Lecture 3 - EDA

探索型資料分析與資料視覺化

了解資料的外觀、維度及變數的分佈等資訊

資料視覺化

在不簡化資訊情況下，降低複雜資料的理解門檻；以較簡單的方式，去理解高維度(複雜)的資料。

- 單變數
  - 類別型變數 `pie()`，`barplot()`
  - 連續型變數 `hist()`
- 雙變數
  - 連續 vs 連續 `plot(x,y)`，`xyplot()`
  - 連續 vs 離散 `boxplot()`，`barplot()`
  - 離散 vs 離散 `mosaicplot()`
- 多變量 `plot(data)`，`cloud()`，`corrgram()`，`heatmap.2()`，`corrplot()`

https://www.analyticsvidhya.com/blog/2015/07/guide-data-visualization-r/ (https://www.analyticsvidhya.com/blog/2015/07/guide-data-visualization-r/)

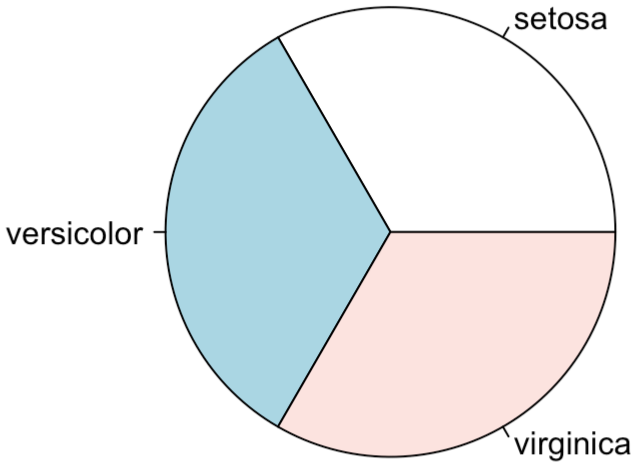
https://rpubs.com/skydome20/R-Note4-Plotting_System (https://rpubs.com/skydome20/R-Note4-Plotting_System)

Visualization of large datasets with `tabplot` https://github.com/mtennekes/tabplot (https://github.com/mtennekes/tabplot)

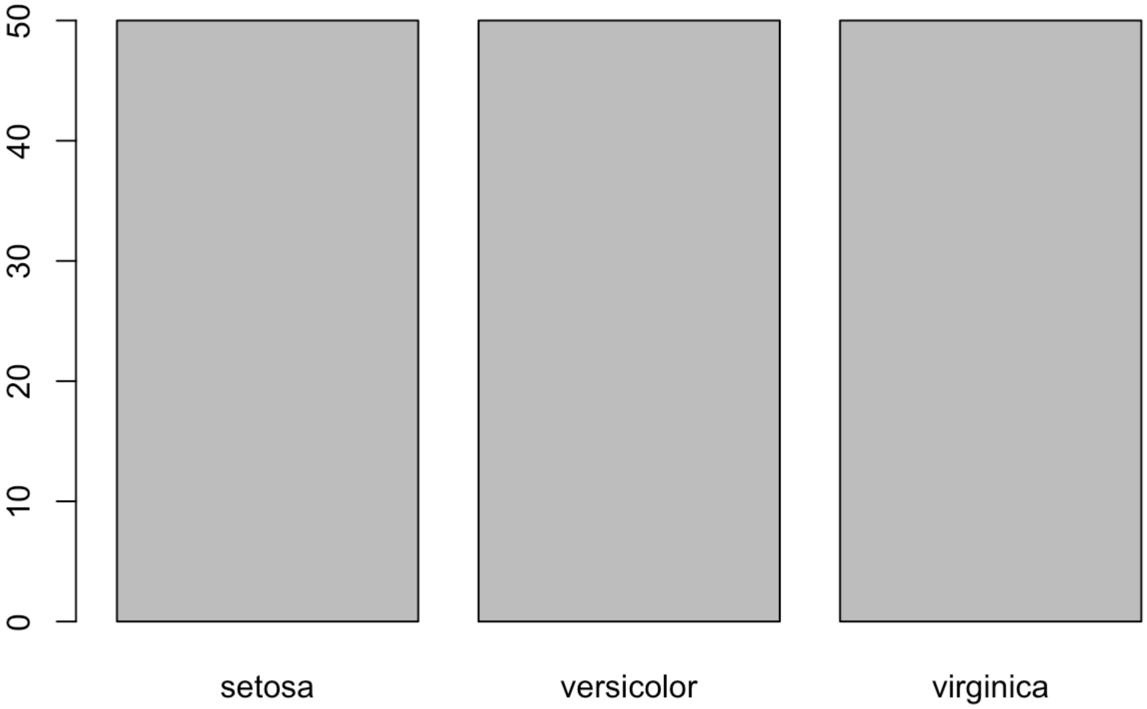
https://mran.microsoft.com/snapshot/2015-11-17/web/packages/tabplot/vignettes/tabplot-vignette.html (https://mran.microsoft.com/snapshot/2015-11-17/web/packages/tabplot/vignettes/tabplot-vignette.html)

Example

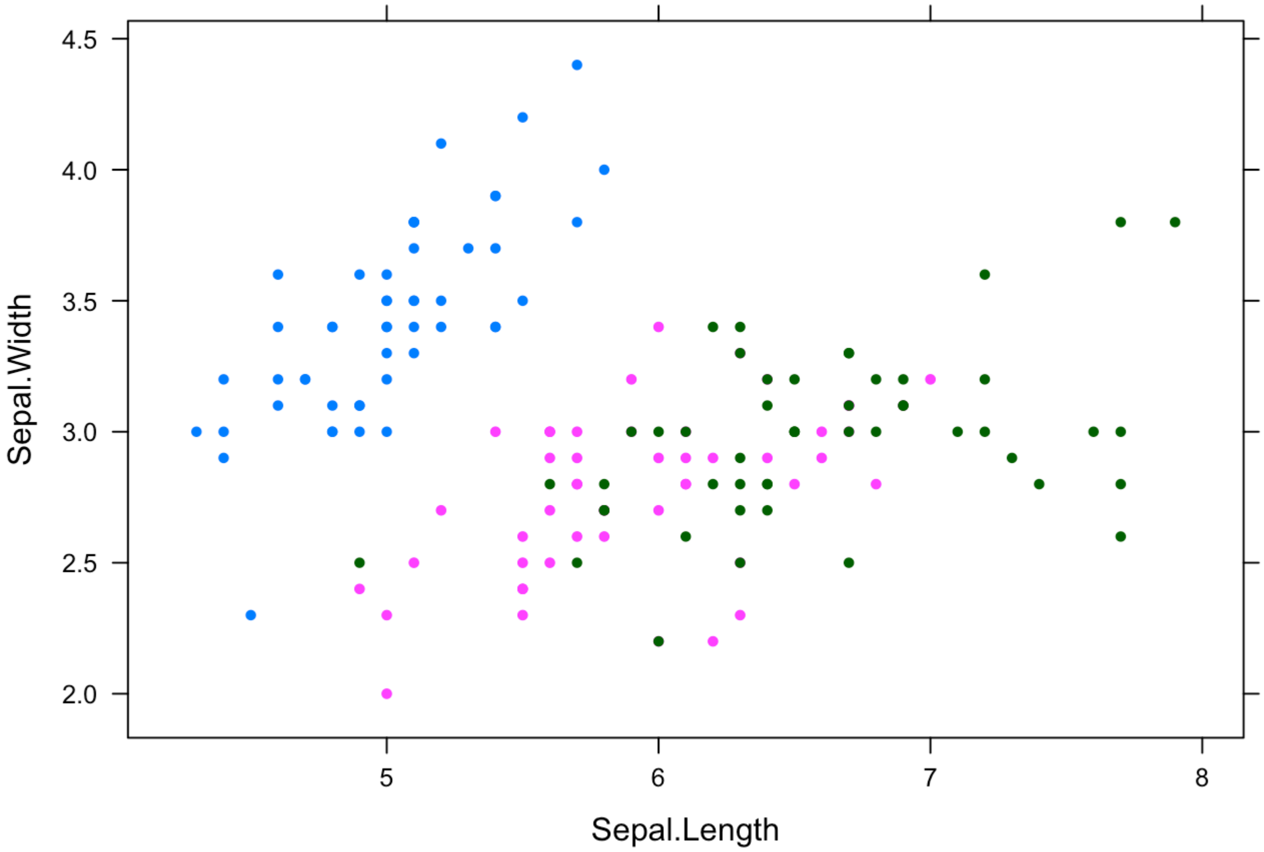
```
# Discrete
pie(table(iris$Species))
```



