all = FALSE)
head(merged)
```

```
##   Item No Year Month      Price List Name UOM Each Qty Sales Amt  MSRP
## 1  00014 2017     9 2016 DOMESTIC PRICE LIST      1    60.24 88.20
## 2  00015 2017     6 2016 DOMESTIC PRICE LIST      1    21.99 30.25
## 3  00019 2017    12 2018 DOMESTIC PRICE LIST      3     8.61  3.75
## 4  00019 2017     5 2016 DOMESTIC PRICE LIST      1     2.68  3.75
## 5  00024 2017    10 2018 DOMESTIC PRICE LIST      1    67.47 88.20
## 6  00030 2017     8 2016 DOMESTIC PRICE LIST      1    11.48 16.00
##      Cost Profit_Margin
## 1 57.3495    34.97789
## 2 20.8905    30.94050
## 3  2.5460    32.10667
## 4  2.5460    32.10667
## 5 57.3495    34.97789
## 6 10.4040    34.97500
```

That displays just the first few rows of our newly created table that is the merged sets.

Unit price etc.

```
merged <- merged %>%
  mutate(Unit_Price = `Sales Amt` / `UOM Each Qty`)

head(merged)
```

##	Item No	Year	Month	Price List Name	UOM	Each Qty	Sales Amt	MSRP
## 1	00014	2017	9	2016 DOMESTIC PRICE LIST		1	60.24	88.20
## 2	00015	2017	6	2016 DOMESTIC PRICE LIST		1	21.99	30.25
## 3	00019	2017	12	2018 DOMESTIC PRICE LIST		3	8.61	3.75
## 4	00019	2017	5	2016 DOMESTIC PRICE LIST		1	2.68	3.75
## 5	00024	2017	10	2018 DOMESTIC PRICE LIST		1	67.47	88.20
## 6	00030	2017	8	2016 DOMESTIC PRICE LIST		1	11.48	16.00

##	Cost	Profit_Margin	Unit_Price
## 1	57.3495	34.97789	60.24
## 2	20.8905	30.94050	21.99
## 3	2.5460	32.10667	2.87
## 4	2.5460	32.10667	2.68
## 5	57.3495	34.97789	67.47
## 6	10.4040	34.97500	11.48

This gives us one more columns on the end just as I did it where we have individual unit cost that will be used later.

```
returns <- merged[merged$`UOM Each Qty` <= 0, ]
head(returns)
```

##	Item No	Year	Month	Price List Name	UOM	Each Qty	Sales Amt	MSRP	Cost
## 215	00374	2017	12	FSS PRICE LIST		-1	-60.26	106.5	57.2470
## 223	00382	2017	12	FSS PRICE LIST		-5	0.00	169.5	92.3495
## 251	00384	2017	11	FSS PRICE LIST		-1	-97.21	169.5	92.3495
## 262	00384	2017	12	FSS PRICE LIST		-1	0.00	169.5	92.3495
## 300	00394	2017	9	FSS PRICE LIST		-1	-60.26	106.5	57.2470
## 679	00630	2017	5	FSS PRICE LIST		-1	0.00	508.5	234.6595

##	Profit_Margin	Unit_Price
## 215	46.24695	60.26
## 223	45.51652	0.00
## 251	45.51652	97.21
## 262	45.51652	0.00
## 300	46.24695	60.26
## 679	53.85261	0.00

This is the set of items from the Sales Data that is returns doesn't do much but nice to see most returned product.

```
merged2 <- merged[merged$`UOM Each Qty` > 0, ]
merged3 <- merged2 %>%
  mutate(UminC = Unit_Price - Cost)
MSRPsale <- merged2[(merged2$Unit_Price - merged2$MSRP) >= 0, ]
sort_ms <- MSRPsale[order(MSRPsale$`Item No`), ]
average_sales <- merged %>%
  group_by(`Item No`) %>%
  summarize(Average_Sale = mean(`Sales Amt`, na.rm = TRUE))
head(sort_ms)
```

```
##      Item No Year Month Price List Name UOM Each Qty Sales Amt  MSRP  Cost
## 7496   10220 2017     2 DAPA PRICE LIST          5   216.25 29.78 43.25
## 7497   10220 2017     2 DAPA PRICE LIST         15   648.75 29.78 43.25
## 7498   10220 2017     2 DAPA PRICE LIST         30  1297.50 29.78 43.25
## 7499   10220 2017     3 DAPA PRICE LIST         24  1038.00 29.78 43.25
## 7500   10220 2017     5 DAPA PRICE LIST         10   432.50 29.78 43.25
## 7501   10220 2017     5 DAPA PRICE LIST          2    86.50 29.78 43.25
##      Profit_Margin Unit_Price
## 7496      -45.2317      43.25
## 7497      -45.2317      43.25
## 7498      -45.2317      43.25
## 7499      -45.2317      43.25
## 7500      -45.2317      43.25
## 7501      -45.2317      43.25
```

```
head(average_sales)
```

```
## # A tibble: 6 x 2
##   'Item No' Average_Sale
##   <chr>         <dbl>
## 1 00014         60.2
## 2 00015         22.0
## 3 00019          5.64
## 4 00024         67.5
## 5 00030         11.5
## 6 00035         11.7
```

From this we get data sets of what products we can sell at or above MSRP and also the average sale according to our Sales data both operations I did in the excel sheets separately as they are done here.

```
Unprofitable1 <- merged[(merged$Unit_Price - merged$Cost) >= 0, ]
head(Unprofitable1)
```

```
##      Item No Year Month      Price List Name UOM Each Qty Sales Amt  MSRP
## 1   00014 2017     9 2016 DOMESTIC PRICE LIST          1   60.24 88.20
## 2   00015 2017     6 2016 DOMESTIC PRICE LIST          1   21.99 30.25
## 3   00019 2017    12 2018 DOMESTIC PRICE LIST          3    8.61  3.75
## 4   00019 2017     5 2016 DOMESTIC PRICE LIST          1    2.68  3.75
## 5   00024 2017    10 2018 DOMESTIC PRICE LIST          1   67.47 88.20
## 6   00030 2017     8 2016 DOMESTIC PRICE LIST          1   11.48 16.00
##      Cost Profit_Margin Unit_Price
## 1 57.3495      34.97789      60.24
## 2 20.8905      30.94050      21.99
## 3  2.5460      32.10667       2.87
## 4  2.5460      32.10667       2.68
## 5 57.3495      34.97789      67.47
## 6 10.4040      34.97500      11.48
```

This shows each unprofitable product by Unit sale price minus cost as that means they are already being sold cheaper than able to make.

```
unmatched <- anti_join(Cost, Sales, by = "Item No")
sortunmatch<- unmatched[order(unmatched$`Item No`), ]
head(sortunmatch)
```

```
## # A tibble: 6 x 4
##   'Item No'   MSRP   Cost Profit_Margin
##   <chr>      <dbl> <dbl>      <dbl>
## 1 00002         2    1.27        36.4
## 2 00003        2.75    1.81        34.0
## 3 000038-000  10     6.89        31.1
## 4 00008       15.8   11.4        27.3
## 5 00009        41    29.8        27.3
## 6 00012       88.2   57.3        35.0
```

Finally this gives us the item numbers that were not matched to the sales data to show what items we have no info on and either need more or will sell at MSRP

End

This is the end of the redoing, I believe I accomplished much of what we talked about on the call though it may not be in complete detail as I did in the excel, I believe this shows off my ability to code in R. Next I will try to complete the Python ask if necessary.