Hands-on Exercise 10: Information Dashboard Design: R methods

Dr. Kam Tin Seong
Assoc. Professor of Information Systems

School of Computing and Information Systems,
Singapore Management University

Overview

By the end of this hands-on exercise, you will be able to:

- create bullet chart by using ggplot2,
- create sparklines by using ggplot2 ,
- build industry standard dashboard by using R Shiny.

Importing Microsoft Access database

The data set

For the purpose of this study, a personal database in Microsoft Access mdb format called *Coffee Chain* will be used.

Importing database into R

In the code chunk below, `odbcConnectAccess()` of **RODBC** package is used used to import a database query table into R.

```r
library(RODBC)
con <- odbcConnectAccess('data/Coffee Chain.mdb')
coffeechain <- sqlFetch(con, 'CoffeeChain Query')
write_rds(coffeechain, "data/CoffeeChain.rds")
odbcClose(con)
```

Note: Before running the code chunk, you need to change the R system to 32bit version. This is because the `odbcConnectAccess()` is based on 32bit and not 64bit

Getting started

For the purpose of this hands-on exercise, the following R packages will be used.

```
packages = c('tidyverse', 'lubridate',
             'ggthemes')

for (p in packages){
  if(!require(p, character.only = T)){
    install.packages(p)
  }
  library(p,character.only = T)
}
```

The code chunk below is used to import *CoffeeChain.rds* into R.

```
coffeechain <- read_rds("data/CoffeeChain.rds"
```

Note: This step is optional if *coffeechain* is already available in R.

Data Preparation

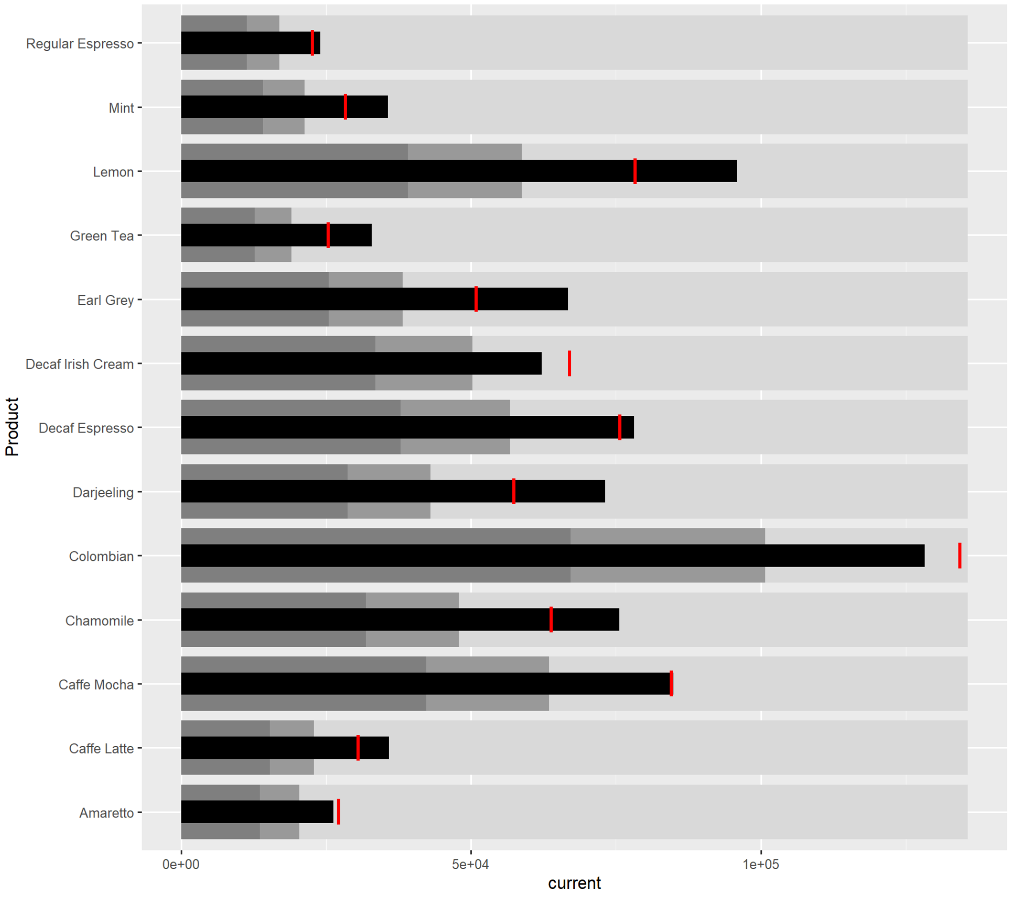
The code chunk below is used to aggregate Sales and Budgeted Sales at the Product level.

```
product <- coffeechain %>%
  group_by(`Product`) %>%
  summarise(`target` = sum(`Budget Sales`),
            `current` = sum(`Sales`)) %>%
  ungroup()
```