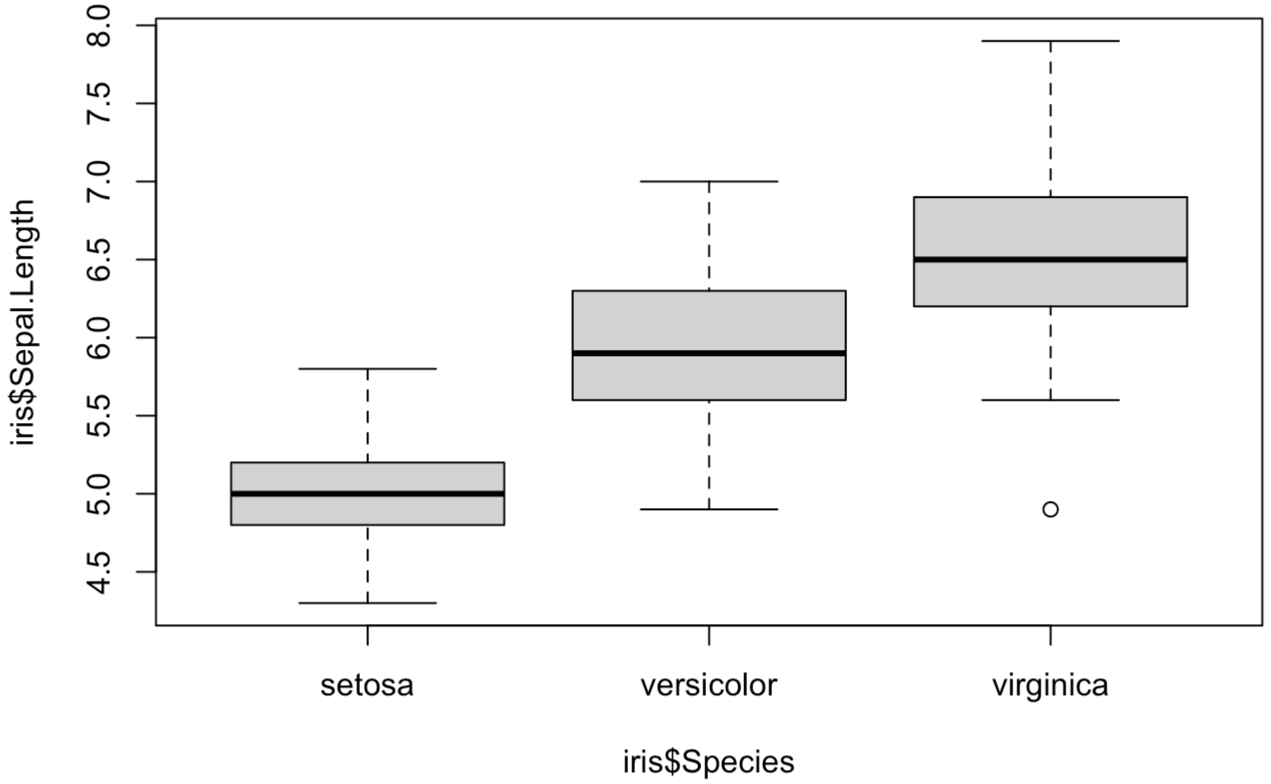
```
barplot(table(iris$Species))
```



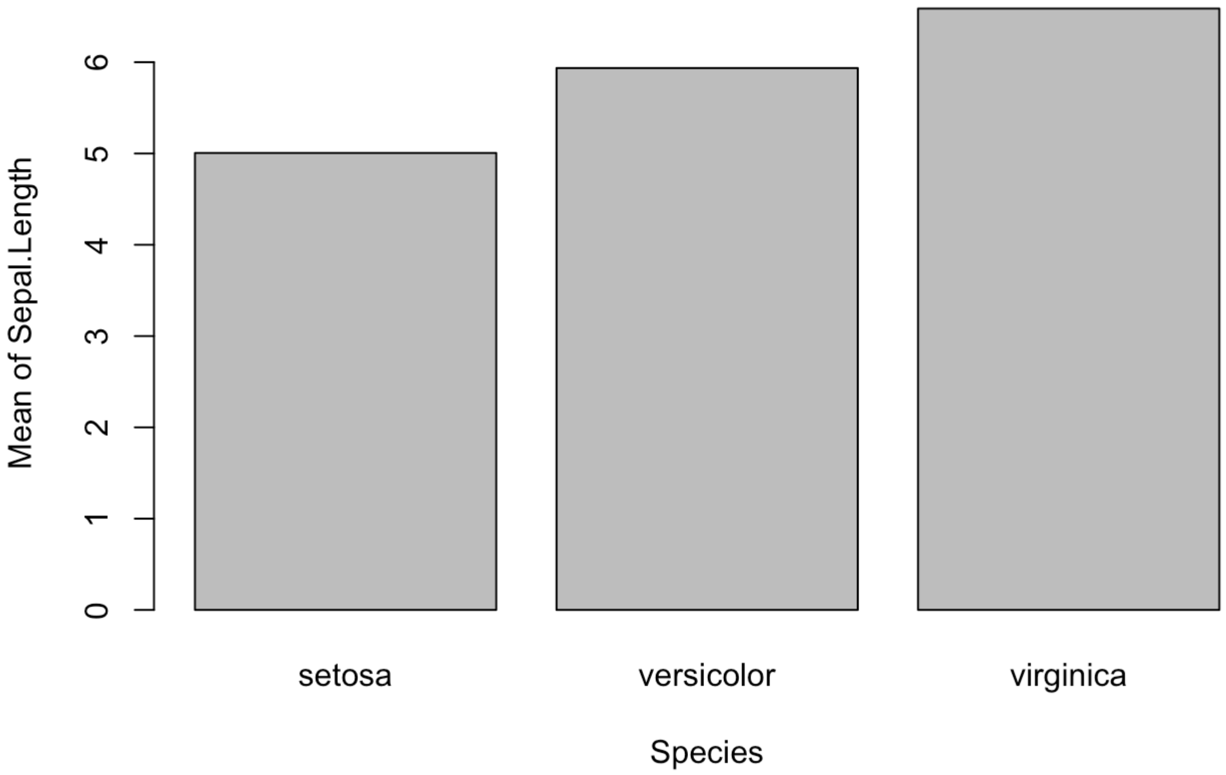
```
# two variables: continuous n continuous
library(lattice)
xyplot(Sepal.Width ~ Sepal.Length, iris, groups = Species, pch= 20)
```



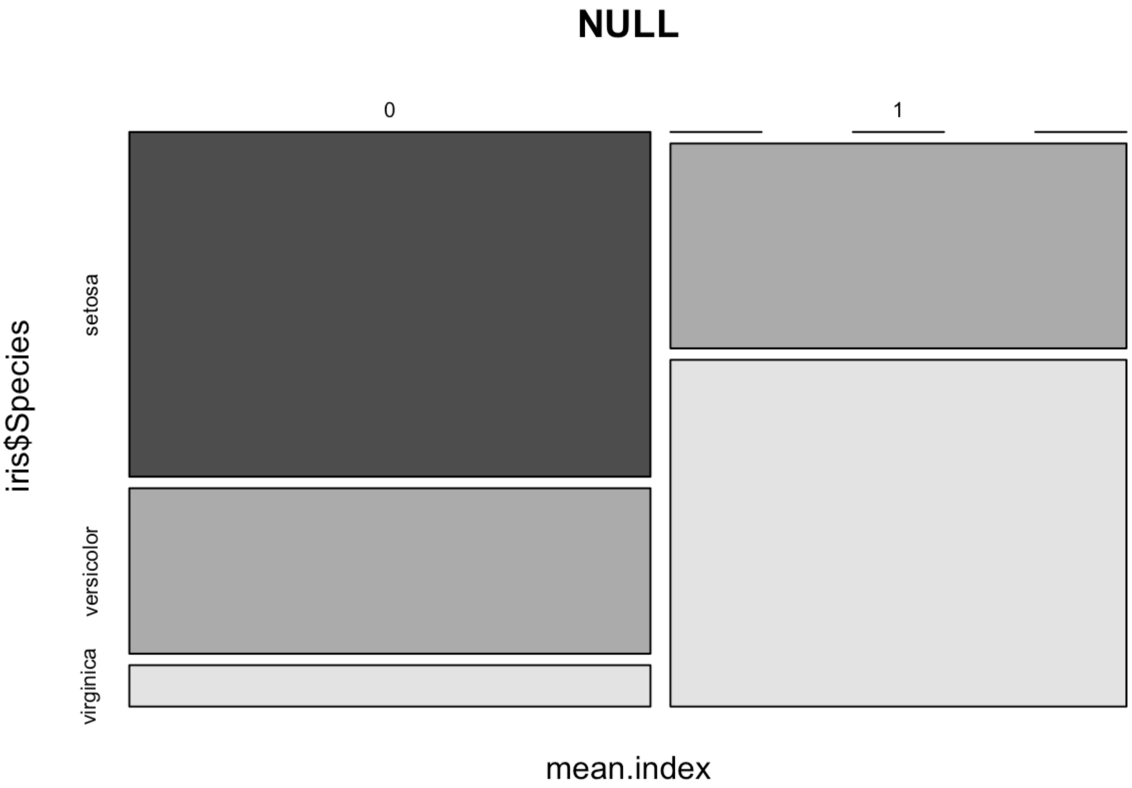
```
# two variables: continuous n discrete
boxplot(iris$Sepal.Length~iris$Species)
```



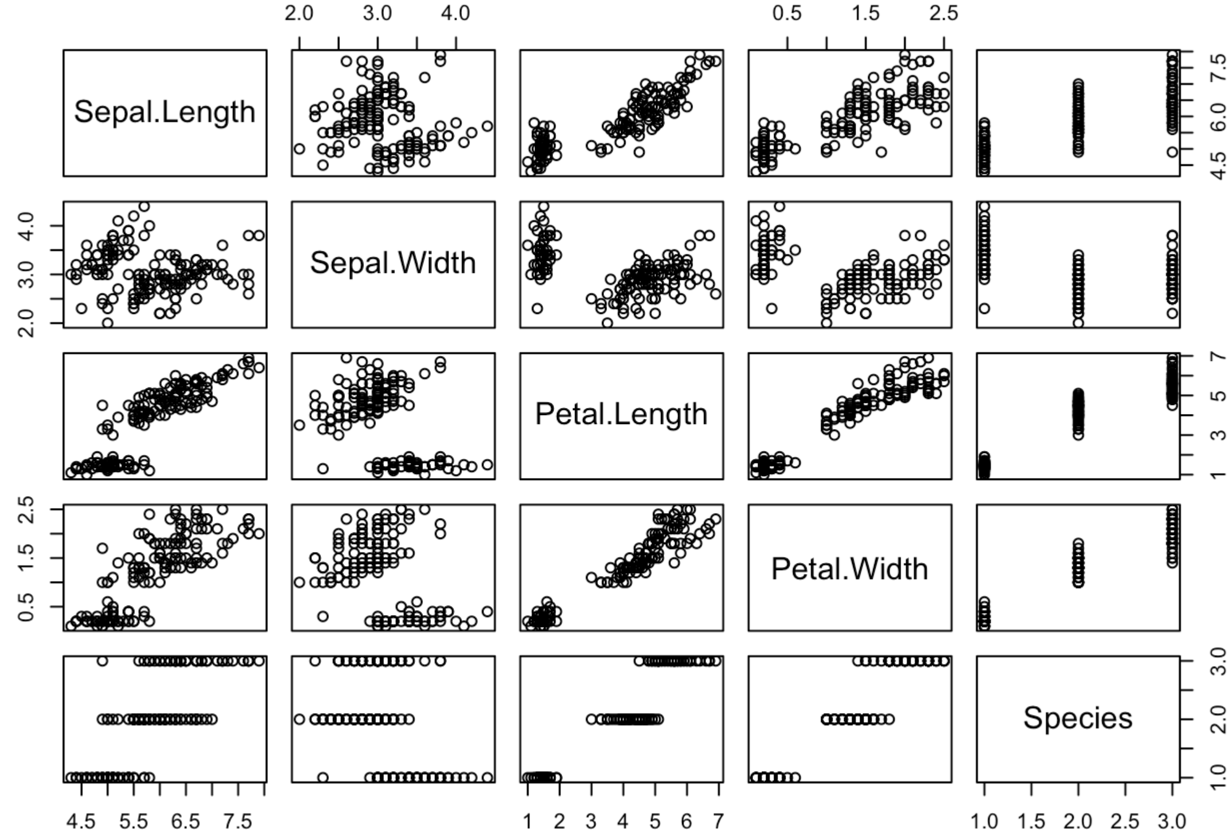
```
length.means = tapply(
  iris$Sepal.Length,
  iris$Species,
  mean)
barplot(length.means,
         xlab = "Species",
         ylab = "Mean of Sepal.Length")
```



```
# two variables: discrete n discrete
mean.index <- ifelse(iris$Sepal.Length>mean(iris$Sepal.Length),1,0)
mosaicplot( ~mean.index + iris$Species, color=T)
```



```
# multivariables
plot(iris)
```



ggplot2

語法

```
ggplot(data,aes(x,y)) +
散佈圖：geom_point()
線圖：geom_line()
直方圖：geom_histogram()
盒鬚圖：geom_boxplot()
長條圖：geom_bar()
```

ref: http://www.sthda.com/english/wiki/ggplot2-essentials (http://www.sthda.com/english/wiki/ggplot2-essentials)

(請參考tidyverse lecture.)

ggtheme

https://ggplot2.tidyverse.org/reference/ggtheme.html (https://ggplot2.tidyverse.org/reference/ggtheme.html)

Background

https://www.r-bloggers.com/adding-custom-fonts-to-ggplot-in-r/ (https://www.r-bloggers.com/adding-custom-fonts-to-ggplot-in-r/) https://wilkelab.org/cowplot/reference/theme_cowplot.html (https://wilkelab.org/cowplot/reference/theme_cowplot.html)

```
library(tidyverse)
```

```
## ─ Attaching packages ──────────────────── tidyverse 1.3.1 ─
```

