ภาคผนวก ข.2

“คู่มือการใช้โปรแกรม R การใช้คำสั่งหรือฟังก์ชันพื้นฐาน”

เป็นส่วนหนึ่งของโครงการ

การพัฒนาการใช้โปรแกรม R ในการวิเคราะห์ข้อมูล

R Program Development for Research Utilization

โดย

หัชชา ศรีปลั่ง และคณะ

เดือน ปี ที่เสร็จโครงการตามสัญญา

ตุลาคม 2549

ต้นฉบับคู่มือ

“การใช้โปรแกรม R การใช้คำสั่งหรือฟังก์ชันพื้นฐาน”

ผู้แต่ง

นราทิพย์ จั่นสกุล

กัลยา นิติเรืองจรัส

คำนำ

มหาวิทยาลัยเป็นสถาบันทางการศึกษาระดับสูง ควรเป็นแบบอย่างที่ดีแก่สังคมในการใช้โปรแกรมทางคอมพิวเตอร์ที่ถูกกฎหมาย แต่ด้วยปัญหาที่โปรแกรมสำเร็จรูปทางสถิติที่ถูกต้องตามกฎหมายมีราคาค่อนข้างแพงและผู้ใช้มีงบประมาณจำกัด ดังนั้นทางเลือกหนึ่งคือการหันไปใช้โปแกรมที่มีการแจกจ่ายให้ฟรีโดยไม่ถือว่าเป็นการละเมิดลิขสิทธิ์ ปัจจุบันได้มีองค์กรต่าง ๆ ทั่วโลกร่วมมือกันสร้างโปรแกรมสำหรับการวิเคราะห์ข้อมูลทางสถิติ ให้เป็นทางเลือกเพื่อแก้ปัญหาดังกล่าว โปรแกรม **R** เป็นโปรแกรมทางเลือกหนึ่งที่ใช้ในการวิเคราะห์ข้อมูลทางสถิติที่สามารถนำมาใช้ได้ฟรีโดยไม่ถือว่าเป็นการละเมิดลิขสิทธิ์ ผู้ใช้สามารถ download ได้จาก web site ของโปรแกรม **R** www.r-project.org ได้โดยตรง แต่โปรแกรม **R** มีวิธีใช้ค่อนข้างใช้ยาก จึงไม่ได้รับความนิยมเท่าที่ควร ทั้งๆที่ประสิทธิภาพในการทำงานไม่ได้ด้อยไปกว่าโปรแกรมทางการค้าทั่วไป อีกทั้งในบางกรณี **R** ยังเป็นโปรแกรมที่มีประสิทธิภาพดีกว่าโปรแกรมอื่นๆ ด้วยซ้ำ

หน่วยระบาดวิทยา คณะแพทยศาสตร์ และสาขาวิชาสถิติ ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ ได้เล็งเห็นถึงปัญหาเหล่านี้ จึงได้พัฒนาโปรแกรม **R** ให้ง่ายต่อการใช้งาน สร้างคู่มือการใช้โปรแกรม **R** เป็นภาษาไทย และส่งเสริมให้คนได้รู้จักและหันมาใช้โปรแกรมนี้ในการวิเคราะห์ข้อมูลเพิ่มขึ้น คู่มือการใช้โปรแกรม **R** สำหรับการวิเคราะห์ข้อมูลสถิติฉบับนี้ ได้รวบรวมคำสั่งหรือฟังก์ชันพื้นฐานสำหรับการวิเคราะห์ข้อมูลพื้นฐานที่มีอยู่ในโปรแกรม **R** และที่คณะผู้วิจัยสร้างขึ้นใหม่ รวมทั้งวิธีการใช้ โดยยกตัวอย่างที่เป็นข้อมูลจริงซึ่งเหมาะกับนักวิจัยที่ทำการวิเคราะห์ข้อมูลด้วยสถิติพื้นฐานและอาจารย์ผู้สอนวิชาสถิติพื้นฐานในระดับปริญญาตรี เพื่อความสะดวกของผู้ใช้ คณะผู้วิจัยได้เตรียมโปรแกรม **R รุ่น 2.3.1** ซึ่งใช้ในการเตรียมต้นฉบับ รวบรวมคำสั่งที่สร้างขึ้นใหม่ไว้เป็น text file ชื่อ StatCan.txt และไฟล์ข้อมูลที่ใช้สาธิตไว้ใน CD ที่แนบมาพร้อมกับคู่มือเล่มนี้

คณะผู้วิจัยขอขอบคุณ เครือข่ายวิจัยสุขภาพ สำนักงานกองทุนสนับสนุนงานวิจัย (สกว) และมูลนิธิสาธารณสุขแห่งชาติ (มสช) ที่สนับสนุนทุนวิจัย ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ และหน่วยระบาดวิทยา คณะแพทยศาสตร์ มหาวิทยาลัยสงขลานครินทร์ ที่ให้ความอนุเคราะห์ในด้านเครื่องมือ และสถานที่ในการทำวิจัย คณะผู้วิจัยหวังเป็นอย่างยิ่งว่าคู่มือการใช้โปรแกรม **R** เล่มนี้จะเป็นประโยชน์สำหรับนักวิจัย อาจารย์และผู้อ่านทุกท่าน

คณะผู้วิจัย

กันยายน 2549

สารบัญ

สารบัญ (ต่อ)

สารบัญ (ต่อ)

บทที่ 1

โปรแกรม R (R Program)

1.1 แนะนำโปรแกรม R

โปรแกรมสำเร็จรูป **R** เป็น Open Source Software ที่นำเสนอให้ฟรี สามารถดัดแปลงได้ตามต้องการอย่างอิสระ แจกจ่ายได้ แต่ห้ามจำหน่ายโดยหวังผลกำไร ผู้เริ่มต้นเขียนโปแกรมนี้ คือ Robert Gentleman และ Ross Ihaka จากภาควิชาสถิติ มหาวิทยาลัยโอ๊คแลนด์ ประเทศนิวซีแลนด์ จากนั้นตั้งแต่กลางปี พ.ศ. 2540 เป็นต้นมาจึงเกิดเป็นทีมผู้พัฒนาโปรแกรม **R** (The **R** Development Core Team) ซึ่งกลุ่มผู้พัฒนาโปรแกรมนี้เป็นกลุ่มเดียวกับผู้พัฒนาโปรแกรมสำเร็จรูป **S Plus** ที่ได้รับการยกย่องว่าเป็นโปรแกรมดีเด่น มีประสิทธิภาพในการคำนวณทางสถิติสูง แต่มีลิขสิทธ์และราคาแพง โปรแกรม **R** มีมูลนิธิ (Software foundation) เป็นผู้สนับสนุนด้านวิชาการทั้งจากสหรัฐอเมริกา, ออสเตรเลีย และญี่ปุ่น เป็นต้น โปรแกรมนี้จึงเป็นทางเลือกหนึ่งที่สามารถนำมาพัฒนาเพื่อใช้ในด้านการเรียนการสอนได้ นอกเหนือจากการเป็นซอฟต์แวร์ฟรี จึงเป็นโอกาสอันดีสำหรับนักวิชาการในประเทศไทยที่จะนำไปใช้ในด้านต่าง ๆ เช่น **R** ช่วยในการเรียนการสอนด้านสถิติให้เข้าใจเนื้อหาทางทฤษฎีได้ลึกซึ้ง เปิดโอกาสในนักวิจัยสามารถเขียนฟังก์ชันและกราฟใหม่ ๆ สำหรับใช้งานเฉพาะด้านของตน และที่สำคัญคือสามารถเข้าร่วมเรียนรู้กับนานาชาติ โดยการร่วมเขียน module ตามโจทย์และศักยภาพที่เรามีอยู่ วิสัยทัศน์ของการพัฒนาและพึ่งตนเองพร้อม ๆ กับประสานงานกับกลุ่มนานาชาติในการทำงานทางวิชาการที่ไม่ผูกติดกับผลประโยชน์ทางการค้า เพื่อพัฒนาสปิริตการสร้างสมบัติวิชาการที่เป็นสาธารณะของโลก ซึ่งประเทศไทยยังขาดการเรียนรู้ด้านนี้มาก จึงเป็นสิ่งที่ควรพัฒนาขึ้น

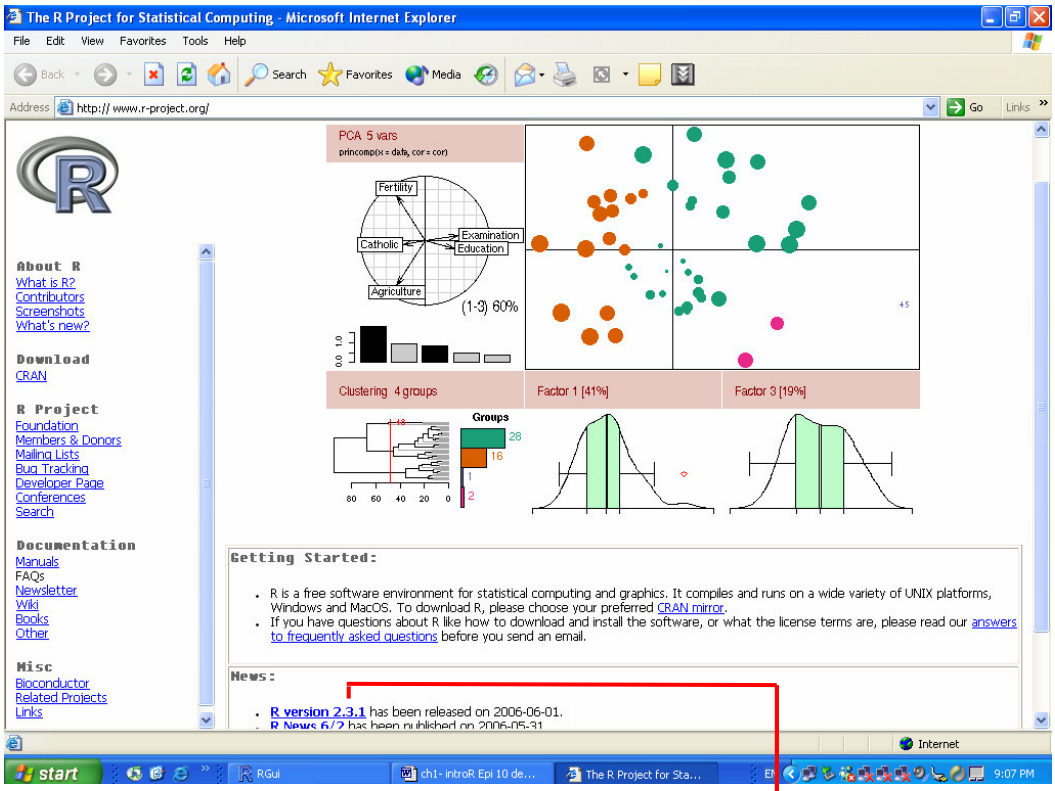
1.2 เหตุผลในการใช้โปรแกรม R

การประกาศนโยบายรับรองทรัพย์สินทางปัญญาเป็นนโยบายหนึ่งของรัฐบาล โดยเฉพาะอย่างยิ่งซอฟต์แวร์ทางคอมพิวเตอร์ โดยมีประกาศให้หน่วยราชการต่าง ๆ ใช้ซอฟต์แวร์ที่ถูกกฎหมายเท่านั้น อย่างไรก็ตาม ในทางปฏิบัติ หน่วยราชการต่าง ๆ ให้มีงบประมาณในการจัดซื้อซอฟต์แวร์จำกัด หากจัดซื้อจริงคาดว่าค่าใช้จ่ายต่างๆ จะสูงขึ้นอย่างมาก โปรแกรมทางคอมพิวเตอร์ที่ใช้ใน

การวิเคราะห์ข้อมูลทางสถิติส่วนใหญ่เป็นโปรแกรมที่มีลิขสิทธิ์ ในปัจจุบันการวิเคราะห์ข้อมูลทางสถิติ ต้องอาศัยโปรแกรมสำเร็จรูป (software) ที่สามารถรองรับข้อมูลที่มีขนาดใหญ่ และระเบียบวิธีทางสถิติที่สลับซับซ้อน อย่างไรก็ตามซอฟต์แวร์ที่มีประสิทธิภาพดังกล่าว ถูกผลิตขึ้นเพื่อการค้า และมีราคาแพง หน่วยงานต่าง ๆ จึงใช้ซอฟต์แวร์อย่างผิดกฎหมาย มหาวิทยาลัยเป็นสถาบันการเรียนการสอน ควรเป็นแบบอย่างที่ดีแก่สังคมภายนอกในการใช้ซอฟต์แวร์อย่างถูกกฎหมาย ดังนั้นการพัฒนาการใช้ซอฟต์แวร์ที่ไม่ใช่เพื่อการค้าสำหรับวิเคราะห์ข้อมูล จึงเป็นสิ่งจำเป็นในการส่งเสริมให้มีการใช้อย่างแพร่หลายขึ้น เพื่อให้หลุดพ้นจากปัญหาเชิงการค้าให้ได้มากที่สุด

1.3 การ download โปรแกรม R

โปรแกรม **R** สามารถเข้าไป download ได้ที่ web site ของ **R** โดยตรงคือ http:// www.r-project.org ดังรูป 1.1 ปัจจุบันโปรแกรม **R** ที่ download จะอยู่ใน version 2.3.1

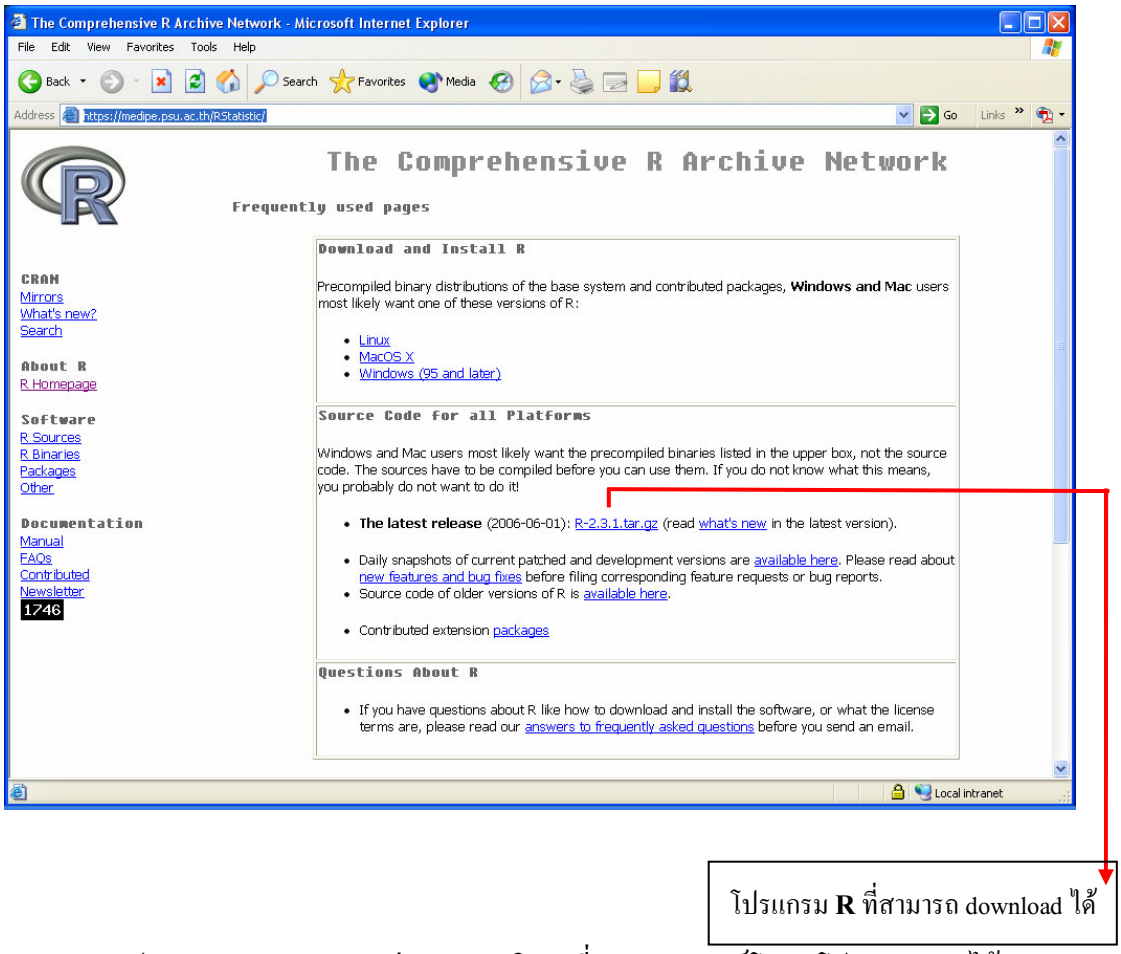


รูป 1.1 Web page **R**

โปรแกรม **R** ที่ให้ download ได้ฟรี

หมายเหตุ โปรแกรม **R** version 2.3.1 ออกเมื่อเดือนมิถุนายน 2549

นอกจากนี้ ผู้ใช้สามารถดาวน์โหลดโปรแกรม **R** จาก website ของหน่วยระบาดวิทยา คือ https://medipe.psu.ac.th/RStatistic/ ดังรูป 1.2



รูป 1.2 Web page ของหน่วยระบาดวิทยาที่สามารถดาวน์โหลดโปรแกรม **R** ได้

1.4 การติดตั้งโปรแกรม R

เมื่อ download โปรแกรมมาจาก web site ก็จะได้ไฟล์ชื่อ R-2.3.1-win32.exe ทำการติดตั้งโปรแกรม **R** ดังนี้

- Double click ที่ไฟล์ชื่อ R-2.3.1-win32.exe
- ทำตามขั้นตอน โดยการคลิก Next ต่อไปเรื่อย ๆ โปรแกรม ก็จะถูกติดตั้ง เมื่อติดตั้งเสร็จแล้ว ก็จะปรากฏ icon ของโปรแกรม ที่ desktop โดยเป็นสัญญักษณ์ตัวอักษร **R** ที่มีสีน้ำเงิน เมื่อดับเบิ้ลคลิกที่ไอคอน โปรแกรม **R** ก็จะถูกเปิดขึ้น

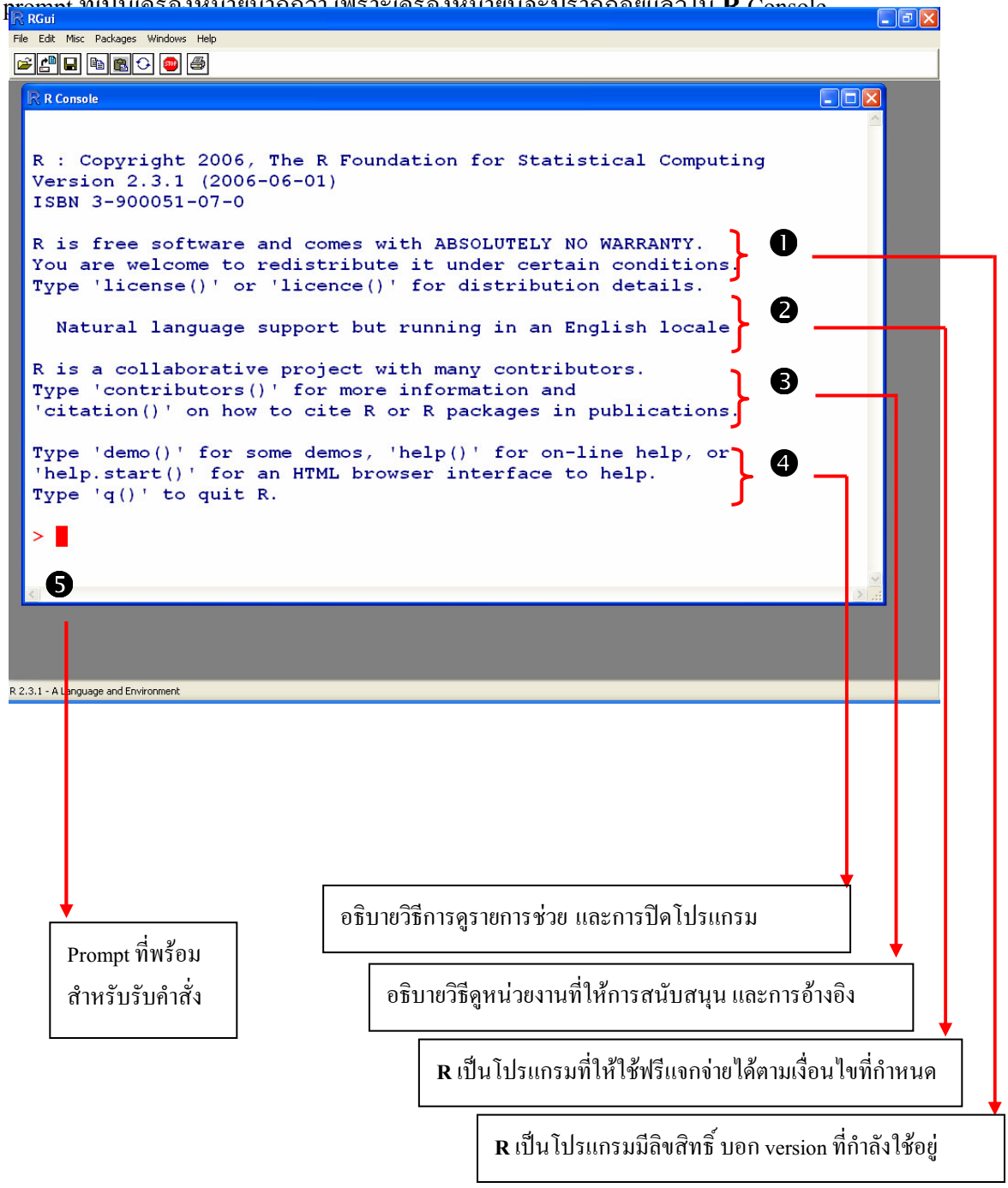
หลังจากการติดตั้ง โปรแกรม **R** จะถูกเก็บไว้ที่ C:\Program Files\R\R-2.3.1\ และ shortcut icon จะถูกสร้างไว้บน desktop โดยอัตโนมัติ แฟ้มข้อมูล ฟังก์ชัน ตัวแปร ผลลัพธ์ เป็นต้น ควรจะเก็บไว้ที่ working folder ที่ผู้ใช้ได้สร้างไว้ ซึ่งควรเก็บไว้ใน folder แยกต่างหากจะสะดวกกว่า เช่นเก็บไว้ที่ C:\RWorkplace โดยใช้ mouse วางที่ไอคอน **R** คลิ๊กที่ปุ่มขวาแล้วเลือก Properties จากนั้นพิมพ์คำว่า C:\RWorkplace ใน 'Start in' ของ Properties ดังรูป 1.3 การสร้าง working folder ควรจะสร้างโดยใช้ Windows Explorer สร้างขึ้นมาก่อน ก่อนที่จะกำหนด working folder ใน Start in ของ Properties ผู้ใช้สามารถสร้าง working folder ใหม่และใช้ชื่อตามที่ต้องการ ในที่นี้ใช้ชื่อ folder ว่า "RWorkplace"

รูป 1.3 การตั้งค่าใน Start in

1.5 การเริ่มทำงานกับโปรแกรม R

เมื่อเปิดโปรแกรม **R** ขึ้นมาโดยการดับเบิ้ลคลิกที่ไอคอน **R** ที่อยู่บน desktop ก็จะปรากฏ **R** Console Window ขึ้นดังรูป 1.4 พร้อมกับ prompt ที่พร้อมสำหรับรับคำสั่ง ซึ่ง prompt ที่สามารถป้อนคำสั่งใช้สัญลักษณ์ > (เป็นสัญลักษณ์เครื่องหมายมากกว่า) พร้อมกับแถบ cursor อยู่หลัง

prompt ในเอกสารเล่มนี้จะแสดง prompt พร้อมกับคำสั่ง หากผู้ใช้ต้องการใช้คำสั่ง ไม่ต้องพิมพ์ prompt ที่เป็นเครื่องหมายนำก่อน เพราะเครื่องหมายนี้จะปรากฏอยู่แล้วใน **R** Console



รูป 1.4 หน้าจอ Rgui (**R** Graphic User Interface)

เนื่องจากโปรแกรม **R** เป็นโปรแกรมที่ต้องป้อนคำสั่ง หากไม่ทราบอะไรเลย สิ่งหนึ่งที่จะช่วยได้คือการเรียกดูรายการช่วย ดังอธิบายไว้แล้วในส่วนที่ ❹ คำสั่งที่จะขอดูรายการช่วย คือ

```
> help.start()
```

หากต้องการออกจากโปรแกรม ก็จะมีบอกไว้แล้วในส่วนที่ ❹ เช่นกัน นั่นคือ พิมพ์คำว่า

```
> q()
```

โปรแกรมจะถามว่า Save work space image? ถ้าเลือก Yes โปรแกรมก็จะ save work space ทุกอย่างที่เราได้สร้างไว้ โดยให้นามสกุลของไฟล์เป็น ".Rdata" ควรจะเลือก Yes ก็ต่อเมื่อต้องการบันทึกงานค้างที่ยังทำไม่เสร็จและอยากกลับมาทำต่อเนื่องในทันทีที่เปิด **R** มาใช้งาน

1.6 คำและความหมายในโปรแกรม R

สำหรับโปรแกรม **R** แล้ว จะมีคำต่าง ๆ ที่ผู้ใช้ไม่คุ้นเคยนัก แต่เพื่อให้ใช้โปรแกรมได้อย่างมีประสิทธิภาพ และมีความเข้าใจในโปรแกรม คำต่าง ๆ ที่ผู้ใช้จำเป็นต้องทราบมีดังนี้

1.6.1 วัตถุ (Object)

เป็นสิ่งที่ **R** สร้างขึ้น จากคำสั่งที่ผู้ใช้ได้พิมพ์ไว้ และถูกเก็บไว้ในหน่วยความจำ โปรแกรม **R** เป็นโปรแกรมชนิด object oriented ทุกสิ่งใน **R** จะมีสภาพเป็นวัตถุ มีคุณสมบัติที่สำคัญ คือ

1. มีที่อยู่ที่อ้างอิงได้ในหน่วยความจำ
2. การอ้างอิงทำได้โดยการ เรียกชื่อวัตถุนั้น และผู้ใช้สามารถสร้างและทำลายวัตถุในหน่วยความจำได้
3. วัตถุไม่ว่าจะเป็นชนิดใด เมื่อเรียกด้วยชื่อ จะมีระดับการอ้างถึงเท่ากันหมด

นอกจากนี้ยังมีคุณสมบัติอีกหลายประการที่จะไม่กล่าวถึงในที่นี้ ผลที่ได้ตามมาจากคุณสมบัติเหล่านี้คือ วัตถุอาจเป็นอะไรก็ได้ เช่น ตัวแปร ตัวเลข เมื่อมีการตั้งชื่อของ เวกเตอร์, กรอบข้อมูล หรือฟังก์ชัน ซ้ำกัน วัตถุเดิมจะถูกทำลาย (ถูกปลดจากการอ้างอิง แล้วถูก garbage collector ทำลายในภายหลัง) ดังนั้น พึงระวังในการตั้งชื่อซ้ำกัน เพราะจะไม่มีการเตือน

ตัวอย่างคำสั่งในการสร้างวัตถุขึ้นมาใหม่ ดังต่อไปนี้

```
> x = 6 + 9        ❶

> x <- 6 + 9       ❷
```

คำอธิบาย

คำสั่ง ❶ และ ❷ มีความหมายเดียวกัน คือ การให้คำนวณ ค่า 6 บวกด้วย 9 และเก็บค่าที่ได้ไว้ในวัตถุที่ชื่อ x หากต้องการดูผลลัพธ์ จะต้องพิมพ์ชื่อวัตถุที่ได้สร้างขึ้น ดังตัวอย่าง

```
> x
[1] 15
```

[1] 15 → คือ ผลลัพธ์ที่ได้จากการคำนวณ

15 → คือ ผลลัพธ์ลำดับที่ 1 ของ x

1.6.2 เวกเตอร์ (Vector)

โครงสร้างข้อมูลพื้นฐานชนิดหนึ่งใน **R** คือ เวกเตอร์ตัวเลข ซึ่งเป็นรูปแบบอย่างง่ายที่สุด ประกอบด้วยลำดับของตัวเลข การสร้างเวกเตอร์จะต้องอาศัย ฟังก์ชัน **c()** ดังตัวอย่าง

```
> c(10, 20, 30, 40, 50)
[1] 10 20 30 40 50
```

คำอธิบาย

ตัวอย่างข้างต้นยังไม่ได้กำหนดชื่อวัตถุที่เก็บผลลัพธ์ จึงมีการแสดงผลลัพธ์ในบรรทัดถัดมา แล้ววัตถุในหน่วยความจำที่ไม่มีชื่อ จะไม่สามารถอ้างอิงเรียกใช้ภายหลังได้ และจะถูกทำลายโดยอัตโนมัติ หากต้องการเก็บผลลัพธ์ไว้ใช้ในอนาคต ควรกำหนดให้เก็บไว้ในวัตถุตามชื่อที่กำหนด

```
> x <- c(10, 20, 30, 40, 50)
> x
[1] 10 20 30 40 50
```
❶

```
> (x <- c(10, 20, 30, 40, 50))
  [1] 10 20 30 40 50
```
❷

คำอธิบาย

จากตัวอย่างคำสั่งที่ ❶ เป็นการกำหนดชื่อวัตถุที่เก็บผลลัพธ์เป็นชื่อ x เมื่อพิมพ์คำว่า x ผลลัพธ์ถูกแสดงในบรรทัดถัดมา คำสั่งที่ ❷ เป็นคำสั่งเช่นเดียวกับคำสั่งที่ ❶ แต่ใส่คำสั่งทั้งหมดไว้ในวงเล็บ โดยย่อมาจากคำสั่ง print(x <- c(10, 20, 30, 40, 50) เพื่อให้โปรแกรมแสดงผลลัพธ์ออกมาทันที โดยไม่ต้องพิมพ์ชื่อวัตถุ ดังคำสั่งที่ ❶

1.6.3 ลิสต์ (List)

ลิสต์เป็นวัตถุที่องค์ประกอบภายในเรียงกันเป็นลำดับ ซึ่งจะเรียกองค์ประกอบภายในนั้นว่า component ไม่มีความจำเป็นที่องค์ประกอบของลิสต์จะต้องเป็นชนิดเดียวกันทั้งหมด ลิสต์สามารถรวมเอาเวกเตอร์ตัวเลข, เมทริกซ์, เวกเตอร์ซ้อน, อาเรย์, ตัวอักษร, ฟังก์ชัน และอื่น ๆ เข้าไว้ด้วยกัน ตัวอย่างต่อไปนี้เป็นตัวอย่างอย่างง่ายสำหรับการสร้างลิสต์

```
> Lst <- list(name="Fred", wife="Mary", no.child=3, child.age=c(4,7,9))
```

```
> Lst
$name
[1] "Fred"
$wife
[1] "Mary"
$no.child
[1] 3
$child.age
[1] 4 7 9
```

แสดงผลลัพธ์ที่ได้จากการพิมพ์ชื่อวัตถุ Lst

คำอธิบาย

ทำการสร้างวัตถุชนิดลิสต์ขึ้นมา ให้ชื่อว่า Lst โดยภายในประกอบด้วยองค์ประกอบ ชื่อ name, wife, no.child และ child.age เมื่อพิมพ์เรียกชื่อวัตถุ Lst รายละเอียด ต่าง ๆ ภายในลิสต์จะปรากฏขึ้น โดยมีเครื่องหมาย $ หน้าชื่อตัวแปรเป็นการบอกให้ทราบว่า ชื่อที่อยู่หลังเครื่องหมายเป็นองค์ประกอบ (component) ที่อยู่ในลิสต์ องค์ประกอบเหล่านี้จะยังไม่เรียกว่าตัวแปร คำว่าตัวแปรจะใช้เมื่อเป็นองค์ประกอบในกรอบข้อมูล เพราะองค์ประกอบในลิสต์ อาจเป็นฟังก์ชัน หรือกรอบข้อมูล ก็ได้

แบบฝึกหัด

ลองทำตามคำสั่งดังต่อไปนี้ และค่าที่ได้จากคำสั่งเหล่านี้คืออะไร

```
> Lst$child.age[1]
```

```
> Lst$child.age[2]
```

1.6.4 กรอบข้อมูล (Data frame)

คือลิสต์ที่อยู่ในคลาส “data.frame” แต่มีข้อจำกัดคือ ลิสต์ที่จะทำเป็นกรอบข้อมูลได้ ต้องประกอบด้วย เวกเตอร์, เมทริกซ์ตัวเลข หรือ กรอบข้อมูล

กรอบข้อมูลใหม่จะได้ตัวแปรมาจากสดมภ์ของเมทริกซ์ องค์ประกอบของลิสต์หรือตัวแปรในกรอบข้อมูลที่นำมารวมกันนั้น จะต้องมีขนาดความยาวเดียวกัน และโครงสร้างของเมทริกซ์ จะต้องมีความยาวของแถวเท่ากัน วัตถุประสงค์ของข้อจำกัดข้างต้น ก็คือ ต้องการทำให้ข้อมูลทั้งหมดบรรจุลงในสดมภ์ (ตัวแปร) ของกรอบข้อมูลได้พอดีด้วยฟังก์ชัน data.frame() ดังนี้

```
> x <- c(10, 5.6, -3, 6.4, 21.7)                    ❶
> y <- c(5.1, 3, 0.5, 11, 2.5)                      ❷
> z <- c("a1", "a2", "a3", "a4", "a5")              ❸

> data.dat <- data.frame(x,y,z)                     ❹
> rm(x, y, z)                                       ❺

> data.dat                                          ❻
     x     y      z
1  10.0   5.1   a1
2   5.6   3.0   a2
3  -3.0   0.5   a3
4   6.4  11.0   a4
5  21.7   2.5   a5

> attributes(data.dat)                              ❼
$names
[1] "x" "y" "z"

$row.names
[1] "1" "2" "3" "4" "5"

$class
[1] "data.frame"
```

❶ ❷ ❸ ความยาว (จำนวนองค์ประกอบ) ของ x, y และ z ต้องเท่ากัน

❻ แสดงรายละเอียดของข้อมูลที่อยู่ภายใน data.dat โดย x, y และ z จะถูกแสดงในแนวตั้ง คือสดมภ์

❼ แสดงชื่อตัวแปรที่มีอยู่ภายใน data.dat

แสดงจำนวนแถวที่มีอยู่ภายใน data.dat

แสดงคลาสของวัตถุ data.dat

คำอธิบาย

❶, ❷ และ ❸ เป็นการสร้างวัตถุ x, y และ z ที่ประกอบด้วยค่าตัวเลข และ ตัวอักษรต่าง ๆ โดยที่ x และ y เป็นวัตถุชนิดเวกเตอร์ตัวเลข ส่วน z เป็นวัตถุชนิดเวกเตอร์ตัวอักษร

❹ สร้างวัตถุ โดยให้ชื่อว่า data.dat ซึ่งก็คือ กรอบข้อมูล (data frame) ที่ประกอบด้วยองค์ประกอบ 3 ตัว ที่มีค่าเท่ากับ x, y และ z

❺ สังเกตว่าวัตถุ x, y และ z เดิมยังคงอยู่ จึงควรลบวัตถุ x, y และ z ออกจากหน่วยความจำ เพราะถ้าหากไม่ลบบอก การทำงานอาจเกิดการสับสนได้ เช่น เมื่อต้องเรียกใช้ x เป็นตัวแปรหนึ่งในกรอบข้อมูล data.dat ทำให้ไปเรียกใช้วัตถุ x ที่อยู่นอกกรอบข้อมูลได้

❻ เป็นการแสดงองค์ประกอบต่าง ๆ ที่อยู่ในกรอบข้อมูล data.dat

❼ เป็นการดูแอ็ททริบิวต์ (attribute) ของกรอบข้อมูล data.dat ซึ่งจะแสดงผลลัพธ์ คือ บอกชื่อตัวแปร (สดมภ์) ที่มีอยู่ในกรอบข้อมูล ถัดมาก็จะบอกถึงจำนวนแถวที่มีอยู่ภายในกรอบข้อมูล และสุดท้ายบอกประเภทของคลาส คือ เป็น กรอบข้อมูล (data.frame)

1.6.5 ฟังก์ชัน (Function)

เป็นคำสั่งที่โปรแกรม **R** ใช้ในการทำงานตามที่ผู้ใช้เลือก ฟังก์ชันของโปรแกรม **R** มีมากมาย จึงยากต่อผู้ใช้ที่จะจดจำได้ นอกจากนี้แล้ว ผู้ใช้ในระดับสูง เช่น นักสถิติ สามารถสร้างฟังก์ชันขึ้นมาใหม่ได้ตามต้องการ และหากต้องการเผยแพร่ ก็จะส่งฟังก์ชันของตัวเองให้กับ CRAN ซึ่งเป็นองค์กรกลางของ **R** เมื่อตรวจสอบการทำงาน และความถูกต้องแล้ว ฟังก์ชันเหล่านี้ก็จะถูกนำมาใส่ไว้ใน **R** ในเวอร์ชันถัดไป **R** จึงมีการพัฒนาและเปลี่ยนแปลงเวอร์ชันบ่อยๆ โดยมีการเปลี่ยนแปลงเวอร์ชันทุก ๆ 1 – 2 เดือน ซึ่งแสดงให้เห็นว่ามีผู้พัฒนา **R** อยู่ทั่วทุกมุมโลก **R** เป็นโปรแกรมที่พัฒนาขึ้นมาใช้ตามวัตถุประสงค์ของตนเองได้ ฟังก์ชันมักจะตามด้วยอากิวเมนต์ (argument) ที่อยู่ในวงเล็บ ซึ่ง argument ในที่นี้อาจจะเป็น ค่าตัวเลข สมการ ชื่อของวัตถุ หรือ ชื่อตัวแปรก็ได้ ตัวอย่างฟังก์ชันใน **R** เช่น

ฟังก์ชัน	ความหมาย	ตัวอย่าง
log()	ลอกการิทึม	log(100)
sqrt()	รากที่สอง	sqrt(16)
exp()	ค่ายกกำลังของ e	exp(5)

ฟังก์ชัน	ความหมาย	ตัวอย่าง
abs()	ค่าสัมบูรณ์	abs(-5)
c()	การนำข้อมูลมาต่อกันเป็นเวกเตอร์	c(10, 20, 30, 40, 50)
cbind()	การนำเวกเตอร์มาต่อกันแนวสดมภ์	cbind(x, y)
rbind()	การนำเวกเตอร์มาต่อกันตามแนวแถว	rbind(x,y)
attach()	การนำกรอบข้อมูลปะติดกับเส้นทางค้นหาของ **R**	attach(data.dat)
detach()	การยกเลิกการปะติดกรอบข้อมูล	detach(data.dat)

ฟังก์ชันพิเศษ attach() และ detach()