Bullet chart in ggplot2

The code chunk below is used to plot the bullet charts using ggplot2 functions.

```
ggplot(product, aes(Product, current)) +
  geom_col(aes(Product, max(target) * 1.01),
           fill="grey85", width=0.85) +
  geom_col(aes(Product, target * 0.75),
           fill="grey60", width=0.85) +
  geom_col(aes(Product, target * 0.5),
           fill="grey50", width=0.85) +
  geom_col(aes(Product, current),
           width=0.35,
           fill = "black") +
  geom_errorbar(aes(y = target,
                    x = Product,
                    ymin = target,
                    ymax= target),
                width = .4,
                colour = "red",
                size = 1) +
  coord_flip()
```



Plotting sparklines using ggplot2

In this section, you will learn how to plot sparklines by using ggplot2.

Preparing the data

```
sales_report <- coffeechain %>%
  filter(Date >= "2013-01-01") %>%
  mutate(Month = month(Date)) %>%
  group_by(Month, Product) %>%
  summarise(Sales = sum(Sales)) %>%
  ungroup() %>%
  select(Month, Product, Sales)
```

The code chunk below is used to compute the minimum, maximum and end othe the month sales.

```
mins <- group_by(sales_report, Product) %>%
  slice(which.min(Sales))
maxs <- group_by(sales_report, Product) %>%
  slice(which.max(Sales))
ends <- group_by(sales_report, Product) %>%
  filter(Month == max(Month))
```

The code chunk below is used to compute the 25 and 75 quantiles.

```
quarts <- sales_report %>%
  group_by(Product) %>%
  summarise(quart1 = quantile(Sales,
                              0.25),
            quart2 = quantile(Sales,
                              0.75)) %>%
  right_join(sales_report)
```