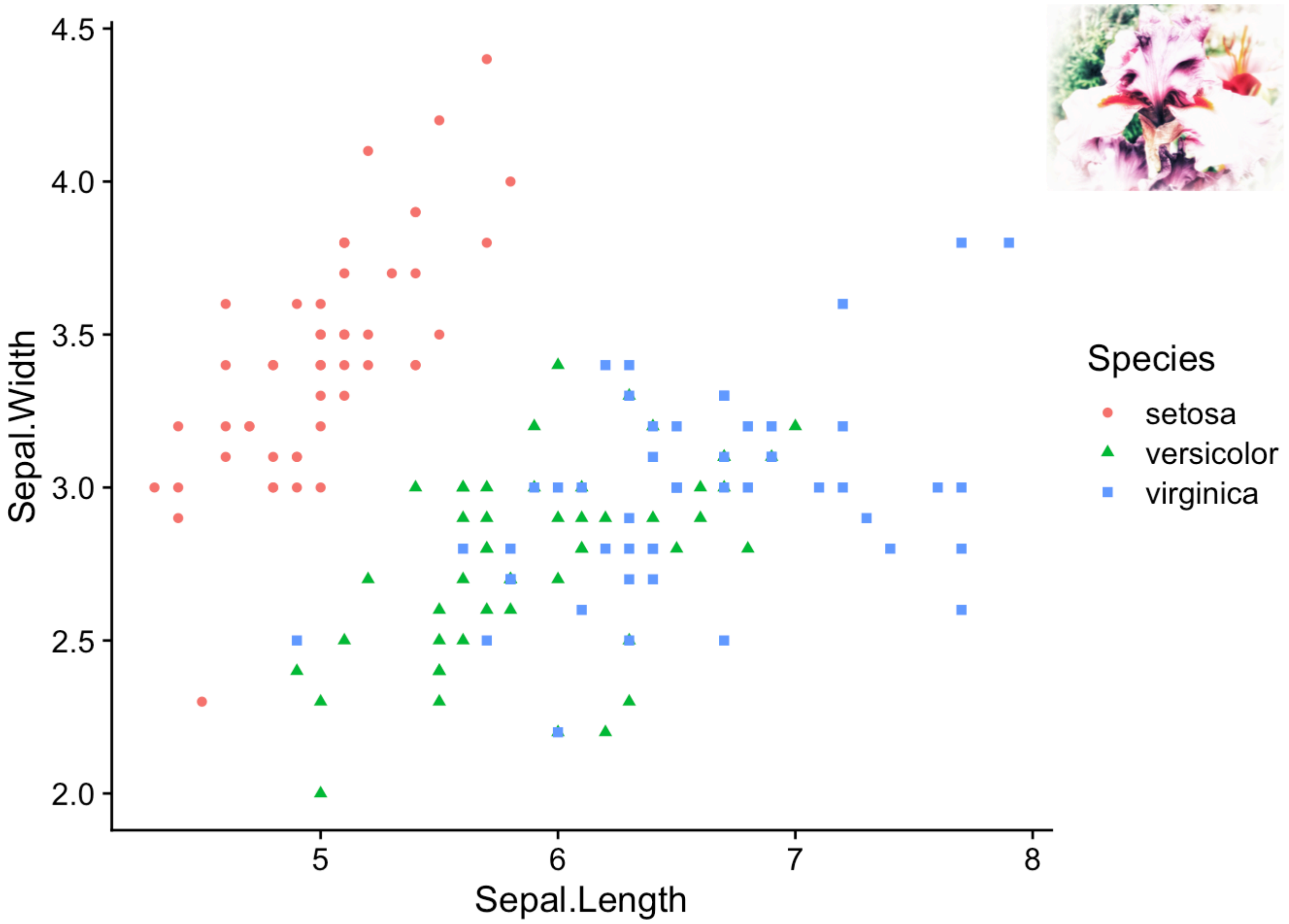
```
## ✓ ggplot2 3.3.3     ✓ purrr   0.3.4
## ✓ tibble  3.1.2     ✓ dplyr   1.0.6
## ✓ tidyr   1.1.3     ✓ stringr 1.4.0
## ✓ readr   1.4.0     ✓ forcats 0.5.1
```

```
## ─ Conflicts ────────────────────── tidyverse_conflicts() ─
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

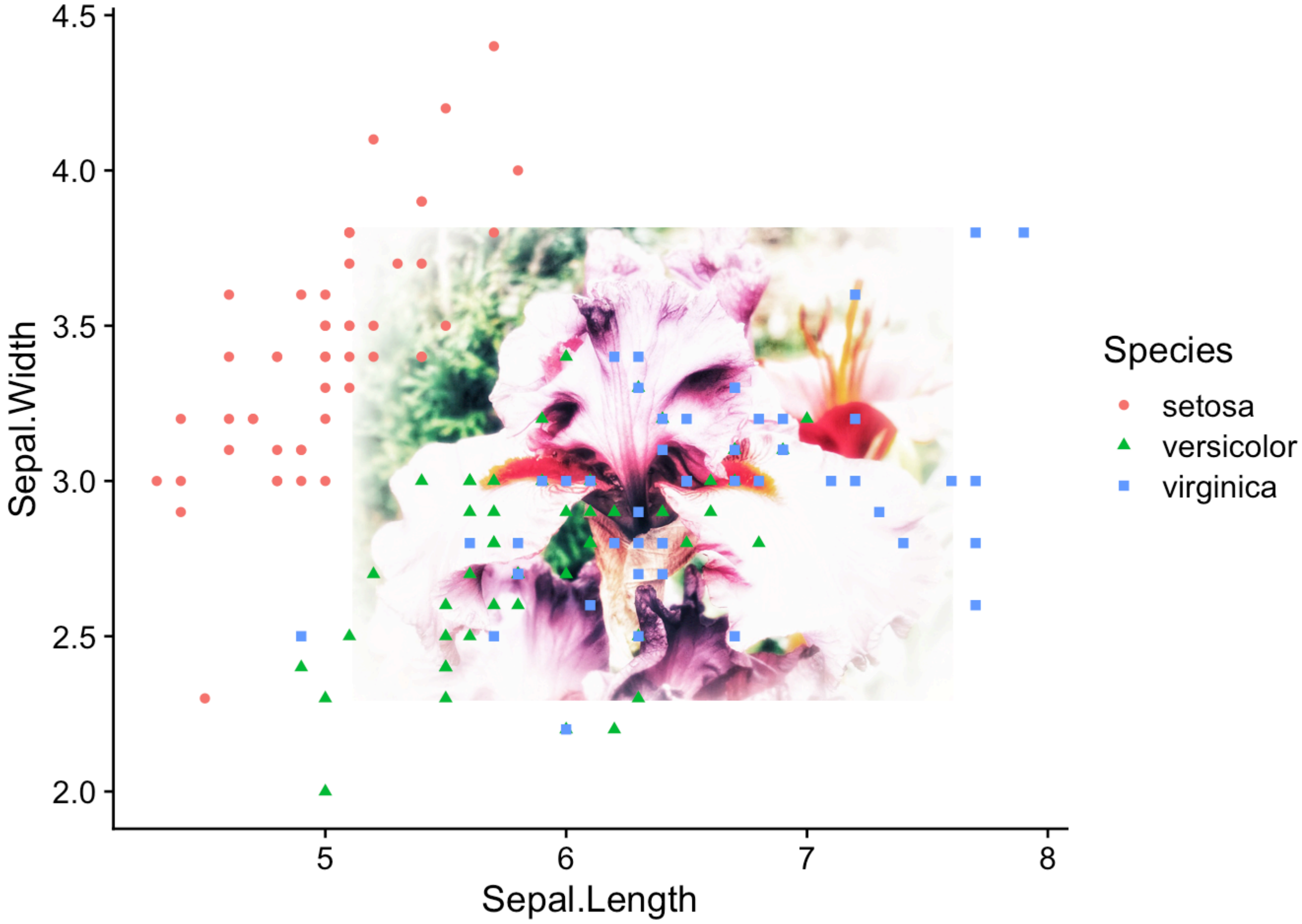
```
library(cowplot)
library(magick)
```

```
## Linking to ImageMagick 6.9.12.3
## Enabled features: cairo, fontconfig, freetype, heic, lcms, pango, raw, rsvg, webp
## Disabled features: fftw, ghostscript, x11
```

```
ii <- iris %>%
  ggplot(aes(x=Sepal.Length, y=Sepal.Width ,color=Species,shape=Species)) +
  geom_point(size=1.5) +
  theme_cowplot()

ggdraw(ii) +
  draw_image("flower.jpeg", x = 1, y = 1, width = 0.2, height = 0.2,hjust = 1, vjust = 1)
```

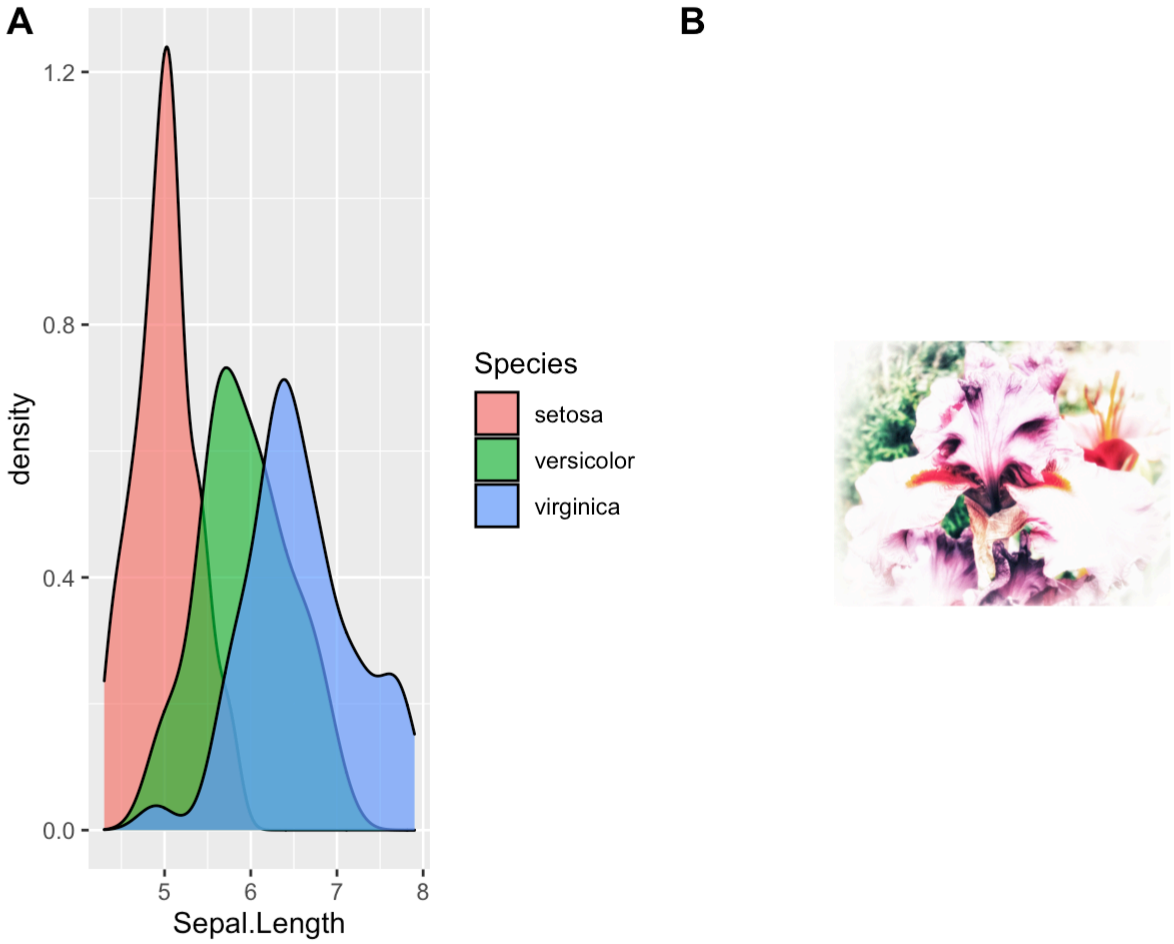