โดยปกติในการเรียกชื่อตัวแปรในกรอบข้อมูล เราจะเรียกชื่อกรอบข้อมูลตามด้วยเครื่องหมาย $ แล้วจึงตามด้วยชื่อตัวแปร (หรือสดมภ์) เช่น เมื่อมีกรอบข้อมูล ชื่อ data.dat ที่ประกอบด้วย x, y และ z ถ้าต้องการเรียกชื่อ ตัวแปร x ภายใน data.dat จะเรียกดังนี้

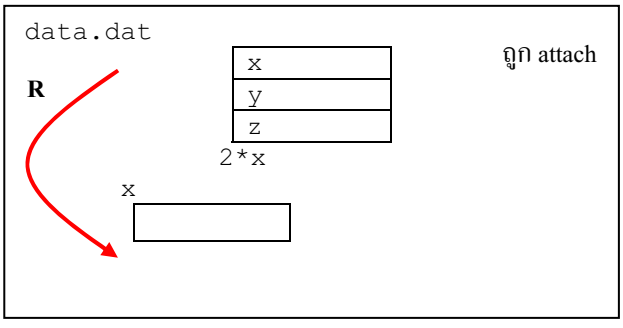
```
> data.dat$x
```

และหากต้องการบวกค่าของ x และ y ใน data.dat เข้าด้วยกัน จะต้องใช้คำสั่งดังนี้

```
> data.dat$x + data.dat$y
```

จะเห็นได้ว่าการเรียกชื่อเช่นนี้จะทำให้ต้องพิมพ์ชื่อยาวมาก ฟังก์ชัน `attach()` จะช่วยแก้ปัญหานี้ โดยเมื่อใส่ชื่อกรอบข้อมูลเป็นอากิวเมนต์ของฟังก์ชัน `attach()` เช่น `attach(data.dat)` ก็จะทำให้ **R** มองเห็นชื่อตัวแปรภายในกรอบข้อมูลที่ได้ attach แล้วนั้นโดยไม่ต้องเรียกชื่อกรอบข้อมูลนำหน้า ดังนั้น การบวก x เข้ากับ y เข้าด้วยกัน จะทำได้ง่ายขึ้น ดังนี้

```
> attach(data.dat)
> x+y
```



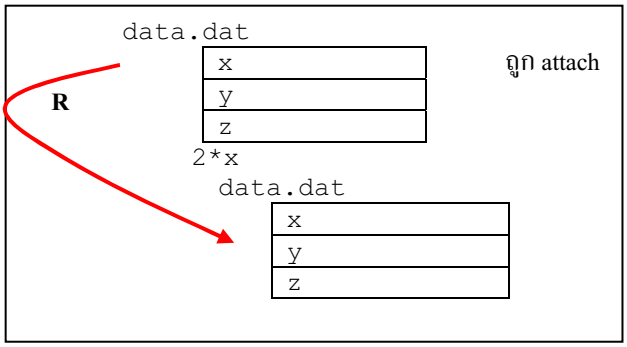
ถึงจุดนี้พึงสังเกตว่าคำสั่งกำหนดค่าต่อไปนี้

```
> x <- 2*x
```

จะไม่เป็นการเปลี่ยนค่า x ใน data.dat ให้เป็น 2*x แต่จะเป็นการสร้างตัวแปร x ขึ้นมาใหม่ นอกกรอบข้อมูล data.dat โดยใช้ค่า x ในกรอบข้อมูล data.dat คูณด้วย 2 คือ มีความหมายเท่ากับ`> x <- 2*data.dat$x`

ถ้าหากต้องการเปลี่ยนค่า x ในกรอบข้อมูล data.dat ให้มีค่าใหม่เป็น 2*x จะต้องใส่ชื่อเดิมของตัวแปร x นั้นไว้ด้านซ้ายมือของเครื่องหมาย <- ดังนี้

```
> data.dat$x <- 2*x
```



อย่างไรก็ตาม พึงจำไว้ว่าขณะนี้ **R** จะยังจำวัตถุกรอบข้อมูล data.dat อันเดิมที่ได้ attach ไว้ก่อนหน้านี้ ถ้าเรียกดู x โดยไม่ใส่ชื่อกรอบข้อมูลนำหน้าจะยังคงได้ค่า x เหมือนเดิม ไม่ใช่ค่าใหม่แต่อย่างใด การที่จะให้ **R** เลิกจำกรอบข้อมูลเดิม (ในหน่วยความจำ) จะต้องใช้คำสั่ง detach() และเมื่อ attach กรอบข้อมูล data.dat ใหม่ ค่า x ที่เรียกได้จึงจะแสดงค่าใหม่ให้เห็น ดังนี้

```
> x
```
ให้ค่า x เดิม

```
> detach(data.dat)
> attach(data.dat)
> x
```
ให้ค่า x ที่คำนวณใหม่

หมายเหตุ	detach กรอบข้อมูลที่ได้ attach ไว้ทุกครั้งที่ทำงานกับกรอบข้อมูลเสร็จแล้ว มิฉะนั้นจะทำงานกับข้อมูลผิดชุดได้ หากต้องการทำงานกับกรอบข้อมูลนั้นอีก ก็ให้ attach ใหม่อีกครั้ง

ฟังก์ชัน search() และ searchpaths()

ฟังก์ชัน search เป็นการค้นหาวัตถุที่มีอยู่ใน **R** console ในขณะที่กำลังเปิดใช้งาน คำสั่งนี้จะแสดง package หรือ library หรือวัตถุ ที่ถูก attach ใน **R** ดังตัวอย่างต่อไปนี้

```
> search()  ❶
[1] ".GlobalEnv"        "package:methods"   "package:stats"
[4] "package:graphics"  "package:grDevices" "package:utils"
[7] "package:datasets"  "Autoloads"         "package:base"

> searchpaths()  ❷
[1] ".GlobalEnv"
[2] "C:/PROGRA~1/R/RW2001/library/methods"
```

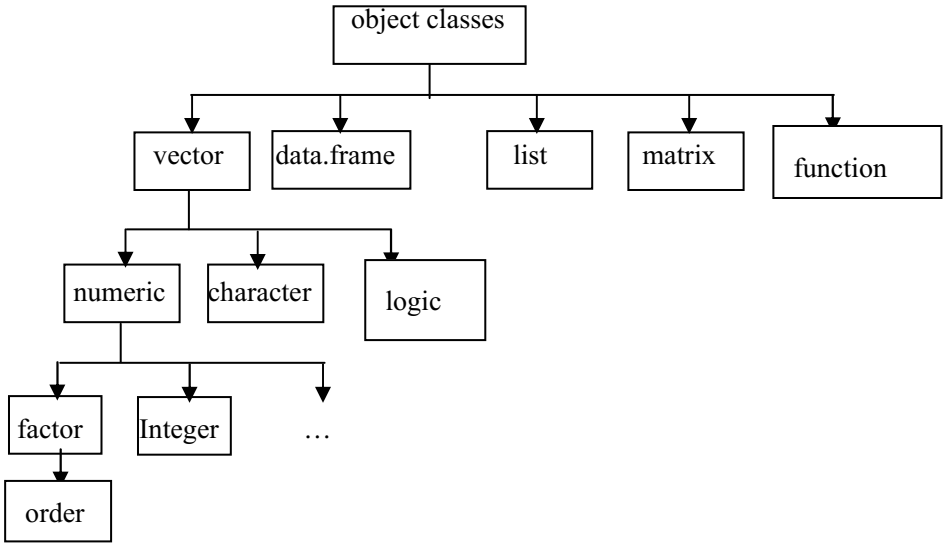
```
[3] "C:/PROGRA~1/R/RW2001/library/stats"
[4] "C:/PROGRA~1/R/RW2001/library/graphics"
[5] "C:/PROGRA~1/R/RW2001/library/grDevices"
[6] "C:/PROGRA~1/R/RW2001/library/utils"
[7] "C:/PROGRA~1/R/RW2001/library/datasets"
[8] "Autoloads"
[9] "C:/PROGRA~1/R/RW2001/library/base"
```

คำอธิบาย

คำสั่ง ❶ แสดงชื่อ packages หรือ library ที่ถูก attach ใน **R** ส่วนคำสั่งที่ ❷ แสดงเส้นทางหรือที่อยู่ใน library นั้น ๆ ว่าอยู่ใน drive ไหน folder อะไร

1.6.6 คลาส (Class)

วัตถุทุกชนิดจะถูกจัดเป็นคลาสประเภทต่าง ๆ เช่น กรอบข้อมูล, ตัวเลข, อักษร, แฟกเตอร์, ลิสต์, เมทริกซ์ เป็นต้น และในคลาสใหญ่คลาสหนึ่งอาจมีคลาสย่อยต่าง ๆ ได้อีก เช่น แฟกเตอร์ เป็นคลาสย่อยของคลาสเวกเตอร์ตัวเลข ยกตัวอย่างดังแผนภาพต่อไปนี้



1.6.7 แฟกเตอร์ (Factor)

แฟกเตอร์ เป็นคลาสย่อยของเวกเตอร์ตัวเลขที่มีคุณสมบัติพิเศษ คือ แต่ละค่าจะเป็นรหัสแทนความหมายใด ๆ เช่น เพศ ให้ตัวเลข 1 และ 2 โดยที่ เลขรหัส 1 คือ ชาย และ 2 คือ หญิง

1.6.8 อาเรย์ (Array)

อาเรย์ หมายถึงชุดของข้อมูลชนิดเดียวที่เรียงเป็นชั้นหลายมิติ **R** จะถือว่ามิติแรกเป็นแถว มิติที่ 2 เป็นสดมภ์ และมิติที่ 3 เป็นชั้น (stratum) แม้ว่าในความเป็นจริง อาเรย์อาจมีมากกว่า 3 มิติ ก็ได้ แต่ในทางสถิติมักจำกัด อาเรย์ไว้เพียง 3 มิติเท่านั้น

1.6.9 เมทริกซ์ (Matrix)

เมทริกซ์ เป็น เวกเตอร์หลาย ๆ เวกเตอร์มาประกอบกัน ซึ่งเวกเตอร์จะมีเพียงแถวเพียงหนึ่งแถว ส่วนเมทริกซ์ เป็นการรวมเวกเตอร์ที่เป็นตัวเลขเข้าด้วยกัน จึงมีแถวหลายแถว ดังนั้นเมทริกซ์จึงเป็นอาเรย์ที่มี 2 มิติ คือ แถวกับสดมภ์ เมทริกซ์ทำให้เป็นกรอบข้อมูลได้ แต่กรอบข้อมูลที่สามารถนำมาเปลี่ยนให้เป็นเมทริกซ์ได้นั้น ตัวแปรทุกตัวต้องเป็นชนิดเดียวกัน

1.6.10 แพคเกจ (Package)

หมายถึง ชุดฟังก์ชัน ซึ่งประกอบไปด้วยหลาย ๆ ฟังก์ชัน สร้างขึ้นเพื่อใช้ในการทำงานตามวัตถุประสงค์นั้น ๆ

1.6.11 ไลบรารี่ (Library)

หมายถึง โครงสร้างที่รวมของ Package, help, demo และ อื่น ๆ ที่เกี่ยวข้องกับ package นั้น ผู้ใช้ในระดับสูงสามารถสร้าง Library ของตัวเองขึ้นมาได้ตามที่ต้องการ เพื่อให้ง่ายต่อการใช้งาน

1.7 สัญลักษณ์ที่ใช้ในโปรแกรม R

ในการเขียนคำสั่ง ในโปรแกรม **R** จะมีสัญลักษณ์พิเศษที่มักจะพบเจอในโปรแกรมนี้ ได้แก่

1.7.1 สัญลักษณ์ทั่ว ๆ ไป

สัญลักษณ์	ความหมาย
<−	Left assignment หมายถึง ให้นำค่าหรือผลลัพธ์จากการคำนวณหรือการทำงานของฟังก์ชันที่ปลายลูกศร ไปใส่ในวัตถุที่เขียนไว้ด้านหน้าหัวลูกศร (ด้านซ้าย)
−>	Right assignment หมายถึง ให้นำค่าหรือผลจากการคำนวณหรือการทำงานของฟังก์ชันที่ปลายลูกศร ไปใส่ในวัตถุที่เขียนไว้ด้านหน้าหัวลูกศร (ด้านขวา)
=	เป็นเครื่องหมาย assignment เช่นเดียวกับ <- แต่ไม่ควรใช้ เพราะจะสับสน

	กับเครื่องหมาย = = ที่เป็นเครื่องหมายเท่ากับทางตรรกศาสตร์

1.7.2 สัญลักษณ์ทางคณิตศาสตร์

สัญลักษณ์	ความหมาย
+	บวก
-	ลบ
*	คูณ
/	หาร
^	ยกกำลัง

1.7.3 สัญลักษณ์ทางตรรกศาสตร์

สัญลักษณ์	ความหมาย
<	น้อยกว่า
>	มากกว่า
==	เท่ากับ
>=	มากกว่าหรือเท่ากับ
<=	น้อยกว่าหรือเท่ากับ
!=	ไม่เท่ากับ
&	และ
|	หรือ

1.8 ข้อควรทราบในการเขียนคำสั่งด้วย R

1.8.1 การใช้ตัวอักษรใน R

โปรแกรม **R** จะใช้อักษรภาษาอังกฤษเป็นภาษาหลักและถือว่าภาษาอังกฤษตัวพิมพ์เล็กและตัวพิมพ์ใหญ่ แตกต่างกัน เช่น

```
> a <- 2 + 5
> A <- 6 + 7
> a==A
[1] FALSE
```

คำอธิบาย	วัตถุ a และ วัตถุ A ข้างต้น **R** จะถือว่าเป็นคนละตัวกัน

1.8.2 การแสดงผลลัพธ์

โดยปกติแล้วการคำนวณใน **R** โดยการให้เก็บผลลัพธ์เป็นชื่อวัตถุ **R** จะไม่แสดงผลลัพธ์ให้เห็น จนกว่าจะพิมพ์ชื่อวัตถุนั้น แต่หากต้องการให้แสดงผลลัพธ์ในทันที และเก็บผลลัพธ์นั้นไว้เป็นวัตถุ ให้พิมพ์วงเล็บคร่อมคำสั่งทั้งหมด ดังตัวอย่าง

```
> (a <- 2 + 5)
[1] 7
```

1.8.3 การกำหนดชื่อของวัตถุ

ชื่อของวัตถุประกอบด้วยตัวอักษรที่มีตั้งแต่หนึ่งตัวขึ้นไป สามารถมีได้ไม่จำกัด แต่ต้องไม่มีการเว้นวรรค หากชื่อมีข้อความต่อกันมาก ๆ สามารถคั่นด้วยจุด (.) หรือใช้ตัวพิมพ์ใหญ่ หรือใช้เครื่องหมาย "_" (underscore) ได้ ตัวอย่าง เช่น

```
> mean.blood.pressure<- (120+130)/2
> mean_blood_pressure <- (120+130)/2
> MeanBloodPressure <- (120+130)/2
```

1.8.4 การรวมคำสั่งไว้ในบรรทัดเดียวกัน

R สามารถที่จะรวมการพิมพ์คำสั่งหลาย ๆ คำสั่งให้อยู่ในบรรทัดเดียวกันได้ โดยการคั่นแต่ละคำสั่งด้วยเครื่องหมาย ; (semi colon)

```
> a <- 2 + 5 ; b <- 5 + 7
> a; b
[1] 7
[2] 12
```

1.8.5 การใช้คำอธิบาย

ผู้ใช้สามารถใส่คำอธิบายต่าง ๆ ในโปรแกรม **R** ได้ ซึ่งสามารถทำได้โดยใช้เครื่องหมาย # ข้อความใด ๆ ก็ตาม ที่ตามหลังเครื่องหมาย # **R** จะไม่นำไปทำงาน การใช้คำอธิบาย เหมาะสำหรับผู้ใช้ที่ต้องการใส่ข้อความหลังคำสั่งที่ใช้ใน **R** ว่าคำสั่งแต่ละคำสั่ง สร้างขึ้นเพื่อให้ทำอะไรบ้าง เช่น

```
> name <- c("Tom", "John", "Smith") # สร้างวัตถุชื่อ name ประกอบด้วยชื่อคน 3 ชื่อ
> age <- c(30, 40, 37)              # สร้างวัตถุชื่อ age ประกอบด้วยอายุ 3 ค่า
> wt <- c(70, 80, 75)               # สร้างวัตถุชื่อ wt ประกอบด้วยน้ำหนัก 3 ค่า
```

```
> ht <- c(180, 185, 179)                # สร้างวัตถุชื่อ ht ประกอบด้วยส่วนสูง 3 ค่า
> person.dat <- data.frame(name, age, wt, ht)
# สร้างกรอบข้อมูล person.dat มีตัวแปรชื่อ name, age, wt และ ht
```

1.8.6 การเรียกใช้คำสั่งเดิม

หากต้องการเรียกใช้คำสั่งเดิมที่ได้ใช้ไปแล้ว สามารถเรียกใช้โดยไม่ต้องพิมพ์คำสั่งเดิมอีก ด้วยการกดลูกศรชี้ขึ้นบนคีย์บอร์ด **R** จะเรียกคำสั่งที่ใช้ไปก่อนหน้า ทีละ 1 คำสั่ง

1.8.7 การเรียกดูวัตถุที่ถูกสร้างไว้แล้วใน R console

การทำงานใน **R** ผู้ใช้มักจะสร้างวัตถุต่าง ๆ ขึ้นมา บางครั้งไม่สามารถจำชื่อได้หมด หรือ ชื่อวัตถุที่ถูกสร้างใหม่มีชื่อเหมือนกับชื่อตัวแปรในกรอบข้อมูล ในกรณีที่ชื่อวัตถุและชื่อตัวแปรในกรอบข้อมูลมีชื่อเหมือนกัน **R** จะเรียกใช้วัตถุที่อยู่นอกกรอบข้อมูลมาทำงานก่อนเสมอ ดังนั้นจึงต้องมีการตรวจสอบเสมอว่า ได้สร้างวัตถุอะไรไปบ้าง ตรวจสอบว่ามีวัตถุอะไรที่อยู่นอกกรอบข้อมูล ด้วยการใช้คำสั่ง **`ls()`**

```
> ls()
 [1] "a"                    "A"                     "age"
 [4] "b"                   "ht"                    "mean.blood.pressure"
 [7] "mean_blood_pressure" "MeanBloodPressure"    "name"
[10] "person.dat"          "wt"
```

คำสั่ง **`ls()`** แสดงให้เห็นเพียงชื่อของวัตถุที่ถูกสร้างขึ้นเท่านั้น แต่ไม่แสดงชื่อ package หรือ library ให้เห็น

1.8.8 การเรียกดูแถวหรือสดมภ์ในกรอบข้อมูล

ลำดับการเรียกข้อมูลใน **R** กำหนดให้ตัวแรกสุดเป็นแถวก่อนเสมอ ตัวถัดมาจะเป็นสดมภ์ ดังคำสั่งต่อไปนี้

```
> person.dat[1,1]                # ให้แสดงข้อมูลในแถวที่ 1 สดมภ์ที่ 1
[1] "Tom"
```

หากต้องการเรียกทุกแถว หรือทุกสดมภ์ ทำได้โดยการพิมพ์เครื่องหมายคอมม่า “,”

```
> person.dat[1,]                 # ให้แสดงข้อมูลในแถวที่ 1 ทุกสดมภ์
  name age wt  ht
1  Tom  30 70 180
```

```
> person.dat[,2]                # ให้แสดงข้อมูลทุกแถวของสดมภ์ที่ 2
[1] 30 40 37
```

หากต้องการเรียกชื่อตัวแปร ให้พิมพ์เครื่องหมาย "$" หลังชื่อกรอบข้อมูลสำหรับกรณีที่ไม่ได้ attach ข้อมูล

```
> person.dat[person.dat$name=="John",]
  name age wt  ht
2 John  40 80 185
```

1.8.9 การลบวัตถุออกจาก R console

วัตถุบางตัว ที่ไม่ต้องการใช้แล้ว หรือป้องกันการสับสนระหว่างชื่อวัตถุกับชื่อตัวแปร ผู้ใช้สามารถลบออกจาก **R** console ได้ ด้วยคำสั่ง **`rm()`**

```
> rm(age, name, wt, ht)
> ls()
[1] "a"                    "A"                      "b"
[4] "mean.blood.pressure" "mean_blood_pressure" "MeanBloodPressure"
[7] "person.dat"
```

เมื่อใช้คำสั่ง **`ls()`** ดูวัตถุที่อยู่ใน **R** console จะเห็นได้ว่า วัตถุที่ชื่อ age name wt และ ht ถูกลบไปแล้ว โดยในกรอบข้อมูล person.dat มีตัวแปรชื่อ age, name, wt และ ht เช่นกัน ดังนั้นการลบวัตถุ age, name, wt และ ht ออก ก็จะช่วยลดการสับสนของการเรียกใช้ตัวแปรในภายหลัง หากต้องการลบวัตถุทุกชนิดออกจาก **R** console ก็สามารถทำได้ด้วยการใช้คำสั่งดังนี้

```
> rm(list=ls())
```

1.8.10 การลบผลลัพธ์ออกจากหน้าจอ R console

ในบางครั้งเมื่อผู้ใช้ทำงานไปนาน ๆ ก็จะมีผลลัพธ์ปรากฏขึ้นหน้าจอเป็นจำนวนมาก หากไม่ต้องการผลลัพธ์ดังกล่าว สามารถลบออกจากหน้าจอ **R** console ได้ โดยการไปเลือกที่เมนู Edit และ เลือก Clear console หรือ กด Ctrl+L

1.8.11 การสลับไปมาระหว่างหน้าจอต่าง ๆ ใน RGui

หน้าจอในการพิมพ์คำสั่งต่าง ๆ ใน **R** คือ **R** Console แต่หากบางครั้งมีการสร้างกราฟ **R** ก็จะปรากฏหน้าจอ **R** Graphics มาให้ หรือการเรียก help ขึ้นมาดู หากต้องการทำงานสลับไปมาระหว่างหน้าจอต่าง ๆ เหล่านี้ ทำได้โดยการกด Alt ค้างไว้ ตามด้วยปุ่ม Tab เพื่อสลับไปยังหน้าจอที่ต้องการ

1.8.12 ความครบถ้วนของคำสั่ง

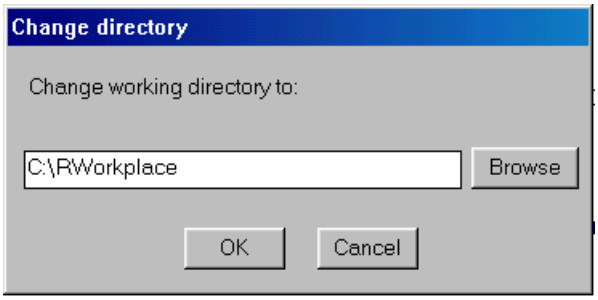
หากคำสั่งที่พิมพ์ไม่ครบถ้วน เมื่อกด Enter **R** จะไม่กระทำตามคำสั่ง แต่จะขึ้นบรรทัดใหม่ พร้อมด้วยเครื่องหมาย + นั่นหมายถึงคำสั่งยังไม่ครบถ้วน เช่น ขาดวงเล็บปิด ก็จะต้องพิมพ์วงเล็บปิดแล้วจึงกด Enter **R** จึงจะกระทำตามคำสั่งนั้น แต่ถ้าหากคำสั่งนั้น ๆ ตกวงเล็บหลายวงเล็บ **R** จะไม่ให้ผู้ใช้แก้ไขคำสั่งนั้น ในกรณีนี้ให้กด Esc เพื่อยกเลิกคำสั่ง

1.8.13 การกำหนดโฟลเดอร์สำหรับการทำงาน

หากไม่ได้กำหนดโฟลเดอร์ที่ต้องการทำงานด้วยการเลือก Properties จากไอคอน **R** สามารถพิมพ์คำสั่งใน **R** Console เพื่อกำหนดไดเรกตอรี่ที่ต้องการทำงานด้วยคำสั่ง **`setwd()`** ดังนี้

```
> setwd("c:/rworkplace")
```

R กำหนดให้ใช้สัญลักษณ์ Forward slash (/) เป็นตัวแบ่งแยก sub-folder ต่าง ๆ เหมือนระบบ Linux ในขณะที่โปรแกรมทั่วไปบน Windows กำหนดให้ใช้สัญลักษณ์ **Back slash (\)** เป็นตัวแบ่งแยก หรือสามารถกำหนดโฟลเดอร์สำหรับการทำงานได้โดยการเลือกเมนู File จากนั้นเลือก Change dir… ก็จะปรากฏดังรูป 1.5 ให้พิมพ์ C:\RWorkplace (ถ้าเป็นการเปลี่ยน Folder หรือ Directory ที่เลือกจากเมนู จะใช้เครื่องหมาย \ เหมือนกับโปรแกรมอื่น ๆ)



รูป 1.5 การเปลี่ยน working directory

1.8.14 การเรียกดูโฟลเดอร์ที่ทำงานอยู่ในปัจจุบัน

ในกรณีที่ต้องการทราบโฟลเดอร์ที่กำลังทำงานอยู่ในปัจจุบันสามารถเรียกดูได้จากคำสั่ง **`getwd()`** ดังนี้

```
> getwd()
```

1.9 การเรียกใช้การช่วยเหลือ

โปรแกรม **R** มีลักษณะพิเศษ คือ ผู้ใช้สามารถขอความช่วยเหลือในขณะที่ใช้ผ่าน online help การดูวิธีการเรียกใช้ help สามารถใช้คำสั่งดังต่อไปนี้

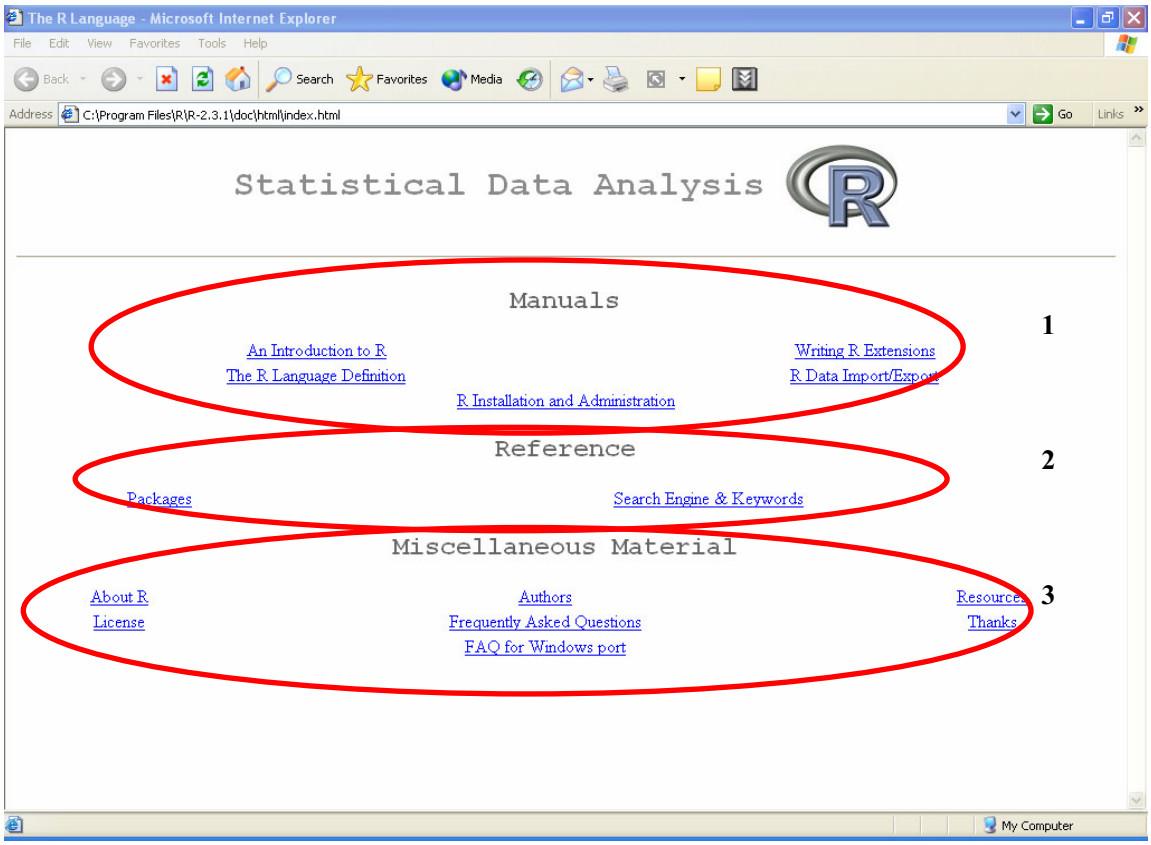
```
> help.start()        ❶ หรือ
> help()              ❷ หรือ
> ?help               ❸
```

คำอธิบาย

คำสั่งที่ ❶ ใช้ในกรณีที่ต้องการทราบรายละเอียดต่าง ๆ ใน **R** เมื่อพิมพ์คำสั่งนี้ หน้าจอ online help ดังรูป 1.6 ก็จะปรากฏขึ้น ซึ่งจะมีให้เลือก 3 ส่วนด้วยกัน คือ

1. Manuals เป็นส่วนที่แสดงคู่มือการใช้โปรแกรม
2. Reference เป็นส่วนที่เก็บรวบรวม packages หรือ libraries ต่าง ๆ และมีส่วนของการค้นหาโดยใช้คำสำคัญ คือ Search Engine & Keywords
3. Miscellaneous Material เป็นส่วนอื่น ๆ ที่เกี่ยวกับโปรแกรม เช่น ประวัติความเป็นมาของการเขียนโปรแกรม ผู้เขียนโปรแกรม ลิขสิทธิ์ เป็นต้น

คำสั่งที่ ❷ และ ❸ ใช้เหมือนกัน โดยคำสั่งที่ ❸ ใช้เครื่องหมาย ? แทนการพิมพ์คำว่า help เมื่อพิมพ์คำสั่งดังกล่าว **R** จะแสดงหน้าจอ online help ขึ้นมาแสดงให้เห็นวิธีการใช้คำสั่ง help



รูป 1.6 หน้าจอแสดงผลการใช้คำสั่ง `help.start()`

1.9.1 กรณีทราบคำสั่งที่มีอยู่แล้วใน R

ถ้าต้องการทราบไวยากรณ์ (Syntax) ของการใช้คำสั่งต่าง ๆ ที่มีอยู่แล้วใน **R** ให้พิมพ์ คำสั่ง **`help()`** ตามด้วยชื่อคำสั่งอยู่ในวงเล็บ หรือ สามารถพิมพ์ เครื่องหมายคำถาม คือ ? ซึ่งใช้แทนคำว่า help จากนั้นจึงตามด้วยคำสั่งที่ต้องการขอความช่วยเหลือ เช่น

```
> help(mean)        ❶
> ?mean             ❷
```

คำอธิบาย

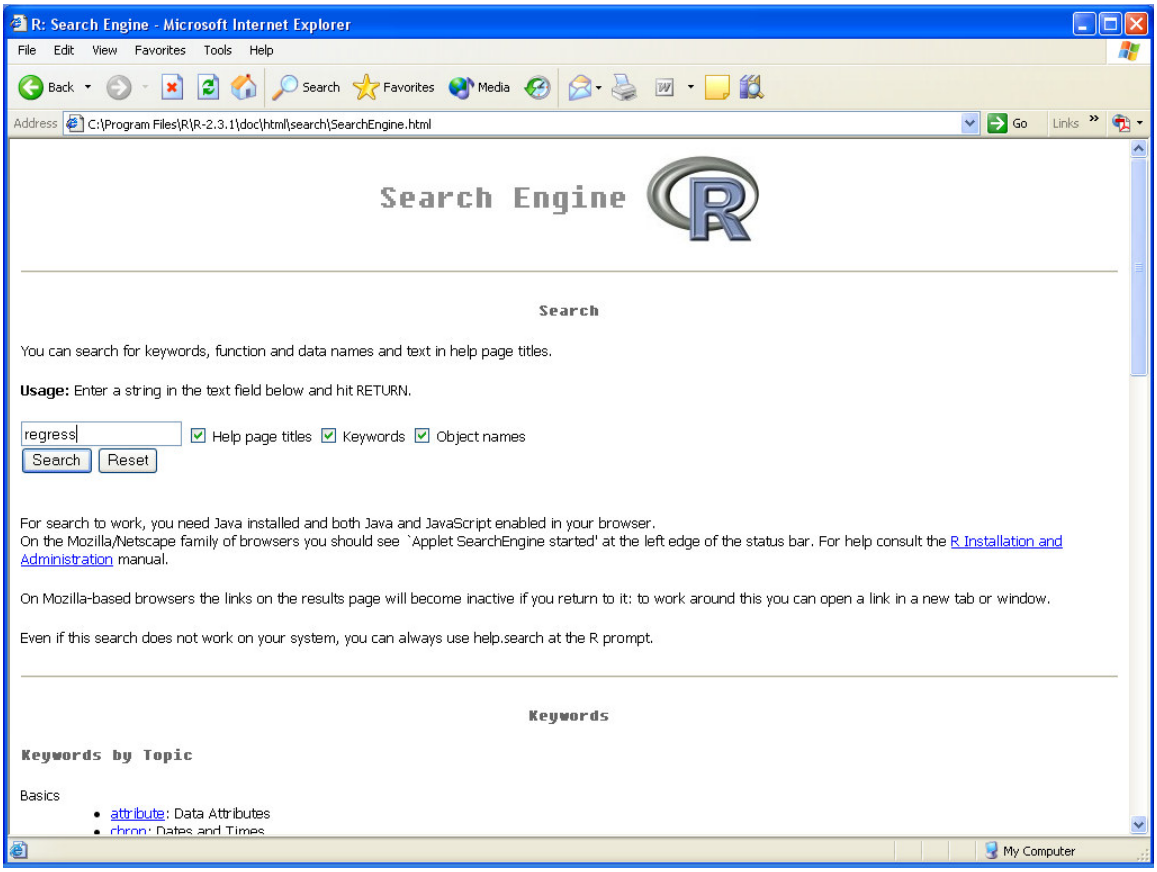
คำสั่งทั้งสองคำสั่งจะแสดงหน้าจอ online help ของการใช้คำสั่ง mean เหมือนกัน ฉะนั้นหากต้องการเรียกความช่วยเหลือที่สั้นกะทัดรัดก็สามารถใช้คำสั่งที่ ❷

1.9.2 กรณีไม่ทราบคำสั่งที่มีอยู่ใน R

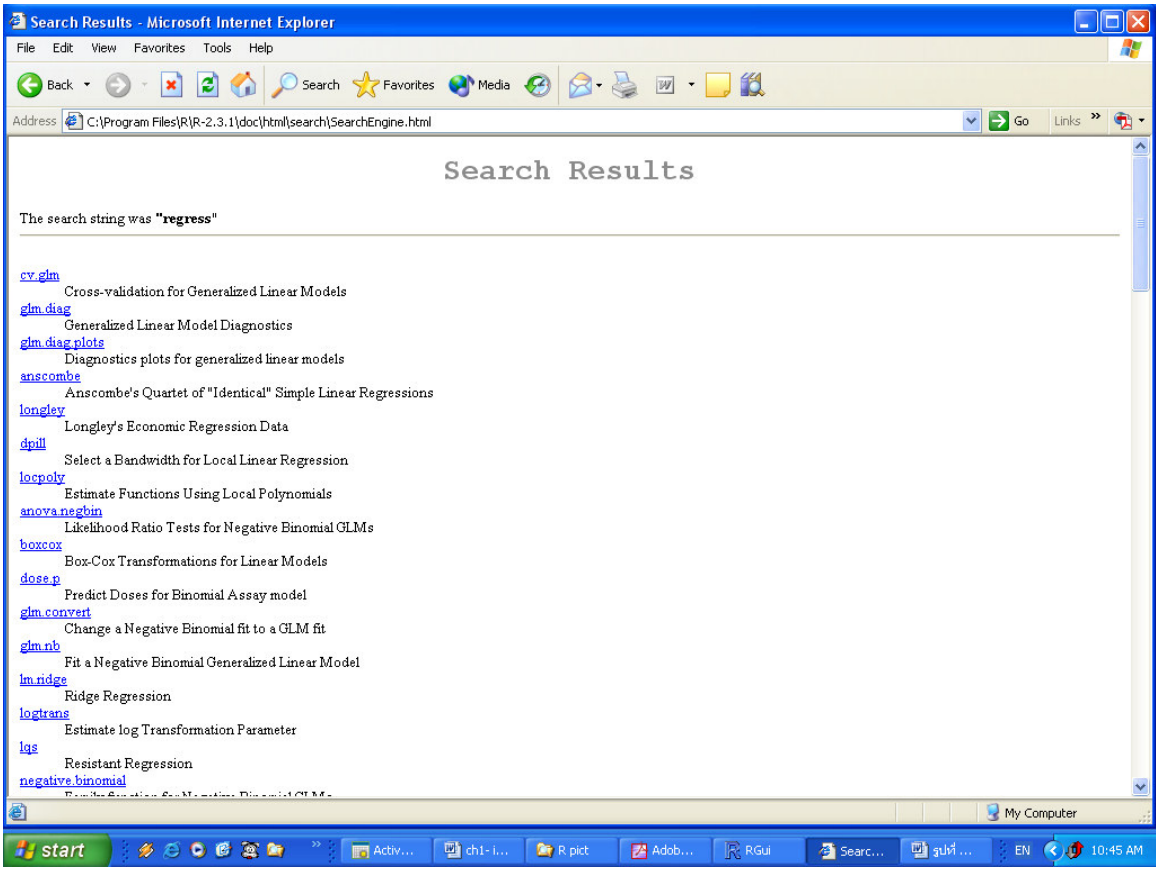
สำหรับกรณีนี้ หากทราบคำบางคำที่เกี่ยวข้องกับงานที่ต้องการใช้ ซึ่งอาจเป็นคำสำคัญ (keyword) เช่น คำว่า test การเรียกดูความช่วยเหลือจะต้องพิมพ์คำนั้นไว้ในเครื่องหมาย " " ดังเช่น

- `help.search("test")`

อย่างไรก็ตาม การพิมพ์ **help.start()** เพื่อเปิด online help ดังรูป 1.6 แล้วเลือก Search Engine & Keywords จะเป็นวิธีที่ง่ายและมีตัวเลือกให้มากกว่า เช่นต้องการค้นคำว่า regression ก็ให้คลิกที่เมนู ดังกล่าวก็จะปรากฏหน้าจอดังรูป 1.7 ขึ้น และให้พิมพ์คำว่า regression ในช่องว่าง จากนั้นจึงกด ปุ่ม Search ผลการค้นหาดังในรูป 1.8