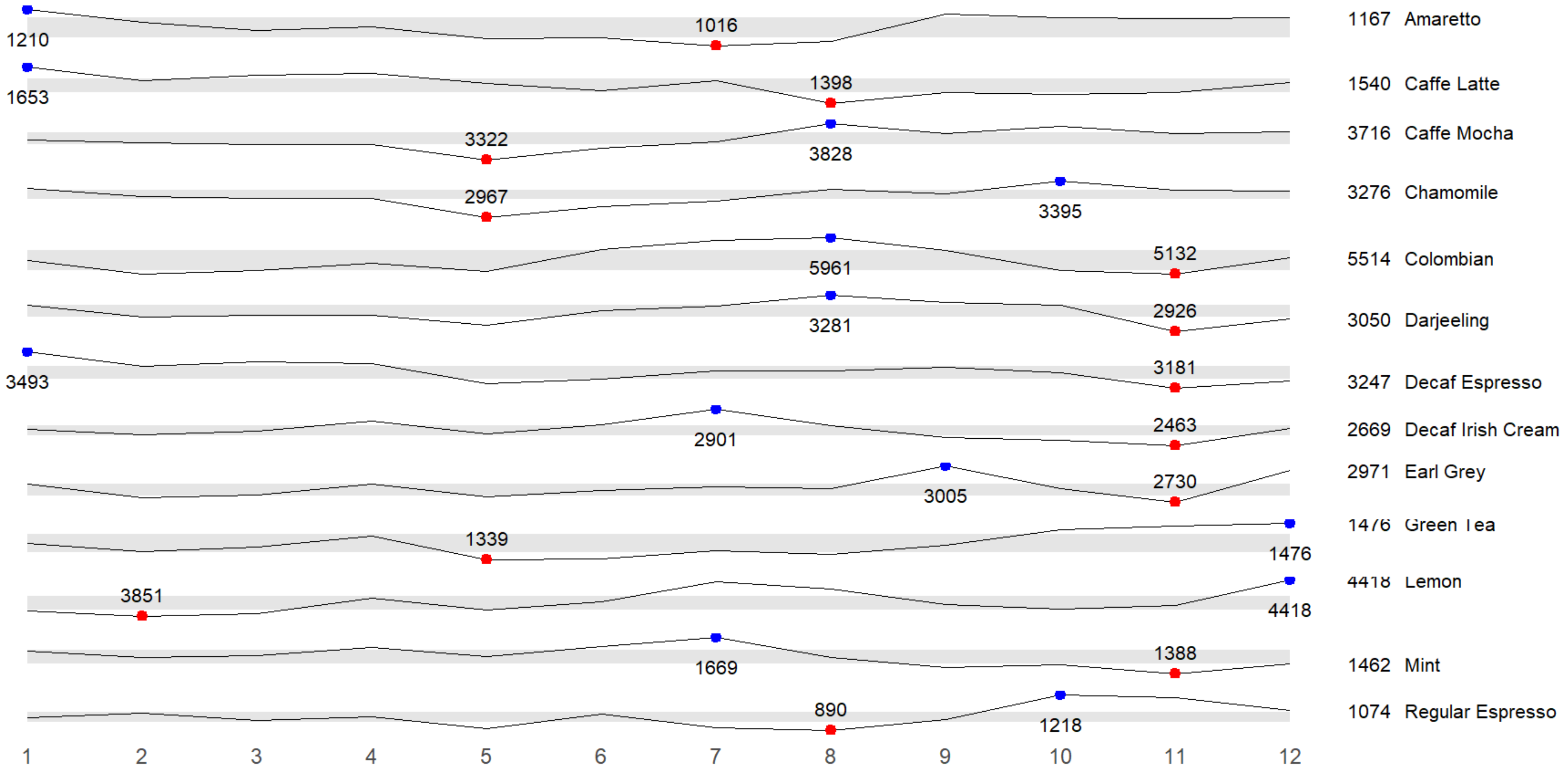
sparklines in ggplot2

The code chunk used.

```r
ggplot(sales_report, aes(x=Month, y=Sales)) +
  facet_grid(Product ~ ., scales = "free_y") +
  geom_ribbon(data = quarts, aes(ymin = quart1, max = quart2),
              fill = 'grey90') +
  geom_line(size=0.3) +
  geom_point(data = mins, col = 'red') +
  geom_point(data = maxs, col = 'blue') +
  geom_text(data = mins, aes(label = Sales), vjust = -1) +
  geom_text(data = maxs, aes(label = Sales), vjust = 2.5) +
  geom_text(data = ends, aes(label = Sales), hjust = 0, nudge_x = 0.5) +
  geom_text(data = ends, aes(label = Product), hjust = 0, nudge_x = 1) +
  expand_limits(x = max(sales_report$Month) +
                  (0.25 * (max(sales_report$Month) - min(sales_report$Month)))) +
  scale_x_continuous(breaks = seq(1, 12, 1)) +
  scale_y_continuous(expand = c(0.1, 0)) +
  theme_tufte(base_size = 3, base_family = "Helvetica") +
  theme(axis.title=element_blank(), axis.text.y = element_blank(),
        axis.ticks = element_blank(), strip.text = element_blank())
```

sparklines in ggplot2

The output


1167 Amaretto
1540 Caffe Latte
3716 Caffe Mocha
3276 Chamomile
5514 Colombian
3050 Darjeeling
3247 Decaf Espresso
2669 Decaf Irish Cream
2971 Earl Grey
1476 Green Tea
4418 Lemon
1462 Mint
1074 Regular Espresso