```
ggdraw() +
  draw_image("flower.jpeg", scale = 0.5) +  # the background for the plot
  draw_plot(ii)
```

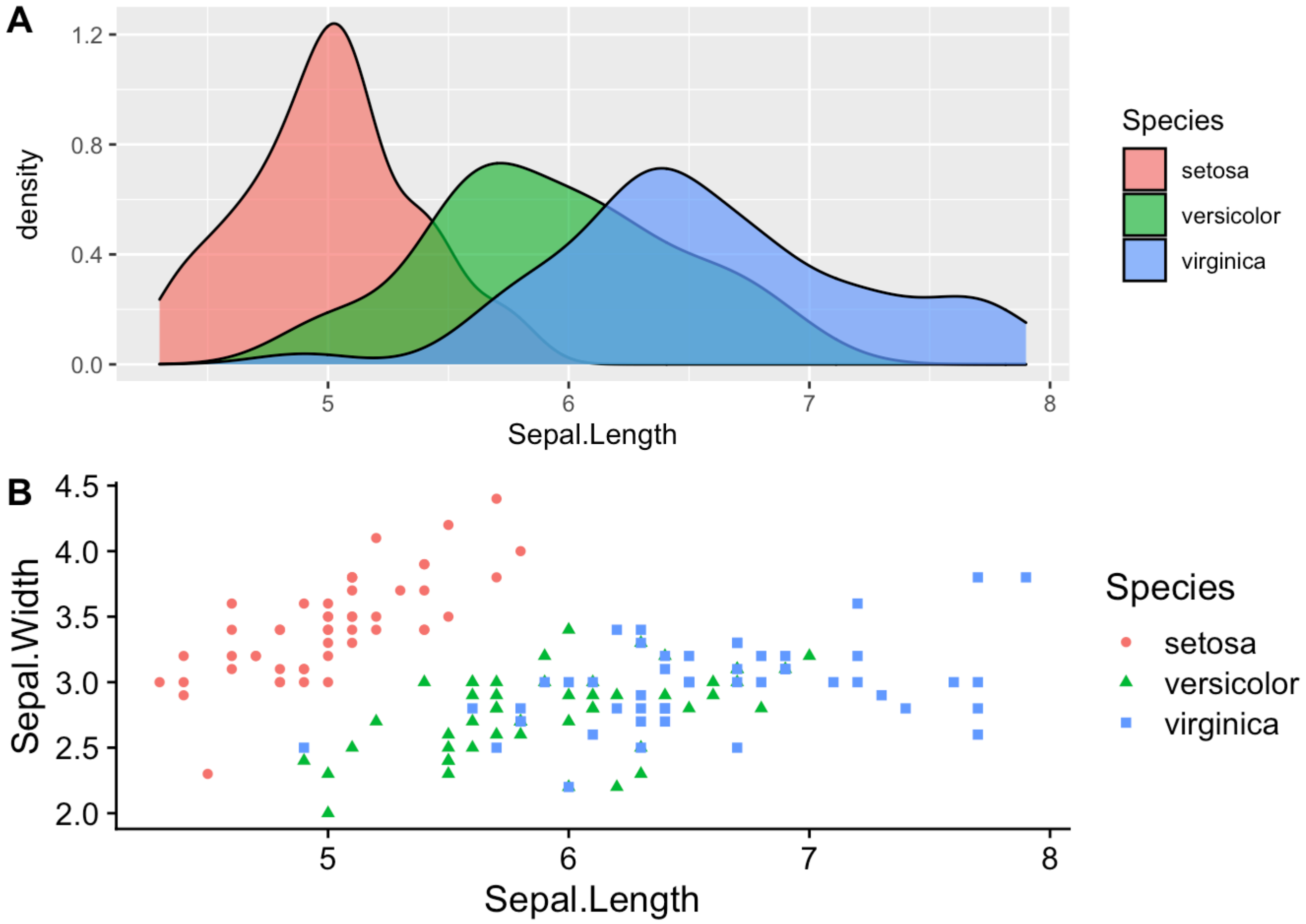


輸出圖的排序

```
p <- ggplot(iris, aes(x = Sepal.Length, fill = Species)) + geom_density(alpha = 0.7)
p2 <- ggdraw() + draw_image("flower.jpeg", scale = 0.5)
plot_grid(p, p2, labels = "AUTO")
```



```
plot_grid(p, ii, labels = c("A","B"), ncol = 1, align = 'v')
```



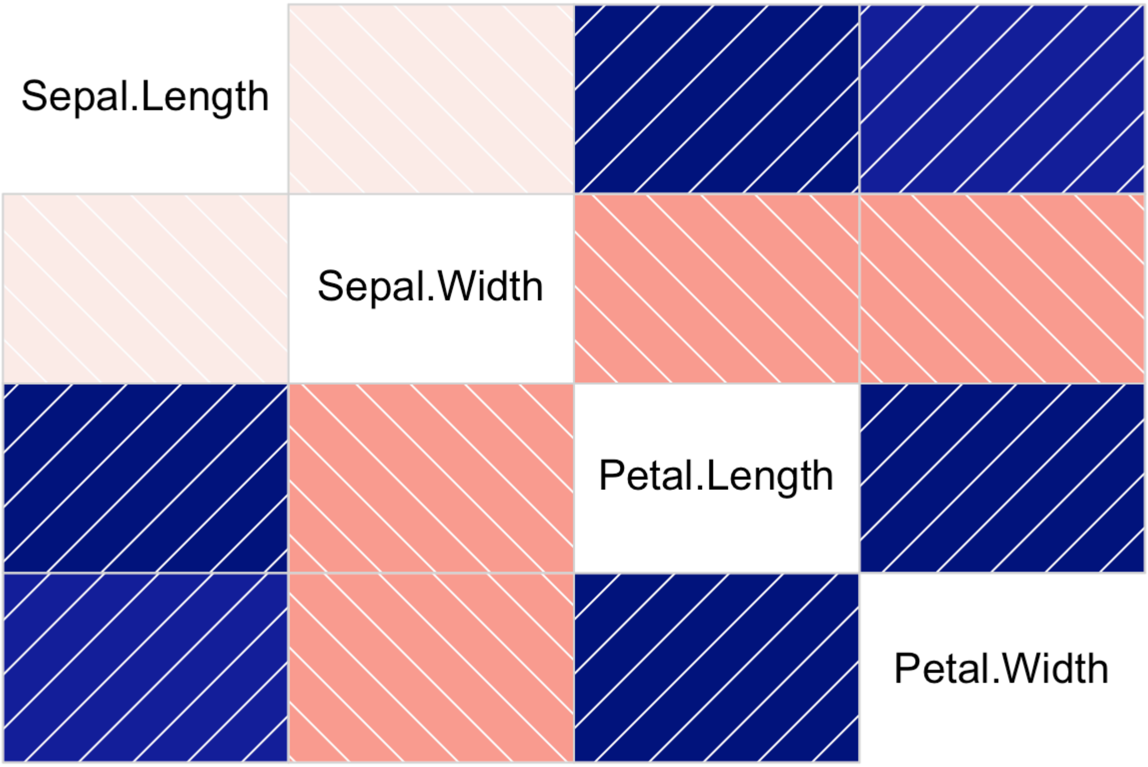
關係圖

```
#install.packages("corrgram")
library(corrgram)
```

```
##
## 載入套件：'corrgram'
```

```
## 下列物件被遮斷自 'package:lattice':
##
##     panel.fill
```

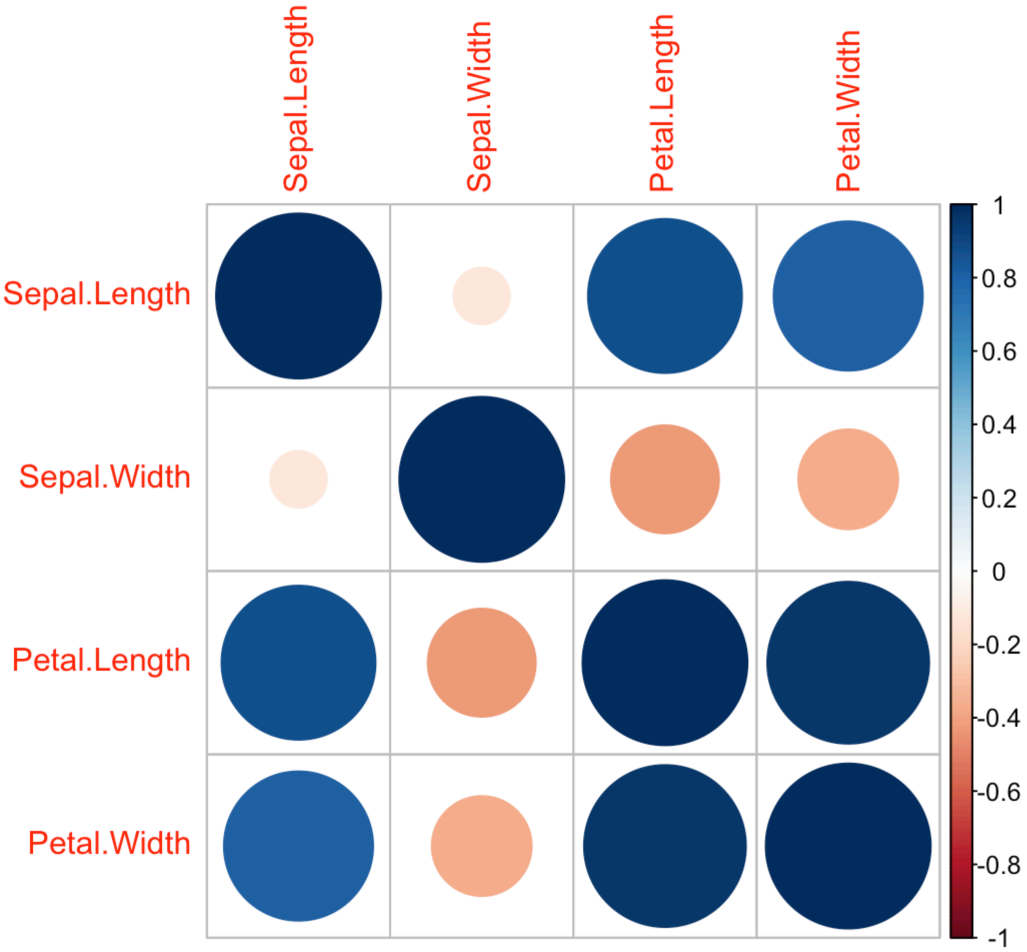
```
corr = cor(iris[,1:4])
corrgram(iris)
```



```
#install.packages("corrplot")
library(corrplot)
```

```
## corrplot 0.90 loaded
```

```
corrplot(corr, method = "circle")
```

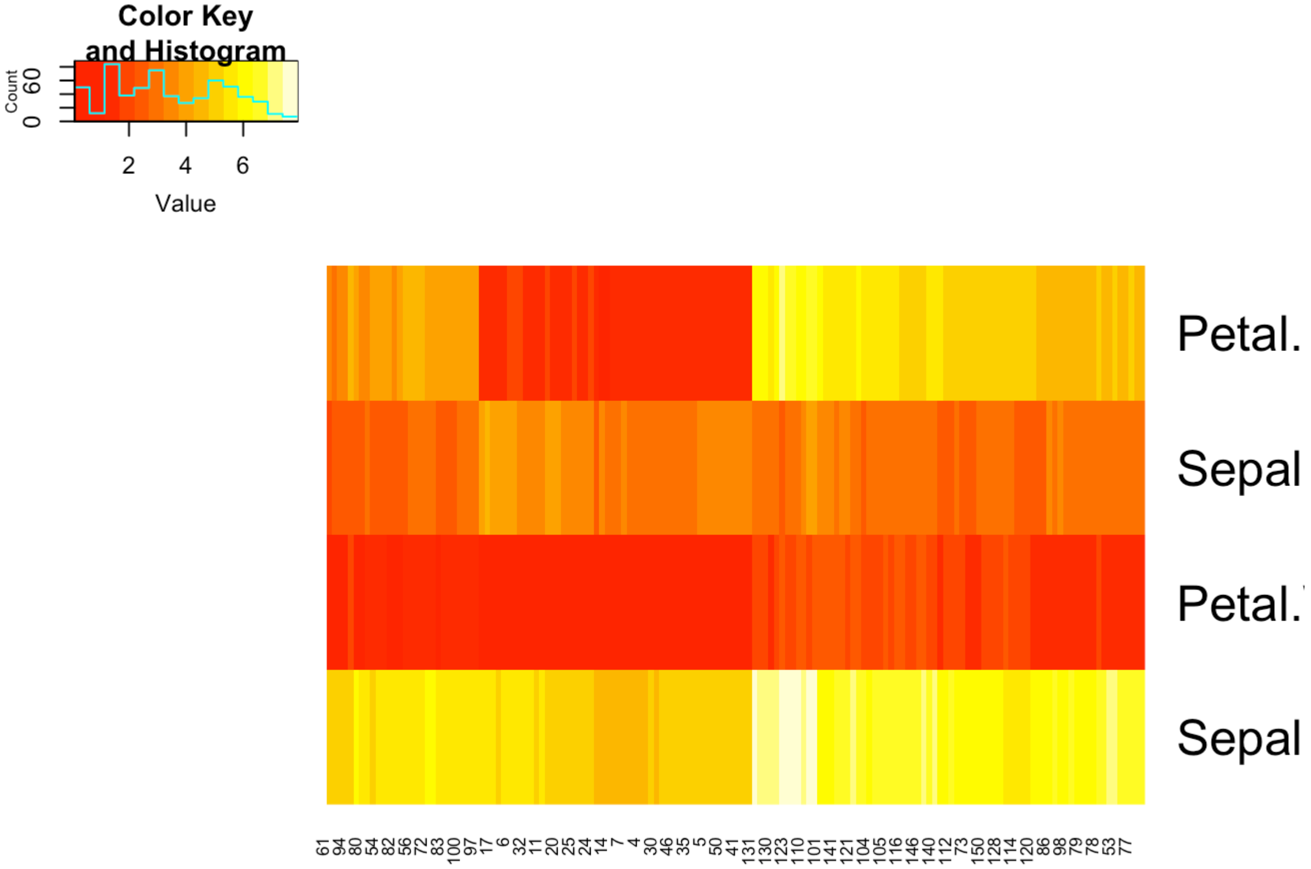