รูป 1.7 หน้าจอ Search Engine ใน **R**



รูป 1.8 ผลการ Search จากคำสำคัญ regression

1.9.3 กรณีคำสั่งที่ต้องการใช้ไม่ได้อยู่ในไลบรารี่ที่เปิดมาพร้อมกับ R console

โดยปกติแล้วหากเปิดโปรแกรม **R** ขึ้นมาใช้งาน library ที่ถูกเปิดขึ้นโดยอัตโนมัติอยู่แล้ว คือ base stats graphics grDevices utils methods และ datasets การขอดูความช่วยเหลือคำสั่งที่อยู่ใน library เหล่านี้ก็จะเรียกได้ทันที แต่หากคำสั่งที่ต้องการใช้อยู่ใน library อื่น จะต้องเปิด library นั้นขึ้นมาก่อน จึงจะสามารถเรียกดูความช่วยเหลือคำสั่งที่อยู่ใน library ที่ต้องการได้ ดังเช่น การดูความช่วยเหลือของคำสั่ง negative.binomial ซึ่งอยู่ใน library MASS จะต้องเปิด library นี้ขึ้นมาก่อน ดังนี้

```
> library(MASS)
> help(negative.binomial)
```

1.9.4 การเรียกตัวอย่างในแฟ้มความช่วยเหลือมาทำงานบน R console

เมื่อต้องการเรียกตัวอย่างคำสั่งมาทำงาน อาจใช้วิธี copy ข้อความมา paste บน **R** console ก็ได้ แต่มีอีกวิธีหนึ่งที่สะดวกคือใช้คำสั่ง **example()** ตามด้วยชื่อคำสั่งที่อยู่ในวงเล็บ ดังเช่น ต้องการดูตัวอย่างการใช้คำสั่ง **summary()** ซึ่งจะได้ผลลัพธ์บน **R** console ดังนี้

```
> example(summary)

summry> summary(attenu, digits = 4)
     event            mag           station         dist
 Min.   : 1.00   Min.   :5.000   117    :  5   Min.   :  0.50
 1st Qu.: 9.00   1st Qu.:5.300   1028   :  4   1st Qu.: 11.32
 Median :18.00   Median :6.100   113    :  4   Median : 23.40
 Mean   :14.74   Mean   :6.084   112    :  3   Mean   : 45.60
 3rd Qu.:20.00   3rd Qu.:6.600   135    :  3   3rd Qu.: 47.55
 Max.   :23.00   Max.   :7.700   (Other):147   Max.   :370.00
                                 NA's   : 16
     accel
 Min.   :0.00300
 1st Qu.:0.04425
 Median :0.11300
 Mean   :0.15422
 3rd Qu.:0.21925
 Max.   :0.81000

summry> summary(attenu$station, maxsum = 20)
  117   1028     113    112    135    475   1030   1083   1093   1095
    5      4       4      3      3      3      2      2      2      2
  111    116    1219   1299    130   1308   1377   1383(Other)   NA's
    2      2       2      2      2      2      2      2    120     16


summry> lst <- unclass(attenu$station) > 20


summry> summary(lst)
   Mode   FALSE    TRUE    NA's
logical      28     138      16

summry> summary(as.factor(lst))
FALSE  TRUE  NA's
   28   138    16
```

ในการใช้คำสั่ง **`example()`** จะทำให้ผลที่ได้จากการทำงานของตัวอย่างมี prompt เป็นชื่อของคำสั่งนั้น ตามด้วยเครื่องหมาย '>' เช่นเป็น 'summary>' ดังตัวอย่างข้างต้น

1.9.5 รูปแบบการแสดงหน้าจอความช่วยเหลือ

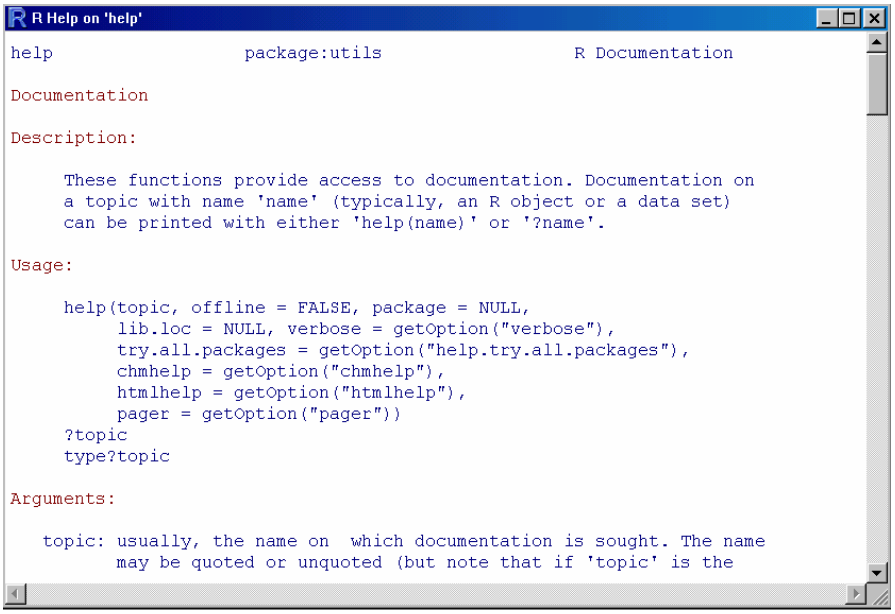
R แสดงหน้าจอ help เป็น 3 รูปแบบด้วยกัน คือ

1. ความช่วยเหลือที่ปรากฏด้วยขึ้นมาภายใต้หน้าจอของ RGUI
2. ความช่วยเหลือที่ปรากฏด้วย Web browser ซึ่งอยู่ในรูปแบบ html
3. ความช่วยเหลือที่ปรากฏด้วย Windows ที่เป็น Complied html ขึ้นมา

โดย **R** จะแสดงหน้าจอ online help ตามรูปแบบที่ผู้ใช้เลือก ตัวอย่างการใช้คำสั่ง help ดังนี้

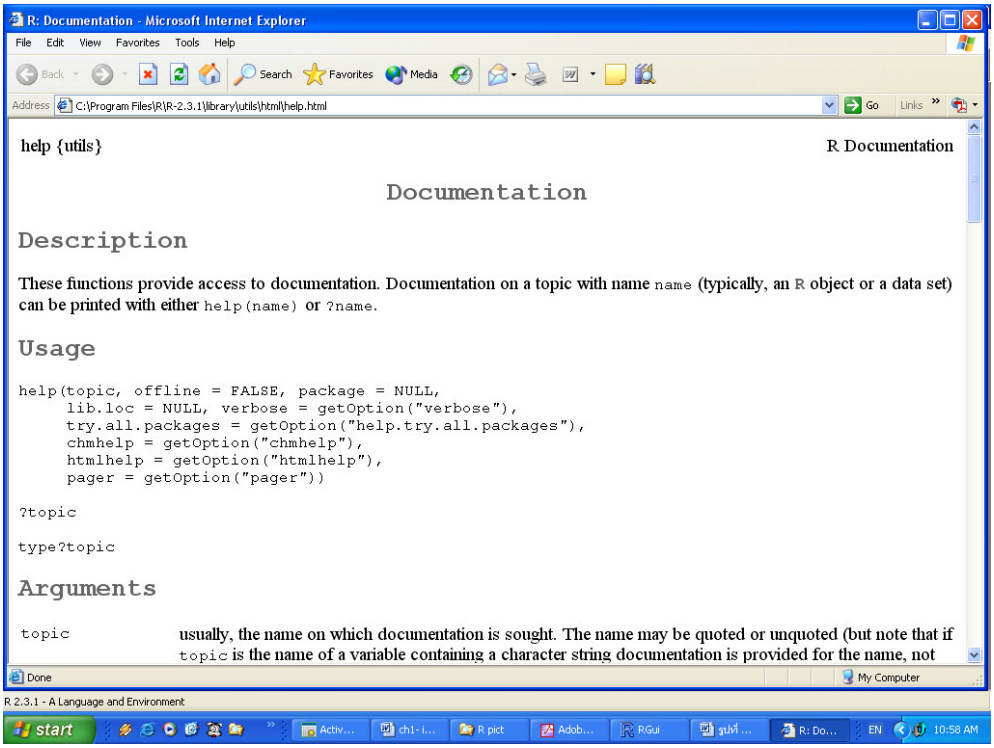
```
> help(help)
```

จากคำสั่งนี้ **R** จะแสดงหน้าจอ help ดังรูป 1.9



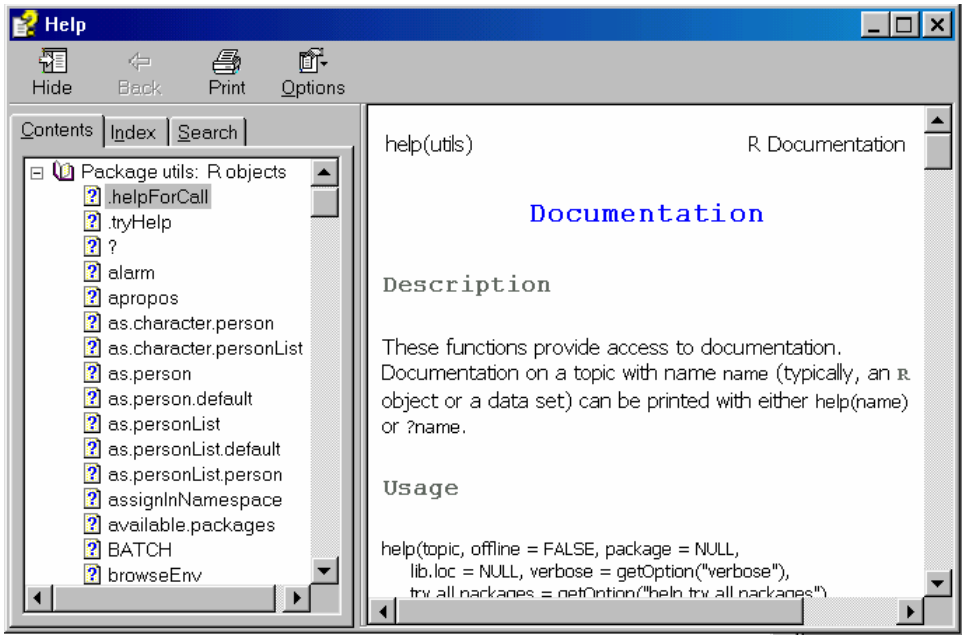
รูป 1.9 หน้าจอ help ภายใต้ RGUI

`> help(help, html=T)` จะแสดงหน้าจอ help ด้วย web browser ดังรูป 1.10



รูป 1.10 หน้าจอ help จาก web browser

`> help(help, chm=T)` จะแสดงหน้าจอ help ดังรูป 1.11



รูป 1.11 หน้าจอ help จาก Complied html

แบบฝึกหัด

จงใช้โปรแกรม **R** คำนวณค่าต่าง ๆ ต่อไปนี้

1. ค่ารากที่สองของ 876 มีค่าเท่าใด
2. ค่าลอกกาลิทึมของ 0.0001 มีค่าเท่าใด
3. จากผลการศึกษาอัตราการเกิดโรคชนิดหนึ่ง เป็นดังนี้

	เป็นโรค	ไม่เป็นโรค	รวม
ชาย	50 (a)	120 (b)	170
หญิง	75 (c)	205 (d)	280
รวม	125	325	450

- จงคำนวณหาอัตราการเป็นโรคในชาย และหญิง
- จงคำนวณค่า Odds ของการเป็นโรคในเพศชาย และหญิง โดยสูตรการคำนวณ คือ $Odds_{ชาย} = a/b$ และ $Odds_{หญิง} = c/d$
- จงคำนวณหาค่า Odds Ratio ของการเป็นโรคดังกล่าวโดยสูตรการคำนวณ คือ $OR = ad/bc$

4. จงสร้าง data frame ของข้อมูลต่อไปนี้

x	y	z	x	y	z
35	65	6	7	9	4
7	9	7	20	35	5
23	41	4	3	1	4
23	41	7	19	33	5
25	45	5	0	0	6
29	53	6	28	51	1
25	45	6	30	55	5

บทที่ 2
การจัดการกับข้อมูล (Data Management)

การจัดการกับข้อมูลเป็นการเตรียมข้อมูลให้พร้อมก่อนการนำไปวิเคราะห์ข้อมูลเพื่ออะไร ซึ่งในบทนี้จะกล่าวถึง การสร้างตัวแปร การสร้างแฟ้มข้อมูลด้วยโปรแกรม **R** การอ่านข้อมูลที่สร้างจากโปรแกรมต่างๆ เข้ามาในโปรแกรม **R** การเก็บบันทึกผลการประมวลผล การปรับเปลี่ยนข้อมูล การเลือกข้อมูล และการสุ่มตัวอย่าง

2.1 คำสั่งใน R ที่ใช้ในการจัดการกับข้อมูล

คำสั่ง	คำอธิบาย
การสร้างวัตถุใน R	
c()	นำข้อมูลหรือวัตถุที่ต้องการมาเรียงต่อกัน วัตถุนั้นจะเป็นค่าคงที่ เวกเตอร์ หรือเป็นวัตถุชนิดอื่นๆก็ได้
seq()	สร้างเวกเตอร์ที่เป็น sequence
rep()	สร้างเวกเตอร์ที่เป็นตัวเลขหรืออักษรซ้ำ ๆ
คำสั่ง	**คำอธิบาย**
factor()	สร้างเวกเตอร์ที่ค่าเป็นตัวแปรเชิงกลุ่มหรือ factor
as.integer()	เปลี่ยนคุณสมบัติของวัตถุที่อยู่ใน () ให้เป็น integer
as.factor()	เปลี่ยนคุณสมบัติของวัตถุที่อยู่ใน ()ให้เป็น factor
as.numeric()	เปลี่ยนคุณสมบัติของวัตถุที่อยู่ใน ()ให้เป็น numeric
data.frame()	สร้างกรอบข้อมูล (แฟ้มข้อมูล)
cbind()	สร้างเมทริกซ์ จากเวกเตอร์หรือกรอบข้อมูลโดยนำมาเรียงต่อกันในแนวคอลัมน์
rbind()	สร้างเมทริกซ์ จากเวกเตอร์หรือกรอบข้อมูลโดยนำมาเรียงต่อกันในแนวแถว
attributes(x)	ตรวจสอบว่าวัตถุ x มีลักษณะใดและมีองค์ประกอบอะไรบ้าง
การอ่านแฟ้มข้อมูล	
library (foreign)	เรียก `library` ชื่อ `foreign` เพื่อเตรียมอ่านแฟ้มข้อมูลที่

	สร้างขึ้นด้วยโปรแกรมอื่น เช่น **SPSS STATA**
read.spss()	อ่านแฟ้มข้อมูล **spss** เข้าใน **R**
read.table()	อ่านแฟ้มข้อมูล ที่มีนามสกุล .txt เข้าใน **R**
read.csv()	อ่านแฟ้มข้อมูล ที่มีนามสกุล .csv เข้าใน **R**
การบันทึกผลการวิเคราะห์	
write.table()	บันทึกผลการวิเคราะห์ไว้ในแฟ้มที่มีนามสกุล .txt
การปรับเปลี่ยนตัวแปร	
round()	ปัดทศนิยม
replace()	แทนที่ค่าในตัวแปรใน () ตามเงื่อนไขที่กำหนด
ifelse()	ปรับเปลี่ยนค่าตัวแปรใน () ตามเงื่อนไขที่กำหนด
cut()	กำหนดจุดตัดของตัวแปรใน () ที่มีค่าต่อเนื่อง ณ จุดที่ระบุ
levels()	อธิบายรหัสตามที่กำหนด
การเลือกข้อมูล/สุ่มตัวอย่าง	
subset()	เลือกข้อมูลบางส่วนจากกรอบข้อมูล ตามเงื่อนไขที่ระบุ
sample()	เลือกตัวอย่างสุ่ม

2.2 การสร้างเวกเตอร์ (Vector) หรือ ตัวแปร (Variable)

เวกเตอร์ (หรือตัวแปร) มีหลายลักษณะ เช่น เป็นตัวเลข (numerical vector) ตัวอักษร (character vector) ตรรก (logical vector) การนำข้อมูลหลายๆค่ามาสร้างเวกเตอร์ต้องอาศัยฟังก์ชัน **c( )**

วิธีใช้

2.2.1 การสร้างเวกเตอร์ตัวเลข (numerical vector)

ตัวอย่าง 2.1 ถ้ามีตัวแปร 2 ตัวที่มีค่าดังต่อไปนี้ x = 5, 10, 15, 20, 25 และ y = 2,4,6,8

สร้าง vector (ตัวแปร) x และ y ได้ดังนี้

```
> x <- c(5,10,15,20,25)
> y<-c(2,4,6,8)
```

สร้าง vector (ตัวแปร) x y ที่มีค่า `0, 5, 10, 15, 20, 25,2,4,6,8`

x y

```
> xy<-c(0,x,y)
```

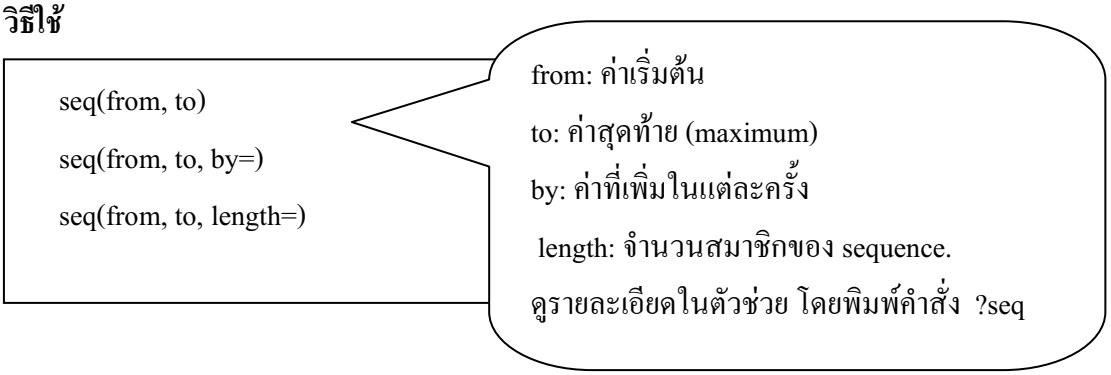
เรียกดูค่าของตัวแปร x และ xy

```
> x
[1] 5 10 15 20 25

> xy
 [1] 0  5 10 15 20 25  2  4  6  8
```

ถ้าพิมพ์เฉพาะคำสั่ง `c(5,10,15,20,25)` โดยไม่กำหนดชื่อ vector ที่จะเก็บค่าเหล่านี้ไว้ เราจะไม่สามารถนำผลลัพธ์จากคำสั่ง `c(5,10,15,20,25)` มาใช้งานในครั้งต่อไปได้

2.2.2 การสร้างเวกเตอร์เป็นตัวเลขที่เป็นระบบ โดยใช้ฟังก์ชัน seq()

วิธีใช้



ตัวอย่าง 2.2 สร้างตัวแปร มีค่าตั้งแต่ 1 ถึง 5 โดยเพิ่มทีละ 1
สามารถใช้คำสั่ง ดังต่อไปนี้ ซึ่งจะให้ผลลัพธ์เหมือนกัน

1) `x<-seq(from=1, to=5)` หรือ 2) `x<-seq(1,5)`
หรือ 3) `x<-seq(1:5)` หรือ 4) `x<- 1:5`

```
> x
[1] 1 2 3 4 5
```

ตัวอย่าง 2.3 สร้างตัวแปร มีค่าตั้งแต่ 1 ถึง 10 โดยเพิ่มทีละ2 หน่วย

```
> x<-seq(from=1, to=10,by=2)
```
หรือเขียนสั้นๆ `x<-seq(1,10,2)`

```
> x
[1] 1 3 5 7 9
```

ตัวอย่าง 2.4 สร้างตัวแปรมีค่าในช่วง a ถึง b โดยมีจำนวนสมาชิกเท่ากับค่าที่ระบุ (length)

```
> x<- seq(from=0, to=20, length=5)
[1]  0  5 10 15 20

> y<- seq(from=0, to=18, length=5)
[1]  0.0  4.5  9.0 13.5 18.0
```

2.2.3 การสร้างเวกเตอร์ที่มีค่าซ้ำ โดยใช้ฟังก์ชัน `rep()`

วิธีใช้

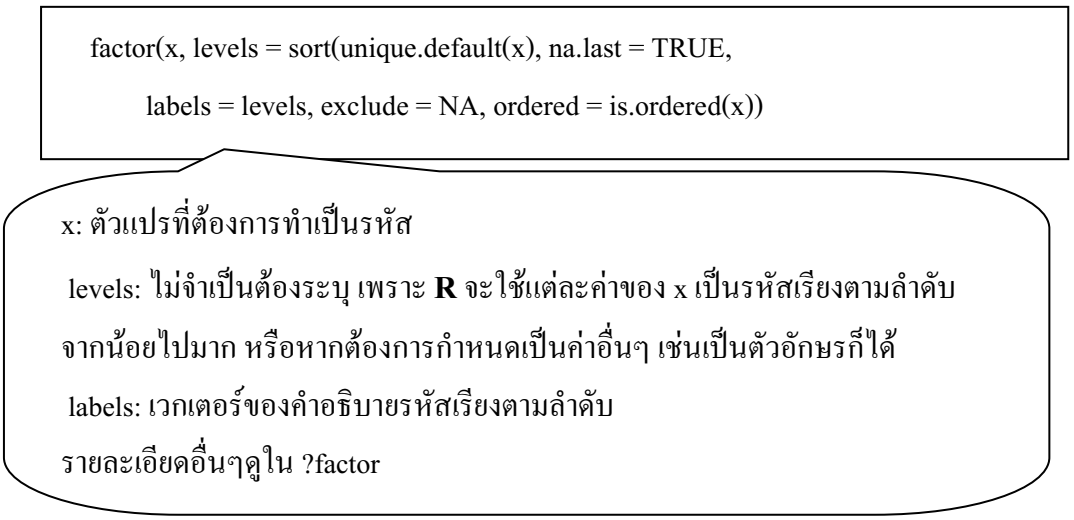
ตัวอย่าง 2.5 สร้างเวกเตอร์ขนาด 10 (มีสมาชิก 10 ตัว) ประกอบด้วยค่า 2 และ4 อย่างละ 5 ตัว

```
> x<-c(2,4)

> y<- rep(x,times=5)          # สร้างเป็นตัวแปร y จากการนำค่าของ x มาต่อกัน 5 ซ้ำ
 [1] 2 4 2 4 2 4 2 4 2 4

> rep(c(3,2),times=5)
 [1] 3 2 3 2 3 2 3 2 3 2

> rep(c(2,4),times=c(5,5))
 [1] 2 2 2 2 2 4 4 4 4 4

> rep(c(2,4),each = 5)
[1] 2 2 2 2 2 4 4 4 4 4
```

2.2.4 การสร้างเวกเตอร์เป็นรหัส (factor)

Factor เป็นคลาสย่อยของเวกเตอร์ตัวเลขที่มีคุณสมบัติพิเศษ คือตัวเลข แต่ละตัวจะเป็นรหัสแทนแต่ละกลุ่มของตัวแปร เช่น ตัวแปรเพศ ให้รหัส 1 แทนชาย และ 2 แทน หญิง สร้าง factor โดยใช้คำสั่ง factor

วิธีใช้



ตัวอย่าง2.6 เปลี่ยนตัวแปรเพศซึ่งเดิมกำหนดค่าเป็นตัวเลข (1,2) ให้เป็นรหัส พร้อมทั้งให้มีคำอธิบายรหัส

```
> sex<-rep(c(1,2),times=c(2,3))
> sex
[1] 1 1 2 2 2

> sex.f <-factor(sex)
> sex.f
[1] 1 1 2 2 2
Levels: 1 2

> levels(sex.f)<-c("male","female")

> sex.f
[1] male male female female female
Levels: male female

> attributes(sex.f)
```

TIP: รวบหลายขั้นตอนภายใต้คำสั่งเดียว

```
>  rm(sex.f)
sex.f<- factor(rep(c(1,2),
     times=c(2,3)),
     labels=c("male","female"))
```

ตรวจสอบว่า sex.f เป็น factor หรือไม่

```
> is.factor(sex.f)
[1] TRUE
```

2.2.5 การสร้างเวกเตอร์ตัวอักษร (character vector)

R จะรับรู้ตัวอักษรภายใต้เครื่องหมาย "" เช่น "1" เป็นตัวอักษรไม่ใช่ตัวเลข

```
> y <- c("m","m","f","f","f")
> y
[1] "m" "m" "f" "f" "f"
```

2.2.6 การเปลี่ยนและการตรวจสอบชนิดของเวกเตอร์

วิธีใช้

as.integer(x, ...)
as.factor(x,…)
as.numeric(x,…)

x เวกเตอร์ที่ต้องการเปลี่ยนให้เป็น integer หรือ factor หรือ numeric

is.integer(x, ...)
is.factor(x,…)
is.numeric(x,…)

x เวกเตอร์ที่ต้องการตรวจสอบว่าเป็น integer หรือ factor หรือ numeric หรือไม่

ตัวอย่าง 2.7

```
x<-c(1.2,2.8,3.0,4.1)
> xi<-as.integer(x)
> xi
[1] 1 2 3 4
```

เปลี่ยนตัวแปร x ให้มีค่าเป็นจำนวนเต็มเก็บไว้ในชื่อ xi

```
> x.f<-as.factor(xi)
> x.f
[1] 1 2 3 4
Levels: 1 2 3 4
> x.fac<-factor(xi)
> x.fac
[1] 1 2 3 4
Levels: 1 2 3 4
```

เปลี่ยนตัวแปร xi ให้เป็นรหัส โดยใช้คำสั่ง as.factor

หรือ factor ซึ่งให้ผลเหมือนกัน

```
> is.integer(x)
[1] FALSE
> is.integer(xi)
[1] TRUE
```

ตรวจสอบว่า ตัวแปรx หรือ xi มีค่าเป็นจำนวนเต็มหรือไม่

```
> is.factor(x.f)
[1] TRUE
> is.factor(x.fac)
[1] TRUE
```

ตรวจสอบว่า ตัวแปร `x.f` หรือ `x.fac` มีค่าเป็นรหัสหรือไม่

2.3 การสร้างแฟ้มข้อมูล (Data File) หรือ กรอบข้อมูล (Data frame)

แฟ้มข้อมูลใน **R** จะมีลักษณะเป็น กรอบข้อมูลซึ่งประกอบด้วยตัวแปร (เวกเตอร์) หลายๆตัว ที่ทุกตัวมีความยาวเท่ากัน หรือ ประกอบด้วยเมทริกซ์ก็ได้ โดยแต่ละสดมภ์ (column) คือ หนึ่งตัวแปร และแต่ละแถว (row) คือลักษณะต่างๆของหนึ่ง case

2.3.1 สร้างแฟ้มข้อมูล (กรอบข้อมูล) ด้วยฟังก์ชัน data.frame ()

วิธีใช้

```
data.frame(..., row.names = NULL, check.rows = FALSE,check.names = TRUE)
```

... วัตถุหรือเวกเตอร์ที่ต้องการนำมารวมเป็นกรอบข้อมูล(แฟ้มข้อมูล)
row.names:เวกเตอร์ที่บอกชื่อแถวของกรอบข้อมูล แต่ไม่ต้องระบุก็ได้
ดูรายละเอียดจาก ?data.frame

ตัวอย่าง2. 8 สร้างกรอบข้อมูล

```
> x<-c(1,2,3,4)
> s<-c("m","m","f","f")
> xs.dat<-data.frame(x,s)
> xs.dat
1 1 m
2 2 m
3 3 f
4 4 f

> attributes(xs.dat)
$names
[1] "x" "s"

$row.names
[1] "1" "2" "3" "4"

$class
[1] "data.frame"
```

ตัวอย่าง2. 9 สร้างกรอบข้อมูลและกำหนดชื่อ row

```
> rname<-c("one","two","three","four")
> xs.dat<-data.frame(x,s,
  row.names = rname)
> xs.dat
       x s
  one  1 m
  two  2 m
  three 3 f
  four  4 f
```

TIP: 1. ตรวจสอบว่าวัตถุมีลักษณะใดได้ด้วยคำสั่ง **attributes(object)**

```
> attributes(xs.dat)
```

2. อ้างอิงสมาชิกในเวกเตอร์ โดยใช้ x[i], x=เวกเตอร์ , i=สมาชิกของ x ตัวที่ i

`> x[3]` หมายถึงสมาชิกตัวที่ 3 ของ x

3. อ้างอิงสมาชิกในกรอบข้อมูล โดยใช้ xs.dat [row,column] , xs.dat คือชื่อกรอบข้อมูล

`> xs.dat [1,]` หมายถึงสมาชิกแถวที่ 1 ในทุกคอลัมน์ของ xs.dat

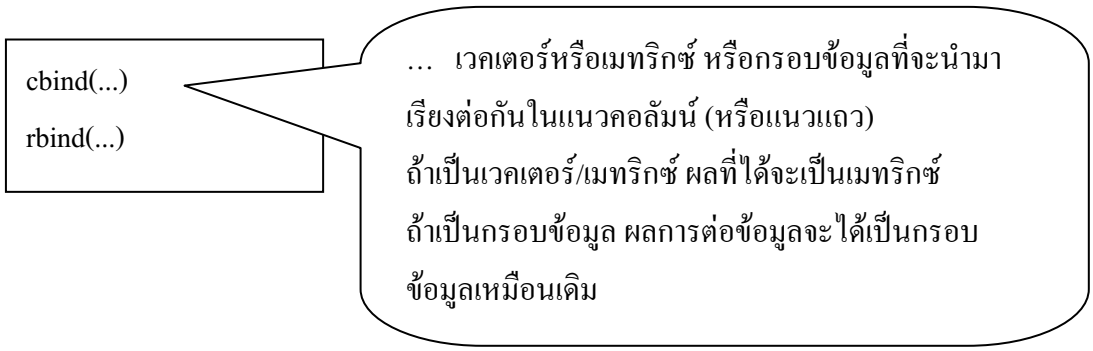
`> xs.dat [ ,2]` หมายถึงสมาชิกทุกแถวในคอลัมน์ 2 ของ xs.dat

`> xs.dat [ 1:4,2]` หมายถึงสมาชิกแถวที่ 1 ถึง 4 ในคอลัมน์ที่ 2 ของ xs.dat

4. เมื่อนำ x, s มาสร้าง dataframe แล้ว ควรลบเวกเตอร์ x, s เดิมทิ้ง โดยใช้คำสั่ง rm(x,s)

2.3.2 สร้างแฟ้มข้อมูลเป็น เมทริกซ์โดยการนำเวกเตอร์มาเรียงต่อกัน

การนำเวคเตอร์มาเรียงต่อกัน ทำได้ 2 แบบ คือ 1) นำตัวแปรมาต่อกัน โดยนำเวคเตอร์มาปะติดกันทางสดมภ์ (combind by column, cbind) และ 2) นำ case ต่อกัน โดยนำเวคเตอร์มาปะติดกันทางแถว (combind by row) คำสั่งที่ใช้ คือ



ตัวอย่าง 2.10 ดูความแตกต่างของฟังก์ชัน c(…),rbind(…), cbind(…)

```
> x<-c(1,2,3,4)
> s<- c("m","m","f","f")
> c(x,s)
[1] "1" "2" "3" "4" "m" "m" "f" "f"      # ผลลัพธ์เป็นเวคเตอร์

> rbind(x,s)
  [,1] [,2] [,3] [,4]
x "1"  "2"  "3"  "4"
s "m"  "m"  "f"  "f"          # ผลลัพธ์เป็นเมทริกซ์ ขนาด 2 x 4
```

```
> xs.mat<-cbind(x,s)
> xs.mat
     x   s
[1,] "1" "m"
[2,] "2" "m"
[3,] "3" "f"
[4,] "4" "f"

> attributes(xs.mat)
$dim
[1] 4 2

$dimnames
$dimnames[[1]]
NULL

$dimnames[[2]]
[1] "x" "s"
```

ผลลัพธ์เป็นเมทริกซ์ ขนาด 4 x 2

แบบฝึกหัด

1. สร้างแฟ้มข้อมูลชื่อ child.dat ที่ประกอบด้วยตัวแปรดังตารางข้างล่างนี้

Id	sex	age	type	status	hour
1	1	10	3	1	3.0
2	1	10	3	1	6.0
3	1	13	1	1	6.0
4	1	16	1	1	6.0
5	2	12	2	2	3.0
6	2	14	3	3	1.0
7	2	12	1	1	6.0
8	2	13	2	2	3.0

sex: 1= male, 2 = female

type (type of work): 1=fishery, 2=fish selection, 3= sea food factory

status (working status): 1= employee, 2= self employ, 3= home helper

2. ตรวจสอบว่าตัวแปรแต่ละตัวเป็นตัวแปรชนิดใด

2.4 การอ่านแฟ้มข้อมูลที่สร้างจากโปรแกมต่างๆ เข้ามาในโปรแกรม R

คุณสมบัติพิเศษอีกอย่างหนึ่งของโปรแกรม **R** คือ สามารถอ่านแฟ้มข้อมูลที่สร้างจากโปรแกรมอื่นได้

2.4.1 แฟ้มข้อมูลที่ใช้สาธิต

ในการสาธิตการใช้ ฟังก์ชันใน **R** เราใช้ Birth weight data ซึ่ง download จาก http://stat-www.berkeley.edu/users/statlabs/lab.html เป็นส่วนใหญ่ โดยข้อมูลดังกล่าวถูกใช้ในการศึกษาน้ำหนักของทารกแรกเกิดกับประวัติหรือภูมิหลังโดยเฉพาะพฤติกรรมการสูบบุหรี่ของมารดา ตัวแปรที่ทำการรวบรวมมี ดังนี้

```
1.  id:         identification number
2.  pluralty:   5= single fetus
3.  outcome:    1= live birth that survived at least 28 days
4.  date:       birth date where 1096=January1,1961
5.  gestation:  length of gestation in days
6.  sex:        infant's sex; 1=male, 2=female, 9=unknown
7.  wt:         birth weight in ounces (999 unknown)
8.  parity:     total number of previous pregnancies including
                fetal deaths and still births, 99=unknown
9.  race:       mother's race; 0-5=white 6=mex 7=black 8=asian 9=mixed
                99=unknown
10. age:        mother's age in years at termination of pregnancy,
                99=unknown
11. ed:         mother's education; 0= less than 8th grade,
                1 = 8th -12th grade - did not graduate,
                2= HS graduate--no other schooling , 3= HS+trade,
                4=HS+some college 5= College graduate, 6&7 Trade
                school HS unclear, 9=unknown
12. ht:         mother's height in inches to the last completed inch,
                99=unknown
13. momwt:      mother prepregnancy wt in pounds, 999=unknown
14. drace:      father's race, coding same as mother's race.
15. dage:       father's age, coding same as mother's age.
16. ded:        father's education, coding same as mother's education.
17. dht:        father's height, coding same as for mother's height
18. dwt:        father's weight coding same as for mother's weight
19. marital:    1=married, 2= legally separated, 3= divorced,
                4=widowed, 5=never married
20. income:     family yearly income in $2500 increments; 0 = under
                2500,1=2500-4999, ..., 8= 102,500-104,999, 9=105,000+,
                98=unknown, 99=not asked
21. smoke:      does mother smoke? 0=never, 1= smokes now, 2=until
                current pregnancy, 3=once did, not now, 9=unknown
22. time:       If mother quit, how long ago? 0=never smoked, 1=still
                smokes,2=during current preg, 3= within 1 yr,
                4 = 1 to 2 years ago,5= 2 to 3 yr ago,6= 3 to 4 yrs
                ago, 7=5 to 9yrs ago, 8=10+yrs ago, 9=quit and don't
                know,98=unknown, 99=not asked
23. number:     number of cigars smoked per day for past and current
                smokers  0=never, 1=1-4, 2=5-9, 3=10-14, 4=15-19,
```

```
5=20-29, 6=30-39, 7=40-60, 8=60+,9=smoke but don't
know, 98=unknown, 99=not asked
```

ในการเตรียมข้อมูลสำหรับบทนี้เราเลือกตัวแปรที่สนใจจำนวน 14 ตัวแปร คือ gestation, sex, wt, parity race ed age marital ht momwt income smoke time และ number และสุ่มมาศึกษา 150 cases เก็บไว้ในแฟ้ม " baby23.txt" กับ " baby23.sav" เพื่อความสะดวกในการใช้แฟ้มข้อมูล ให้ copy แฟ้มข้อมูลทั้งหมดจาก CD ที่เตรียมให้ ไปยัง Rworkplace

2.4.2 การอ่านข้อมูลที่สร้างจากโปรแกรมต่างๆ

หากมีแฟ้มข้อมูลที่จัดเก็บด้วยโปรแกรม SPSS (*.sav), EXCEL (*.xls หรือ *.csv) หรือ text file (*.txt) สามารถอ่านเข้ามาในโปรแกรม R ได้ ด้วยคำสั่ง read.spss, read.table, read.csv, read.dta หรือ read.mtp แล้วแต่ชนิดของโปรแกรม แต่คำสั่งเหล่านี้ อยู่ใน library ชื่อ foreign ซึ่งต้องเปิดใช้ libraryนี้ก่อนที่จะใช้คำสั่งอ่านแฟ้มข้อมูลดังที่กล่าวมา

```
> library (foreign)
```

2.4.2.1 การอ่านข้อมูลจากไฟล์ SPSS (*.sav) เข้ามาใน R โดยใช้ฟังก์ชัน read.spss()

วิธีใช้

```
read.spss("file", use.value.labels=TRUE, to.data.frame=FALSE)
```

file: ชื่อแฟ้มข้อมูลที่ต้องการนำเข้าใน **R** ภายใต้เครื่องหมาย ""

use.value.labels: ระบุให้ใช้คำอธิบายรหัสแทนที่จะเป็นค่ารหัส

to.data.frame: ถ้า = TRUE ระบุว่าให้นำเข้าข้อมูลเป็น Data frame ซึ่งตัวแปรแต่ละตัวจะต้องมีจำนวนสมาชิกเท่ากัน กรณีที่ตัวแปรมีสมาชิกไม่เท่ากัน **R** จะกำหนดให้ตัวแปรทุกตัวมีสมาชิกเท่ากับจำนวนสมาชิกของตัวแปรที่มีสมาชิกมากที่สุด โดยจะให้ค่าที่ขาดหายไปของตัวแปรบางตัวเป็น NA แต่ถ้า to.data.frame = FALSE ข้อมูลจะถูกนำเข้าเป็น list ซึ่งค่อนข้างยุ่งยากในการนำมาใช้

ตัวอย่าง2.1 1 อ่านแฟ้มข้อมูล "baby23.sav" ที่อยู่ใน working directory

```
> library (foreign)
> baby.dat <- read.spss("baby23.sav", use.value.labels=TRUE,
              to.data.frame=T)

> baby.dat[1:2,1:5]
          ID GESTAION SEX  WT PARITY
1 "1"              256   1 117      3
2 "2"              293   1 119      0
```

TIP: 1. **R** จะไม่รู้จักค่าสูญหายจาก SPSS จึงต้องตรวจสอบข้อมูลอีกครั้งหลังจากอ่านเข้ามาใน **R** แล้วจัดการกับค่าสูญหายให้เรียบร้อย

2. ชื่อตัวแปรใน **R** ที่อ่านเข้ามา จาก SPSS จะเป็นอักษรตัวพิมพ์ใหญ่

2.4.2.2 การอ่านข้อมูลจาก text file (*.txt)