```
# heatmap

library(gplots)
```

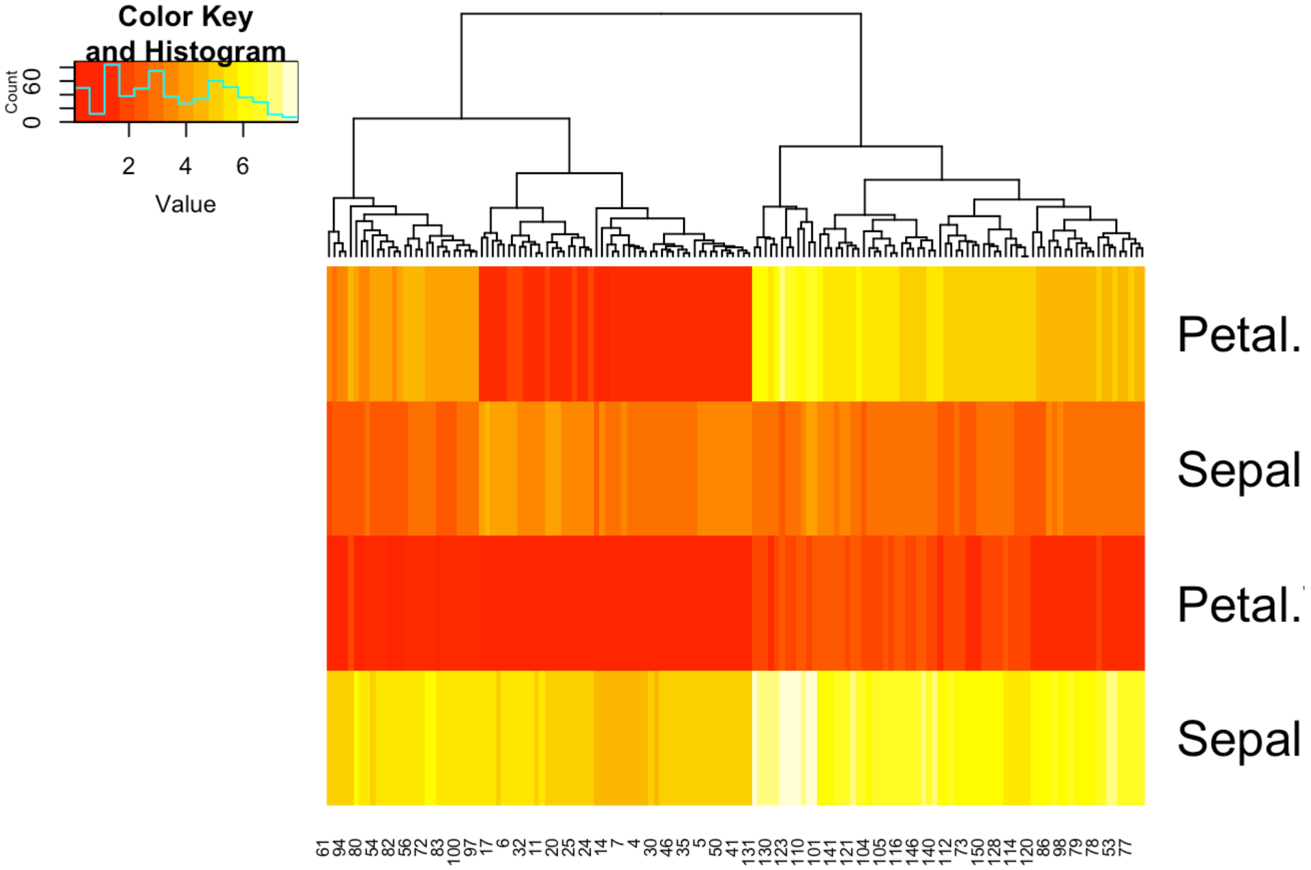
```
##
## 載入套件：'gplots'
```

```
## 下列物件被遮斷自 'package:stats':
##
##     lowess
```

```
heatmap.2(as.matrix(t(iris[,1:4])),dendrogram ="none",trace="none")
```



```
heatmap.2(as.matrix(t(iris[,1:4])),dendrogram ="column",trace="none")
```



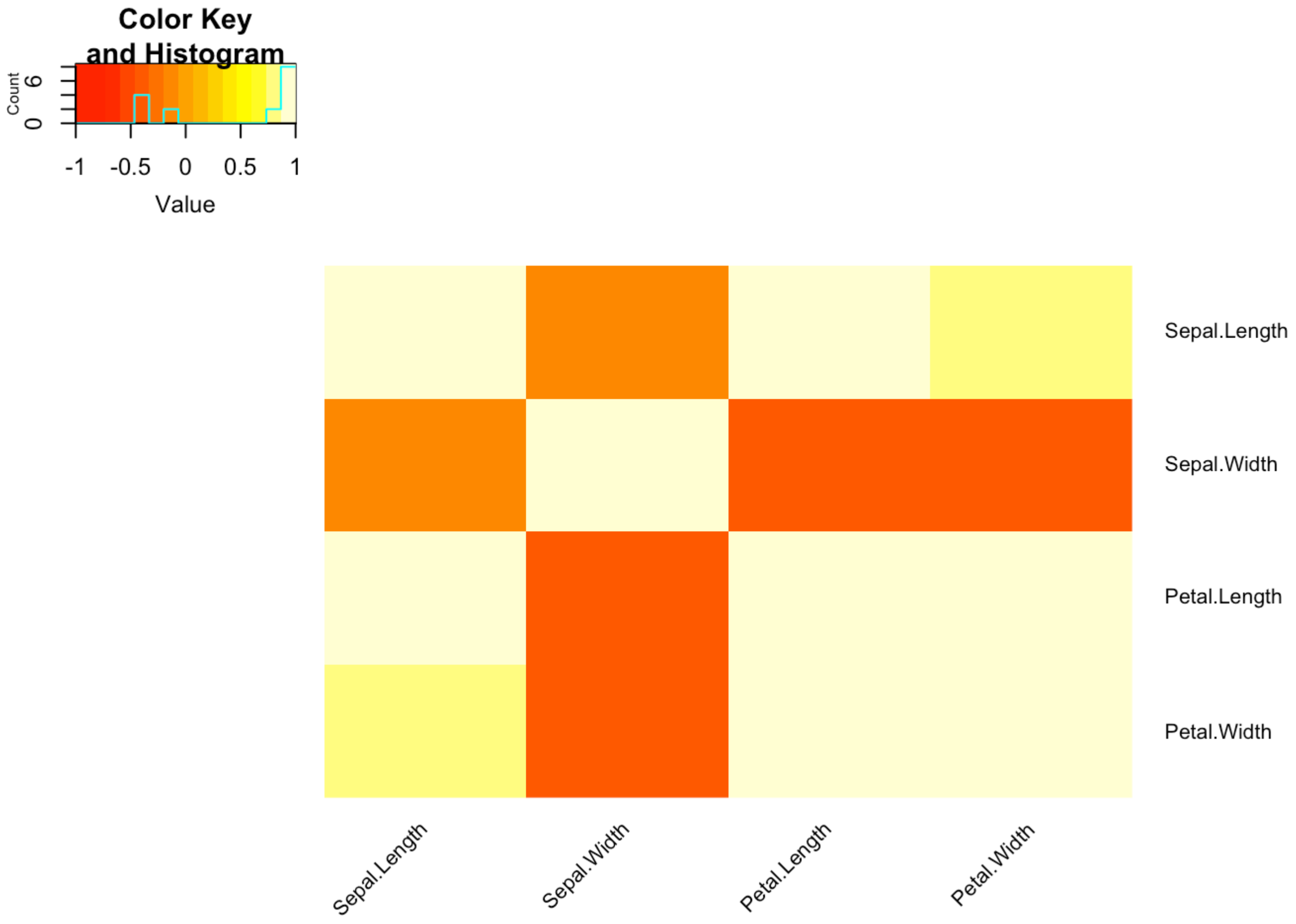
```
symnum(corr)
```

```
##                S.L S.W P.L P.W
## Sepal.Length 1
## Sepal.Width      1
## Petal.Length +   .   1
## Petal.Width  +   .   B   1
## attr(,"legend")
## [1] 0 ' ' 0.3 '.' 0.6 ',' 0.8 '+' 0.9 '*' 0.95 'B' 1
```

```
heatmap.2(corr, Rowv=FALSE, symm=TRUE, trace="none" ,cexRow=0.8, cexCol=0.8,srtCol=45,srtRow
=0)
```

```
## Warning in heatmap.2(corr, Rowv = FALSE, symm = TRUE, trace = "none", cexRow
## = 0.8, : Discrepancy: Rowv is FALSE, while dendrogram is `both'. Omitting row
## dendogram.
```

```
## Warning in heatmap.2(corr, Rowv = FALSE, symm = TRUE, trace = "none", cexRow =
## 0.8, : Discrepancy: Colv is FALSE, while dendrogram is `column'. Omitting column
## dendogram.
```



互動的ggplot

ploty

```
install.packages("plotly")
install.pacakges("tidyverser")
library(plotly)
library(tidyverse)
plot_ly()
```

EDA Example

輸入資料 `salesdata.csv`

```
saledata <- read.csv("salesdata.csv", sep=",")
head(saledata)
```

	Store	Product	Client	UnitPrice	Quantity	Region
	<chr>	<int>	<int>	<int>	<int>	<chr>
1	A	101	1	4	20	Taiwan
2	A	102	1	5	4	Taiwan
3	A	103	1	6	22	Taiwan
4	B	104	1	7	66	Taiwan
5	A	101	2	4	44	USA
6	A	102	2	5	3	USA

6 rows

```
str(saledata)
```

```
## 'data.frame':    39 obs. of  6 variables:
##  $ Store    : chr  "A" "A" "A" "B" ...
##  $ Product  : int  101 102 103 104 101 102 103 104 105 106 ...
##  $ Client   : int  1 1 1 1 2 2 2 2 2 2 ...
##  $ UnitPrice: int  4 5 6 7 4 5 6 7 8 9 ...
##  $ Quantity : int  20 4 22 66 44 3 8 4 6 10 ...
##  $ Region   : chr  "Taiwan" "Taiwan" "Taiwan" "Taiwan" ...
```

變數：

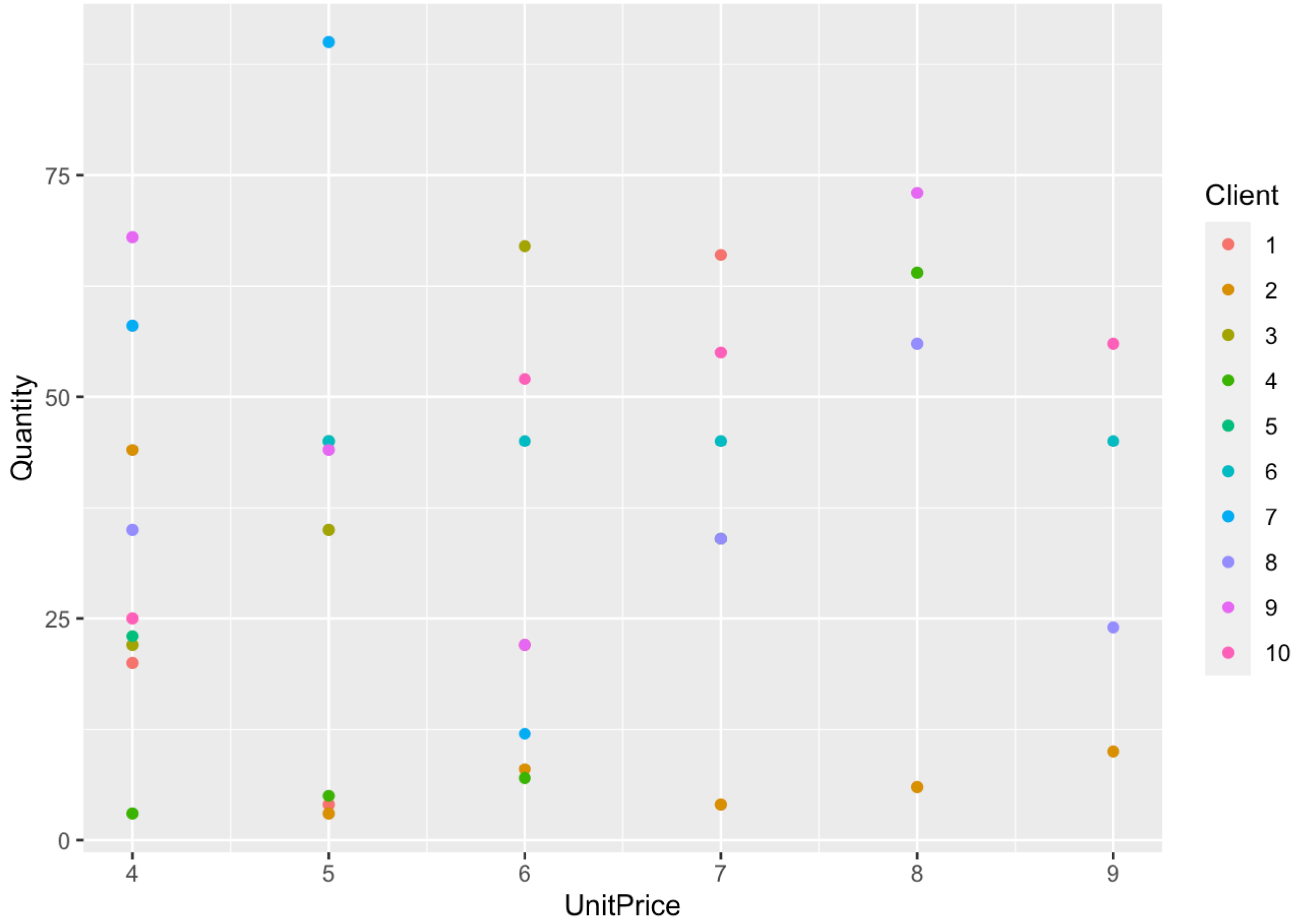
- Store: A、B，2種通路
- Product: 101~106，6種商品
- Client: 1~10，10個顧客
- UnitPrice: 物品單價，單位1000
- Quantity：購買數量

- Region: 10個國家

```
saledata$Product <-as.factor(saledata$Product)
saledata$Client <-as.factor(saledata$Client)
```

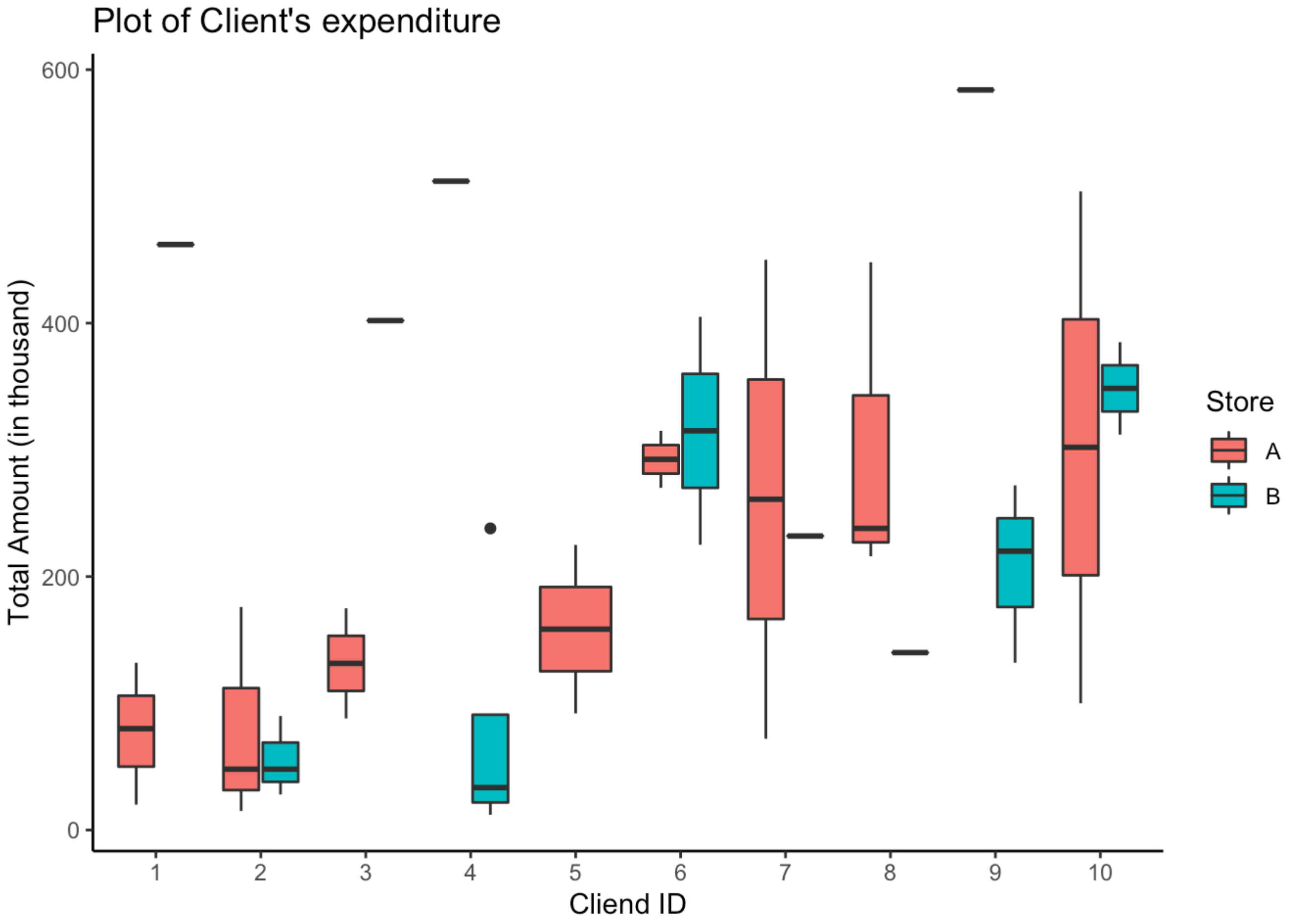
單價和銷售？

```
library(tidyverse)
saledata %>%
  ggplot(aes(x=UnitPrice, y=Quantity, color=Client))+
  geom_point()
```

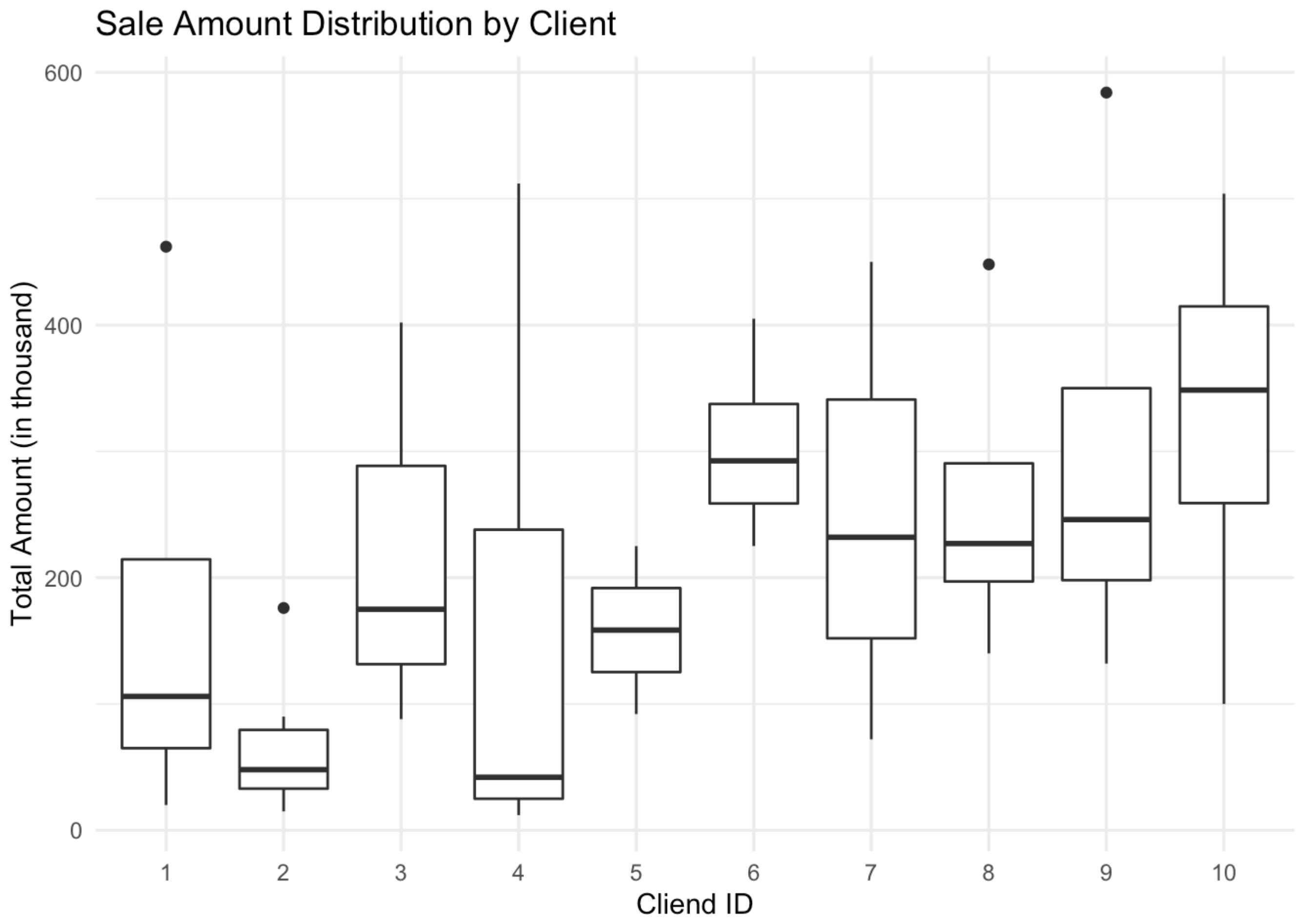


每個顧客在不同通路消費概況？每個顧客每次消費狀況？