แฟ้มข้อมูลที่เป็น text หรือ ASCII ที่เก็บข้อมูลในลักษณะที่คั่นข้อมูลแต่ละตัวด้วย space จะอ่านเข้ามาในโปรแกรม **R** ด้วย คำสั่ง read.table()

วิธีใช้

```
read.table(file, header = FALSE, sep = "", quote = "\"'",
     dec = ".", row.names, col.names, as.is = FALSE,
     na.strings = "NA",  colClasses = NA, nrows = -1,
     skip = 0, check.names = TRUE, fill = !blank.lines.skip,
     strip.white = FALSE, blank.lines.skip = TRUE,comment.char = "#")
```

หรือใช้คำสั่งที่ระบุตัวเลือกเท่าที่จำเป็น ดังนี้

```
read.table("file", header = FALSE, sep = " ", fill = TRUE,
      blank.lines.skip = TRUE)
```

File: ชื่อแฟ้มข้อมูลที่ต้องการอ่าน

header =TRUE บอกให้ **R** อ่านชื่อตัวแปรใน file เดิมเข้ามาด้วย

fill = TRUE ใช้ในกรณีที่แต่ละตัวแปรใน file มีจำนวนข้อมูลไม่เท่ากัน **R** จะให้ค่าแถวที่ไม่มีข้อมูลเป็น NA ไม่เช่นนั้นจะเกิด error

blank.lines.skip = TRUE ให้ข้ามแถวที่ไม่มีข้อมูล

ตัวอย่าง 2.12 อ่านแฟ้ม `birthwei23.txt` ซึ่งอยู่ใน working directory แล้ว

```
> baby23.dat<-read.table("baby23.txt ", header=TRUE,fill=TRUE)
> attach(baby23.dat)
```

```
> baby23.dat[1:2,1:10]
  gestation sex  wt parity race ed age marital ht momwt
1       256   1 117      3    0  2  37       1 65   132
2       293   1 119      0    0  4  23       1 65   127
```

2.4.2.3 การอ่านข้อมูลจาก EXCEL (*.xls หรือ *.csv)

ตามปกติแฟ้มข้อมูลที่สร้างใน EXCEL จะมีนามสกุลเป็น .xls แต่ถ้าเป็นนามสกุลนี้ จะไม่สามารถอ่านเข้าใน **R** ได้ ดังนั้น เพื่อให้ง่ายต่อการอ่านแฟ้มข้อมูลจาก EXCEL เข้าใน **R** มีวิธีทำ 2 แบบ คือ

1) บันทึกแฟ้มข้อมูลในEXCEL เป็น *.txt แล้วใช้คำสั่งอ่านแฟ้มเช่นเดียวกับการอ่าน text file ใน หัวข้อ 2.4.2.2

2) บันทึกข้อมูลในEXCEL เป็น *.csv (comma delimeted) แล้วอ่านด้วยคำสั่ง read.csv()

วิธีใช้

```
read.csv("file", header = TRUE, sep = ",",  fill = TRUE, ...)
```

ตัวอย่าง 2.13

```
> read.csv("matched.csv", header = TRUE, sep = ",",  fill = TRUE)
```

2.4.2.4 การอ่านข้อมูลจากโปรแกรม STATA

แฟ้มข้อมูลที่สร้างขึ้นโดยใช้โปรแกรม STATA สามารถอ่านเข้ามาใช้ในโปรแกรม **R** ด้วย คำสั่ง read.dta()

วิธีใช้

```
read.dta(file, convert.dates = TRUE, tz = NULL,
          convert.factors = TRUE, missing.type = FALSE,
            convert.underscore=TRUE, warn.missing.labels=TRUE)
```

file: ชื่อแฟ้มข้อมูลที่ต้องการอ่าน

convert.dates: เปลี่ยน วันที่ ใน Stata เป็นวันที่ ใน **R**

convert.factors: ใช้คำอธิบายรหัสใน STATA เป็น factors ใน **R** (version 6.0or later)

รายละเอียดดูใน ?read.dta

โครง

ตัวอย่าง 2.14 อ่านแฟ้มข้อมูล choles.dta

```
> choles.sta<-read.dta("choles.dta")
```

2.4.2.5 การอ่านแฟ้มข้อมูลจาก Minitab

แฟ้มข้อมูลที่สร้างขึ้นโดยใช้โปรแกรม MINITAB และบันทึกเป็น Minitab portable file (*.mtp) สามารถอ่านเข้ามาใช้ในโปรแกรม **R** ด้วย คำสั่ง read.mtp

วิธีใช้

```
read.mtp("file")
```

TIP:

1. เมื่ออ่านแฟ้มข้อมูลเข้ามาใน โปรแกรม **R** แล้ว ควร attach แฟ้มข้อมูลดังกล่าวเพื่อให้ สามารถระบุชื่อตัวแปรจากแฟ้มนี้โดยตรงได้ เช่น gestation ไม่เช่นนั้นต้องระบุชื่อกรอบข้อมูลตามด้วย $ และชื่อตัวแปร ดังเช่น baby23.dat$gestation
2. กรณีที่ชื่อแฟ้มข้อมูลที่สร้างใหม่มีชื่อเหมือนกับตัวแปรที่มีอยู่ใน **R** ควรลบตัวแปรต่างๆ เหล่านั้นออกจาก Rworkplace ก่อน เพื่อกันการสับสน เพราะ **R** จะเรียกตัวแปรชื่อเดียวกันที่อยู่นอกแฟ้มข้อมูลมาใช้

```
> attach(baby23.dat)
```

แบบฝึกหัด

1. จากข้อมูลแบบฝึกหัด ในหน้า 36 ให้สร้างแฟ้มข้อมูลด้วยโปรแกรม EXCEL, แล้วอ่านแฟ้มข้อมูลที่สร้างนั้น เข้ามาใน **R**
2. Label ตัวแปรเชิงคุณภาพในแฟ้มข้อมูลที่สร้างขึ้น

2.5 การเก็บบันทึกผลการประมวลผลจากโปรแกรม R

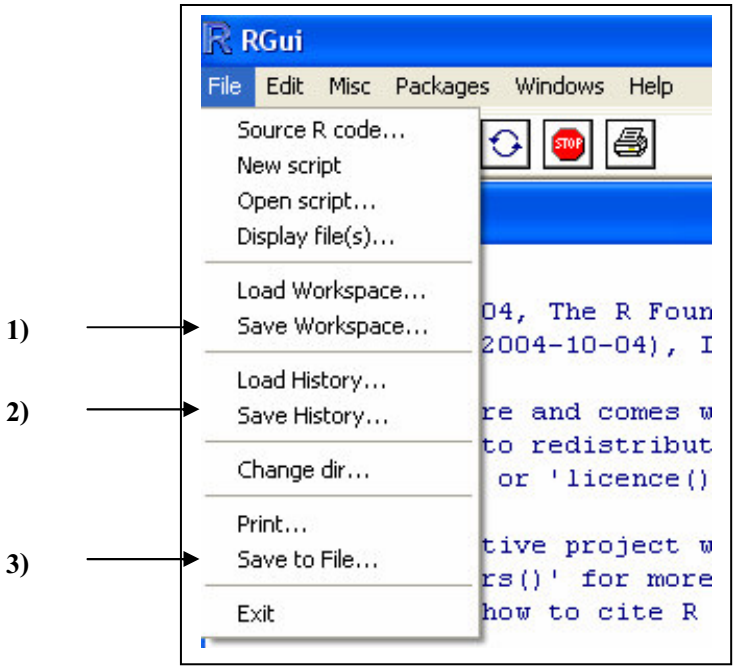
การบันทึกผลจากการประมวลผลด้วยโปรแกรม **R** อาจแบ่งเป็น 4 อย่าง คือ

1) Save workspace บันทึกคำสั่ง และวัตถุที่เป็นผลจากการประมวลผลทุกอย่างที่เกิดขึ้นจากการประมวลผลแต่ละครั้ง (แต่ละsession)

2) Save History บันทึกประวัติการใช้คำสั่งที่เคยทำมาแล้วทั้งหมดเท่าที่เคยสั่งประมวลผล โดยไม่รวมผลลัพธ์ลงในแฟ้ม ซึ่งสามารถเปิดด้วย notepad เพื่อแก้ไขแล้วนำมาประมวลผลใหม่ โดยใช้คำสั่ง source("file", echo = T) ถ้าให้ echo=F จะไม่ปรากฏผลลัพธ์จากคำสั่งทางหน้าจอ นอกจากนี้ ยังสามารถ load เข้ามาใน **R** เพื่อเรียกคำสั่งมาใช้ใหม่ได้ด้วยการกดลูกศรบนแป้นพิมพ์ (ขึ้น หรือลง)

3) Save to file ถ้าต้องการบันทึกผลการวิเคราะห์ข้อมูลเพื่อนำไปประกอบรายงาน ทำได้โดยการใช้เมนู save to file บันทึกได้เป็น text file และสามารถเปิดกับโปรแกรม notepad หรือ Microsoft word รวมทั้งสามารถแก้ไข ตัดต่อรายงานเพิ่มเติมได้

วิธีที่ 1) ถึง 3) สามารถใช้จากเมนู ดังรูป 2.1



รูป 2.1

4) บันทึกวัตถุที่ต้องการเก็บเป็น text file โดยใช้ฟังก์ชัน write.table

ผลลัพธ์ที่ได้จากการประมวลผลด้วยโปรแกรม **R** จะถูกเก็บเป็นวัตถุใน working directory ดังนั้น ถ้าต้องการบันทึกผลลัพธ์ดังกล่าวเพื่อนำไปใช้ภายหลัง เช่น สร้างกรอบข้อมูลแล้ว ต้องการเก็บเป็น text file (*.txt) หรือ (*.csv) เพื่อนำไปใช้กับโปรแกรมอื่นๆ ทำได้ ดังนี้

วิธีใช้

```
write.table(x, file = "", append = FALSE, quote = TRUE, sep = " ",
            eol = "\n",  na = "NA", dec = ".", row.names = TRUE,
            col.names = TRUE, qmethod = c("escape", "double"))
```

หรือใช้คำสั่งเท่าที่จำเป็น ดังนี้

```
write.table(x, file = "file.txt",  sep = " ", na = "NA" ,row.names = F)
```

```
write.table(x, file = "file.csv ", sep = ",", row.names = F)
```

x: วัตถุ (หรือผลลัพธ์)ที่ต้องการบันทึก ส่วนใหญ่จะเป็นกรอบข้อมูล แต่ถ้าไม่ใช่ R จะถือเสมือน x เป็นกรอบข้อมูล

file: ชื่อแฟ้มข้อมูลที่จะเก็บผลลัพธ์ ให้ใส่นามสกุลตามชนิดของโปรแกรมที่ต้องการเก็บบันทึกข้อมูล เช่น เป็น text file ใช้ .txt เป็น EXCEL ใช้ .cvs

sep: ตัวแบ่งค่าของแต่ละตัวแปร ใส่ไว้ภายใน " " ถ้าบันทึกเป็น *.txt ควรใช้ blank แต่ถ้าบันทึกใน EXCEL ควรใช้ comma (,)

row.names = F ระบุไม่ต้องพิมพ์ชื่อแถว

ตัวอย่าง 2.15 บันทึกข้อมูลที่เลือกเฉพาะตัวแปรที่1 ถึง 3, 5-7, 11 จากแฟ้ม baby23.dat ไว้ในกรอบข้อมูล baby7.txt แล้วอ่านกลับมาใน **R**

```
> baby7.dat<-data.frame(gestation,sex,wt,race,ed, age,smoke)
> baby7.dat[1:3,]
  gestation sex  wt race ed age smoke
1       256   1 117    0  2  37     1
2       293   1 119    0  4  23     3
3       291   1 148    4  2  21     0

> write.table(baby7.dat,file="baby7.txt",na="NA",sep=" ",row.names=F)
```

ผลที่ได้ คือ text file ชื่อ baby7.txt เก็บใน working directory ของ **R** และเมื่อต้องการอ่านเข้ามาใช้งานใน **R** อีกครั้ง สามารถทำได้ด้วยคำสั่ง read.table(…)

```
> read.table("baby7.txt ", header = TRUE)
```

ตัวอย่าง2.16 บันทึกข้อมูลจากแฟ้มชื่อ baby7.dat ไปอยู่ในโปรแกรม EXCEL ให้ชื่อแฟ้ม baby7.cvs แล้วอ่านกลับมาใน **R**

```
> write.table(baby7.dat,file = "baby7.csv",row.names =F, sep = ",")
> read.table("baby7.csv", header = TRUE, sep = ",")
```

แบบฝึกหัด

สร้างแฟ้มข้อมูล โดยตัดเฉพาะตัวแปรที่ 1 ถึง 5 จากแฟ้ม baby23.dat แล้ว save ไว้ใน baby5.txt, baby5.csv และอ่านกลับเข้ามาใน **R**

2.6 การปรับเปลี่ยนข้อมูล (Data manipulations)

ข้อมูลที่จะนำมาวิเคราะห์ นั้น บางส่วนพร้อมที่จะนำไปวิเคราะห์ได้เลย แต่บางส่วนจะต้องนำมาปรับเปลี่ยนเพื่อให้อยู่ในรูปแบบที่เหมาะสมกับการวิเคราะห์ ดังนั้น การปรับเปลี่ยนข้อมูล (data manipulation) จึงเป็นเรื่องจำเป็น ลักษณะการปรับเปลี่ยนข้อมูลที่จะกล่าวถึงในบทนี้ มีดังนี้

1) การจัดการกับค่าสูญหาย (Missing values)
2) คำนวณค่า (Data operations)
3) การปรับเปลี่ยนค่าของตัวแปรเชิงคุณภาพ (recoding)
4) การจัดกลุ่มตัวแปรเชิงปริมาณ (Grouping)
5) การอธิบายรหัสที่ใช้แทนกลุ่มต่างๆ ของตัวแปร (Value labeling)

2.6.1 การจัดการกับค่าสูญหาย

R จะใช้ NA แทนค่าสูญหาย ซึ่งค่า NA นี้ บางครั้งจะก่อให้เกิดความยุ่งยากในการประมวลผล เพราะบางคำสั่งจะให้ผลเป็น NA ถ้าข้อมูลมีค่า NA รวมอยู่ การจัดการกับค่า NA เหล่านี้ทำได้โดยใส่ argument " na.rm=TRUE" เพื่อให้ตัดค่า NA ออกก่อนประมวลผล แต่บางคำสั่งจะไม่มี argument นี้ให้เลือก ดังนั้นหากจำเป็นต้องตัดค่า NA ออกจากข้อมูลทั้งชุด ทำได้โดยใช้คำสั่ง na.omit(object)

ตัวอย่าง 2.17 ให้ตัดข้อมูลทุกแถวที่ตัวแปรใดๆมีค่า NA

```
> baby.nna.dat<-na.omit(baby23.dat) # จะได้แฟ้มข้อมูลที่ไม่มีค่า NA เหลืออยู่เลย
```

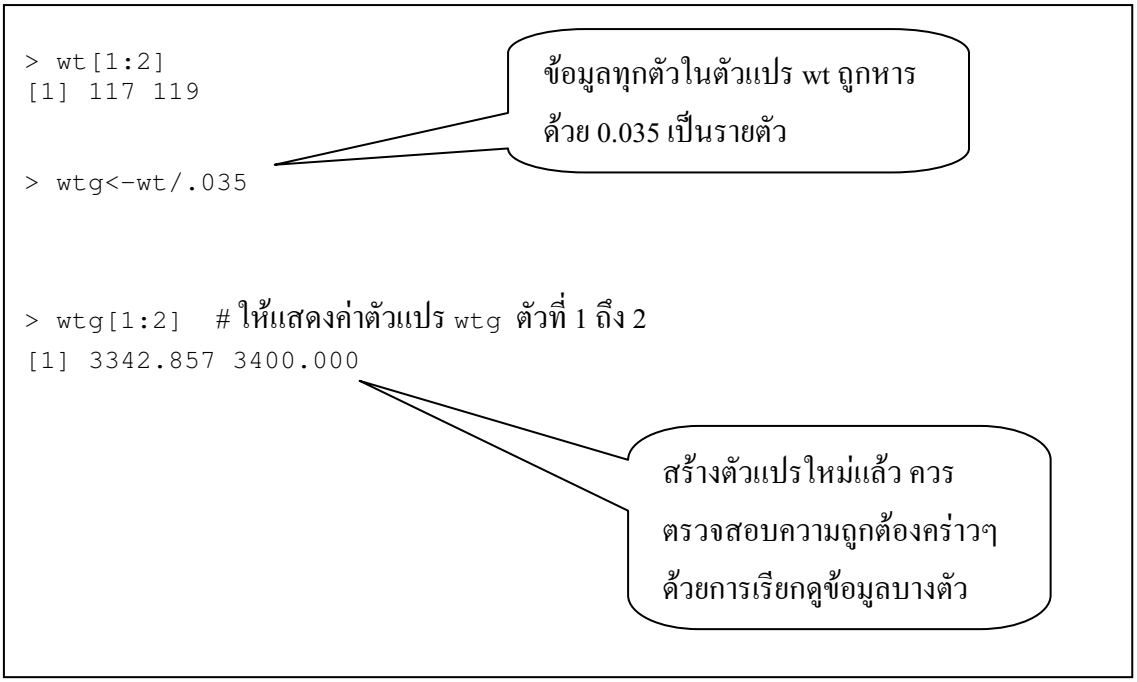
2.6.2 การคำนวณ

การทำงานของ **R** ในการคำนวณทางคณิตศาสตร์ (เช่น บวก ลบ คูณ หาร ยกกำลัง ฯลฯ) โดยใช้ operators หรือ function ที่มีลักษณะการทำงานดังตัวอย่าง

Operators		functions	
บวก +	หาร /	ถอดรากที่ 2 ของ x	sqrt (x)
ลบ -	ยกกำลัง ^	ลอกการิทึม x	log(x) (Natural log)
คูณ *		e ยกกำลัง x	exp(x)

ค่าตัวแปร (เวกเตอร์) ในพจน์ทางคณิตศาสตร์ (expression) จะถูกนำมาคำนวณตัวต่อตัว

ตัวอย่าง 2.18 น้ำหนักแรกเกิดของทารก (wt) เดิมมีหน่วยเป็น ounces แต่เราต้องการ wt ที่มีหน่วยเป็นกรัม (1 g =.035 ounces) เพื่อจะนำไปเปรียบเทียบกับเกณฑ์มาตรฐานที่วัดว่าเด็กอยู่ในกลุ่ม low birth weight (<2500 g) หรือไม่



ตัวอย่าง 2.19 ต้องการคำนวณค่า Body mass index, BMI = weight (kg) / height^2 (m^2) ของมารดา

```
> wtkg<-momwt/2.2    # เปลี่ยนหน่วยจากปอนด์เป็นกิโลกรัม
>  wtkg[1:3]
[1] 60.00000 57.72727 52.27273

> htm<-ht*2.5/100  # เปลี่ยนหน่วยจากนิ้วเป็นเมตร
> htm[1:3]
[1] 1.625 1.625 1.575

> BMI<-wtkg/htm^2
> BMI[1:3]
[1] 22.72189 21.86122 21.07240
```

ค่าใน wtkg จะถูกหารด้วย htm^2 เป็นรายคู่ ดังนี้

wtkg	htm^2	BMI
60.00	1.625^2	60.00/1.625^2 = 22.722
57.73	1.625^2	57.73/1.625^2 = 21.861

TIP:

ถ้าต้องการปัดจุดทศนิยม BMI ให้เหลือเพียง 2 ตำแหน่ง

หรือปัดให้เป็นจำนวนเต็ม ใช้ฟังก์ชัน round(x,digit)

```
> BMI<-round(BMI,2)
> BMI[1:2]
[1] 22.72 21.86
```

round(x, digits = 0)

x คือ เวคเตอร์ หรือ data frame

digits คือจำนวนตำแหน่งทศนิยม

2.6.2.2 การคำนวณค่าตัวแปร 2 ตัว ที่มีขนาดไม่เท่ากัน

ในการนำตัวแปร 2 ตัว (หรือมากกว่า) มาบวก ลบ คูณ หรือหารกันนั้น ความยาว (จำนวนข้อมูลในตัวแปร) ของตัวแปรทั้ง 2 ตัว อาจจะไม่เท่ากันก็ได้ ผลลัพธ์จะเป็นตัวแปรที่มีความยาวเท่ากับตัวแปรที่ยาวที่สุด

ตัวอย่าง 2.20

```
> y<-c(4,6,9,12,14,15)
> x<-c(1,2,3)
> z<-y/x
> z
[1]  4  3  3 12  7  5
```

ลักษณะการทำงาน

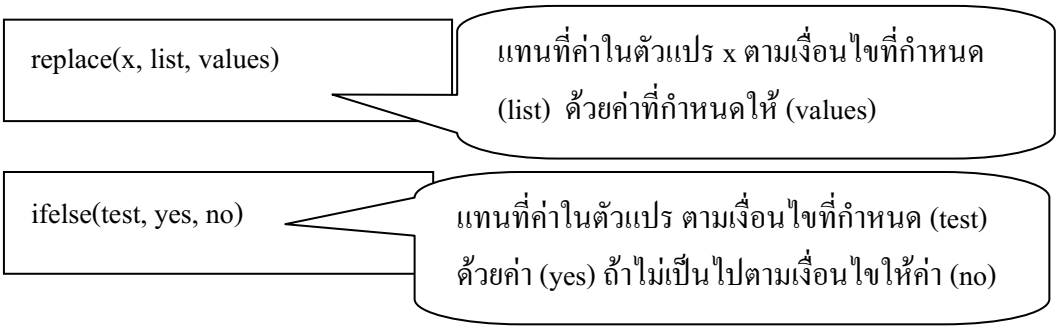
y	x	z
4	1	4
6	2	3
9	3	3
12	1	12
14	2	7
15	3	5

ค่าของ x (ตัวที่มีความยาวน้อยกว่า) จะถูกนำมาวนใช้อีกรอบ

ตัวอย่าง 2.21 บวกตัวแปร 2 ตัวที่มีความยาวไม่เท่ากัน

```
> x <-c(1,2,3)
> y<-c(4,5,6,7)
> xy<-x+y

Warning message:
longer object length
        is not a multiple of shorter object length in: x + y
> xy
[1] 5 7 9 8
```

2.6.3 การเปลี่ยนค่าของตัวแปรเชิงคุณภาพ (Recoding)



ตัวอย่าง 2.22 แฟ้ม `baby23.dat` มีตัวแปร smoke (0=never sm, 1= current sm, 2=sm at pregnancy, 3= Ever sm, NA=unknown) ต้องการเปลี่ยนค่า smoke = 3 ให้เป็น 0 ทำได้ 2 วิธี ดังนี้

```
> attach(baby23.dat)
> table(smoke) # ตรวจสอบตัวแปรก่อนเปลี่ยนรหัส
smoke
 0  1  2  3
56 68 15  9
```

วิธีที่ 1 ใช้คำสั่ง replace(x, list, values)

```
> smoke<-replace(smoke,smoke==3,0)
> table(smoke)  # ตรวจสอบจำนวนหลังเปลี่ยน
smoke
 0  1  2
65 68 15
```

ถ้า smoke= =3 (ใช้ = = เพราะเป็น logic) แทนที่ตัวแปร smoke ด้วยค่า 0 ถ้าไม่เป็นไปตามเงื่อนไข ยังคงเป็นค่าเดิมของตัวแปร smoke (0, 1,2)

มาจาก 9+56 ตามเงื่อนไข

วิธีที่ 2 ใช้คำสั่ง ifelse(test, yes, no)

```
> rm(smoke)
> smoke<-ifelse(smoke==3, 0, smoke)
> table(smoke)    # ตรวจสอบจำนวนหลังเปลี่ยน
smoke
 0  1  2
65 68 15
```

ถ้า smoke=3 เปลี่ยนค่าเป็น 0 แต่ถ้าไม่เป็นไปตามเงื่อนไข ยังคงเป็นค่าเดิมของตัวแปร smoke (0, 1,2)

ตัวอย่าง 2.23 ต้องการจัดกลุ่มตัวแปร smoke ให้เหลือเพียง 2 กลุ่ม คือ 0= non smoker (เดิมมีค่า 0,2,3) และ 1= smoker

```
> rm(smoke)
> smoke.new<-replace(smoke,smoke!=1,0)        # != หมายถึง not equal
> table(smoke.new)
Smoke.new
 0  1
80 68

>rm(smoke.new)
> smoke.new<-ifelse(smoke==1, smoke,0)
> table(smoke.new)
smoke.new
 0  1
80 68
```

TIP: ในการเปลี่ยนค่าโดยแทนที่ค่าใหม่ลงในตัวแปรเดิม อาจทำให้สับสนหรือเข้าใจผิดว่าตัวแปร smoke ในแฟ้ม baby23.dat ได้เปลี่ยน 3 เป็น 0 แล้ว แต่แท้จริงแล้ว smoke ที่ปรับเปลี่ยนแล้วเป็นวัตถุใหม่ที่อยู่ใน working directory ไม่ใช่ baby23.dat$smoke ซึ่งเป็นตัวแปร smoke ใน baby23.dat

ถ้าจะเก็บค่า smoke.new ที่เปลี่ยนแปลงแล้วเข้าแทนที่ในไฟล์baby23.dat ให้ใช้คำสั่ง

```
> detach(baby23.dat)
> baby23.dat$smoke<-smoke.new
> attach(baby23.dat)
```

2.6.4 การจัดกลุ่มตัวแปรเชิงปริมาณ (Grouping)

การเปลี่ยนข้อมูลเชิงปริมาณเป็นข้อมูลเชิง คุณภาพ อาจทำได้ 2 วิธี ดังนี้

2.6.4.1 กำหนดจุดตัดของแต่ละกลุ่มในเวกเตอร์ แล้วใช้คำสั่ง cut()

วิธีใช้

```
cut(x, breaks, labels = NULL,include.lowest = FALSE, right = TRUE, dig.lab = 3, ...)
```

x: เป็นเวกเตอร์ที่เป็นค่าตัวแปร

bre: เวกเตอร์ที่บอกช่วงของกลุ่มที่จะแบ่งข้อมูล

right: เงื่อนไขที่ระบุการรวมจุดปลายทางขวามือ หรือไม่ (ค่าปกติจะรวม)

labels: อธิบายชื่อของกลุ่มข้อมูลที่แบ่งแล้ว

ตัวอย่าง 2.24 ต้องการแบ่งช่วงอายุออกเป็น 2 ช่วงคือ อายุ 0-<20 ปี และ อายุ 20-<45 ปี โดยกำหนดตัวแปร bre เป็นจุดตัด และใช้คำสั่ง cut แบ่งกลุ่ม และสรุปข้อมูลที่แบ่งกลุ่มแล้วในตาราง

```
> summary(age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  17.00   23.00   26.00   26.99   30.00   42.00

> bre <- c(0,20,45)
```

กำหนดจุดตัดเป็น 2 ช่วงคือ [0 - 20) และ [20 - 45) ค่าปกติของคำสั่ง cut จะรวมค่า upper limit ไว้ในช่วงนั้นด้วย หากต้องการเปลี่ยนให้เพิ่ม option ของคำสั่ง cut (right=F หรือ left=T)

```
> table (cut (age, bre, right= F))
 [0,20) [20,45)
      8     142
```

#เพิ่มคำอธิบายความหมายของกลุ่ม

```
> table(cut (age, bre,right= F, labels = c("0-<20", "20-<45")))

 0-<20 20-<45
     8    142
> agegr <-cut(age,bre,right=F,labels=c("0-<20","20-<45"))
> table(agegr)
agegr
 0-<20 20-<45
     8    142
```

2.6.4.2 ใช้คำสั่ง ifelse(test, yes, no)

ตัวอย่าง 2.25 ต้องการจัดกลุ่มตัวแปร gestation (วัน) ให้เป็น 3 กลุ่ม คือ

preterm: < 37 weeks; at term: 37- 42 weeks; post term: 42+

```
> gesweek<-gestation/7 # เปลี่ยนหน่วยจากวันเป็นสัปดาห์
> summary(gesweek)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  32.00   39.00   40.00   39.85   41.14   45.57    1.00
> ges1<-ifelse(gesweek<37,1,0)
> ges2<-ifelse(gesweek>=37&gesweek<=42,2,0)
> ges3<-ifelse(gesweek>42,3,0)
> term<-ges1+ges2+ges3
> table(term)
term
  1   2   3
 12 126  11
```

2.6.5 การอธิบายความหมายของรหัส (levels)

ตัวแปรที่จะนำมาอธิบายรหัส ต้องมีสถานะเป็น factor

วิธีใช้

ตัวอย่าง 2.26 ต้องการอธิบายค่าจากตัวแปร term จากตัวอย่างข้างบน

```
> term.f<- factor(term)
 levels(term.f)<- c("preterm","at-term","post-term")

> table(term.f)
Term.f
  preterm   at-term post-term
       12       126        11
```

แบบฝึกหัด

ให้ label ค่าของ smoke เก็บไว้ในชื่อ smoke.f (0 : Never sm, 1: Current-sm, 2: SM at pregnant, และ 3: Ever SM) เพื่อไว้ใช้ต่อไป

2.7 การเลือกข้อมูล และการสุ่มตัวอย่าง

2.7.1 การเลือกข้อมูลบางส่วน (Subset)

ในการเลือกข้อมูลมาบางส่วนตามเงื่อนไขที่ต้องการ สามารถใช้คำสั่ง subset**(…)**

วิธีใช้

```
subset(x, ...)
    ## Default S3 method:
    subset(x, subset, ...)
    ## S3 method for class 'data.frame':
    subset(x, subset, select, drop = FALSE, ...)
```

x: กรอบข้อมูลที่ต้องการนำมาเลือกข้อมูลบางส่วน

subset: ระบุเงื่อนไขที่ต้องการให้เลือก

select: ระบุว่าจะเลือก column ใดของกรอบข้อมูล หรือระบุเป็นชื่อตัวแปรก็ได้

ตัวอย่าง 2.27 subset ข้อมูลจากไฟล์ baby23.dat มาเก็บใน wt.dat โดยเลือกเฉพาะตัวแปร wt และ smoke เมื่อ smoke=1

```
> wt.dat <-subset(baby23.dat,smoke==1,select=c(wt,smoke))
> summary(wt.dat)

 wt              smoke
 Min.   : 86.0   Min.   :1
 1st Qu.:103.5   1st Qu.:1
 Median :112.0   Median :1
 Mean   :113.4   Mean   :1
 3rd Qu.:122.5   3rd Qu.:1
 Max.   :154.0   Max.   :1
 NA's   :  1.0
```

ตัวอย่าง 2.28 subset ข้อมูลจากไฟล์ baby23.dat มาเก็บใน wt.sub ถ้า smoke=1 โดยเอาทุกตัวแปรใน baby23.dat

```
> wtsub<-subset(baby23.dat,smoke==1)
```

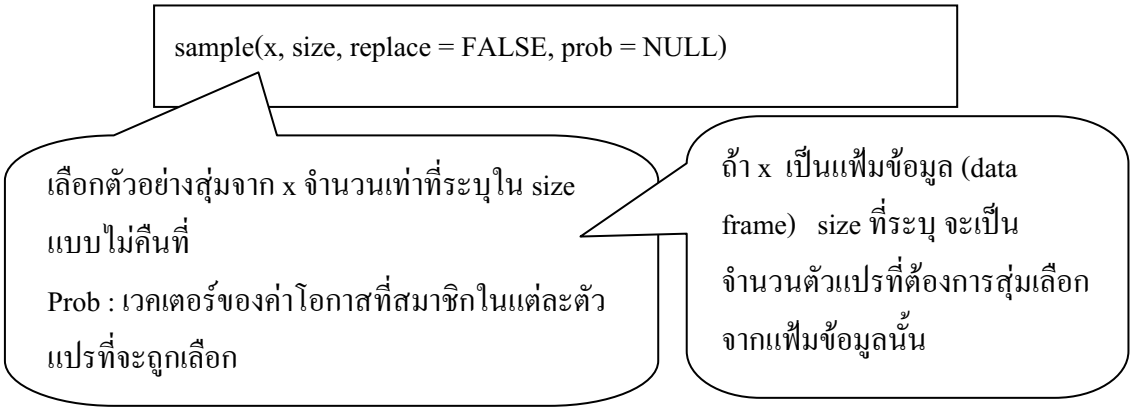
ซึ่งจะได้ผลเหมือนกับคำสั่งข้างล่างนี้

```
> bwts<- baby23.dat[smoke==1,]
```

2.7.2 การสุ่มตัวอย่าง (Random Sampling)

การสุ่มตัวอย่างในที่นี้ จะหมายถึงการสุ่มหน่วยตัวอย่างมาบางส่วนมาโดยที่ทุกหน่วยมีโอกาสจะถูกเลือกเท่ากัน (Equally likely random sampling) โดยเลือกแบบคืนที่ (with replacement) หรือไม่คืนที่ (without replacement) ก็ได้

วิธีใช้



ตัวอย่าง 2.29 เลือกตัวอย่างสุ่ม ขนาด 10 หน่วย จากตัวแปร wt แบบไม่คืนที่

```
 > sample(wt, 10, replace = FALSE)
```

ตัวอย่าง 2.30 เลือกตัวแปร 3 ตัวอย่างสุ่ม จากแฟ้ม baby23.dat ผลที่ได้แต่ละครั้งจะมีชุดตัวแปร 3 ตัวที่ไม่เหมือนกัน

```
> sample(baby23.dat,3, replace = T)
```

ตัวอย่าง 2.31 เลือกตัวอย่างสุ่ม ขนาด 10 หน่วย จากตัวแปร wt ที่ smoke=1

```
> sample(wt[smoke==1], 10, replace = FALSE)
```

TIP:

หากการเลือกตัวอย่างในครั้งต่อไป ต้องการให้ ได้ตัวอย่างชุดเดิม ทำได้โดยการเพิ่มคำสั่ง set.seed( ) คู่กับคำสั่ง sample ดังนี้

```
> set.seed(123456789)
> sample(wt,10)
 [1] 126 109  98 128 104  96 117 143 120 174
```

ตัวอย่าง 2.32 ในกรณีที่ต้องการสุ่มตัวอย่างที่มีตัวแปรทุกตัวในแฟ้มข้อมูล การเลือกตัวอย่าง จะซับซ้อนขึ้น

```
> i<-seq(1:150)
> s2<-sample(i,50)
> baby50.dat<- baby23.dat[s2,]
> length(baby50.dat$wt)
```

- สร้างตัวแปร i เสมือนติดฉลากให้ข้อมูลในแฟ้มตัวที่ 1 ถึง 150
- เลือกตัวอย่างสุ่มจากฉลาก (ตัวแปร i)มา 50 ตัวเก็บในชื่อ s2
- เลือกตัวอย่างจากแฟ้ม baby23.dat โดยเลือกเฉพาะตัวที่ตรงกับฉลากที่เลือกไว้ใน s2
- ผลที่ได้ คือตัวอย่างขนาด50 ที่ประกอบด้วยทุกตัวแปรในแฟ้มข้อมูล baby23.dat

แบบฝึกหัด

1. ให้ subset ข้อมูลจาก baby23.dat มาเก็บในไฟล์ wt3.dat โดยเลือกเฉพาะรายที่ wt < 85.7 ตัวแปรที่ต้องการ คือ wt, smoke และ parity
2. ให้ตรวจสอบผลที่ได้ว่าเป็นไปตามเงื่อนไขที่ต้องการหรือไม่
3. จงแบ่งช่วงอายุของมารดาออกเป็น 4 ช่วงคือ 15-20 35-26 25-21 และตั้งแต่ 36ปีขึ้นไป โดยกำหนดตัวแปร bre เป็นจุดตัด ใช้คำสั่ง cut แบ่งกลุ่ม และสรุปข้อมูลที่แบ่งกลุ่มแล้วในตาราง

บทที่ 3

การสรุปข้อมูล (Data Summary)

การสรุปข้อมูลในบทนี้ จะกล่าวถึงรายละเอียดต่อไปนี้

1) สรุปคำสั่งใน **R** สำหรับการสรุปข้อมูล

2) การสรุปข้อมูลด้วยค่าสถิติพื้นฐานต่างๆ

3) การสรุปข้อมูลด้วยตาราง

3.1 คำสั่งใน R สำหรับการสรุปข้อมูล

คำสั่ง	คำอธิบาย
summary( )	สรุปค่าสถิติพื้นฐานของตัวแปร
mean()	คำนวณค่า mean
median()	คำนวณค่า median
max()	คำนวณค่า maximum
min()	คำนวณค่า minimum
round()	ปัดตำแหน่งทศนิยมของเวกเตอร์
quantile( )	หาค่า quartile, decile หรือ percentile
var( )	คำนวณค่า variance หรือ covariance
sd( )	คำนวณค่าstandard deviation
tapply ( )	ทำตารางสรุปข้อมูลเชิงปริมาณจำแนกตามตัวแปรเชิงกลุ่ม
by()	
table()	สร้างตารางสรุปข้อมูลเชิงคุณภาพ
ฟังก์ชันที่เขียนขึ้นใหม่ ในชุด statcan	
pct.oneway()	คำนวณร้อยละของตารางทางเดียว
twoway.tab()	คำนวณค่าร้อยละและทดสอบ chi-squares test
ftable()	หาค่า quantile ตำแหน่งที่ต้องการ
freq.table()	สร้างตารางสำหรับตัวแปรเชิงปริมาณ

3.2 สรุปข้อมูลด้วยค่าสถิติพื้นฐาน (Typical values)

3.2.1 การสรุปข้อมูลด้วยค่าสถิติพื้นฐานโดยใช้ฟังก์ชัน summary(…)

```
summary(object, ...)


     ## S3 method for class 'factor':
summary(object, maxsum = 100, ...)
```

object: วัตถุที่ต้องการนำมาสรุปค่าสถิติพื้นฐานต่างๆ อาจจะเป็นเวกเตอร์ กรอบข้อมูล แฟกเตอร์ หรือ เมทริกซ์ก็ได้

maxsum: จำนวนเต็มบอกจำนวนกลุ่มของ แฟกเตอร์ ที่ต้องการให้แสดงในตาราง

รายละเอียดดูใน ?summary

ตัวอย่าง 3.1 สรุปข้อมูลทุกตัวในแฟ้ม baby23.dat ด้วยคำสั่ง summary()

```
> attach(baby23.dat)
> summary(baby23.dat[,1:3])
   gestation            sex                wt
 Min.   :224.0          Min.   :1          Min.   : 71.0
 1st Qu.:273.0          1st Qu.:1          1st Qu.:106.0
 Median :280.0          Median :1          Median :117.0
 Mean   :279.0          Mean   :1          Mean   :118.6
 3rd Qu.:288.0          3rd Qu.:1          3rd Qu.:129.0
 Max.   :319.0          Max.   :1          Max.   :174.0
 NA's   :1.0                               NA's   :  3.0
```

ตัวอย่าง 3.2 summary ตัวแปรเชิงปริมาณ (เช่น wt)

```
> summary(wt)
   Min. 1st Qu.  Median    Mean  3rd Qu.    Max.    NA's
   71.0  106.0   117.0   118.6   129.0    174.0     3.0
```