```
saledata = saledata %>%
  mutate(Spend = UnitPrice*Quantity)

saledata %>%
  ggplot(aes(x=Client, y=Spend, fill=Store)) +
  geom_boxplot() +
  labs(title="Plot of Client's expenditure",x="Cliend ID", y = "Total Amount (in thousand)")+
  theme_classic()
```

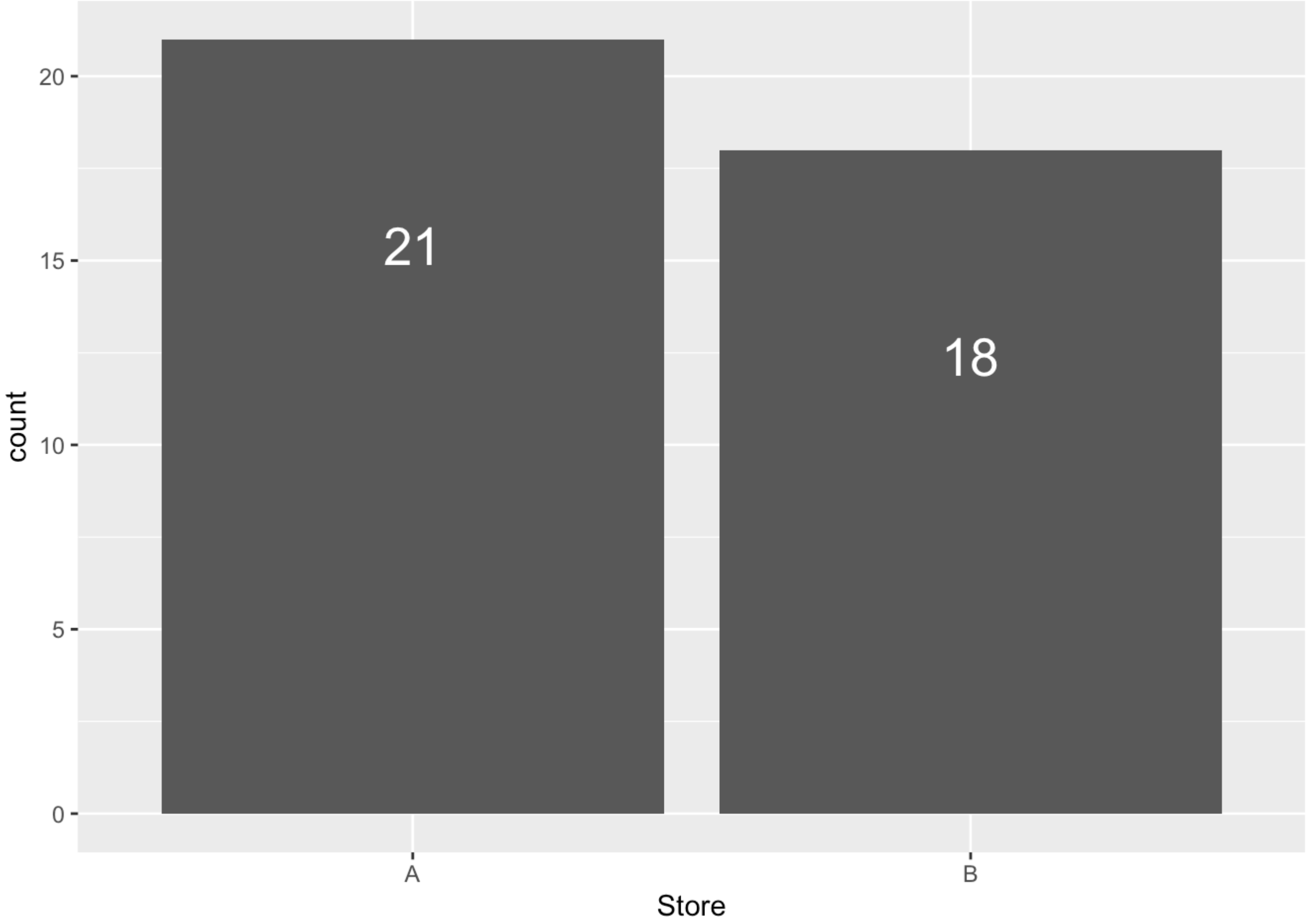


```
saledata %>%
  ggplot(aes(x=Client, y=Spend)) +
  geom_boxplot() +
  labs(title="Sale Amount Distribution by Client",x="Cliend ID", y = "Total Amount (in thousa
nd)")+
  theme_minimal()
```



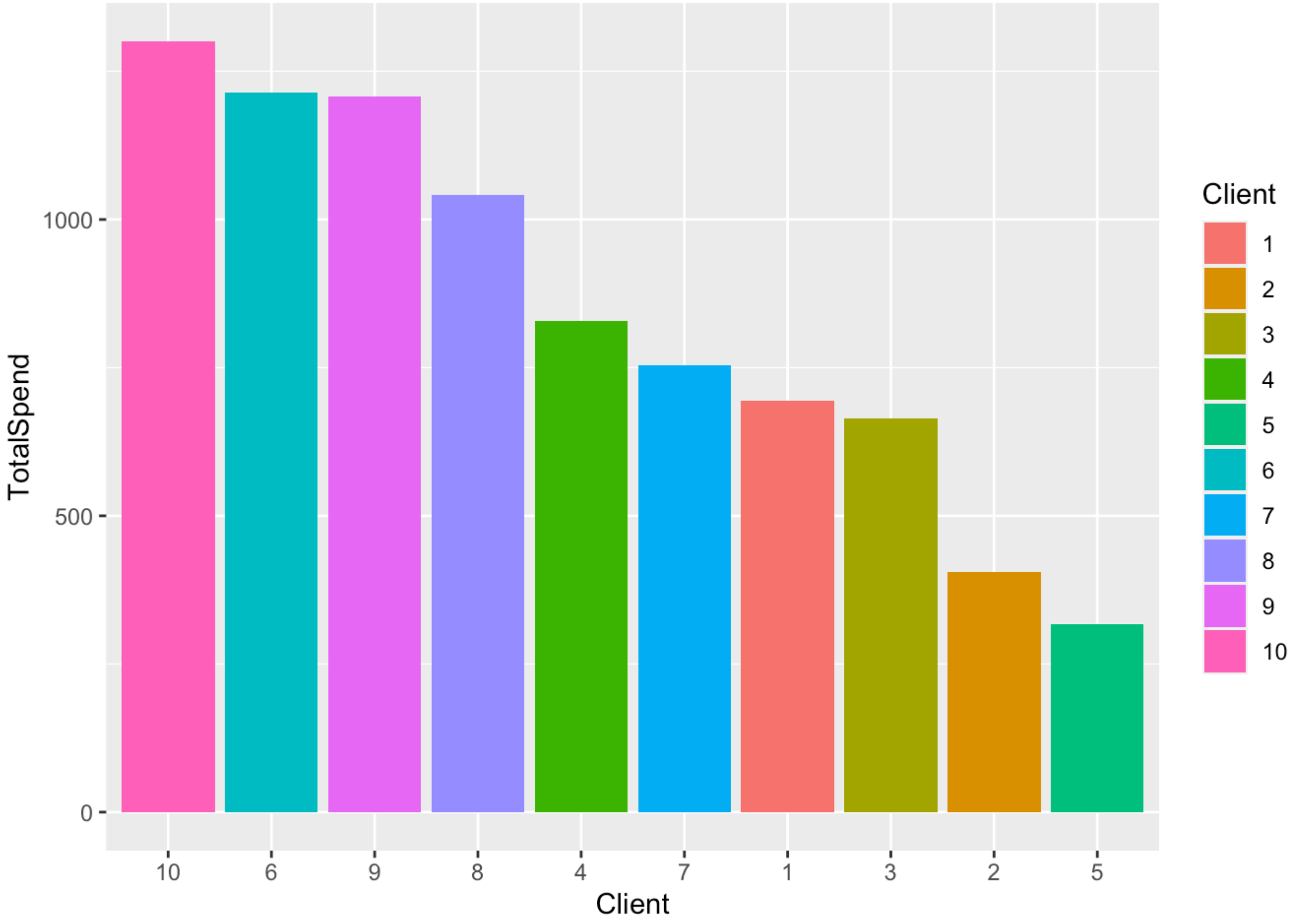
通路

```
ggplot(saledata, aes(x=Store)) +
  geom_bar() +
  geom_text(stat="count",aes(label=..count..),vjust=6, color=I("white"),size=7)
```



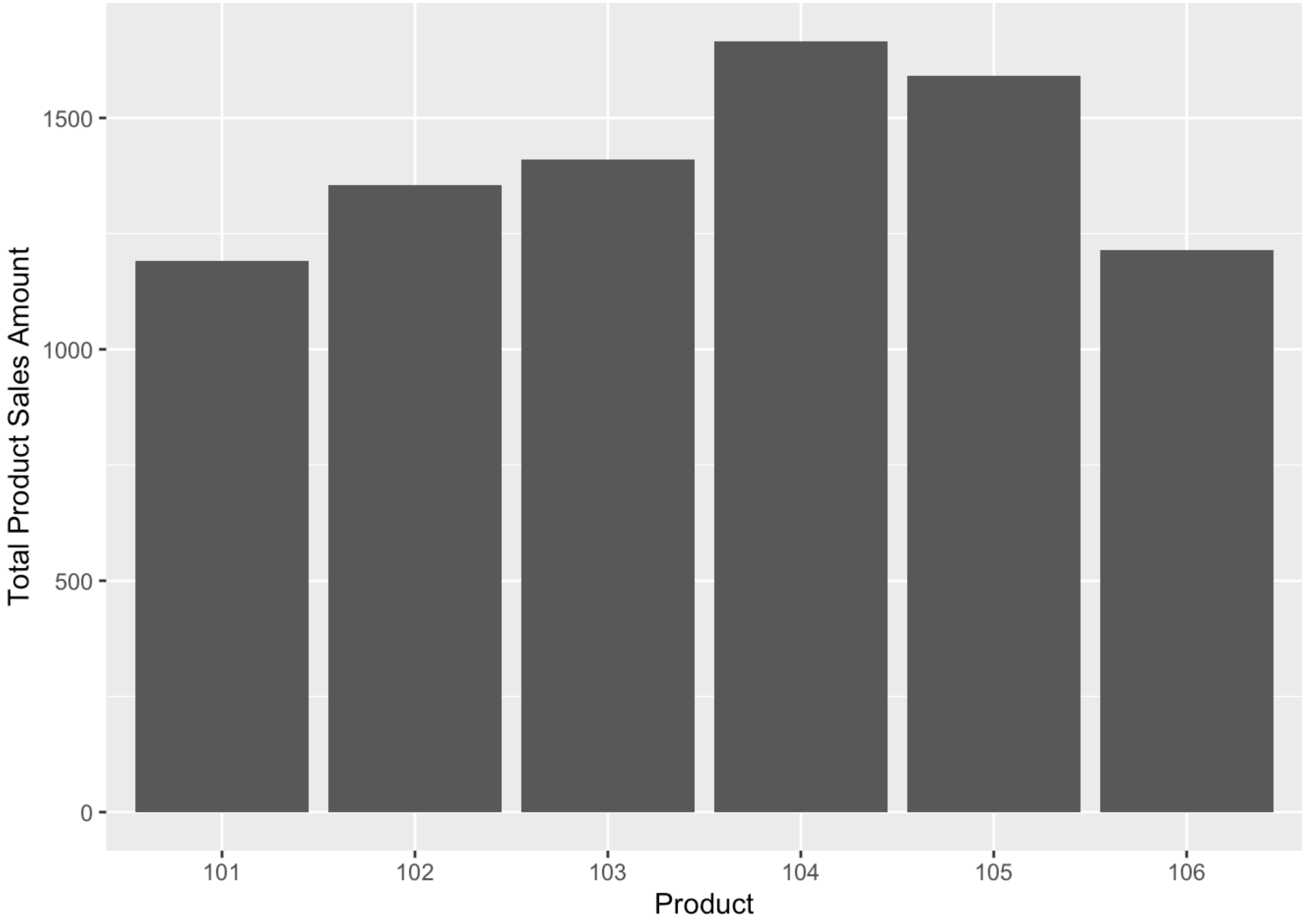
顧客消費能力

```
TotalSales <- saledata %>%
  group_by( Client) %>%  #根據顧客
  summarise( TotalSpend = sum(Spend)) %>% #把Spend加總
  arrange(desc(TotalSpend)) #排大到小