ตัวอย่าง3.3 summary ตัวแปรเชิงคุณภาพ (เช่น smoke)

```
> smoke.f <-factor(smoke)
> summary(smoke.f)
   0    1    2    3 NA's
  56   68   15    9    2
> summary(smoke.f,maxsum=4)
      1       0   (Other)    NA's
     68      56        24       2
```

ทำตัวแปร smoke ให้เป็น factor เก็บไว้ในชื่อ smoke.f

smoke.f เดิมมี 5 กลุ่ม แต่เราระบุ mxsum=4 จึงจัดกลุ่มในตารางให้มีเพียง 4 กลุ่ม

3.2.2 ค่าสถิติบอกตำแหน่ง

```
mean(x, trim = 0, na.rm = FALSE, ...)
median(x, na.rm = FALSE)
max(..., na.rm=FALSE)
min(..., na.rm=FALSE)
```

x: คือ เวกเตอร์ หรือ ตัวแปรเชิงปริมาณ, ถ้าเป็น data frame จะคำนวณค่าเฉลี่ยของตัวแปรทุกตัวในกรอบข้อมูลนั้น แต่ จะใช้ฟังก์ชัน median,max, min กับกรอบข้อมูลไม่ได้

na.rm = FALSE ระบุให้รวมค่าสูญหายในการคำนวณ mean, median, max, min ซึ่งผลลัพธ์จะได้เป็นค่าสูญหาย ดังนั้น จึงควรกำหนด na.rm = TRUE

trim: กำหนดสัดส่วนของข้อมูลที่ต้องการตัดออก (จากค่าต่ำสุด และค่าสูงสุด) ใช้ในกรณีที่มีค่าสุดโต่ง กำหนดค่า trim ได้ในช่วง 0-0.5

ดูรายละเอียดใน ?mean ?median, ?max, ?min

ตัวอย่าง 3.4 ใช้ข้อมูลจาก baby23.dat คำนวณ ค่า mean และ median ของตัวแปร wt

```
> mean(wt)
 [1] NA
> mean(wt,na.rm=T)
[1] 118.6122
> round(mean(wt,na.rm=T),2)  # ปัดทศนิยมเหลือ 2 ตำแหน่งด้วยฟังก์ชัน
                             round(x,decimal)
[1] 118.61

> median(wt,na.rm=T)
[1] 117
```

ตัวอย่าง 3.5 คำนวณ ค่า mean, median, max, min จากแฟ้ม baby23.dat ปัดทศนิยม 2 ตำแหน่ง

```
> round(mean(baby23.dat,na.rm=T),2)
 gestation    sex     wt    parity    race     ed     age   marital
    278.97   1.00   118.61  1.80      3.39     2.91   26.99    1.07
     momwt     income      smoke      time      number
    128.36       3.48       0.84       0.97       2.08

> median(baby23.dat,na.rm=T)
Error in median(baby7.dat) : need numeric data

>max(baby23.dat,na.rm=T)
[1] 319
```

ให้ค่า max ของตัวแปรตัวแรกเท่านั้น

TIP: ตัวแปร ที่ใช้กับฟังก์ชัน mean จะใช้มากกว่า 1 ตัว พร้อมกันไม่ได้ แต่เลี่ยงโดยการใช้เป็นกรอบข้อมูลซึ่งสามารถเลือกเอาเฉพาะตัวแปรบางตัวในกรอบข้อมูลนั้นได้ แต่วิธีนี้นำมาใช้กับฟังก์ชัน median, max, min ไม่ได้

```
> mean(gestation,wt,na.rm=T)
```
[1] 280 # ผลคือ คำนวณเฉพาะ mean ของตัวแปรตัวแรกเท่านั้น

```
> mean(baby23.dat[,c(1,3)],na.rm=T)
gestation        wt
 278.9664   118.6122
```

หมายเหตุ Mean ของ gestation ในการคำนวณ 2 ครั้งไม่เท่ากันเพราะครั้งที่2 มีการตัดข้อมูลแถวที่มีค่าสูญหายไม่เฉพาะของ gestation เท่านั้นออกด้วย

3.2.3 ควันไทล์ Quantiles หมายถึงการนำข้อมูลที่มีการเรียงลำดับมาแบ่งออกเป็นหลายส่วน แต่ละส่วนมีจำนวนข้อมูลเท่าๆ กัน เช่น ควอไทล์ quatiles แบ่งข้อมูลออกเป็น 4 ส่วนเท่าๆ กัน เดไซล์ Deciles แบ่งข้อมูลเป็น 10 ส่วนเท่าๆ กัน หรือเปอร์เซ็นไทล์ แบ่งข้อมูลเป็น 100 ส่วนเท่าๆ กัน

วิธีใช้

```
quantile(x, ...)
quantile(x, probs = seq(0, 1, 0.25), na.rm = FALSE, names =
TRUE, type = 7, ...)
```

x: เวกเตอร์ตัวเลข (ใช้กับ list ของตัวแปรไม่ได้)

probs: ตำแหน่งควันไทล์ที่ต้องการ ระบุในรูปของความน่าจะเป็นซึ่งมีค่าในช่วง [0,1]

na.rm: ถ้า True, ค่า 'NA' and 'NaN"s จะถูกตัดออกจาก 'x' ก่อนการคำนวณ ถ้าตัวแปรมี NA ผลลัพธ์ที่ได้จะเป็น NA ดูรายละเอียดใน ?quantile

ตัวอย่าง 3.6 คำนวณ quantile สำหรับ wt

```
> quantile(wt,na.rm=T)
  0%  25%  50%  75% 100%
  71  106  117  129  174

> quantile(wt,prob=c(.1,.3,.5,.7,.9), na.rm=T)
  10%   30%   50%   70%   90%
 97.6 109.0 117.0 127.0 143.0
```

3.2.4 ค่าสถิติบอกการกระจายของข้อมูล (dispersion)

```
var(x, y = NULL, na.rm = FALSE)
sd(x, na.rm = FALSE)
```

var ใช้คำนวณค่า variance ของตัวแปร x

sd ใช้คำนวณค่า Standard deviation ของตัวแปร x

x คือ เวกเตอร์ หรือ กรอบข้อมูล หรือ เมทริกซ์ ที่ต้องการนำมาคำนวณค่า variance, sd

y คือ กรอบข้อมูลหรือเมทริกซ์ที่มีขนาดเท่ากับ x ถ้าไม่มี y แสดงว่าคำนวณ variance ของ x ถ้า x,y คือ เมทริกซ์ คำนวณค่า covariance ระหว่างคอลัมน์ของ x และ y

na.rm ถ้า True, ค่า 'NA' and 'NaN"s จะถูกตัดออกจาก 'x' ก่อนการคำนวณ แต่ถ้าไม่ระบุ (na.rm= FALSE) และตัวแปรมี NA ผลลัพธ์ที่ได้จะเกิด error

รายละเอียดดูใน ?var

ตัวอย่าง 3.7 คำนวณ covariance ของ gestation, wt

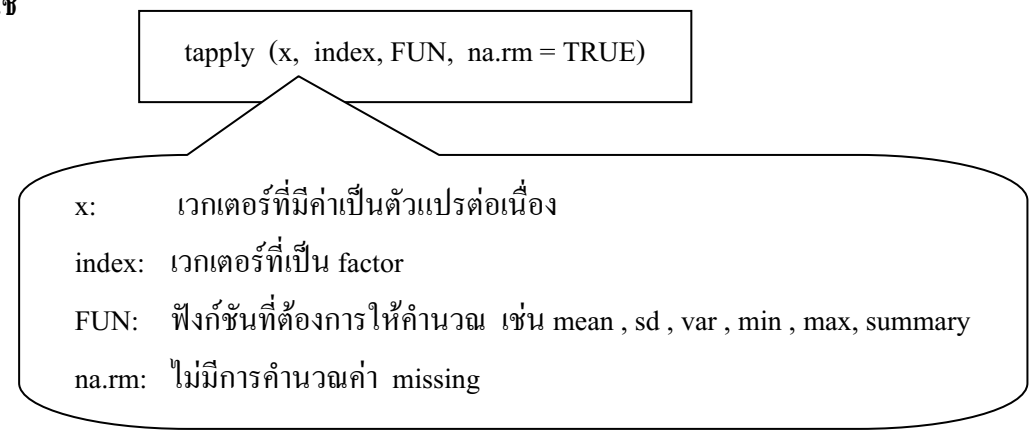
```
> var(gestation,na.rm=T)
[1] 186.8975

> var(wt,na.rm=T)
[1] 318.8418

> var(gestation,wt,na.rm=T)
[1] 113.5416


> M=cbind(gestation,wt)  # สร้างเมทริกซ์ขอของ gestation และ wt

> var(M,na.rm=T) # covariance ของ gestation, wt

           gestation       wt
gestation  185.9106 113.5416
wt         113.5416 305.2996
```

3.2.5 การสรุปลักษณะของตัวแปรเชิงปริมาณจำแนกตามกลุ่ม

การสรุปข้อมูลโดยการคำนวณค่าเฉลี่ย (mean) ,ค่าเบี่ยงเบนมาตรฐาน (sd) , ค่ามัธยฐาน (median) , ค่าสูงสุด (max) ค่าต่ำสุด (min) และ summary ของตัวแปรเชิงปริมาณจำแนกตามลักษณะของตัวแปรเชิงกลุ่ม (factor) สามารถใช้คำสั่ง tapply(…) หรือ by(…) ในการสรุปข้อมูลได้ดังนี้

วิธีที่ 1 การคำนวณหาค่าสถิติของตัวแปรเชิงปริมาณ จำแนกตามตัวแปรเชิงกลุ่ม (factor) โดยใช้คำสั่ง tapply

วิธีใช้

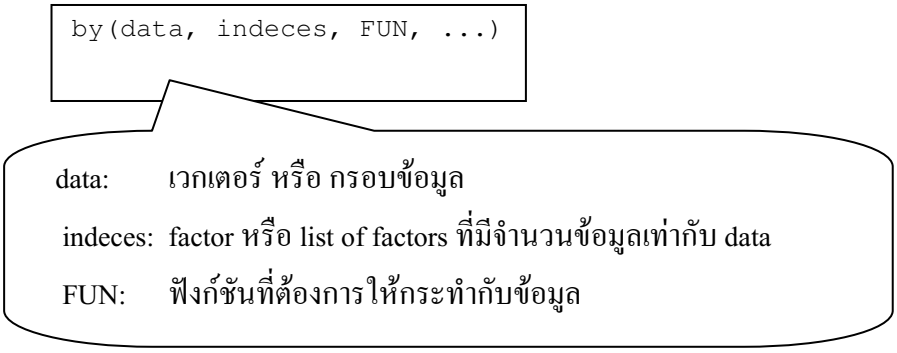


ตัวอย่าง 3.8 สรุปข้อมูล wt จำแนกตาม smoke

```
> smoke.f <-factor(smoke, labels=c("Never sm","Current-sm","Sm at
pregnant","Ever sm"))

> tapply(wt,smoke.f,summary)

$"Never sm"
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   71.0   110.5   123.0   122.1   132.5   174.0     1.0
$"Current-sm"
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   86.0   103.5   112.0   113.4   122.5   154.0     1.0
$"SM at pregnant"
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   87.0   105.5   121.0   122.8   136.0   157.0
$"Ever SM"
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  106.0   119.0   123.0   128.7   142.0   148.0
```

วิธีที่ 2 การคำนวณหาค่าสถิติของตัวแปรเชิงปริมาณ จำแนกตามตัวแปรเชิงกลุ่ม (factor) โดยใช้คำสั่ง by(…)

วิธีใช้



ตัวอย่าง 3.9 Summary wt แบ่งตาม smoke.f

```
> by(wt,smoke.f,summary)

INDICES: Never sm
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   71.0   110.5   123.0   122.1   132.5   174.0     1.0
------------------------------------------------------------
INDICES: Current-sm
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   86.0   103.5   112.0   113.4   122.5   154.0     1.0
------------------------------------------------------------
INDICES: SM at pregnant
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   87.0   105.5   121.0   122.8   136.0   157.0
------------------------------------------------------------
INDICES: Ever SM
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  106.0   119.0   123.0   128.7   142.0   148.0
```

3.3 การสรุปข้อมูลด้วยตาราง

ข้อมูลที่จะนำมาสรุปด้วยตาราง จะเป็นข้อมูลเชิงคุณภาพ หรือหากเป็นข้อมูลเชิงปริมาณจะต้องนำมาจัดกลุ่มเสียก่อน (วิธีการจัดกลุ่มในบทที่ 2) คำสั่งหลัก คือ table(…)

วิธีใช้

```
table(..., exclude = c(NA, NaN))
      as.table(x, ...)
      is.table(x)
```

...: เวกเตอร์ หรือ กรอบข้อมูล ที่ข้อมูลเป็นเชิงคุณภาพ (อาจจะมีลักษณะเป็น factors หรือ character strings ก็ได้)

exclude: ค่า หรือ ระดับของ factor ที่ไม่ต้องการนำมาสรุปในตาราง

x: วัตถุที่มี class เป็น "'table"' ที่ต้องการทำให้เป็นตาราง

3.3.1 ตารางทางเดียว

วิธีใช้

```
table(x, exclude = c(NA, NaN))
```

ตารางทางเดียวหรือตารางแจกแจงความถี่ สร้างโดยระบุตัวแปรในคำสั่ง table เพียงตัวเดียว เราจะไม่สามารถคำนวณค่ายอดรวม (total) และร้อยละ (row percent, column percent) ภายใต้คำสั่งนี้

ตัวอย่าง 3.10 สร้างตารางแจกแจงความถี่สำหรับตัวแปร smoke.f และเก็บผลลัพธ์ไว้เป็นวัตถุที่มี class เป็น table

```
> table(smoke.f)
smoke.f
      Never sm     Current-sm SM at pregnant        Ever SM
            56             68             15              9
> smf.table<- table(smoke.f)    # เก็บผลลัพธ์ไว้ใน smf.table
> attributes(smf.table)

$dim
[1] 4
$dimnames
$dimnames$smoke.f
[1] "Never sm"       "Current-sm"     "SM at pregnant" "Ever SM"
$class
[1] "table"
```

3.3.1.1 การคำนวณยอดรวม และร้อยละในตารางทางเดียว

ชุดคำสั่งที่สามารถเขียนขึ้นมาใหม่เพื่อคำนวณยอดรวมสำหรับตารางทางเดียว ทำได้ ดังนี้

ตัวอย่าง 3.11 เขียนชุดคำสั่งคำนวณยอดรวมสำหรับตารางทางเดียว

```
> total<- sum(smf.table)
> percent<-round(c(smf.table[1]*100/total,smf.table[2]*100/total,
            smf.table[3]*100/total,smf.table[4]*100/total),2) ……. (1)

> percent
      Never sm     Current-sm SM at pregnant        Ever SM
         37.84          45.95          10.14           6.08
```

> rbind(smf.table,percent) # นำค่าความถี่ และร้อยละมาต่อเป็นตารางเดียวกัน

```
          Never sm Current-sm SM at pregnant Ever SM
smf.table    56.00      68.00          15.00    9.00
percent      37.84      45.95          10.14    6.08
```

ถ้าต้องการนำเสนอตาราง โดยให้แต่ละกลุ่มอยู่ในแนว row

```
> smf.frame<-data.frame(smf.table)
> smf.frame
         smoke.f  Freq
1       Never sm   56
2     Current-sm   68
3  SM at pregnant   15
4        Ever SM    9
> cbind(smf.frame,percent)
smoke.f                 Freq      percent
1       Never sm         56         37.84
2     Current-sm         68         45.95
3 SM at pregnant         15         10.14
4        Ever SM          9          6.08
```

หมายเหตุ ในการคำนวณร้อยละนั้น ถ้ามีตารางขนาดใหญ่กว่านี้ จะไม่สะดวกในการใส่คำสั่ง(…1) จึงลองสร้างฟังก์ชันเพื่อคิดร้อยละของตารางทางเดียว ดังตัวอย่างต่อไปนี้

ฟังก์ชัน pct.oneway, x หมายถึงวัตถุที่เก็บค่าตารางทางเดียวจากคำสั่ง table( )

```
pct.oneway<-function( x  ) {
cpct <- NULL
for(i  in 1:length(x))
{
total<-sum(x)
pct<-round(x[i]*100/total,2)
cpct <- c(cpct,pct)
}
tab<-rbind(x,cpct)
print(tab)
}
```

.....

ตัวอย่าง 3.12 วิธีใช้ฟังก์ชัน pct.oneway

```
smf.table<- table(smoke.f)
table.percent<- pct.oneway(smf.table)
```

ใช้ฟังก์ชัน pct.onewy()คำนวณร้อยละในตารางทางเดียวซึ่งเก็บอยู่ในวัตถุชื่อ smf.table แล้วเก็บผลลัพธ์เป็นเมทริกซ์ชื่อ table.percent

3.3.2 ตาราง 2 ทาง

ตาราง 2 ทางเป็นตารางไขว้ที่ประกอบด้วยตัวแปรเชิงคุณภาพ 2 ตัว

วิธีใช้

```
table(x,y,exclude = c(NA, NaN))
```

ตัวอย่าง 3.13 สร้างตาราง 2 ทางเพื่อสรุปตัวแปร `term.f` และ `smoke.f` (จากที่สร้างไว้ในบทที่ 2)

```
> term.sm.table<- table(term.f,smoke.f)
> term.sm.table
            smoke.f
term.f          Never sm Current-sm SM at pregnant Ever SM
  preterm      5         6          1              0
  at- term    44        56         13              9
  post- term   6         6          1              0
```

3.3.2.1 การคำนวณยอดรวม และร้อยละรวมทั้งการทดสอบสมมติฐานสำหรับตาราง 2 ทาง

โดยฟังก์ชัน twoway.tab ซึ่งเขียนโดย ดร.นราทิพย์ จั่นสกุล (ให้ copy ฟังก์ชัน twoway.tab ซึ่งอยู่ในแฟ้ม StatCan.txt มายัง R work directory แล้ว Run ฟังก์ชัน twoway.tab ก่อนจะนำมาใช้ ด้วยคำสั่ง

> source("StatCan.txt")

วิธีใช้

```
twoway.tab(x,y, pct = c("row", "column", "total"),
chi.test = FALSE)
```

x,y : เวกเตอร์ที่จะนำมาสรุปในตาราง

pct: เลือกให้คำสั่งคำนวณค่าร้อยละตามแนว row, col, total

chi.test: ถ้า TRUE ให้ทดสอบความสัมพันธ์ระหว่าง x,y โดย chi-squares test

ตัวอย่าง 3.14 สร้างตาราง 2 ทางเพื่อสรุปตัวแปร `term.f` และ `smoke.f` ด้วยฟังก์ชัน `twoway.tab`

```
> twoway.tab(term.f,smoke.f,pct="row", chi.test=TRUE)

< A two-way table >
           Never sm Current-sm Sm at pregnant Ever sm Total
preterm           5          6              1       0    12
at- term         45         58             13       9   125
post- term        5          4              1       0    10
Total            55         68             15       9   147
< Row percent >
           Never sm Current-sm Sm at pregnant Ever sm Total
preterm      41.667     50.000          8.333   0.000   100
at- term     36.000     46.400         10.400   7.200   100
post- term   50.000     40.000         10.000   0.000   100
Total        37.415     46.259         10.204   6.122   100

Independent test
Pearson's Chi-squared = 2.299978 , df = 6 , p-value = 0.8901474

Warning message:
Chi-squared approximation may be incorrect in: chisq.test(...)
```

3.3.3 ตาราง 3 ทาง

ตาราง 3 ทางเป็นตารางไขว้ที่ประกอบด้วยตัวแปรเชิงคุณภาพ 3 ตัว

วิธีใช้

```
table(x,y,z, exclude = c(NA, NaN))
ftable(x,y,z, ...)
     ## Default S3 method:
     ftable(..., exclude = c(NA, NaN), row.vars = NULL,
col.vars = NULL)
```

คำสั่ง table (x,y,z) สร้าง ตาราง x * y จำแนกตาม z ซึ่งผลลัพธ์จะเป็นตาราง 2 ทางจำนวนเท่ากับ levels ของ z แต่หากต้องการตาราง 3 ทางที่กระทัดรัดกว่า (Flate table) นั่นคือ แสดงผลตารางของ 3 ตัวแปรอยู่ในตารางเดียวกัน ให้ใช้คำสั่ง ftable(z,x,y)

ตัวอย่าง 3.15 สร้างตาราง 3 ทางจากตัวแปร smoke.f, parity, term.f ด้วยคำสั่ง table(…)

```
> table(smoke.f,parity,term.f)
, , term.f = preterm

                parity
smoke.f           0  1  2  3  4  5  6  7
  Never sm        1  3  0  0  0  1  0  0
  Current-sm      0  3  2  1  0  0  0  0
  Sm at pregnant  0  0  1  0  0  0  0  0
  Ever sm         0  0  0  0  0  0  0  0

, , term.f = at- term

                parity
smoke.f           0  1  2  3  4  5  6  7
  Never sm        7 13 12  4  4  2  2  1
  Current-sm     15 16 12  9  2  2  1  1
  Sm at pregnant  7  2  2  0  1  0  1  0
  Ever sm         2  1  3  2  0  1  0  0

, , term.f = post- term
```

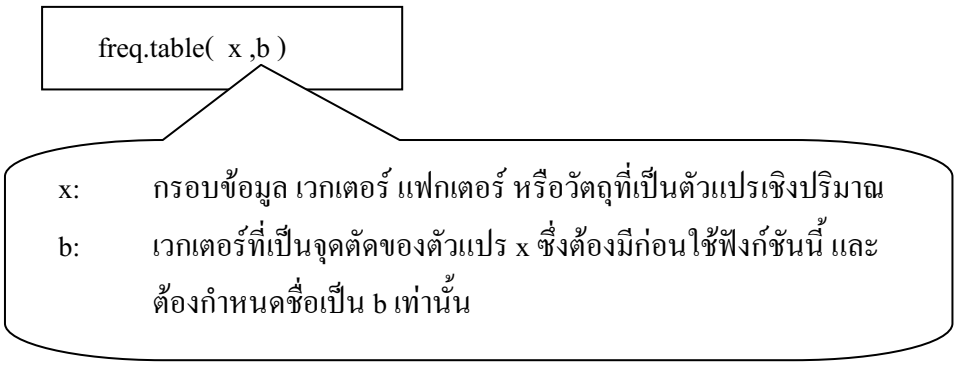
ตัวอย่าง3.16 สร้างตาราง 3 ทางจากตัวแปร smoke.f,parity,term.f ด้วยคำสั่ง ftable

```
> ftable(term.f,smoke.f,parity)

                        parity  0  1  2  3  4  5  6  7
term.f     smoke.f
preterm    Never sm             1  3  0  0  0  1  0  0
           Current-sm           0  3  2  1  0  0  0  0
           Sm at pregnant       0  0  1  0  0  0  0  0
           Ever sm              0  0  0  0  0  0  0  0
at- term   Never sm             7 13 12  4  4  2  2  1
           Current-sm          15 16 12  9  2  2  1  1
           Sm at pregnant       7  2  2  0  1  0  1  0
           Ever sm              2  1  3  2  0  1  0  0
post- term Never sm             2  0  1  1  1  0  0  0
           Current-sm           1  2  0  0  0  0  1  0
           Sm at pregnant       0  1  0  0  0  0  0  0
           Ever sm              0  0  0  0  0  0  0  0
```

3.3.4 ตารางความถี่สำหรับตัวแปรเชิงปริมาณ

ในกรณีที่ข้อมูลเป็นตัวแปรเชิงปริมาณจะทำการนับจำนวนข้อมูลของตัวแปรซึ่งจัดกลุ่มตามจุดตัดที่กำหนด (b) แล้วใช้ฟังก์ชัน freq.table ซึ่งปรับเปลี่ยนมาจากฟังก์ชัน freq. tab เรียกใช้ฟังก์ชันนี้ได้จากแฟ้มข้อมูล StatCan.txt

วิธีใช้



ตัวอย่าง 3.17 สร้างตาราง แจกแจงความถี่ของตัวแปร age โดยแบ่งช่วง age เป็น 4 ช่วง

```
> b<-c(10,20,30,40,50)
> freq.table(age,b)
        freq cfreq pcent cpcent
[10,20)    8     8  5.33   5.33
[20,30)   98   106 65.33  70.67
[30,40)   39   145 26.00  96.67
[40,50)    5   150  3.33 100.00
Total    150   150 99.99 100.00
```

แบบฝึกหัด

1. สรุปข้อมูลจากแฟ้มข้อมูล child.dat ที่สร้างขึ้นในแบบฝึกหัดท้ายบทที่ 2

บทที่ 4
การสร้างกราฟเบื้องต้น (Introduction to Graphical Procedure)

ในบทนี้ จะกล่าวถึงการสร้างกราฟในแง่มุมต่างๆ ดังนี้

1) การกำหนดสภาพแวดล้อมในการสร้างกราฟ ด้วยคำสั่ง par
2) High level plotting
3) Low level plotting
4) การสร้างกราฟชนิดต่างๆ

4.1 การกำหนดสภาพแวดล้อมในการสร้างกราฟ ด้วยคำสั่ง par()

การสร้างกราฟมีส่วนประกอบต่าง ๆ หลายส่วน (graphical parameters) กล่าวคือ มีสิ่งแวดล้อมอื่นๆ นอกเหนือจากตัวกราฟ สิ่งแวดล้อมของกราฟสามารถกำหนดได้ด้วยคำสั่งหลัก คือ par(…)

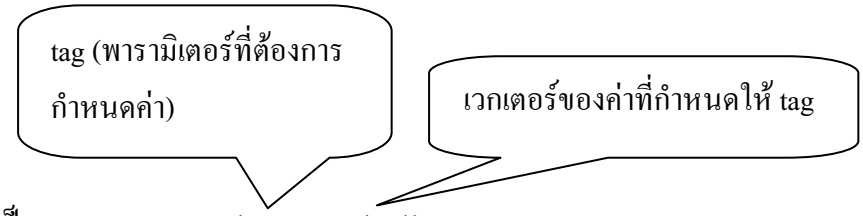
วิธีใช้

```
par(..., no.readonly = FALSE)
<highlevel plot> (..., <tag> = <value>)
```

par: เป็นคำสั่งที่ใช้กำหนดสภาพแวดล้อมของกราฟด้วยค่าพารามิเตอร์ต่างๆ ซึ่งเป็นรายละเอียดที่จะเพิ่มเติมให้กราฟสมบูรณ์มากขึ้น รูปแบบในการกำหนด คือ 'tag = value' หรือ list เมื่อ tags หมายถึง graphical parameters

รายละเอียดของกราฟจะปรากฏให้เห็นตามพารามิเตอร์ที่ระบุในคำสั่ง par หรือ plot แทนที่ของเดิมซึ่งกำหนดให้เป็นค่าปกติของโปรแกรม ถ้าต้องการยกเลิกการตั้งค่าพารามิเตอร์ดังกล่าว ทำง่ายๆ โดยการปิดจอกราฟ

ตัวอย่างรูปแบบการใช้คำสั่ง par()



ระบุพารามิเตอร์เป็น vector par(mfrow = c(2,2))

ระบุพารามิเตอร์เป็น logical par(xlog=T)

ระบุพารามิเตอร์เป็น string par(lty="dotted")

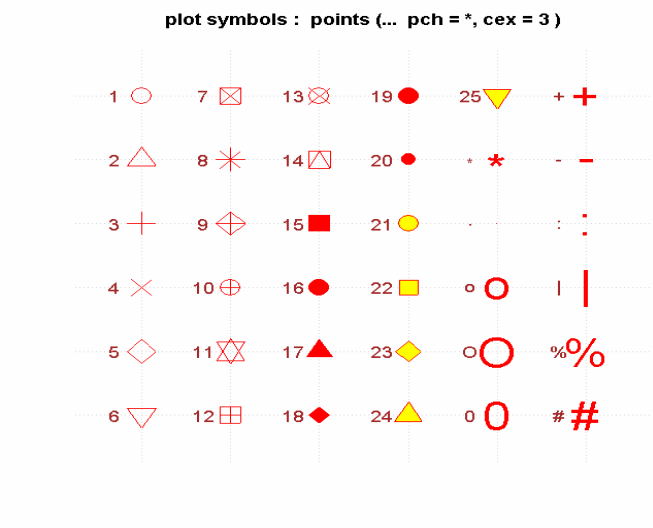
ระบุพารามิเตอร์เป็น single value par(lty=2)

พารามิเตอร์สำหรับกราฟ (graphical parameters)

พารามิเตอร์สำหรับกราฟ ที่จะกำหนดด้วยคำสั่ง par นี้ ใช้ได้กับกลุ่มคำสั่งที่เกี่ยวข้องกับกราฟใน **R** ซึ่งได้แก่คำสั่ง 'plot', 'points', 'lines', 'title', 'text', 'mtext' ตามวิธีใช้คำสั่ง นั้นๆ

การกำหนดสัญลักษณ์ในกราฟ

'pch' ตัวเลข หรือ ตัวอักษรที่กำหนดสัญลักษณ์แทนค่าข้อมูลในกราฟ ค่าปกติ คือ จุด



'type' กำหนดสไตล์ของกราฟ เช่น เป็นจุด, เป็นเส้น หรือให้มีแต่แกน ค่าปกติคือจุดกลม

"p" plot จุดกลม (*p*oints,)

"l" plot กราฟเส้น (เชื่อมจุดทุกจุดต่อกันเป็นเส้น) (*l*ines)

"b" plot ทั้งจุดและเส้น (*b*oth)

"c" เส้นสั้นๆ สำหรับแต่ละจุด

"o" จุดและลากเส้นทับเชื่อมต่อกันเป็นเส้นทับบนจุด (*o*verplotted")

"h" ลากเส้นยาวแนวตั้งเป็นแท่งคล้าย histogram (* "*h*istogram)

"s" ลากเส้นเป็นขั้นบันได (*s*teps)

"S" ลากเส้นเป็นขั้นบันไดแต่กลับด้านกับเมื่อใช้ “s”

"n" ไม่มีการ plot กราฟ มีแต่แกนของกราฟ

หากใช้สัญลักษณ์อื่นนอกเหนือจากนี้ R จะให้คำเตือน

'lty' ชนิดของเส้น กำหนดเป็นตัวเลข หรือตัวอักษรก็ได้

(0=ว่าง, 1=เส้นทึบ, 2=เส้นประ, 3=เส้นจุด, 4=เส้นจุดต่อกับประ, 5=เส้นประยาว, 6= เส้นประคู่)

("blank", "solid", "dashed", "dotted", "dotdash", "longdash", or "twodash")

font ตัวเลขกำหนดชนิดของตัวอักษรที่ใช้กับข้อความในกราฟ เลข 1= อักษรธรรมดา, 2 = ตัวหนา, 3 = ตัวเอน และ 4 =ตัวหนาและเอน

cex ขนาดของสัญญลักษณ์ในกราฟ ถ้าตั้งค่าน้อยกว่า 1 จะช่วยให้การลงจุดกราฟไม่ทับซ้อนกัน

การกำหนดสี

col.axis ตัวเลขกำหนดสีของแกน

col.lab ตัวเลขกำหนดสีของ labels บนแกน

กำหนดสีได้หลายวิธี วิธีที่ง่าย คือบอกสี เช่น'"red"' ซึ่งอาจจะระบุเป็น list ของสีด้วยฟังก์ชัน 'colors' หรืออีกวิธี คือ การระบุเป็นตัวเลข index (Index '0' หมายถึงสีของ background, 1 = สีดำ, 2= สีแดง, 3= สีเขียว, 4= สีน้ำเงิน, 5= สีฟ้า)

สามารถกำหนด 'NA' เป็น '"transparent"' ซึ่งมีประโยชน์ในกรณีที่ใช้กับพื้นหลัง เพื่อมิให้มองเห็นเส้น หรือข้อความที่ไม่ต้องการแสดงให้เห็นในกราฟ

การกำหนดแกน

'lab' numerical vector ในรูป 'c(x, y)' ซึ่งกำหนดจำนวน tickmarks บนแกน x และ y

ค่าปกติ คือ 'c(5, 5)'

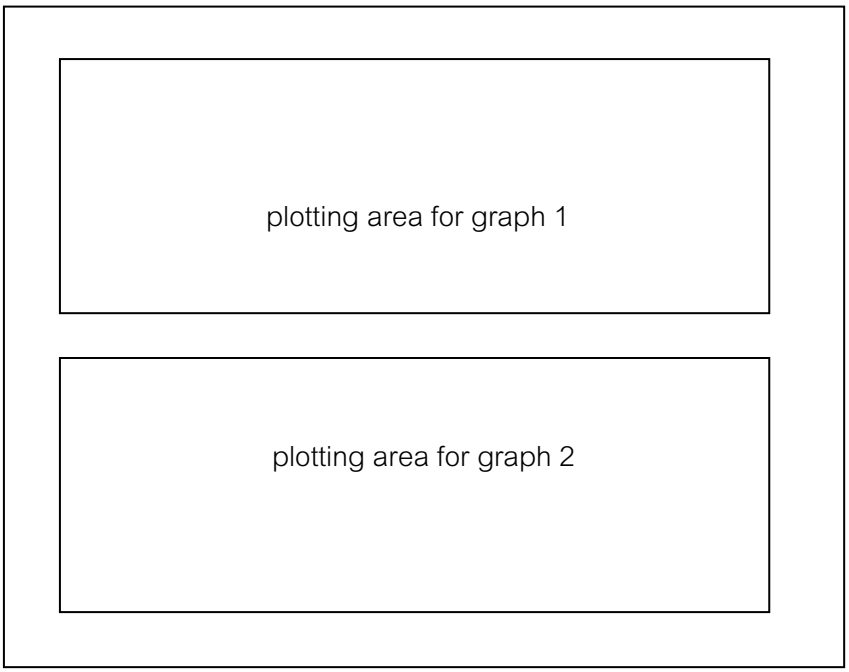
'xlog' ,'ylog' ถ้า 'xlog=TRUE', คำนวณค่า log สำหรับแกน x (หรือ y) ค่าปกติ คือ 'FALSE'

‘xlim', ‘ylim’ : กำหนดช่วงสูงสุด-ต่ำสุดของค่าบนแกน x,y ให้กราฟในรูปแบบ xlim=c(min,max)

การกำหนดพื้นที่สำหรับสร้างกราฟ

'mfcol, mfrow' เป็น vector ในรูปแบบ 'c(r, c)' ซึ่งแบ่งพื้นที่ทั้งหมดของกราฟเป็นเมทริกซ์ ขนาด r x c (r=จำนวน row, c= จำนวนcolumn) เพื่อแสดงกราฟได้จำนวน r x c รูป

รูปแสดงการแบ่งพื้นที่เป็น par(mfrow=c(2,1))



การกำหนดชื่อกราฟ

title กำหนดชื่อหลักของกราฟ (ดูใน low level plot)

main กำหนดชื่อหลักของกราฟ ในรูปแบบ main=" text"

sub กำหนดชื่อรองของกราฟ ในรูปแบบ main=" text"

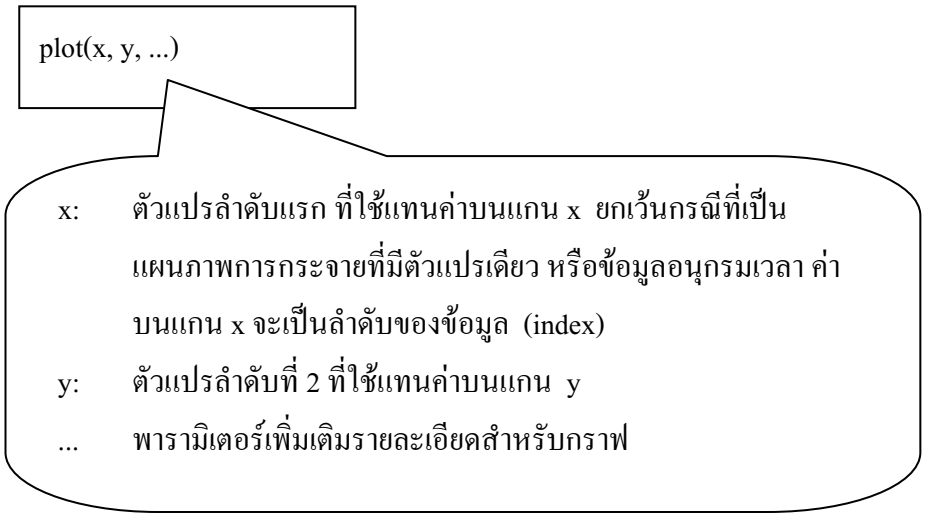
'xlab', 'ylab': ให้เขียนคำอธิบายแกน x (หรือ คำอธิบายแกน y) ในรูปแบบ xlab="text"

หมายเหตุ การกำหนดค่าพารามิเตอร์ทุกค่าในกราฟ จะทำให้ยากต่อการที่จะทราบว่าจะเกิดการเปลี่ยนแปลงใดหรือไม่ ในความพยายามที่จะทำสิ่งเดียวกันตามคำสั่งหลายๆแบบ ผลลัพธ์ที่ได้ คือ **R** จะทำตามคำสั่งหลังสุด (เรียงตามอักษร)

4.2 High level plot

High level plot หมายถึงฟังก์ชันสำหรับสร้างกราฟที่สมบูรณ์โดยตัวมันเอง คำสั่งหลักคือ plot() ซึ่งใช้สร้างกราฟชนิดต่างๆ จะเป็นกราฟชนิดใดนั้น จะขึ้นอยู่กับข้อมูลนำเข้าตัวแรกที่ระบุในฟังก์ชัน (x) สามารถเพิ่มรายละเอียดในกราฟได้ โดยกำหนดค่าพารามิเตอร์สำหรับกราฟ (พารามิเตอร์บางตัวกำหนดด้วยคำสั่ง par() แต่บางตัวกำหนดในคำสั่ง par() หรือ plot() ก็ได้)

วิธีใช้

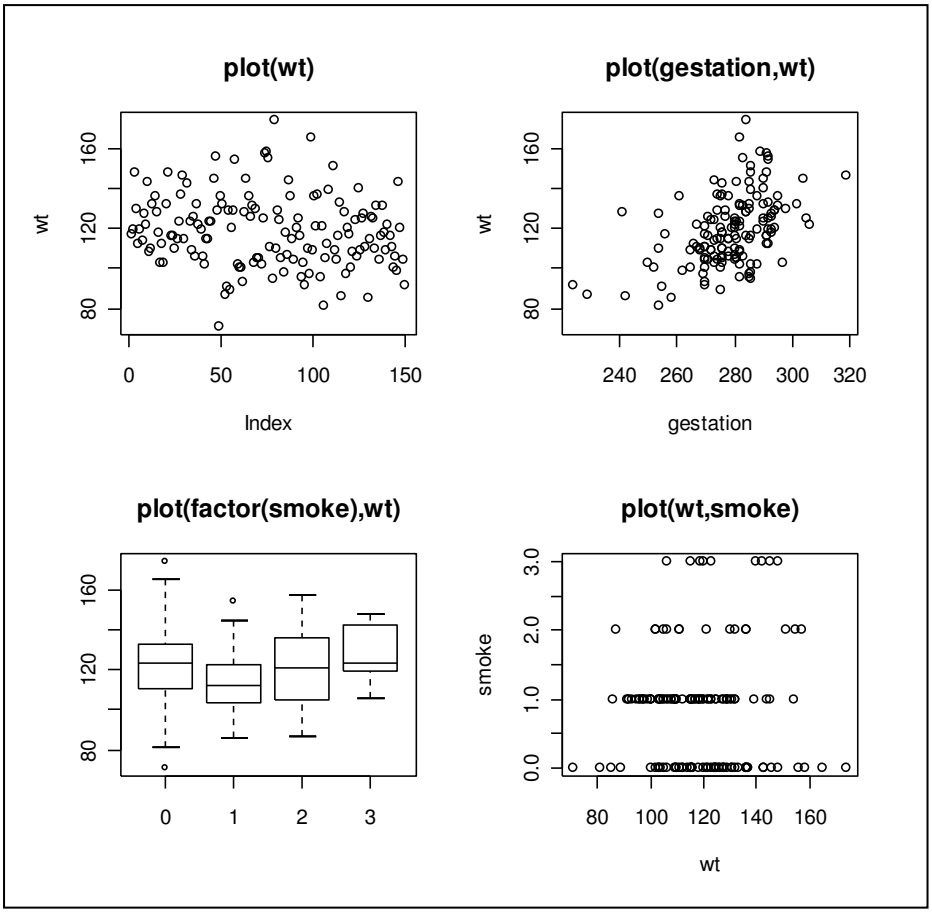


การสร้างกราฟชนิดต่างๆ ตามค่าปกติที่โปรแกรมตั้งไว้ โดยไม่ต้องระบุพารามิเตอร์ใดๆ (นอกจากชื่อหลักของกราฟ) มีดังนี้

1. plot(x) เมื่อ x คือเวกเตอร์ที่เป็นข้อมูลเชิงปริมาณหรืออนุกรมเวลา จะเป็นการ plot ค่าของ x กับลำดับที่ของข้อมูล (plot ของคู่ลำดับ (index,x))
2. plot(x) เมื่อ x เป็น data frameที่ประกอบด้วยตัวแปรเชิงปริมาณมากกว่า 2 ตัว จะเป็น matrix ของกราฟ ที่จับกันเป็นคู่ๆ
3. plot(x,y) เมื่อ x,y เป็น เวกเตอร์ที่เป็นข้อมูลเชิงปริมาณจะเป็น scatterplot ของ x กับ y (plot ของคู่ลำดับ (x,y))
4. plot(f,y) เมื่อ f เป็น factor และ y เป็นเวกเตอร์ที่เป็นข้อมูลเชิงปริมาณ R จะสร้าง boxplot ของ y จำแนกตามกลุ่มของ f

ตัวอย่าง4.1 ใช้คำสั่ง plot() ตามค่าปกติของโปรแกรม

```
>par(mfrow=c(2,2))
>plot(wt, main="plot(wt)")
>plot(gestation,wt, main="plot(gestation,wt)")
>plot(factor(smoke) ,wt , main= "plot(factor(smoke),wt)")
>plot(wt,smoke, main= "plot(wt,smoke)")
```



แบบฝึกหัด ใช้คำสั่ง plot() สร้างกราฟจากข้อมูล column ที่ 1 ถึง 3 ของกรอบข้อมูล baby23.dat

4.3 Low level plot

Low level plot หมายถึงฟังก์ชันที่ไม่ได้สร้างกราฟโดยตัวเอง แต่จะเป็นตัวช่วยให้กราฟที่สร้างด้วย high level plot มีความสมบูรณ์ขึ้น ในการใช้คำสั่งสร้าง low level plot นั้น จะทำได้ภายหลังคำสั่งสร้าง high level plot ดังตัวอย่าง การใช้คำสั่ง plot() สร้างเฉพาะแกนของกราฟ แล้วตามด้วยการลงจุดด้วยคำสั่ง points() (ซึ่งเป็น low level plot)

```
>plot(gestation,wt, type="n")  #
>points(gestation[smoke==1],wt[smoke==1],col="red")
```

รูปแบบอย่างง่ายของคำสั่งสร้าง low level plot ได้แก่

points(x,y) ลงจุด (x,y) ด้วยสัญญลักษณ์ตามที่กำหนดให้

lines(x,y) ลากเส้นตรงเชื่อมต่อจุดต่างๆของกราฟ

segments(x1,y1,x2,y2) ลากเส้นจากจุด (x1,y1) ไปยังจุด (x2,y2)

text(x,y,labels) พิมพ์ตัวอักษร หรือตัวเลข เพื่ออธิบายค่า ณ ตำแหน่งที่กำหนด

title("character ") พิมพ์ชื่อหลัก

mtext(label,side,line) พิมพ์ชื่อรอง ณ ตำแหน่งต่างๆ(side) ได้ 4 ตำแหน่ง คือ side=1 =ด้านล่าง, side=2=ด้านซ้าย, side=3=ด้านบน, side=4=ด้านขวา

legend (x,y,"text") เพิ่มคำอธิบายกราฟ ณ ตำแหน่ง (x,y)

รายละเอียดเพิ่มเติมของฟังก์ชันเหล่านี้ ดูได้จาก online help ของแต่ละฟังก์ชัน

```
points(x, ...)
      ## Default S3 method:
      points(x, y = NULL, type = "p", pch = par("pch"),col =
par("col"), bg = NA, cex = 1, ...)
```

```
lines(x, ...)
     ## Default S3 method:
     lines(x, y = NULL, type = "l", col = par("col"), lty =
par("lty"), ...)
```

```
segments(x0,y0,x1,y1,col=par("fg"),lty=par("lty"),lwd = par("lwd"), ...)
```

```
text(x, ...)
     ## Default S3 method:
     text(x,y = NULL,labels= seq(along = x),adj= NULL, pos = NULL,
 vfont = NULL, cex = 1, col = NULL, font = NULL, xpd = NULL, ...)
```

```
title(main = NULL, sub = NULL, xlab = NULL, ylab = NULL,
           line = NA, outer = FALSE, ...)
```

```
legend(x, y = NULL, legend, fill = NULL, col = "black",
            lty, lwd, pch, …
```

4.4 การสร้างกราฟชนิดต่างๆ

4.4.1 Scatter plot

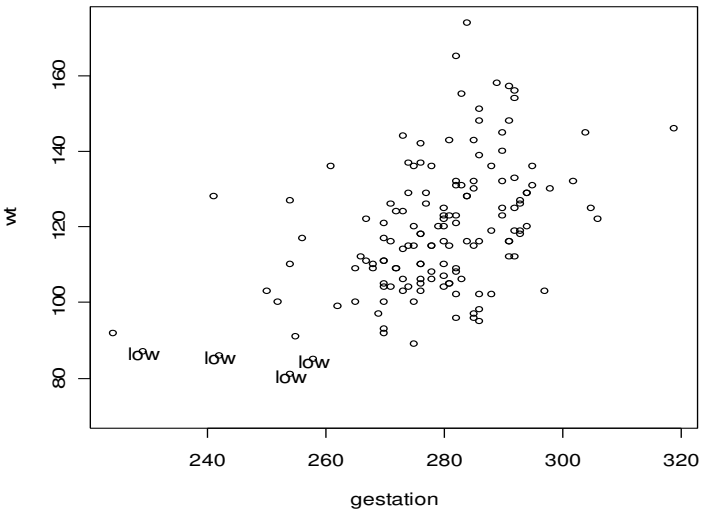
โดยปกติ Scatter plot เป็นกราฟที่ใช้ตรวจสอบความสัมพันธ์ระหว่างตัวแปรเชิงปริมาณ 2 ตัว **R** สามารถสร้าง scatter plot ด้วยคำสั่ง plot(x,y) ซึ่งเป็น high level plot และสามารถใช้คำสั่งนี้ควบคู่กับ low level plot เพื่อเพิ่มรายละเอียดของกราฟให้สมบูรณ์ยิ่งขึ้นหรือเพื่อเน้นจุดสำคัญที่ต้องการ ดังตัวอย่าง

ตัวอย่าง4.2 scatter plot ดูความสัมพันธ์ระหว่าง gestation กับ wt โดยเน้นตัวที่ smoke เท่ากับ 1 ด้วยสีน้ำเงิน

```
> plot(gestation,wt, pch="o")
> points(gestation[smoke==1],wt[smoke==1],col="blue")
```

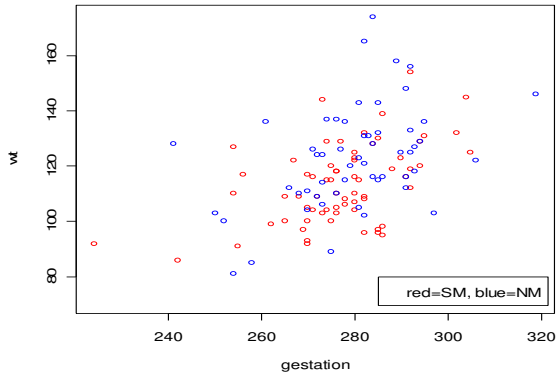
ตัวอย่าง4.3 scatter plot ดูความสัมพันธ์ระหว่าง gestation กับ wt โดยเน้นตัวที่ wt < 87.5 ด้วยคำว่า low

```
> h<-wt<87.5
> wt[h]
[1] NA NA 71 87 81 NA 86 85
> plot(gestation,wt)
> text(gestation[h],wt[h],labels="low")
```



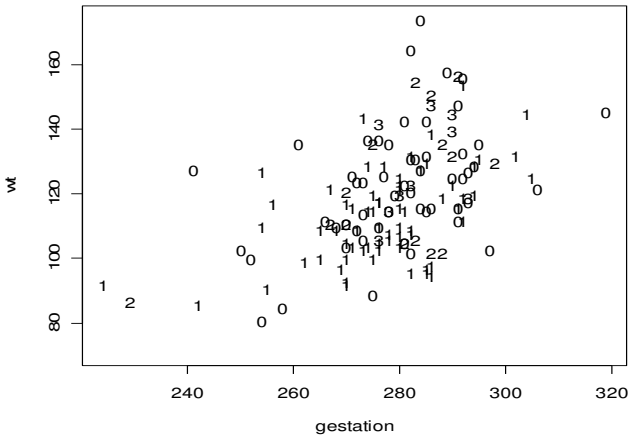
ตัวอย่าง4.4 scatter plot ดูความสัมพันธ์ระหว่าง gestation กับ wt ของเด็กที่แม่สูบบุหรี่ (smoke=1) ด้วยสีแดง และที่ smoke=0 ด้วยสีน้ำเงิน

```
>plot(gestation,wt, type="n")
>points(gestation[smoke==1],wt[smoke==1],col="red")
>points(gestation[smoke==0],wt[smoke==0],col="blue")
>legend(285,80,"red=SM, blue=NM")
```



ตัวอย่าง4.5 scatter plot ดูความสัมพันธ์ระหว่าง gestation กับ wt โดยใช้ค่าของ smoke เป็นสัญญลักษณ์

```
>plot(gestation,wt, type="n") # plot เฉพาะแกน
>text(gestation,wt,smoke)   # label จุดด้วยค่าของ smoke
```



ตัวอย่าง4.6 scatter plot ดูความสัมพันธ์ระหว่าง gestation กับ wt โดยขีดเส้นคั่นที่ wt=120

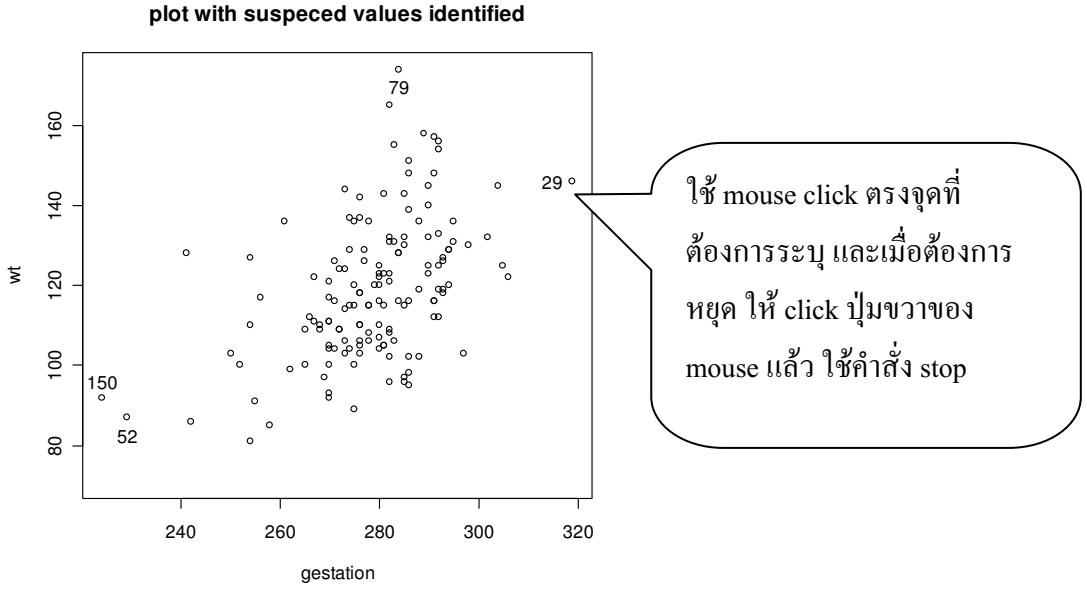
>a<-rep(120,150) # สร้างเวกเตอร์ที่มีค่าเท่ากับ 120 จำนวน 150 ตัว

```
>length(a)
[1] 150
>plot(gestation,wt)
>lines(gestation,a,col=”green”)
```

เราสามารถให้ **R** ระบุจุดที่เราต้องการทราบว่าเป็นข้อมูลตัวใดใน scatter plot ในลักษณะ interacting ได้ ด้วยคำสั่ง identify(x,y) แล้วคลิกตรงจุดที่ต้องการระบุว่าเป็นข้อมูลตัวใด เมื่อเสร็จแล้วใช้คำสั่ง stop จากการclick ปุ่มขวาของ mouse ดังตัวอย่าง 4.7

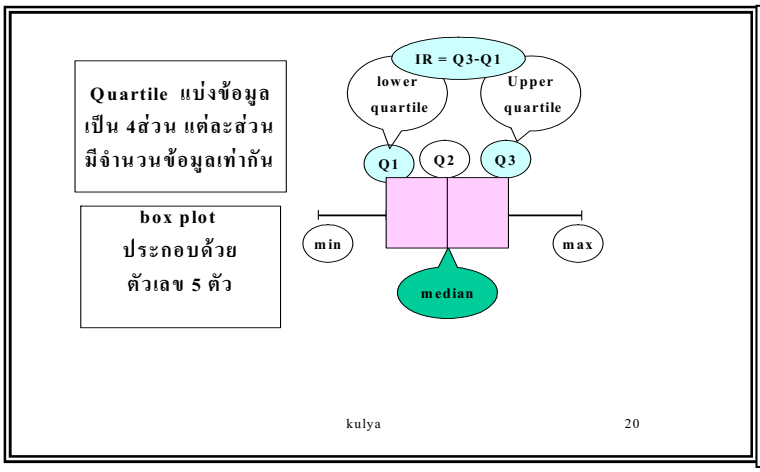
ตัวอย่าง4.7 สร้าง scatter plot ดูความสัมพันธ์ระหว่าง gestation กับ wt โดยระบุข้อมูลตัวที่มีค่าสูงหรือต่ำจนน่าสงสัย

```
>plot(gestation,wt,main="plot with suspeced values identified")
>identify(gestation,wt)
[1]  29  52  79 150
```



4.4.2 Box plot

Box plot เป็นกราฟที่นิยมใช้สรุปข้อมูลเชิงปริมาณด้วยตัวเลข 5 ตัว และยังใช้ตรวจสอบการแจกแจงของข้อมูลและค่า outlier ได้ด้วย



ข้อมูลที่มีค่ามากกว่า Q3+1.5 IR จะถือว่าข้อมูลค่านั้นเป็น Upper Outlier แต่ถ้าน้อยกว่า Q1-1.5 IR จะถือว่าข้อมูลค่านั้นเป็น lower Outlier

วิธีใช้

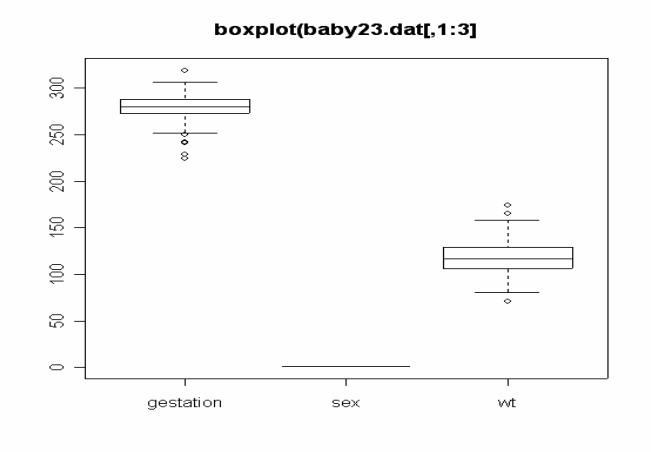
```
plot(f,y)
เมื่อ f เป็น factor และ y เป็นเวกเตอร์ที่เป็นข้อมูลเชิงปริมาณ

boxplot(x, ...)
    ## S3 method for class 'formula':
    boxplot(formula, data = NULL, ..., subset, na.action = NULL)
```

x:	เวกเตอร์ หรือ กรอบข้อมูล หรือ list ที่จะนำมาสร้าง box plot มีค่า NA ได้
… :	พารามิเตอร์ที่ใช้ได้กับการสร้างกราฟ
formula	สูตร, เช่น 'y ~ x' เมื่อ y เป็นตัวแปรเชิงปริมาณที่นำมาสร้าง box plot จำแนกตามกลุ่มของ x
data	data.frame (หรือ list) ที่ประกอบด้วยตัวแปรที่นำมาใช้ในสูตร
subset:	เวกเตอร์ที่มีข้อมูลที่เลือกมาบางส่วนเพื่อนำมาสร้าง boxplot

รายละเอียดดูใน ?boxplot

ตัวอย่าง 4.8 สร้าง box plot ของ baby23.dat[,1:3]

>boxplot(baby23.dat[,1:3],main="boxplot(baby23.dat[,1:3]")

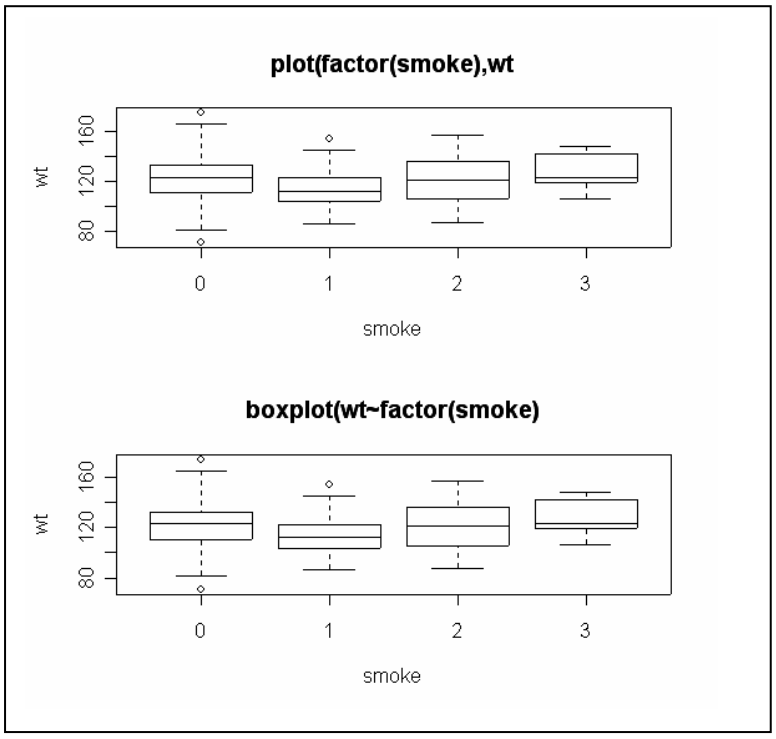


ตัวอย่าง 4.9 สร้าง box plot ของ wt จำแนกตาม smoke ทำได้ 2 แบบ ดังนี้

```
>par(mfrow=c(2,1))
>plot(factor(smoke),wt,
main="plot(factor(smoke),wt)",xlab="smoke",ylab="wt")
```

หรือ

```
>boxplot(wt~factor(smoke),
main="boxplot(wt~factor(smoke)",xlab="smoke",ylab="wt")
```



4.4.3 Histogram

histogram ใช้กับข้อมูลเชิงปริมาณ 1 ตัว โดยการแบ่งข้อมูลเป็นช่วงๆ แสดงบนแกน x แล้วนับความถี่ของข้อมูลแต่ละช่วงนั้นนำมาแสดงบนแกน y ใช้ตรวจสอบการแจกแจงข้อมูลอย่างหยาบๆ

วิธีใช้

```
hist(x, ...)
  ## Default S3 method:
     hist(x, breaks = "Sturges", freq = NULL, probability = !freq,
          include.lowest = TRUE, right = TRUE, density = NULL,
          angle = 45, col = NULL, border = NULL, main =
          paste("Histogram of" , xname), xlim = range(breaks),
          ylim = NULL, xlab = xname, ylab, axes = TRUE,
          plot = TRUE, labels = FALSE,  nclass = NULL, ...)
```

Histogram ใน **R** จะสร้างโดยการ plot ความถี่ของข้อมูลภายในแต่ละอันตรภาคชั้นที่ถูกแบ่งด้วยค่า breaks ดังนั้น ความสูงของแท่งจะเป็นสัดส่วนกับจำนวนข้อมูลที่ตกอยู่ในอันตรภาคชั้นนั้นๆ โดยปกติจะกำหนดให้แต่ละอันตรภาคชั้น (แท่ง) มีช่วงเท่ากัน แต่ถ้าต้องการให้ช่วงของอันตรภาคชั้นเป็นสัดส่วนกับความถี่ของข้อมูลก็สามารถทำได้

x: เวกเตอร์ที่ต้องการนำมาสร้าง histogram

breaks: ตัวกำหนดจุดตัดของแท่ง histogram ซึ่งอาจอยู่ในรูปแบบใดรูปแบบหนึ่ง ต่อไปนี้

* เวกเตอร์ที่ให้ค่า ณ จุดตัดต่างๆของแท่ง,
* ตัวเลขเดี่ยวที่บอกจำนวนแท่ง,
* ฟังก์ชันที่ใช้คำนวณจำนวนแท่งของกราฟ

freq: ตรรกที่บอกว่าความสูงของแท่งกราฟใช้แทนความถี่ หรือความถี่สัมพัทธ์ (ความน่าจะเป็น) ถ้า ' freq = TRUE', แท่ง histogram แทนความถี่ (Defaults); แต่ถ้า ' freq =FALSE', แท่งจะแทนความถี่สัมพัทธ์ เมื่อไม่ระบุค่าใน break และ probability

include.lowest: ตัวเลือกว่าจะนับข้อมูลตัวที่มีค่าเท่ากับขีดจำกัดล่าง ของอันตรภาคชั้นหรือไม่ ถ้า ' include.lowest =TRUE' หมายถึงให้รวมข้อมูลที่มีค่าเท่ากับขีดจำกัดล่าง

right: ตัวเลือกว่าจะนับความถี่ของข้อมูลรวมตัวที่มีค่าเท่ากับขีดจำกัดบนของอันตรภาคชั้นหรือไม่

ถ้า 'right = TRUE' (default), จะนับความถี่ของข้อมูลในอันตรชั้นโดยรวมข้อมูลมีค่าเท่ากับขีดจำกัดบนไว้ด้วย แต่ไม่รวมข้อมูลที่มีค่าเท่ากับขีดจำกัดล่าง '(a, b]' ยกเว้นกรณีที่ระบุ 'include.lowest = 'TRUE' จึงจะนับรวมข้อมูลที่เท่ากับขีดจำกัดล่าง ในทางกลับกัน

ถ้า 'right = FALSE', นับความถี่โดยรวมข้อมูลที่มีค่าเท่ากับขีดจำกัดล่างของอันตรภาคชั้นนั้น '[a, b)' และ 'include.lowest' กรณีนี้ คือให้นับรวมข้อมูลที่มีค่าเท่ากับขีดจำกัดบนไว้ด้วย

density: กำหนดให้มีการแรเงาแท่งกราฟด้วยเส้นทแยง โดย ให้ 'density= c' เมื่อ c คือค่าคงที่ที่กำหนดจำนวนเส้นต่อความยาว 1 นิ้ว default คือ NULL ไม่มีการแรเงาแท่งกราฟ

angle: กำหนดองศาของมุมของเส้นทแยงที่แรเงาแท่งกราฟ

col: ระบุสีของแท่ง default คือไม่มีสี

plot: ตัวเลือกที่จะให้แสดงกราฟบนหน้าจอหรือไม่ ถ้า 'plot=TRUE' (default) จะปรากฏ histogram บนหน้าจอ ไม่เช่นนั้น จะให้ค่าของพารามิเตอร์ที่เกี่ยวข้องกับ histogram

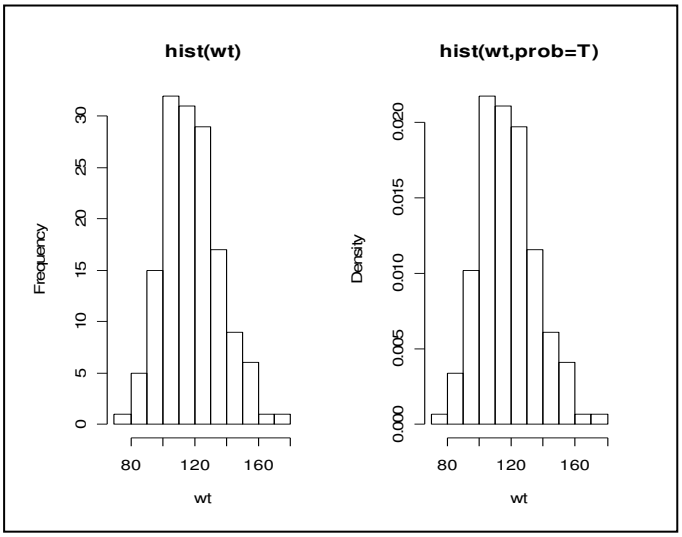
labels: ตัวเลือกที่จะให้ label ค่า หรือคำอธิบายบนแท่งกราฟหรือไม่ default คือ ไม่ต้อง label

... ตัวเลือกอื่นๆเกี่ยวกับชื่อ และแกนของการสร้างกราฟ

รายละเอียดอื่นๆดูใน ?hist

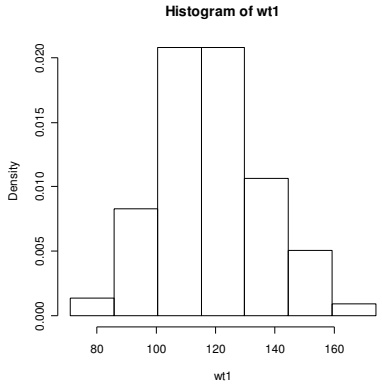
ตัวอย่าง4.10 สร้าง histogram เปรียบเทียบกัน 2 รูป โดยรูปหนึ่งแสดง frequency อีกรูปแสดงค่า density บนแกน y

```
> par(mfrow=c(1,2))
> hist(wt)
> hist(wt,probability=TRUE)
```



ตัวอย่าง4.11 สร้าง histogram โดยกำหนดจุด breaks ด้วยการแบ่งค่า wt1 (wt ที่ตัดค่า na ออก) เป็น 7 ส่วนเท่าๆกัน

```
> wt1<-na.omit(wt)
> bre<-seq(min(wt1),max(wt1),by=((max(wt1)-min(wt1))/7))
> hist(wt1,breaks=bre,probability=TRUE)
```



ตัวอย่าง4.12 histogram จำแนกตามตัวแปรกลุ่ม 2 วิธี ที่ให้ผลต่างกัน

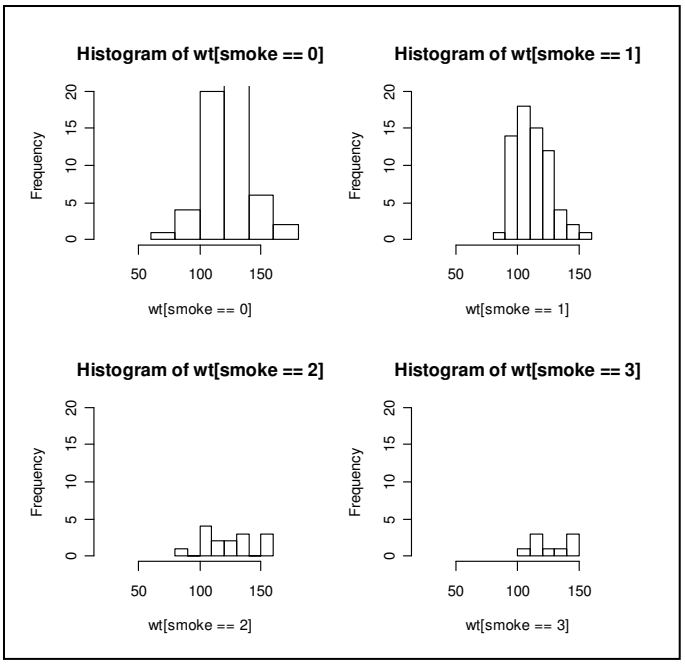
```
>par(mfrow=c(2,2))
```

1) จะได้ histogram ที่สแกลไม่เหมือนกัน ซึ่งไม่เหมาะสำหรับการเปรียบเทียบระหว่างกลุ่ม

```
>by(wt,smoke,hist,main=" by(wt,smoke,hist)")
```

2) จะได้ histogram ที่สแกลเหมือนกัน

```
>hist(wt[smoke==0],ylim=c(0,20),xlim=c(20,180))
>hist(wt[smoke==1],ylim=c(0,20),xlim=c(20,180))
>hist(wt[smoke==2],ylim=c(0,20),xlim=c(20,180))
>hist(wt[smoke==3],ylim=c(0,20),xlim=c(20,180))
```



4.4.4 stem and leaf plot

ใช้กับข้อมูลเชิงปริมาณ 1 ตัว แสดงข้อมูลโดยเรียงลำดับจากน้อยไปหามาก และแบ่งแต่ละค่าออกเป็น 2 ส่วน คือ stem (ตัวเลขทางซ้ายของ |) และ leaf ซึ่งจะแทนด้วยตัวเลขตัวเดียวทางขวาของ | ใช้ในการดูการแจกแจงของข้อมูล พร้อมทั้งรายละเอียดของข้อมูลเป็นรายตัว

วิธีใช้

```
stem(x, scale = 1, width = 80)
```

x: เวกเตอร์ที่เป็นข้อมูลเชิงปริมาณ

scale: ค่าที่ควบคุมจำนวนแถวของแต่ละ stem ค่าปกติ คือ 1 แถวต่อค่า stem 1 ค่า

width: ค่าที่ระบุความกว้างของหน้าจอที่จะสร้างกราฟ

```
> stem(wt)

  The decimal point is 1 digit(s) to the right of the |

   7 | 1
   8 | 15679
   9 | 12235667789
  10 | 000022233334444555566678899999
  11 | 000001112224555555666666778888999
  12 | 00001122233334455555666778889999
  13 | 001112222366666779
  14 | 0233455688
  15 | 145678
  16 | 5
  17 | 4
```

```
> stem(wt,scale=2)
  The decimal point is 1 digit(s) to the right of the |

   7 | 1
   7 |
   8 | 1
   8 | 5679
   9 | 1223
   9 | 5667789
  10 | 000022233334444
  10 | 5555666678899999
  11 | 000001112224
  11 | 55555566666677888999
  12 | 000011222333344
  12 | 5555666778889999
  13 | 0011122223
  13 | 66666779
  14 | 02334
  14 | 55688
  15 | 14
  15 | 5678
  16 |
  16 | 5
  17 | 4
```

4.4.5 Barplot

กราฟแท่งแสดงค่าเฉลี่ยหรือความถี่ของข้อมูลจำแนกตามกลุ่มต่างๆ โดยข้อมูลที่จะนำมาสร้าง barplot นี้ จะต้องทำให้อยู่ในรูปของตารางทางเดียวแสดงค่าที่จะมาสร้างเป็นความสูงของแท่ง

วิธีใช้

```
barplot(height, width = 1, space = NULL, names.arg = NULL,
     legend.text = NULL, beside = FALSE,  horiz = FALSE,
     density = NULL, angle = 45,col = NULL, main = NULL,
     sub = NULL, xlab = NULL, ylab = NULL,  xlim = NULL,
     ylim = NULL,...)
```

วิธีใช้อย่างง่าย

barplot(height, …)

height:: เวกเตอร์ หรือ กรอบข้อมูล หรือตารางทางเดียว ที่จะนำข้อมูลมาสร้าง bar plot ความสูงของแท่งกราฟใน Barplot ใช้แทนค่าข้อมูล ซึ่งโดยทั่วไปจะป้อนข้อมูลเป็นค่าเฉลี่ย หรือ เป็นความถี่จำแนกตามกลุ่ม

ตัวอย่าง4.13 สร้าง bar plot แสดงค่าฉลี่ยของอายุ จำแนกตาม smoke

เตรียมข้อมูลให้อยู่ในรูปตาราง

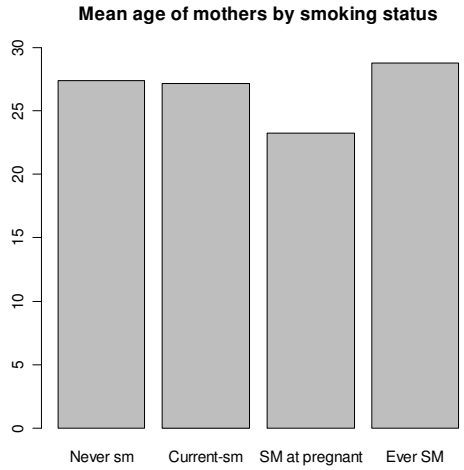
```
> smoke.f<- factor(smoke)
> levels(smoke.f)<-c("Never sm","Current-sm","SM at pregnant","Ever
SM")
> table(smoke.f)
smoke.f
      Never sm     Current-sm SM at pregnant        Ever SM
            56             68             15              9
> a<- by(age,smoke.f,mean)
INDICES: Never sm
[1] 27.44643
------------------------------------------------------------
INDICES: Current-sm
[1] 27.16176
------------------------------------------------------------
INDICES: SM at pregnant
[1] 23.26667
------------------------------------------------------------
INDICES: Ever SM
[1] 28.77778
```

สร้าง barplot

```
> barplot(a,ylim=c(0,30),main="Mean age of mothers by smoking
status")
```

หรือ

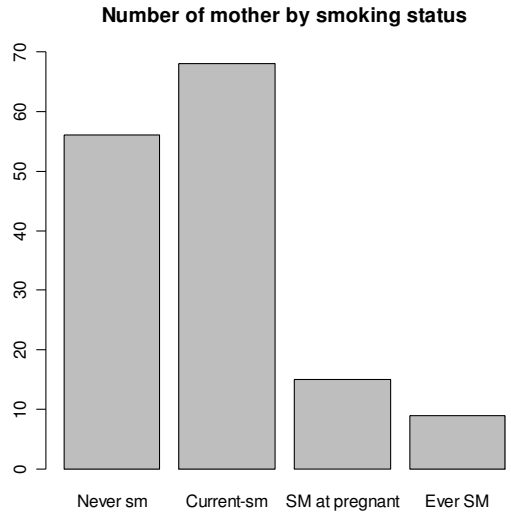
```
> barplot(by(age,smoke.f,mean),ylim=c(0,30),main="Mean age of mothers
by smoking status")
```



ตัวอย่าง 4.14 สร้าง bar plot แสดงจำนวนมารดาจำแนกตาม smoke

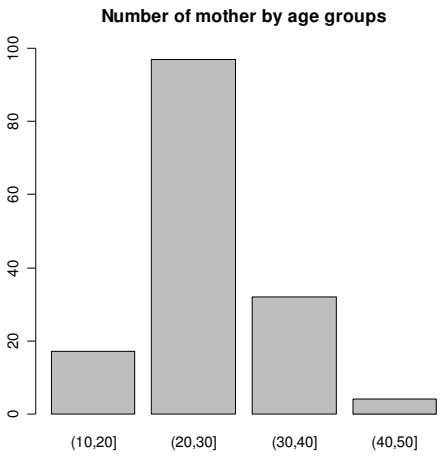
สร้างตาราง t เก็บค่าจำนวนมารดาจำแนกตาม smoke

```
> t<-table(smoke.f)
> t
smoke.f
      Never sm     Current-sm SM at pregnant        Ever SM
            56             68             15              9
> barplot(t,ylim=c(0,70),main="Number of mother by smoking status")
```



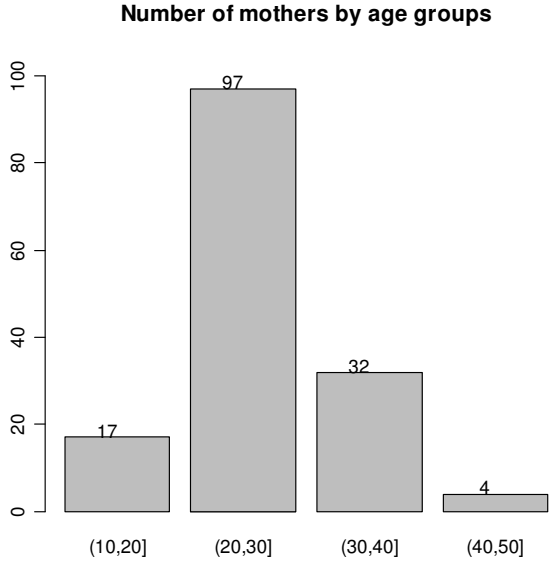
ตัวอย่าง4.15 สร้าง barplot โดยให้ความสูงของ bar แทนความถี่ของข้อมูลที่มีค่าในช่วงที่กำหนดให้ สร้าง barplot โดยกำหนดจุดตัด (bre) เพื่อแบ่งกลุ่มข้อมูล (คำสั่ง cut())

```
> bre<-c(10,20,30,40,50)
> b<-table(cut(age,bre))
> b
(10,20] (20,30] (30,40] (40,50]
     17      97      32       4
> barplot(b,ylim=c(0,100),
main="Number of mother by age groups")
```



ตัวอย่าง4.16 สร้าง barplot ตามตัวอย่าง 4.15 โดยแสดงค่าเป็นตัวเลขเพิ่มบนแท่ง

```
> x<-c(0.6,0.6*3,0.6*5,0.6*7)
> barplot(b,ylim=c(0,105),main="Number of mothers by age groups")
> text(x,b+2,c(17,97,32,4))
```