TotalSales %>%
  ggplot(aes(x=Client, y=TotalSpend, fill=Client)) +
    geom_bar( stat = 'identity') +
    scale_x_discrete(limits = TotalSales$Client)
```



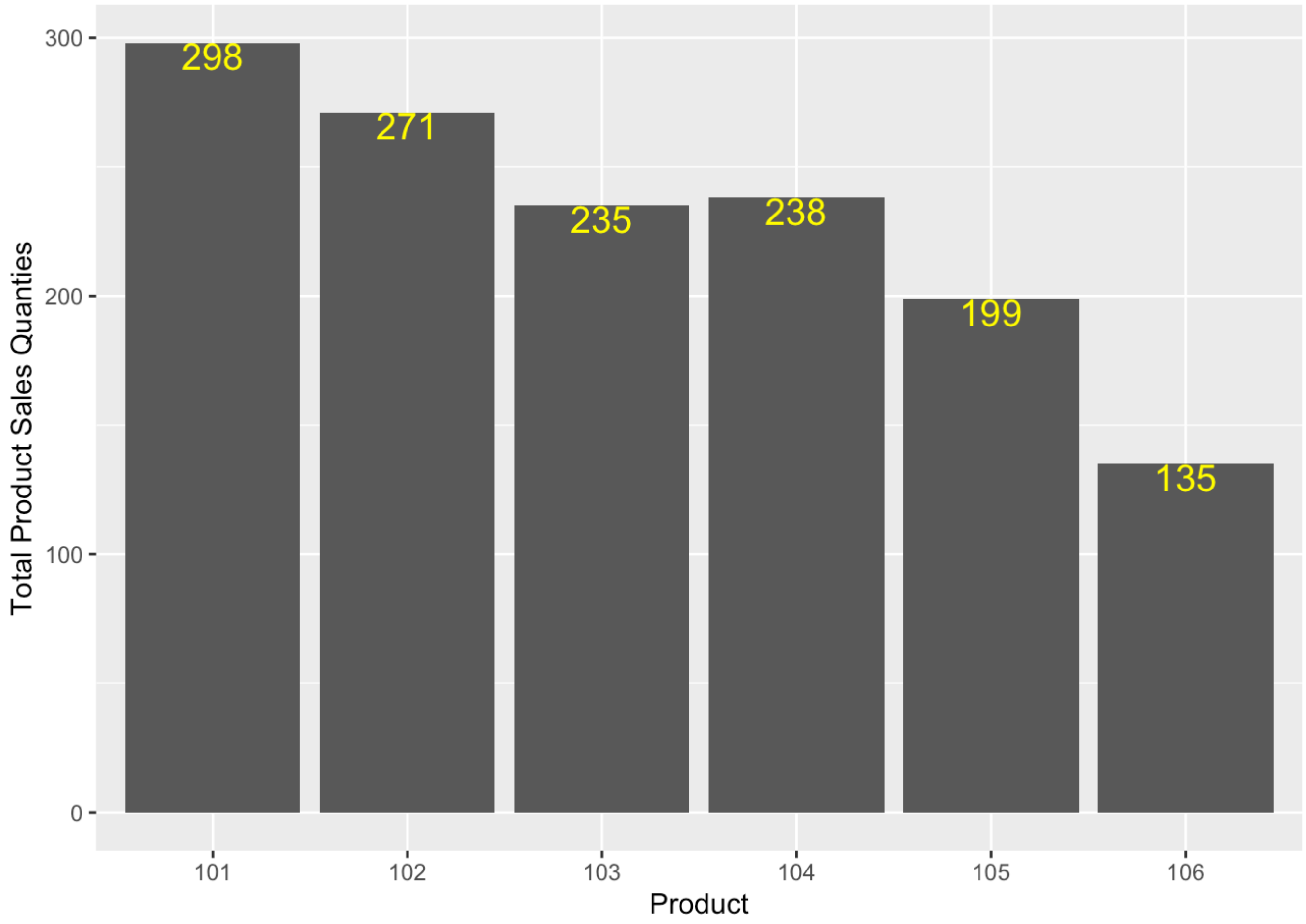
各產品銷售概況？

```
ProductSale <- saledata %>%
  group_by( Product) %>%
  summarise( TotalPSale = sum(Spend))

ProductSale %>%
  ggplot(aes(x= Product, y=TotalPSale)) +
  geom_bar(stat="identity") +
  labs(y="Total Product Sales Amount")
```



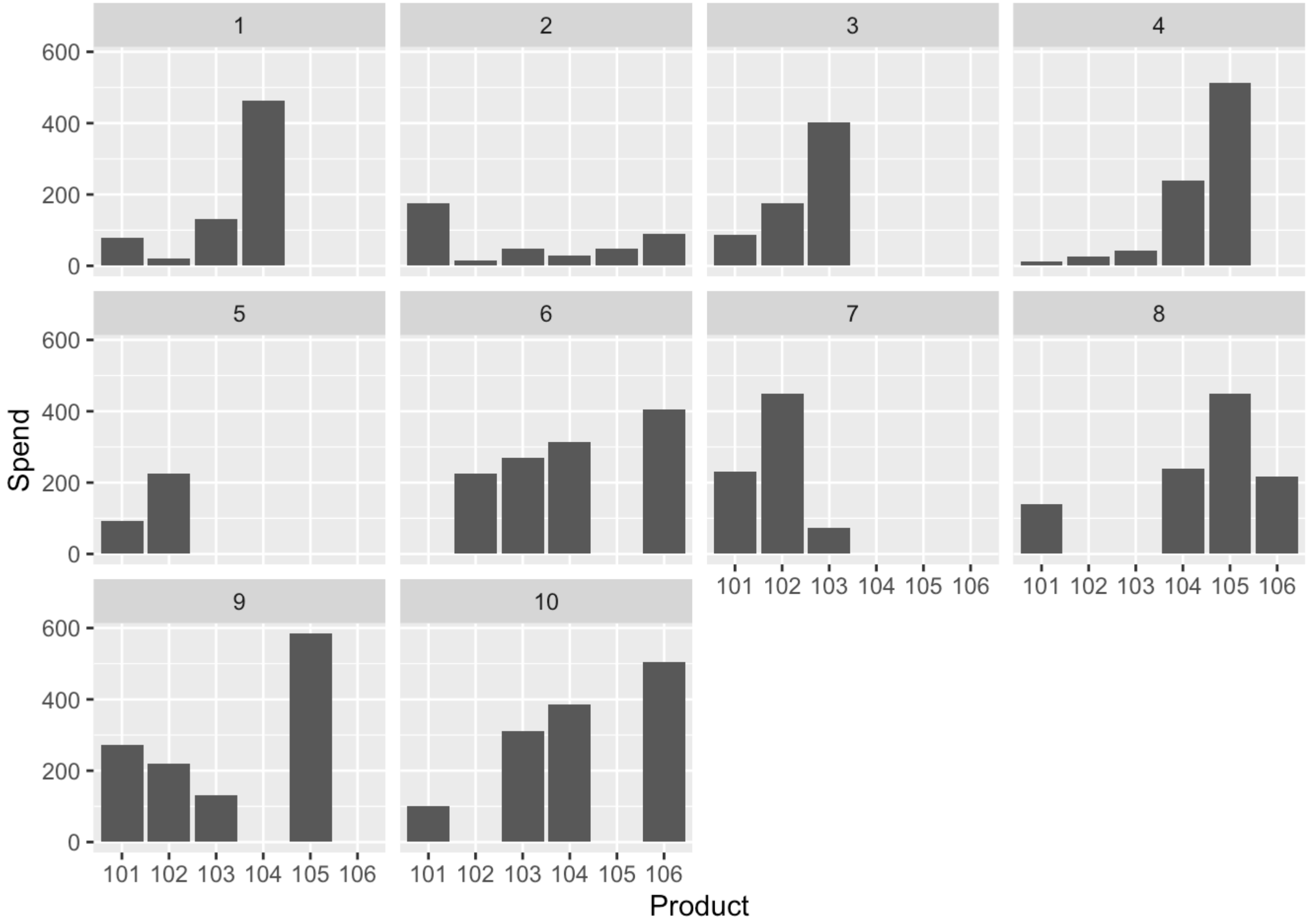
各產品總銷售？

```
ProductSale <- saledata %>%
  group_by( Product) %>%
  summarise( TotalQuan = sum(Quantity))

ProductSale %>%
  ggplot(aes(x= Product, y=TotalQuan)) +
  geom_bar(stat="identity") +
  labs(y="Total Product Sales Quanties") +
  geom_text(stat="identity",aes(label=TotalQuan),vjust=1, color=I("yellow"),size=5)
```



每個顧客購買能力？買什麼？

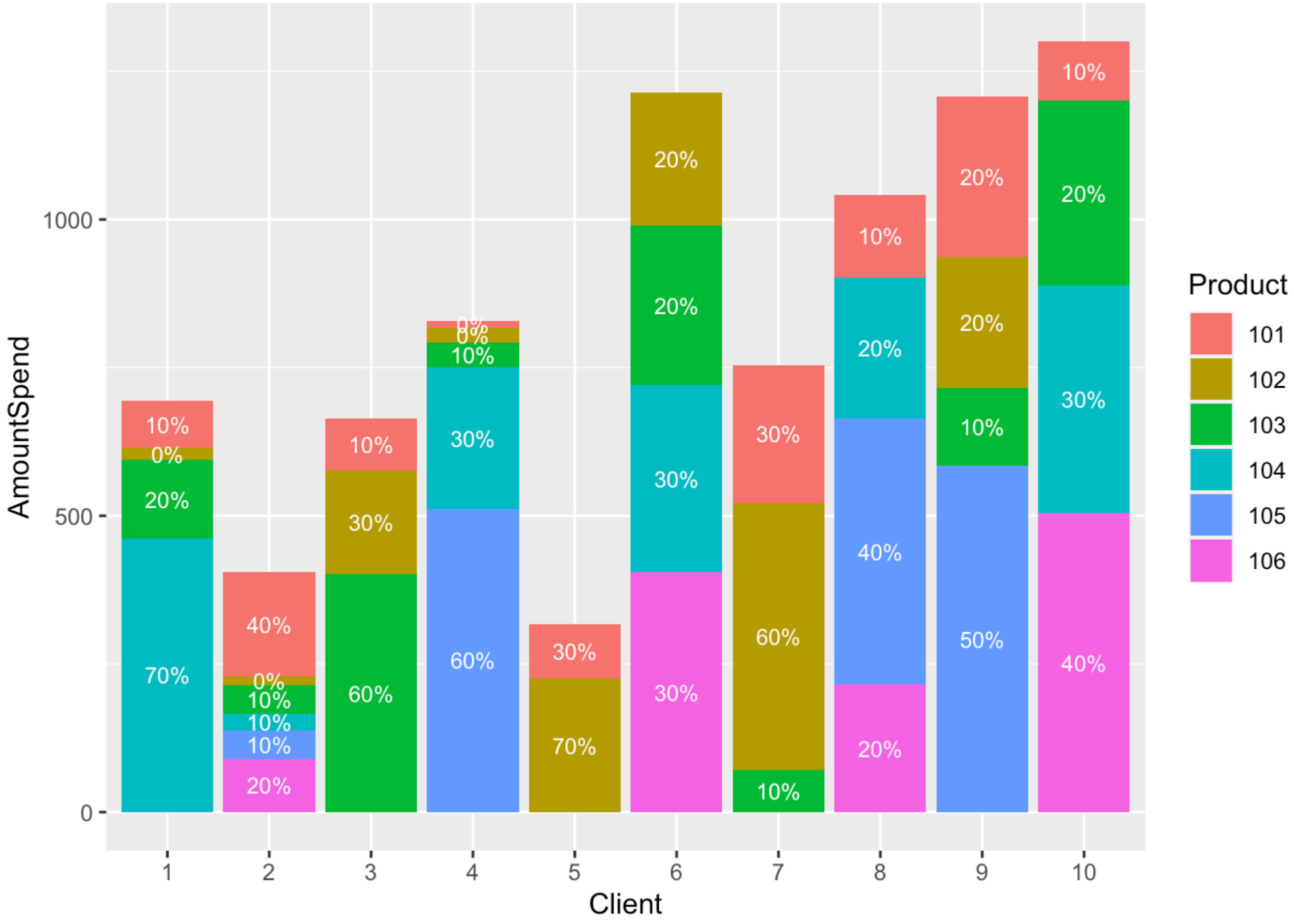
```
ggplot( data = saledata) +
  geom_bar( aes( x = Product,
                 y = Spend),
            stat = 'identity') +
  facet_wrap( ~ Client)
```



```
Product <- saledata %>%
  group_by(Client, Product) %>%
  summarise(AmountSpend = sum(Spend)) %>%
  mutate( Proportion = round(AmountSpend / sum(AmountSpend),1)*100)
```

```
## `summarise()` has grouped output by 'Client'. You can override using the `.groups` argumen
t.
```

```
ggplot( data = Product, aes( x = Client, y = AmountSpend, fill = Product)) +  geom_bar(stat
="identity") +
 geom_text(aes( x = Client, y = AmountSpend,  label= paste(Proportion, '%', sep='')), positio
n = position_stack(vjust = 0.5), color = I("white"), size = 3)
```



Ref: https://ggplot2.tidyverse.org/reference/position_stack.html (https://ggplot2.tidyverse.org/reference/position_stack.html)