4.4.6 Dotchart

กราฟที่แทนค่าของข้อมูล จำแนกตามกลุ่ม โดยการสร้างจุด (dot) (แทนที่จะสร้างเป็นแท่งดังเช่น barplot)

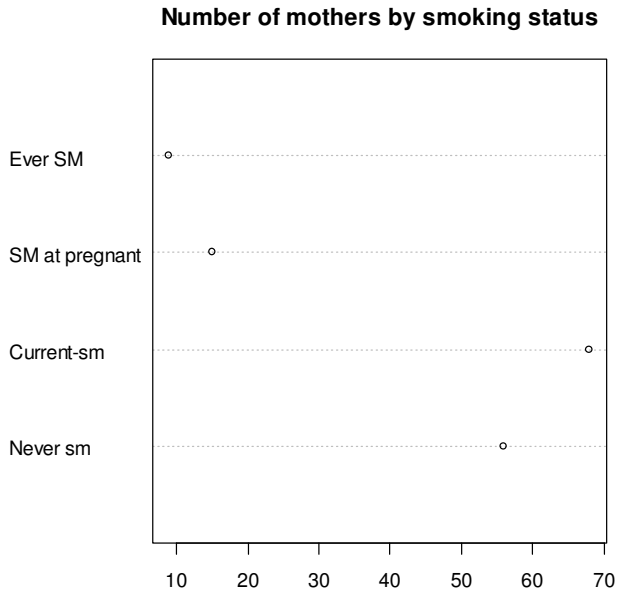
วิธีใช้

dotchart(x, ...)

x: เป็นได้ทั้งเวกเตอร์ หรือเมทริกหรือตาราง (ในทำนองเดียวกับ barplot) ที่จะนำข้อมูลมาลงจุด (dot)

ตัวอย่าง4.17 ใช้ข้อมูลจากตาราง t (ตัวอย่าง 4.14 สร้าง barplot) มาสร้าง dotchart

```
> dotchart(t, main="Number of mothers by smoking status" )
```



4.4.7 Pie chart

กราฟวงกลมแสดงสัดส่วนของข้อมูลจำแนกตามกลุ่ม ข้อมูลที่ใช้จะมีลักษณะเดียวกับข้อมูลที่ใช้ทำ barplot และ dotchart (แต่ข้อมูลนี้จะถูกนำมาคำนวณค่าสัดส่วนของ x[i]/sum(x) ก่อนโดยคำสั่ง pie())

วิธีใช้

```
pie(x, labels = names(x), edges = 200, radius = 0.8,clockwise = FALSE,
        init.angle = if(clockwise) 90 else 0, density = NULL, angle = 45, col = NULL,
        border = NULL,  lty = NULL, main = NULL, ...)
```

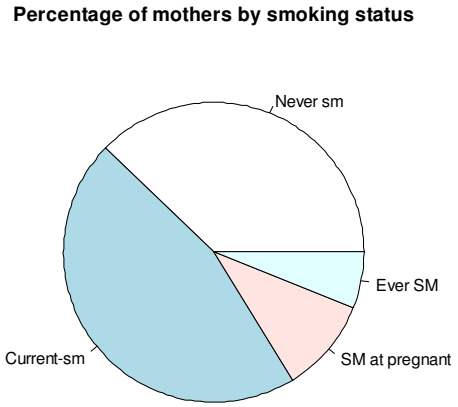
วิธีใช้อย่างง่าย

```
pie(x,  …)
```

x: เป็นเวกเตอร์ที่จะนำข้อมูลมาสร้าง pie chart (เป็นลักษณะเดียวกับข้อมูลที่ใช้กับ bar plot)

ตัวอย่าง4.18 สร้าง pie chart แสดงสถานภาพการสูบบุหรี่ smoke จากข้อมูลในตาราง t

```
> t
smoke.f
      Never sm     Current-sm SM at pregnant        Ever SM
            56             68             15              9
> pie(t, main="Percentage of mothers by smoking status")
```



บทที่ 5
การแจกแจงความน่าจะเป็น Probability Distribution

ในบทนี้ จะกล่าวถึงการแจกแจงความน่าจะเป็นของตัวแปรสุ่มและการแจกแจงความน่าจะเป็นของของค่าเฉลี่ยตัวอย่าง ($\bar{x}$) ซึ่งเป็นทฤษฎีพื้นฐานของการอนุมานสถิติ จุดที่ต้องการเน้น คือการคำนวณค่าความน่าจะเป็นและค่าความน่าจะเป็นสะสมของตัวแปรและของค่าเฉลี่ยตัวอย่าง ($\bar{x}$) เพื่อประโยชน์ในการหาค่า p-value ของการทดสอบสมมติฐาน นอกจากนี้ ยังแสดงวิธีการสร้างเลขสุ่มที่มีการแจกแจงแบบต่างๆ อีกด้วย

การแจกแจงดังกล่าวได้แก่

1) การแจกแจงทวินาม Binomial Distribution

2) การแจกแจงปังส์ซง Poisson Distribution

3) การแจกแจงปกติ Normal Distribution

4) การแจกแจงของค่าเฉลี่ยตัวอย่าง Sampling distribution of the sample mean

5.1 การแจกแจงทวินาม (Binomial distribution)

ถ้า X เป็นตัวแปรที่มีการแจกแจงแบบทวินามใช้ X แทนจำนวนผลสำเร็จที่เกิดขึ้นในการทดลองหรือในเหตุการณ์ ที่เกิดซ้ำ ๆ อย่างเป็นอิสระต่อกันจำนวน n ครั้ง หรือ (size = n) และความน่าจะเป็นที่จะเกิดผลสำเร็จในแต่ละครั้งมีค่าเท่ากันคือเท่ากับ p หรือ (prob =p) เมื่อ $0 \le p \le 1$ เขียนการแจกแจงของ X ด้วยสัญญลักษณ X~B(x,n,p)

ฟังก์ชันความน่าจะเป็น f(x) หมายถึง ค่าความน่าจะเป็นที่ X มีค่าต่างๆ เขียนแทนด้วย P(X=x) นั่นคือ

$$f(x) = P(X=x) = \begin{cases} \binom{n}{x} p^x q^{n-x} & , \ x = 0, \ldots, n \\ 0 & , \ x \text{ มีค่าอื่น ๆ} \end{cases}$$

ฟังก์ชันความน่าจะเป็นสะสม F(x) หมายถึง ความน่าจะเป็นสะสมที่ X มีค่าน้อยกว่าหรือเท่ากับ x $P(X \le x)$ นั่นคือ $P(X \le x) = \sum_{x_i \le x} f(x_i)$

การคำนวณค่า f(x) ของการแจกแจงทวินาม จำเป็นต้องคำนวณค่า $\binom{n}{x}$ ซึ่งโปรแกรมสำเร็จรูปอื่นๆ อาจจะคำนวณโดยตรงไม่สะดวก เพราะต้องอาศัยสมการ x! = gamma(x+1) มาช่วยคำนวณ แต่ใน **R** สามารถใช้ฟังก์ชัน choose(n,x) คำนวณ $\binom{n}{x}$ ได้โดยตรง

การคำนวณค่า $\binom{5}{3} = \frac{5!}{3!(5-3)!}$

```
> choose(5,3)
[1] 10
> gamma(5+1)/(gamma(3+1)*gamma(2+1))
[1] 10
```

5.1.1 การคำนวณความน่าจะเป็นและความน่าจะเป็นสะสมของตัวแปรสุ่มทวินาม

ชุดคำสั่งใน **R** สำหรับการแจกแจงทวินาม มีดังนี้

วิธีใช้

คำนวณค่าf(x)=P(X=x): dbinom(x, size, prob, log = FALSE)

คำนวณค่าF(x)= P(X≤ q): pbinom(q, size, prob, lower.tail = TRUE, log.p = FALSE)

x: คือเวกเตอร์ที่มีค่าเป็นตัวเลขแทนจำนวนผลสำเร็จ (ตามปกติจะมีค่าที่เป็นไปได้คือ0,1,2,...,n) ที่ต้องการหาค่าความน่าจะเป็น

size: จำนวนครั้งที่เกิดเหตุการณ์ จะเท่ากับ n

prob: ความน่าจะเป็นที่จะเกิดผลสำเร็จในแต่ละครั้ง

lower.tail: ถ้า TRUE ให้ค่า P(X≤ q)(เป็นค่าปกติซึ่งไม่ต้องระบุก็ได้) แต่ถ้า เป็น F จะให้ค่า P(X> q)

ตัวอย่าง 5.1 ถ้า X~B(x,5,0.4) ให้แจกแจงความน่าจะเป็นของ X, P(X=x) และ แจกแจงความน่าจะเป็นสะสม P(X≤ x)

```
> x <- 0:5
>  dbinom(x, size=5, prob=0
```

สร้างเวกเตอร์ x ซึ่งประกอบด้วยค่าที่เป็นไปได้ทั้งหมดของ x คือ

หรือใช้แบบง่ายๆ

```
> dbinom(x, 5, 0.4)
```

คำนวณค่าf(x)=P(X=x) สำหรับทุกค่าของ x

```
[1] 0.07776 0.25920 0.34560 0.23040 0.07680 0.01024
```

```
> pbinom(x, 5, 0.4)
```

คำนวณค่า F(x)= P(X≤ x)

```
[1] 0.07776   0.33696   0.68256   0.91296   0.98976   1.00000
```

ตัวอย่าง 5.2 ถ้า X~B(x,5,0.4) คำนวณความน่าจะเป็น P(2 < X < 5)

```
> pbinom(4,5,0.4)-pbinom(2,5,0.4)
```

P(2 <x <5) = P(X≤ 4) - P(X≤ 2)

```
[1] 0.3072
```

```
> dbinom(3,5,0.4)+dbinom(4,5,0.4)
```

P(2 < X <5) = P(X= 3) + P(X= 4)

```
[1] 0.3072
```

ตัวอย่าง 5.3 ถ้า X~B(x,5,0.4) คำนวณความน่าจะเป็น P(X > 2)

```
> pbinom(2, 5, 0.4, lower.tail=F)
 [1] 0.31744
```

ซึ่งมีค่าเท่ากับ

```
> dbinom(3,5,0.4) + dbinom(4,5,0.4)+ dbinom(5,5,0.4)
[1] 0.31744
```

ตัวอย่าง 5.4 จากตัวอย่าง 5.1 เสนอผลลัพธ์ในรูปตารางแจกแจงความน่าจะเป็น

```
> fx<-dbinom(x, 5, 0.4)
> Fx<-pbinom(x, 5, 0.4)
> cbind(x,fx,Fx)
x      fx      Fx
[1,] 0 0.07776 0.07776
[2,] 1 0.25920 0.33696
[3,] 2 0.34560 0.68256
[4,] 3 0.23040 0.91296
[5,] 4 0.07680 0.98976
[6,] 5 0.01024 1.00000
```

5.1.2 กราฟของการแจกแจงความน่าจะเป็นและการแจกแจงสะสมทวินาม

```
# ฟังก์ชันเพื่อวาดกราฟของการแจกแจงความน่าจะเป็นแบบทวินาม
Binom.plot <- function(x, n, p) {
y1<- NULL
x1<- NULL
for(i in 1 : length(x))
        {
         x1 <- c(x[i],x1)
        y1 <- c(dbinom(x[i],n,p),y1)
        }
        plot(x1,y1, ylab=" density", xlab=”number of success” , main = "Graph of
Binomial Distribution")
         segments(x1,0,x1,y1)
        }
```

วิธีใช้ฟังก์ชัน Binom.plot(x,n,p)

1. กำหนดค่าของ x, n, p

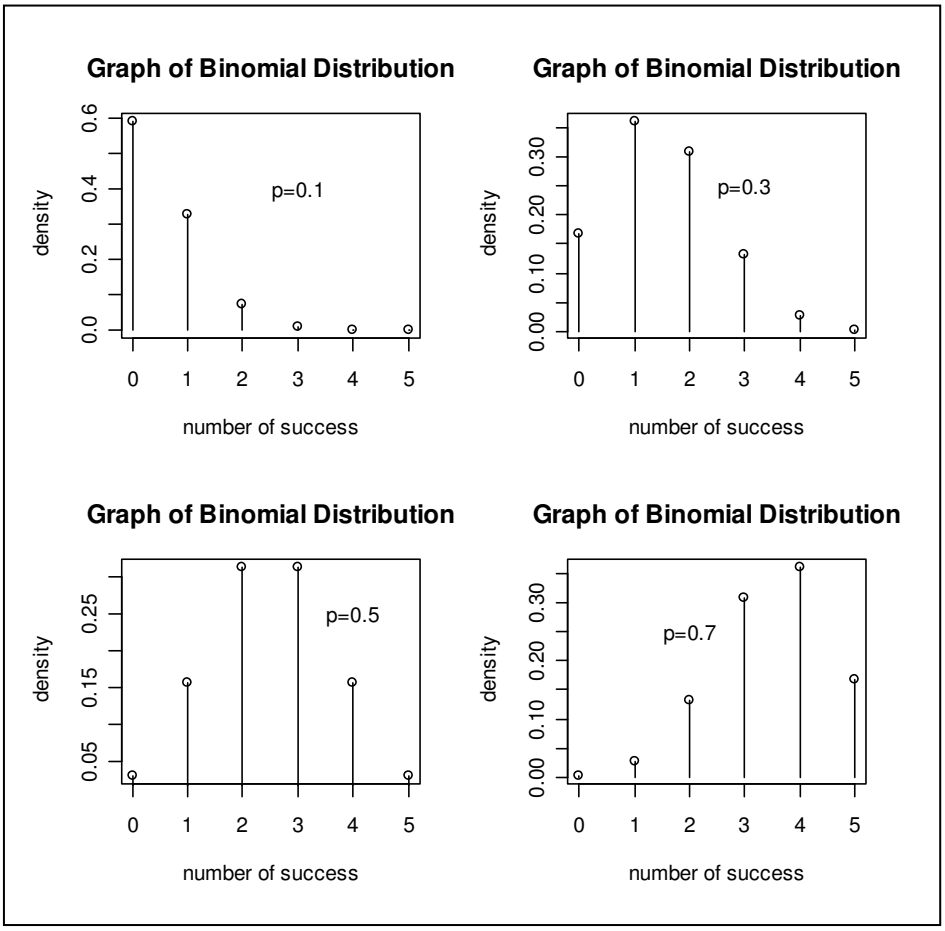
2. ใช้คำสั่ง > Binom.plot(x, n,p)

หรือสามารถกำหนดค่า x, n,p ในตัวฟังก์ชันได้เลย

> Binom.plot(0:5,5,0.4)

ตัวอย่าง 5.5 วาดกราฟของการแจกแจงทวินาม X~B(x;5,0.1) , X~B(x;5,0.3), X~B(x;5,0.5), X~B(x;5,0.7) เมื่อ x=(0:5)

```
>par(mfrow=c(2,2))
>Binom.plot(0:5, 5,.1)
>text(3,0.4,”p=0.1”)
>Binom.plot(0:5, 5,.3)
>text(3,0.25,”p=0.3”)
>Binom.plot(0:5, 5,.5)
>text(4,0.25,”p=0.5”)
>Binom.plot(0:5, 5,.7)
>text(2,0.25,”p=0.7”)
```

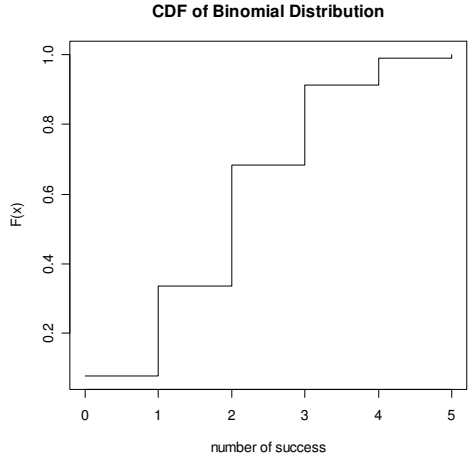


ข้อสังเกต

จะเห็นได้ว่า กราฟมีลักษณะสมมาตรเมื่อค่า p= 0.5 กราฟเบ้ขวา ถ้า p< 0.5 และเบ้ซ้าย ถ้า p> 0.5

ตัวอย่าง 5.6 วาดกราฟของการแจกแจงสะสมทวินาม X~B(x;5,0.4)

```
> plot(0:5, pbinom(0:5,5, 0.4),type="s", ylab="F(x)",main="CDF of Binomial
Distribution",xlab="number of success")
```



5.1.3 สร้างเวกเตอร์เลขสุ่มจำนวน n ตัว ที่มีการแจกแจงแบบทวินาม

วิธีใช้

```
rbinom(n, size, prob)
```

n: จำนวนเลขสุ่มที่ต้องการสร้าง
size: จำนวนครั้งที่เกิดเหตุการณ์ซ้ำ
p: ค่าความน่าจะเป็นที่จะเกิดผลสำเร็จแต่ละครั้ง

ตัวอย่าง 5.7 สร้างเลขสุ่มที่มีการแจกแจงแบบทวินาม X~B(x;10, 0.5)

1) ให้มีจำนวนเลขสุ่ม 5 ตัว 3 ชุด

```
>  rbinom(5, size=10, prob=.5)
[1] 2 4 4 3 6

>  rbinom(5, size=10, prob=.5)
[1] 4 3 7 4 6

> rbinom(5, size=10, prob=.5)
[1]  3  6  5  4 10
```

จะได้เลขสุ่ม 3 ชุดที่มีค่าต่างกัน

หากต้องการสร้างเลขสุ่มที่มีสมาชิกชุดเดียวกัน ให้กำหนดค่า seed ของเลขสุ่มด้วยคำสั่ง set.seed(99999) ควบคู่กับคำสั่ง rbinom(15, size=10, prob=0.5)

```
> set.seed(99999)
> rbinom(15, size=10, prob=0.5)
 [1] 4 7 5 7 4 3 6 5 8 2 8 3 4 5 4
```

เลขอะไรก็ได้ที่มีค่าเป็นจำนวนเต็มบวก
ดูรายละเอียดที่ “?set.seed”

ตัวอย่าง 5.8 สร้างเลขสุ่ม X~B(x; 5 , 0.5)

1) ให้มีจำนวนเลขสุ่ม 30 ตัว และสรุปค่าของเลขสุ่มเป็นตาราง

```
> set.seed(11111)

> table( rbinom(30, size=5, prob=.5))
 1  2  3  4  5
 6  5 11  6  2
```

ค่าของเลขสุ่ม

ความถี่

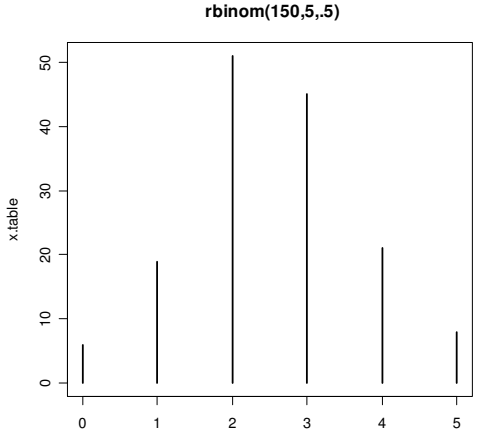
2) ให้มีจำนวนเลขสุ่ม 150 ตัว และสรุปผลลัพธ์เป็นตารางและกราฟ

```
> set.seed(11111)
> x.table<-table( rbinom(150, size=5, prob=.5))
```

```
> x.table
 0  1  2  3  4  5
 6 19 51 45 21  8

> pct.oneway(x.table)
     0     1  2  3  4    5
x    6 19.00 51 45 21 8.00
cpct 4 12.67 34 30 14 5.33
```

pct.oneway เป็นฟังก์ชันที่เขียนขึ้นเพื่อคำนวณร้อยละ(จากบทที่ 3)

```
> plot(x.table,main=" rbinom(150,5,.5)")
```



5.2 การแจกแจงปัวซง Poisson distribution

ตัวแปรสุ่ม X จะเป็นตัวแปรสุ่มที่มีการแจกแจงแบบปัวส์ซง ถ้า X เป็นจำนวนผลลัพธ์ของเหตุการณ์เชิงสุ่มที่เกิดในขอบเขตที่กำหนด (อาจเป็นช่วงเวลาหรือพื้นที่ก็ได้) โดยที่เหตุการณ์ที่เกิดขึ้นแต่ละครั้งจะเป็นอิสระต่อกัน การนับจำนวนสิ่งที่สนใจจะถือว่าสิ่งที่เกิดขึ้นอยู่ในช่วงเวลาที่ไม่คาบเกี่ยวกัน (non overlapping intervals) หรืออีกนัยหนึ่งคือ ความน่าจะเป็นที่สิ่งที่สนใจจะเกิดขึ้นตั้งแต่สองครั้งขึ้นไปในช่วงเวลาสั้นๆหรือในอาณาบริเวณเล็กๆจะมีค่าน้อยมากจนตัดทิ้งได้ (มีค่า = 0) และความน่าจะเป็นที่สิ่งที่สนใจเกิดขึ้นหนึ่งครั้งในช่วงเวลาหรือในอาณาบริเวณจะแปรผันโดยตรงกับความยาวของช่วงเวลาหรือขนาดของพื้นที่

ฟังก์ชันความน่าจะเป็นของ X คือ

$$f(x) = \frac{e^{-\lambda}\lambda^{x}}{x!}$$

เมื่อ $x = 0, 1, 2, \ldots$

$e = 2.71828 \ldots$

λ = ค่าเฉลี่ยของจำนวนครั้งของเหตุการณ์ที่เกิดในช่วงเวลาที่กำหนด, $\lambda \geq 0$

[เขียนเป็นสัญญลักษณ์ $X \sim p(x; \lambda)$]

5.2.1 การคำนวณความน่าจะเป็นและความน่าจะเป็นสะสม

การคำนวณความน่าจะเป็นและความน่าจะเป็นสะสมของการแจกแจงแบบปังส์ซง จะทำได้ทำนองเดียวกับการแจกแจงแบบทวินาม เพียงแต่เปลี่ยนฟังก์ชันและค่าพารามิเตอร์ของการแจกแจง

ชุดคำสั่งใน R สำหรับการแจกแจงปัวส์ซง มีดังนี้

คำนวณค่า $f(x)=P(X=x)$: dpois(x, lambda, log = FALSE)

คำนวณค่า $F(x)= P(X \leq q)$: ppois(q, lambda, lower.tail = TRUE, log.p = FALSE)

x:	เวกเตอร์ที่เป็นค่าของตัวแปรสุ่ม (มีค่าเป็นบวก)
lambda:	เวกเตอร์ของค่าเฉลี่ย (มีค่าเป็นบวก)
lower.tail:	ถ้าเป็น T (ค่าปกติที่ไม่ต้องระบุก็ได้), จะให้ค่า $P[X \leq q]$, แต่ถ้าเป็น F จะให้ค่า $P[X > q]$

ตัวอย่าง 5.9 คำนวณค่าความน่าจะเป็นและความน่าจะเป็นสะสมของ X~ p(x;5) เมื่อ X=(0,1,…,8)

```
> round(dpois(0:8, lambda=5),5)
[1] 0.00674   0.03369  0.08422   0.14037   0.17547   0.17547
0.14622   0.10444   0.06528

> round(ppois(0:8, lambda=5),5)
[1] 0.00674 0.04043 0.12465 0.26503 0.44049 0.61596 0.76218 0.86663
0.93191
```

5.2.2 กราฟของการแจกแจงความน่าจะเป็นและการแจกแจงสะสมแบบปัวส์ซง

```
# ฟังก์ชันเพื่อวาดกราฟของการแจกแจงความน่าจะเป็นแบบปัวส์ซง


Poisson.plot <- function(x, lambda) {
y1<- NULL
x1<- NULL
for(i in 1 : length(x))
      {
       x1 <- c(x[i],x1)
      y1 <- c(ppois(x[i], lambda),y1)  }
      plot(x1,y1, ylab= "density", xlab="number of occurents" ,
         main = "Graph of Poisson Distribution")
      segments(x1,0,x1,y1)
       }
```

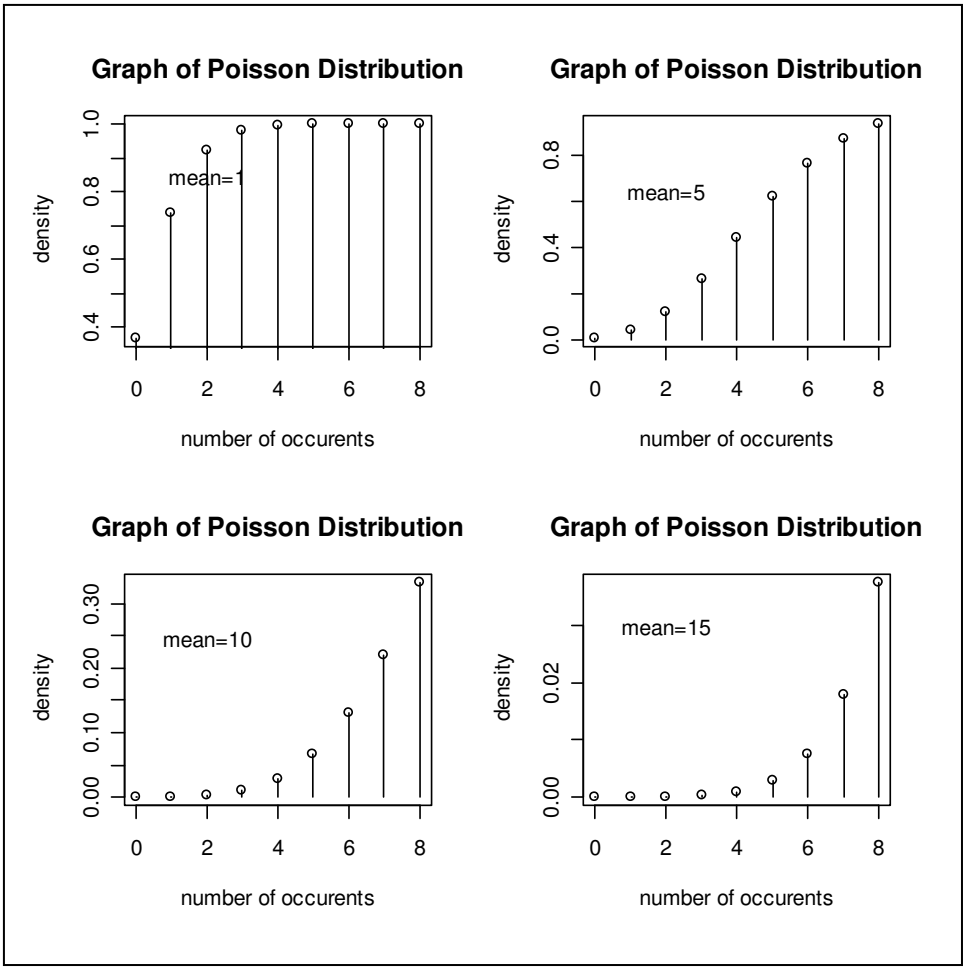
การใช้ฟังก์ชัน Poisson.plot(x, lambda)

กำหนดค่า x และ lambda ลงในฟังก์ชัน เช่น

```
> Poisson.plot(0:8,5)
```

ตัวอย่าง 5.10 ใช้ฟังก์ชัน Poisson.plot(x, lambda) วาดกราฟความน่าจะเป็นของ X~ p(x,1), X~ p(x,5), X~ p(x,10), X~ p(x,15) เมื่อ X= (0:8)

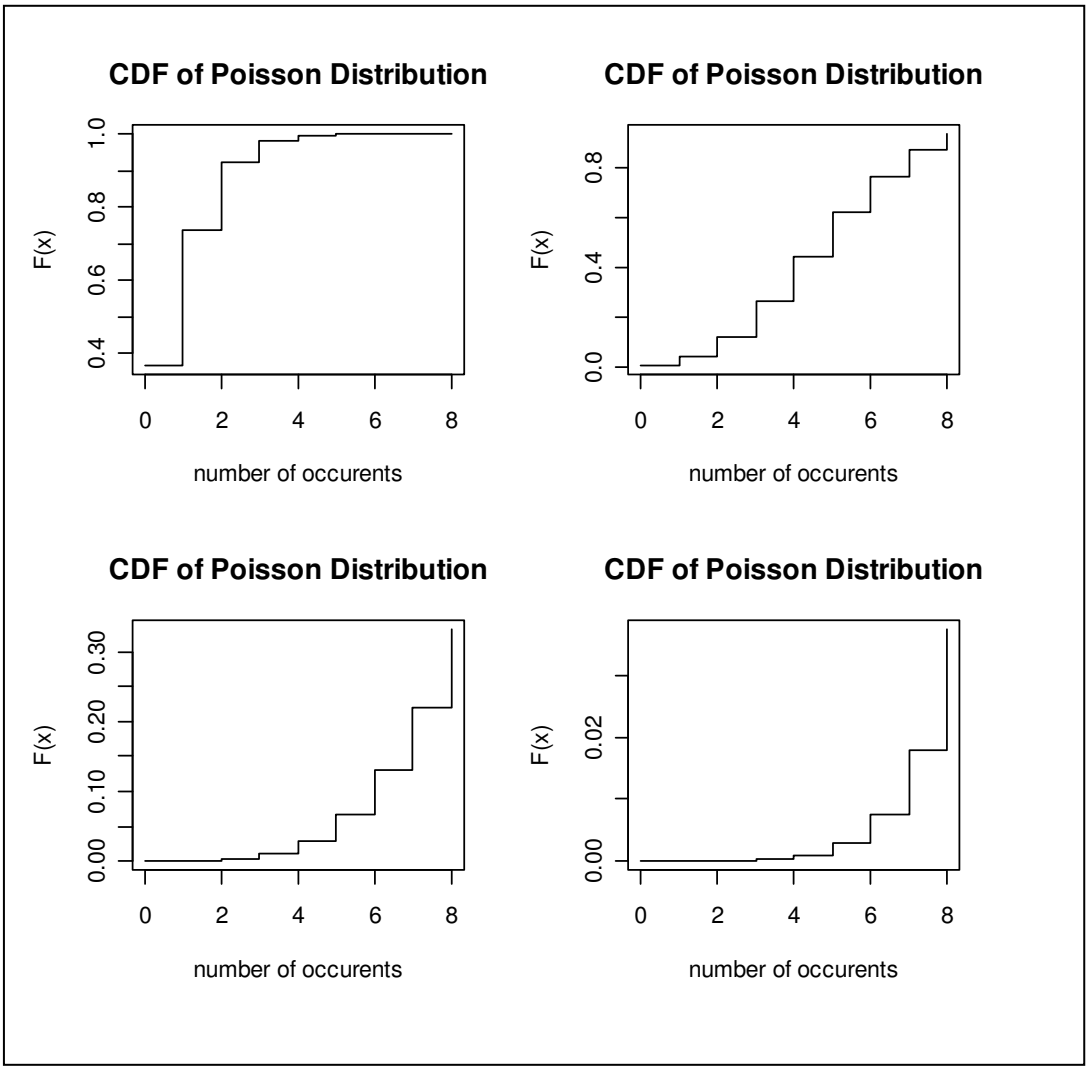
```
> par(mfrow=c(2,2))

> Poisson.plot(0:8,1)

> text(2,0.85,"mean=1")

> Poisson.plot(0:8,5)

> text(2,0.65,"mean=5")

> Poisson.plot(0:8,10)

> text(2,0.25,"mean=10")

> Poisson.plot(0:8,15)

> text(2,0.03,"mean=15")
```



ตัวอย่าง 5.11 วาดกราฟความน่าจะเป็นสะสมของ X~ p(x,1), X~ p(x,5), X~ p(x,10), X~ p(x,15), เมื่อ X= (0:8)

>par(mfrow=c(2,2))

```
> plot(0:8, ppois(0:8,1), type="s", ylab="F(x)", main=" CDF of
Poisson Distribution ", xlab="number of occurents")

> plot(0:8, ppois(0:8,5), type="s", ylab="F(x)", main=" CDF of
Poisson Distribution ", xlab="number of occurents")

> plot(0:8, ppois(0:8,10), type="s", ylab="F(x)", main=" CDF of
Poisson Distribution ", xlab="number of occurents")

> plot(0:8, ppois(0:8,15), type="s", ylab="F(x)", main=" CDF of
Poisson Distribution ", xlab="number of occurents")
```



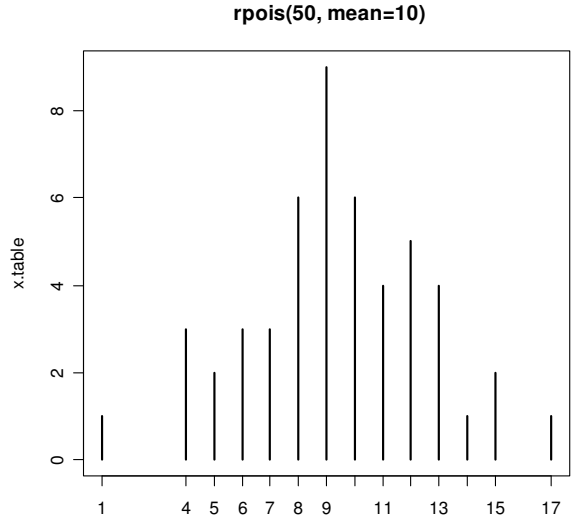
5.2.3 สร้างเวกเตอร์เลขสุ่มจำนวน n ตัว ที่มีการแจกแจงแบบปัวส์ซง

วิธีใช้

ตัวอย่าง 5.12 สร้างเลขสุ่มที่มีการแจกแจงแบบปัวส์ซง ที่มีค่าเฉลี่ยเท่ากับ 10 จำนวน 50 ตัว แล้วสรุปเลขสุ่มที่ได้เป็นตารางแสดงความถี่ และกราฟ

```
> set.seed(99999)

> x.table<-table(rpois(50, lam= 10))
```

```
> pct.oneway(x.table)
> plot(x.table,main=" rpois(50,mean=10)")
```



เมื่อ $\lambda \geq 10$ การแจกแจง Poisson จะมีความใกล้เคียงกับการแจกแจงปกติ

5.3 การแจกแจงปกติ (Normal distribution)

การแจกแจงปกติจะมีลักษณะเป็นรูประฆังคว่ำ (bell-shaped curve) เส้นโค้งสมมาตรตามแนวค่าเฉลี่ย μ (symmetric about the mean) มีค่าความเบ้เท่ากับ 0 และมีค่าความโด่งเท่ากับ 3 ปลายทั้งสองข้างของเส้นโค้งจะไม่แตะแกน X แต่จะค่อยๆลาดไปจรดแกนที่ $-\infty$ และ ∞ ตำแหน่งของจุดศูนย์กลางของโค้งจะถูกกำหนดโดยค่า μ และความกว้างจะถูกกำหนดด้วยค่า σ นั่นคือ โค้งปกติจะขึ้นอยู่กับพารามิเตอร์ 2 ตัว คือ ค่าเฉลี่ยของประชากร (μ) และส่วนเบี่ยงเบนมาตรฐาน (σ)

ถ้า μ มีค่าต่างกัน : ตำแหน่งของโค้งปกติต่างกัน

σ มีค่าต่างกัน : ความกว้างของโค้งปกติต่างกัน

ถ้า X เป็นตัวแปรสุ่มที่มีการแจกแจงแบบปกติ มีค่าเฉลี่ยท่ากับ μ และความแปรปรวนเท่ากับ σ^2 เขียนในรูปสัญญลักษณ์ $X \sim N(\mu, \sigma^2)$

ฟังก์ชันความน่าจะเป็น (Probability density function, p.d.f.) คือ f(x)

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-1/2\left(\frac{x-\mu}{\sigma^2}\right)^2}$$

$\pi = 3.1416$, $e = 2.7182$, $-\infty < x < \infty$

พื้นที่ใต้โค้งทั้งหมดมีค่าเท่ากับ 1 (โดยแต่ละซีกนับจากค่า μ มีพื้นที่ = 0.5)

โค้งปกติมาตรฐาน คือการแจกแจงปกติที่มีค่าเฉลี่ยเท่ากับ 0 และความแปรปรวนเท่ากับ 1 จะใช้ Z แทนตัวแปรสุ่มปกติมาตรฐาน (หรือคะแนนมาตรฐาน Standard score) เขียนเป็น $Z \sim N(0,1)$

ความสัมพันธ์ระหว่าง $X \sim N(\mu, \sigma^2)$ กับ $Z \sim N(0,1)$ คือ

$$Z = \frac{x-\mu}{\sigma}$$

ตัวอย่าง 5.13 ถ้า $X \sim N(10,4)$ จงหาค่า Z เมื่อ x= 8

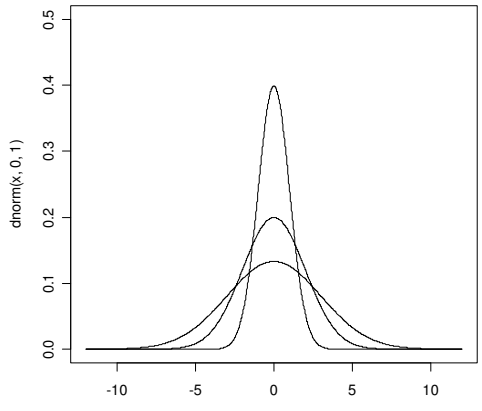
```
> Z<- (8-10)/2
> Z
[1] -1
```

5.3.1 กราฟของการแจกแจงแบบปกติ

สามารถวาดกราฟของการแจกแจงปกติได้โดยคำนวณค่า density ด้วยฟังก์ชัน dnorm(…) ในช่วงค่าที่กำหนด

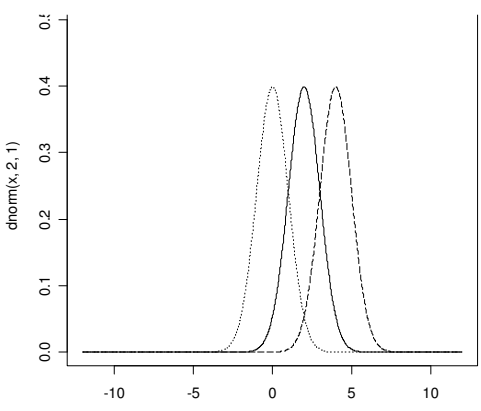
ตัวอย่าง 5.14 วาดกราฟของการแจกแจงปกติเพื่อเปรียบเทียบรูปร่างกราฟที่มี μ ต่างกัน แต่ σ เท่ากันและที่ μ เท่ากันแต่ σ ต่างกัน

```
> x<- seq(from=-12,to=12,0.01)
> plot(x,dnorm(x,0,1),type="n",
    ylim=c(0,.5))
> points(x,dnorm(x,0,1),type="l'
> points(x,dnorm(x,0,2),type="l'
> points(x,dnorm(x,0,3),type="l'
```



μ เท่ากันแต่ σ ต่างกัน

```
> plot(x,dnorm(x,2,1),type="n",
    ylim=c(0,.5))
> points(x,dnorm(x,2,1),type="l")
> points(x,dnorm(x,4,1),type="l",lty=5)
> points(x,dnorm(x,0,1),type="l",lty=3)
```

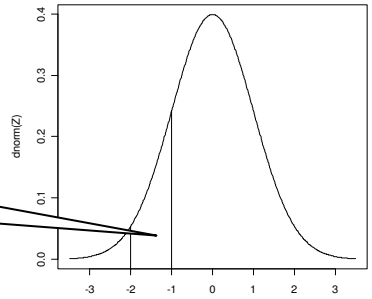


μ ต่างกันแต่ σ เท่ากัน

ตัวอย่าง 5.15 วาดกราฟของ Z~ N(0,1) และลากเส้นแสดงพื้นที่ใต้โค้งที่อยู่ระหว่าง -2 กับ -1

```
> Z<- seq(from=-3.5,to=3.5,0.01)
> plot(Z,dnorm(Z),type="l")
> segments(-1,-0.2,-1,dnorm(-1))
> segments(-2,-0.2,-2,dnorm(-2))
```

พื้นที่ใต้โค้งที่อยู่
ระหว่าง -2 กับ -1



5.3.2 การคำนวณค่าความน่าจะเป็นและความน่าจะเป็นสะสม

ความน่าจะเป็นที่ X จะอยู่ระหว่าง [a,b] หรือ P(a < X < b) คือพื้นที่ที่ล้อมด้วยเส้นโค้งปกติที่อยู่ระหว่างค่า a ถึง b

ชุดคำสั่งที่ใช้ใน R

วิธีใช้

คำนวณค่าf(x)=P(X=x): dnorm(x, mean=0, sd=1, log = FALSE)

คำนวณค่าF(x)=P(X ≤ q): pnorm(q, mean=0, sd=1, lower.tail = TRUE)

หาค่า x ที่ P(X ≤ x)= p: qnorm(p, mean=0, sd=1, lower.tail = TRUE, log.p

x,q: ค่าของตัวแปรสุ่ม

p: ค่าความน่าจะเป็นที่ระบุ P(X ≤ x)

ตัวอย่าง 5.16 คำนวณความน่าจะเป็นสะสม P(X ≤ -1.96) และ P(X ≥ 1.96)

หาค่า x เมื่อ P(X ≤ x) = 0.025 และ หาค่า x เมื่อ P(X ≥ 1.96)=0.025

1) P(X ≤ -1.96)

```
> round(pnorm(-1.96,0,1),5)
[1] 0.025
```

2) หาค่า x เมื่อ P(X ≤ x) = 0.025

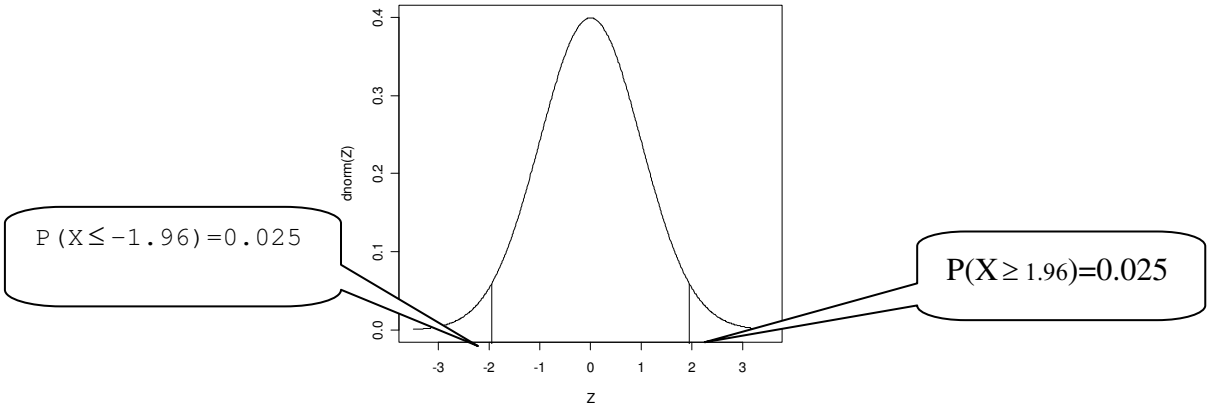
```
> qnorm(0.025, mean=0, sd=1)
[1] -1.959964
```

3) P(X ≥ 1.96)

```
> round(pnorm(1.96,0,1,lower.tail=F),5)
[1] 0.025
```

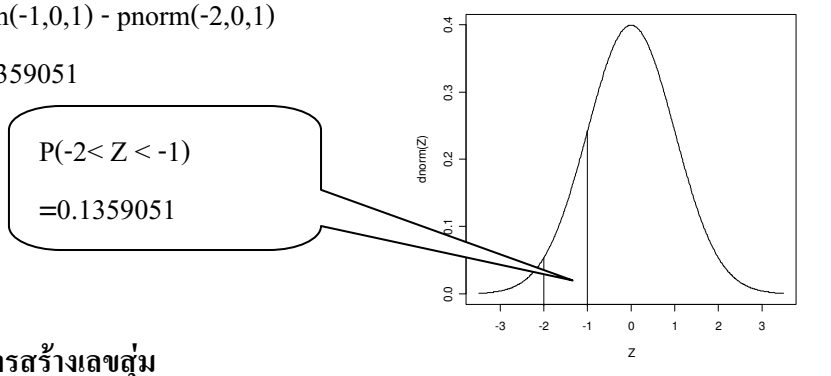
4) หาค่า x เมื่อ P(X ≥ 1.96)=0.025

```
qnorm(0.025,0,1, lower.tail = F)
[1] 1.959964
```



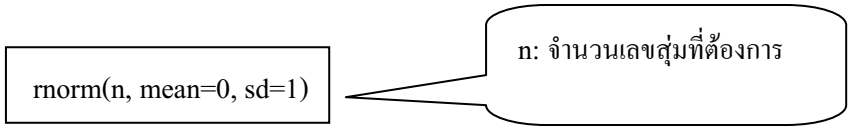
ตัวอย่าง 5.17 คำนวณค่า P(-2< Z < -1)

```
> pnorm(-1,0,1) - pnorm(-2,0,1)
 [1] 0.1359051
```



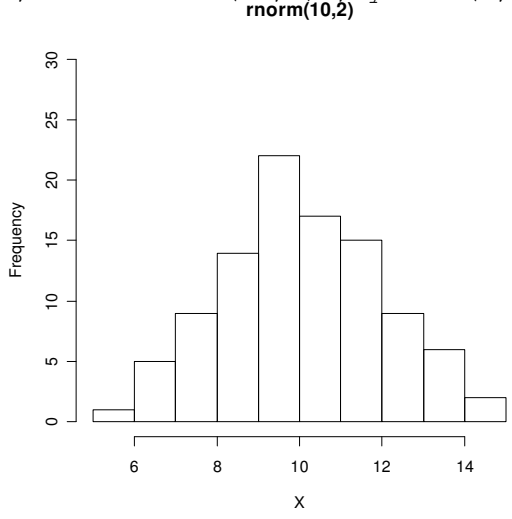
5.3.3 การสร้างเลขสุ่ม

วิธีใช้



ตัวอย่าง 5.18 สร้างเลขสุ่มที่มีการแจกแจงแบบปกติที่มีค่าเฉลี่ย= 10, sd = 2 จำนวน 100 ตัว

```
> X<- rnorm(100, mean=10, sd=2)
> hist(X, main="rnorm(10,2)", ylim=c(0,20))
```



5.4 การแจกแจงความน่าจะเป็นของค่าเฉลี่ยตัวอย่าง ($\bar{X}$) (Sampling distribution of the sample mean)

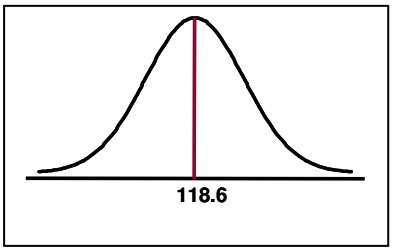
5.4.1 การแจกแจงของค่าเฉลี่ยตัวอย่างที่สุ่มมาจากประชากรที่มีการแจกแจงปกติและ ทราบค่า σ

ถ้ามีประชากรขนาด N ที่มีค่าเฉลี่ยเท่ากับ μ และ แปรปรวนเท่ากับ σ^2 (แทนการแจกแจงของประชากรด้วย $X \sim N(\mu, \sigma^2)$ หากเราเลือกตัวอย่างอย่างสุ่มขนาด n หน่วยจากประชากรชุดนี้ เราจะได้ว่า $\bar{x}_i$ ของตัวอย่างทุกชุดที่เป็นไปได้จะมีการแจกแจงแบบปกติ $\bar{X} \sim N(\mu_{\bar{X}}, \sigma^2_{\bar{X}})$ ที่ ค่าเฉลี่ยของค่าเฉลี่ยตัวอย่าง ($\mu_{\bar{x}}$) มีค่าเท่ากับค่าเฉลี่ยของประชากร (μ) นั่นคือ $\mu_{\bar{x}} = \mu$ และความแปรปรวนของค่าเฉลี่ยตัวอย่าง $\sigma^2_{\bar{x}}$ มีค่าเท่ากับ $\frac{\sigma^2}{n}$ และความคลาดเคลื่อนมาตรฐาน (Standard Error, SE) $\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{n}}$

เมื่อ $\bar{X} \sim N(\mu_{\bar{X}}, \sigma^2_{\bar{X}})$ ดังนั้น

$$Z = \frac{\bar{x} - \mu_{\bar{x}}}{\sigma_{\bar{x}}} = \frac{\bar{x} - \mu}{\sigma/\sqrt{n}}$$

ตัวอย่าง 5.19 เรามีข้อมูลน้ำหนักเด็กแรกเกิด (Birth weight, wt) จำนวน 150 คน ที่มีการแจกแจงแบบปกติ มี μ = 118.6 ออนซ์, σ = 17.8 ออนซ์
เราวาดกราฟดูการแจกแจงได้ดังรูป
เราจะถือว่าประชากรของเรา คือ เด็กแรกเกิด 150 คน
และลักษณะของประชากรที่เราสนใจ คือ น้ำหนัก(wt)



ถ้าเลือกตัวอย่างขนาด 10 คน จากเด็กกลุ่มนี้ เราจะได้ว่า
$\bar{x}$ จะมีการแจกแจงแบบปกติ มี $\mu_{\bar{x}} = \mu$ = 118.6 ออนซ์, $\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{n}} = \frac{17.8}{\sqrt{10}} = 5.63$ ออนซ์
จากตัวอย่างนี้ เราเลือกตัวอย่างสุ่มมาเพียงชุดเดียว แต่เราทราบการแจกแจงของ $\bar{x}$ ได้จากทฤษฎีทางสถิติดังกล่าวข้างต้น แต่ตัวอย่างต่อไปจะแสดงให้เห็นว่าทฤษฎีเกี่ยวกับการแจกแจงของ $\bar{x}$ เป็นจริงได้โดยการเลือกตัวอย่างสุ่มหลายๆ ชุด

ตัวอย่าง 5.20 เลือกตัวอย่างอย่างสุ่ม (แบบไม่ใส่คืน, with out replacement) มา 150 ชุด แต่ละชุดมีขนาดตัวอย่างเท่ากับ 10 หน่วย (n = 10)
คำนวณค่าเฉลี่ยได้ $\overline{x}_1, \ldots, \overline{x}_k$ นำค่าเฉลี่ยตัวอย่าง $\overline{x}_i$ เหล่านี้มาคำนวณค่าเฉลี่ย $\mu_{\overline{x}}$, ส่วนเบี่ยงเบนมาตรฐาน (ความคลาดเคลื่อนมาตรฐาน SE, $\sigma_{\overline{x}}$) และวาด histogram เพื่อดูการแจกแจง ได้โดยใช้ฟังก์ชันที่เขียนขึ้นดังนี้

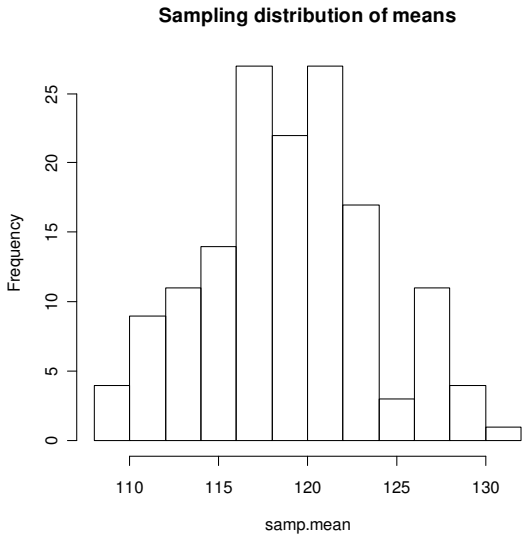
```
## function called 'srs.out'
srs.out<-function(x,n,R){
        cmx <- NULL
        sdmx <- NULL
        for(i in 1:R){
             srs.x <- sample(x,n)
              mx <- mean(srs.x, na.rm=T)
             cmx <- c(cmx,mx)
             sdx <- sd(srs.x,na.rm=T )
            sdmx <- c(sdmx,sdx)
                    }
        samp.mean <- cmx ; samp.sd <- sdmx
         samp <- cbind(samp.mean, samp.sd)
        samp
         hist(samp.mean,main="Sampling distribution of means")
        print(mu.xbar<-mean(samp.mean))
        print(se<- sd(samp.mean))
}
```

การใช้ function: `srs.out(x,n,R)`; x = population to be sampled
n= sample size, R =no. of repeating

```
> srs.out(baby23.dat,10,50)
> attach(b.dat)
> srs.out(wt,10,150)
```

```
[1] 118.6758
```
→ มีค่าใกล้เคียงกับ μ คือ 118.6, σ = 17.8

```
[1] 5.568346
```
→ มีค่าใกล้เคียงกับ 17.8/sqrt10)=[1] 5.628854



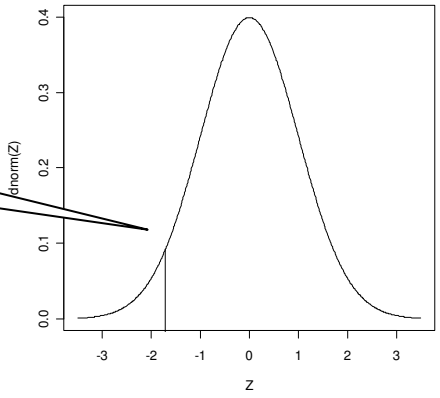
ตัวอย่าง 5.21 ถ้าเลือกตัวอย่างจำนวน 30 หน่วยมาอย่างสุ่มจากประชากรที่มีการแจกแจงปกติที่มี μ เท่ากับ 20 และ σ เท่ากับ 4.8 ให้คำนวณความน่าจะเป็นที่จะเลือกได้ตัวอย่างที่มีค่าเฉลี่ยต่ำกว่า 18.5

```
 z <- (18.5-20)/(4.8/sqrt(30))

> z
[1] -1.711633
```

เปลี่ยนค่า $\bar{x}$ เป็นคะแนนมาตรฐาน Z แล้วหาพื้นที่ที่ $\bar{x} < 18.5$ ซึ่งก็คือ $P(\bar{x} < 18.5)$

```
> pnorm(-1.711633)
[1] 0.04348216
```

$P(\bar{x} < 18.5) = 0.04348$



5.4.2 การแจกแจงของค่าเฉลี่ยตัวอย่างที่สุ่มมาจากประชากรที่มีการแจกแจงปกติแต่ไม่ทราบค่า σ

ในกรณีที่ไม่ทราบค่าส่วนเบี่ยงเบนมาตรฐานของประชากร (σ) เราจะประมาณค่า σ ด้วยส่วนเบี่ยงเบนมาตรฐานของตัวอย่าง (s) และการแจกแจงของ $\bar{x}$ จะเปลี่ยนเป็นการแจกแจงแบบ student – t ดังนั้น ข้อสรุปจากข้อมูลตัวอย่างไปยังประชากร จะเป็นดังนี้ 1) ค่าเฉลี่ยของค่าเฉลี่ยตัวอย่าง ($\mu_{\bar{x}}$) มีค่าเท่ากับค่าเฉลี่ยของประชากร (μ) [i.e., $\mu_{\bar{x}} = \mu$] 2) ค่าประมาณของความแปรปรวนของค่าเฉลี่ยตัวอย่าง $s^2_{\bar{x}}$ มีค่าเท่ากับ $\frac{s^2}{n}$ [i.e., $s^2_{\bar{x}} = \frac{s^2}{n}$] และ ค่าประมาณของส่วนเบี่ยงเบนมาตรฐานของ $\bar{x}$ หรือเรียกชื่อใหม่ว่า ความคลาดเคลื่อนมาตรฐาน มีค่าเท่ากับ $\frac{s}{\sqrt{n}}$ [i.e., $s_{\bar{x}} = \frac{s}{\sqrt{n}}$] และ 3) ถ้าประชากรมีการแจกแจงแบบปกติ ค่าเฉลี่ยตัวอย่าง ($\bar{x}$) จะมีการแจกแจงแบบ student-t ที่มีค่าขึ้นอยู่กับ ค่า degree of freedom ซึ่งมีค่าเท่ากับ n-1 ดังนั้น

$$t = \frac{\bar{x}-\mu}{s/\sqrt{n}}, \quad df = n-1$$

5.5 การหาค่าตัวแปรสุ่มและความน่าจะเป็นของตัวแปรสุ่มที่มีการแจกแจงแบบ student - t

ชุดคำสั่งที่ใช้ใน **R** มีวิธีใช้ทำนองเดียวกับการแจกแจงแบบปกติเพียงแต่ต้องระบุค่า degree of freedom ของการแจกแจงแบบ student – t แทนที่จะระบุค่า m s ดังเช่นการแจกแจงแบบปกติ

วิธีใช้

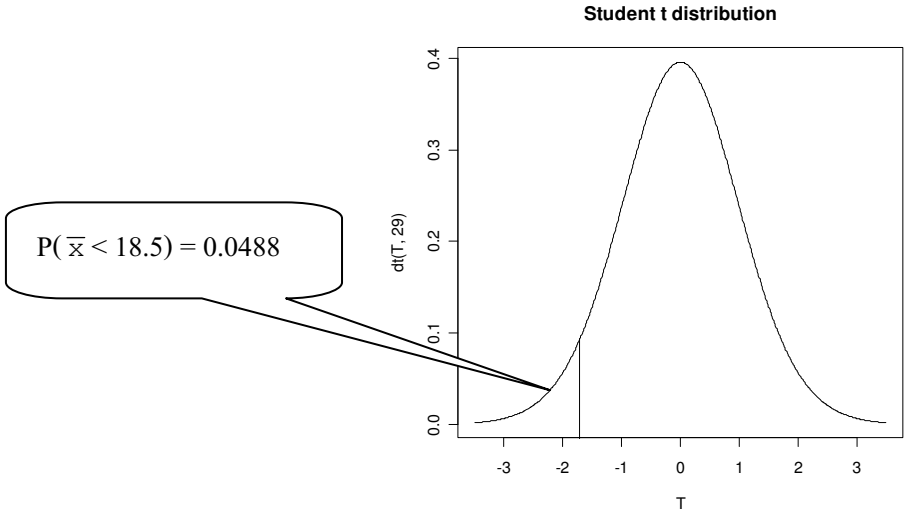
คำนวณค่า `f(x)=P(X=x)` เมื่อ `df=n-1: dt(x, df, log = FALSE)`
คำนวณค่า `F(x)=P(X≤q): pt(q, df, lower.tail = TRUE, log.p = FALSE)`
หาค่า `x` ที่ `P(X≤x)= p: qt(p, df, lower.tail = TRUE, log.p = FALSE)`
สร้างตัวแปรสุ่ม: `rt(n, df)`

ตัวอย่าง 5.22 ถ้าเลือกตัวอย่างจำนวน 30 หน่วยมาอย่างสุ่มจากประชากรที่มีการแจกแจงปกติที่มี μ เท่ากับ 20 แต่ไม่ทราบค่า σ จากตัวอย่างพบว่าค่า s = 4.8 ให้คำนวณความน่าจะเป็นที่จะเลือกได้ตัวอย่างที่มีค่าเฉลี่ยต่ำกว่า 18.5

```
>T <- seq(from=-3.5,to=3.5,0.01)
>plot(T,dt(T,29),type="l",
    main="Student t distribution")
>t<- (18.5-20)/(4.8/sqrt(30))
>t
[1]  -1.711633
>segments(t,t,t,dt(t,29))
```

วาดกราฟการแจกแจง t ที่ df = 29 และ หาค่า $P(\bar{x} < 18.5)$

```
> pt(-1.711633,29)
[1] 0.04882049
```



แบบฝึกหัด

1. ถ้าข้อมูลประชากรที่สนใจ คือข้อมูลในแฟ้ม baby23.dat เลือกตัวอย่างขนาด 25 หน่วย 1 ชุด จากแฟ้มข้อมูลนี้ จากนั้น

   1) ตรวจสอบการแจกแจงของประชากร (ตัวแปรที่สนใจคือ gestation)

   2) จงหา ค่าเฉลี่ย และ sd ของ gestation ในกลุ่มตัวอย่าง

   3) หาการแจกแจงของค่าเฉลี่ยตัวอย่างของ gestation (ชนิดการแจกแจง, $\mu_{\bar{x}}$, $\sigma^2_{\bar{x}}$)

   4) หาความน่าจะเป็นที่ค่าเฉลี่ยตัวอย่าง $\bar{x}$ ที่เลือกมาได้ จะมีค่าต่างจาก ค่า μ ไป 5 หน่วย

บทที่ 6
การทดสอบสมมติฐานและประมาณค่าพารามิเตอร์ของประชากร
(Hypothesis Testing & Interval Estimation)

การทดสอบสมมติฐานและประมาณค่าพารามิเตอร์ของประชากรเป็นระเบียบวิธีวิเคราะห์ข้อมูลขั้นพื้นฐานที่สำคัญที่ใช้ข้อมูลจากตัวอย่างสุ่มขนาด n ซึ่งเรียกว่า สถิติ (statistic) ไปสรุปถึงพารามิเตอร์ของประชากร โดยสามารถบอกระดับของความผิดพลาดที่อาจเกิดขึ้นได้ ค่าพารามิเตอร์ของประชากรที่พิจารณาในบทนี้ ได้แก่ ค่าเฉลี่ย (μ) ค่าสัดส่วน (p) และค่าความแปรปรวน (σ^2)

6.1 ฟังก์ชัน R สำหรับการทดสอบสมมติฐานและการประมาณค่าแบบช่วงของประชากรหนึ่งและสองกลุ่ม

โปรแกรมสำเร็จรูป **R** รวบรวมฟังก์ชันหรือคำสั่งสำหรับ การทดสอบสมมติฐานและการประมาณค่าแบบช่วงพารามิเตอร์ของประชากรไว้ในคลังคำสั่งชื่อ stats (ตรวจสอบว่าคลังคำสั่ง stats รวบรวมฟังก์ชันอะไรไว้บ้างให้ใช้คำสั่ง help.start() ฟังก์ชัน **R** สำหรับการทดสอบสมมติฐานและการประมาณ ค่าแบบช่วงของ ค่าเฉลี่ย ค่าสัดส่วน และค่าความแปรปรวน ของประชากร มีดังนี้

คำสั่ง/ฟังก์ชันเขียนที่มีใน R

ฟังก์ชัน/คำสั่งใน R	คำอธิบาย
t.test()	ทดสอบและประมาณค่าแบบช่วงค่าเฉลี่ยของประชากรเมื่อประชากรมีการแจกแจงแบบปกติและใช้ตัวอย่างขนาดเล็ก
var.test()	ทดสอบและประมาณค่าแบบช่วงค่าความแปรปรวนของประชากร
prop.test	ทดสอบและประมาณค่าแบบช่วงค่าสัดส่วนของประชากรโดยใช้แบบทดสอบไคกำลังสอง
binom.test()	ทดสอบและประมาณค่าแบบช่วงค่าสัดส่วนของประชากรทวินาม

คำสั่ง/ฟังก์ชันเขียนขึ้นใหม่โดยทีมวิจัย

ฟังก์ชัน/คำสั่งใน R	คำอธิบาย
z.test	ทดสอบและประมาณค่าแบบช่วงค่าเฉลี่ยของประชากรเมื่อประชากรมีการแจกแจงแบบปกติและทราบค่าความแปรปรวนของประชากร หรือใช้ตัวอย่างขนาดใหญ่ สำหรับกรณีที่มีข้อมูลดิบ
onemean.ztest()	ทดสอบและประมาณค่าแบบช่วงค่าเฉลี่ยของประชากร 1 กลุ่มเมื่อประชากรมีการแจกแจงแบบปกติและทราบค่าความแปรปรวนของประชากร หรือใช้ตัวอย่างขนาดใหญ่ สำหรับกรณีที่ไม่มีข้อมูลดิบ
onemean.ttest()	ทดสอบและประมาณค่าแบบช่วงค่าเฉลี่ยของประชากร 1 กลุ่ม เมื่อประชากรมีการแจกแจงแบบปกติและใช้ตัวอย่างขนาดเล็ก สำหรับกรณีที่ไม่มีข้อมูลดิบ
oneprop.ztest()	ทดสอบและประมาณค่าแบบช่วงค่าสัดส่วนของประชากร 1 กลุ่ม สำหรับกรณีที่ไม่มีข้อมูลดิบ
twomean.ttest()	ทดสอบและประมาณค่าแบบช่วงค่าเฉลี่ยประชากร 2 กลุ่ม สำหรับกรณีที่ไม่มีข้อมูลดิบ
twoprop.ztest()	ทดสอบและประมาณค่าแบบช่วงค่าสัดส่วนประชากร 2 กลุ่ม สำหรับกรณีที่ไม่มีข้อมูลดิบ

6.2 Birth weight data

ในการสาธิตการใช้ ฟังก์ชัน **R** สำหรับการทดสอบสมมติฐานและการประมาณค่าแบบช่วงค่าเฉลี่ยของประชากร เราใช้ Birth weight data ซึ่ง download จาก http://stat-www.berkeley.edu/users/statlabs/lab.html ข้อมูลดังกล่าวถูกใช้ในการศึกษาน้ำหนักของทารกแรกเกิดกับประวัติหรือภูมิหลังโดยเฉพาะพฤติกรรมการสูบบุหรี่ของมารดา ในการเตรียมข้อมูลสำหรับ workshop นี้เราเลือกตัวแปรที่สนใจจำนวน 13 ตัวแปร คือ gestation, sex, wt, parity, race, ed, age, marital, momwt, income, smoke, time และ number และสุ่มมาเป็นกรณีศึกษาเพื่อสาธิตการใช้โปรแกรม **R** จำนวน 150 cases และมีบาง cases ที่มี missing values ซึ่ง **R** จะรับรู้ว่าเป็น missing values โดยใช้รหัส NA ข้อมูลดังกล่าวถูกเก็บไว้ในลักษณะ text file ชื่อ "`baby23.txt`" ซึ่งคำอธิบายเกี่ยวกับตัวแปรได้แสดงไว้แล้วในหัวข้อ 2.4.1

6.3 การทดสอบสมมติฐานและการประมาณค่าแบบช่วงค่าเฉลี่ยของประชากร

6.3.1 การทดสอบสมมติฐานและการประมาณค่าแบบช่วงค่าเฉลี่ยของประชากรหนึ่งกลุ่ม

การทดสอบสมมติฐานเกี่ยวกับค่าเฉลี่ยสำหรับของประชากรหนึ่งกลุ่มเป็นการทดสอบว่าค่าเฉลี่ยของตัวแปรใดตัวแปรหนึ่งที่สนใจของประชากรซึ่งแทนด้วยสัญลักษณ์ μ มีค่าแตกต่างจากค่าเฉลี่ยที่คาดเดาไว้หรือไม่ โดยสถิติที่ใช้ทดสอบได้แก่ z-test หรือ t-test สำหรับโปรแกรม **R** คำสั่งที่สอดคล้องคือ **z.test()** (ฟังก์ชันนี้เขียนขึ้นใหม่โดยทีมวิจัย) ซึ่งรวบรวมอยู่ใน text file ชื่อ **Statcan.txt** หรือ **t.test() ซึ่งมีอยู่แล้วใน R** ตามลำดับ คำสั่งทั้งสองมีรูปแบบการใช้ที่เหมือนกัน ดังนี้

z.test วิธีใช้

```
z.test(x, y = NULL, alternative = c("two.sided", "less", "greater"),
 mu = 0, paired = FALSE,  var.equal = FALSE, conf.level = 0.95, ...)
```

t.test วิธีใช้

```
t.test(x, y = NULL, alternative = c("two.sided", "less", "greater"),
  mu = 0, paired = FALSE,  var.equal = FALSE, conf.level = 0.95, ...)
```

x : ตัวแปรหรือเวคเตอร์ของของตัวแปรสุ่มแบบต่อเนื่องจากประชากรหรือกลุ่มที่หนึ่ง

y : ตัวแปรหรือเวคเตอร์ของของตัวแปรสุ่มแบบต่อเนื่องจากกลุ่มที่สอง (กรณีทดสอบผลต่างของค่าเฉลี่ย)

alternative: ระบุสมมติฐานทางเลือก "two.sided" (ทดสอบแบบสองหาง), "greater" (ทดสอบแบบหางเดียวทางขวา) หรือ "less" (ทดสอบแบบหางเดียวทางซ้าย) อย่างใดอย่างหนึ่ง นอกจากนี้ผู้ใช้สามารถระบุโดยใช้เพียง อักษรตัวแรก คือ " t ", " g " หรือ " l " ก็ได้

mu: ตัวเลขหรือสเกลาร์หรือค่าของค่าเฉลี่ย (หรือผลต่างของค่าเฉลีย) ของประชากรที่ต้องการทดสอบ

paired: เงื่อนไขที่ระบุว่า เป็นข้อมูลแบบจับคู่ (paired sample) ค่าปกติคือ FALSE ซึ่งหมายถึงตัวอย่างสุ่มทั้งสองกลุ่มเป็นอิสระต่อกัน

var.equal: เงื่อนไขที่ระบุว่า ความแปรปรวนของประชากรสองกลุ่มเท่ากัน หรือไม่ ถ้าระบุว่าเท่า (var.equal = TRUE) ก็จะมีการคำนวณ ความแปรปรวนร่วม หรือ pooled variance แต่ค่าปกติคือ FALSE

conf.level: ระดับความเชื่อมั่นที่ใช้ทดสอบ ค่าปกติคือ 0.95

ตัวอย่าง 6.1 Birth weight data : ต้องการทดสอบว่าน้ำหนักของทารกแรกเกิดโดยเฉลี่ยของประชากรศึกษามีค่ามากกว่า 3,285 กรัม หรือ 115 ออนซ์ หรือไม่ กำหนดระดับนัยสำคัญ 0.05

สมมติฐานของการศึกษาคือ น้ำหนักเฉลี่ยของทารกแรกเกิดของประชากรศึกษามีค่ามากกว่า 3,285 กรัม หรือ 115 ออนซ์

Data preparation

```
> baby23.dat <- read.table("baby23.txt", header=T)
> attach(baby23.dat)
> # เรียกชุดคำสั่งที่รวบรวมไว้ใน StatCan.txt มาใช้ใน R
> source("StatCan.txt")
Department of Mathematics, Prince of Songkla University,
StatCan version 1.1.1 alpha, 2006-08-09
```

ทดสอบโดยใช้ z.test

$H_0 : \mu = 115$ vs $H_1 : \mu > 115$

```
> # import StatCan.txt into R working directory
> z.test(wt, alternative ="g", mu = 115, conf.level = 0.95)

        One Sample z-test

data:  wt

❶ z = 2.4527, p-value = 0.007089
  alternative hypothesis: true mean is greater than 115
  95 percent confidence interval:
  116.1898      Inf
  sample estimates:
  mean of x
❷ 118.6122
```

$\bar{X} - z_{0.95} s_{\bar{X}}$

= 118.6122 - orm(.95)*sd(wt)/sqrt(length(wt))

❶ ค่าสถิติที่ใช้ทดสอบ $z = \dfrac{\bar{X} - \mu}{S/\sqrt{n}}$

❷ ค่าเฉลี่ยของ birth weights $\bar{X} = \sum \dfrac{x}{n}$

ทดสอบโดยใช้ t.test

ดำเนินการทดสอบสมมติฐานข้างต้นโดยใช้ **t.test** ดังแสดงข้างล่าง ก็จะได้ผลสรุปทำนองเดียวกัน เพียงแต่ค่าแตกต่างกันเล็กน้อย

$H_0 : \mu = 115$ vs $H_1 : \mu > 115$

```
> t.test(wt, alternative ="g", mu = 115, conf.level = 0.95)

        One Sample t-test

data:  wt
t = 2.4527, df = 146, p-value = 0.007678
alternative hypothesis: true mean is greater than 115
95 percent confidence interval:
 116.1743      Inf
sample estimates:
mean of x
 118.6122
```

$\bar{X} - t_{0.95,128} s_{\bar{X}}$

= 118.4186-qt(0.95, 128)*sd(wt)/sqrt(length(wt))

สรุปผล จากข้อมูลที่รวบรวมจากตัวอย่างสุ่มสรุปได้ว่าน้ำหนักเฉลี่ยของทารกแรกเกิดจากประชากรศึกษามีค่ามากกว่า 3,285 กรัม หรือ 115 ออนซ์ อย่างมีนัยสำคัญทางสถิติ (ระดับนัยสำคัญ 0.05)

หมายเหตุ

1) ถ้าระบุ alternative = "two.sided" ช่วงความเชื่อมั่นของ μ จะถูกคำนวนโดยใช้สูตร

$$\left(\bar{X} - Z_{1-\frac{\alpha}{2}} \frac{S}{\sqrt{n}}, \bar{X} + Z_{1-\frac{\alpha}{2}} \frac{S}{\sqrt{n}} \right)$$

2) ถ้าระบุ alternative = "less" ช่วงความเชื่อมั่นของ μ จะถูกคำนวนโดยใช้สูตร

$$\left(-\infty, \bar{X} + Z_{1-\alpha} \frac{S}{\sqrt{n}} \right)$$

3) ถ้าระบุ alternative = "greater" ช่วงความเชื่อมั่นของ μ จะถูกคำนวนโดยใช้สูตร

$$\left(\bar{X} - Z_{1-\alpha} \frac{S}{\sqrt{n}}, \infty \right)$$

ในกรณีที่เรามีค่าเฉลี่ยและค่าความแปรปรวนของตัวอย่างพร้อมกับขนาดของตัวอย่างอยู่ในมือและต้องการทดสอบสมมติฐาน พร้อมคำนวณช่วงความเชื่อมั่น $100(1-\alpha)\%$ (ใช้แนวคิดเดียวกับ "two.side" ทุกกรณี) เราสามารถเขียนฟังก์ชันขึ้นใช้เอง เช่น `onemean.ztest` () ถ้าใช้ z-test หรือ `onemean.ttest()` ถ้าใช้ t-test ดังตัวอย่างดังต่อไป นี้

ใช้ z-test

```
onemean.ztest <- function(mx, vx, mu = 0, n,
     alternative = c("two.sided", "less","greater"), conf.level=0.95)
{

# mx: mean of x or a sample mean;
# vx: variance of x or a sample variance
# n: sample size;
# mu: a number indicating the true value of the mean
# alternative: a character string specifying the alternative
# hypothesis must be one of '"two.sided"','"greater"'or '"less"'.

   stderr <- sqrt(vx/n)
   zstat <- (mx-mu)/stderr
     if (alternative == "less") {
      pval <- pnorm(zstat, lower.tail = T)
    }
    else if (alternative == "greater") {
      pval <- pnorm(zstat, lower.tail = F)
    }
    else {
      pval <- 2*pnorm(abs(zstat), lower.tail=F)
    }
      cint <- cbind(mx - qnorm((1-conf.level)/2,lower.tail = F)*stderr ,
              mx + qnorm((1-conf.level)/2,lower.tail = F)*stderr )
      colnames(cint) <- c("lwr", "uppr")


    rval <- list(statistic = zstat, Pvalue = pval, estimate=mx,
            null.value = mu, confidence.level=conf.level,
      confidence.interval=cint, alternative=alternative)
      return(rval)
 }
```

ใช้ t-test

```
onemean.ttest <- function(mx, vx, mu = 0, n,
   alternative = c("two.sided", "less", "greater"), conf.level = 0.95)
{

 # mx: mean of x or a sample mean;
 # vx: variance of x or a sample variance
 # n: sample size;
 # mu: a number indicating the true value of the mean
 # alternative: a character string specifying the alternative
 # hypothesis must be one of '"two.sided"', '"greater"' or '"less"'.

   stderr <- sqrt(vx/n)
   zstat <- (mx-mu)/stderr

     if (alternative == "less") {
      pval <- pt(zstat,n-1, lower.tail = T)
    }
    else if (alternative == "greater") {
      pval <- pnorm(zstat,n-1, lower.tail = F)
    }
    else {
      pval <- 2*pt(abs(zstat),n-1, lower.tail=F)

    }
      cint <- cbind(mx - qt((1-conf.level)/2,n-1,lower.tail = F)*stderr ,
                mx + qt((1-conf.level)/2,n-1,lower.tail = F)*stderr )

      colnames(cint) <- c("lwr", "uppr")

    rval <- list(statistic = zstat, Pvalue = pval,
            confidence.level=conf.level, confidence.interval=cint,
            estimate=mx, null.value = mu, alternative=alternative)
    return(rval)
}
```

หมายเหตุ ชุดคำสั่งทั้งสองชุดคำสั่งนี้ได้ถูกรวบรวมไว้ใน StatCan.txt แล้ว

ตัวอย่างการใช้ `onemean.ztest()` และ `onemean.ttest()`

ในกรณีของของตัวอย่าง 6.1 สมมติเราทราบว่า น้ำหนักเฉลี่ยและความแปรปรวนของน้ำหนักของของทารกตัวอย่าง 147 (ตัด case ที่มี NA) คน เป็น 118.6122 ounces และ 318.8418 ounces^2 ตามลำดับ จงทดสอบว่าน้ำหนักของทารกแรกเกิดโดยเฉลี่ยของประชากรศึกษามีค่ามากกว่า 3,285 กรัม หรือ 115 ออนซ์ หรือไม่ กำหนดระดับนัยสำคัญ 0.05

```
> # Import "StatCan.txt" into R
> source("StatCan.txt") ; mweight <- 118.6122
> vweight <-  318.8418; n <- 147

>## use z-test
> onemean.ztest(mweight, vweight, mu = 115, n,
                alternative = "greater")
$statistic
[1] 2.452722

$Pvalue
[1] 0.007089002

$estimate
[1] 118.6122

$null.value
[1] 115

$confidence.level
[1] 0.95

$confidence.interval
          lwr     uppr
[1,] 115.7257 121.4987

$alternative
[1] "greater"

>## use t-test
> onemean.ttest(mweight, vweight, mu = 115, n,
               alternative = "greater")
$statistic
[1] 2.452722

$Pvalue
[1] 1

$confidence.level
[1] 0.95

$confidence.interval
          lwr     uppr
[1,] 115.7015 121.5229

$estimate
[1] 118.6122

$null.value
[1] 115

$alternative
[1] "greater"
```

หมายเหตุ ผลลัพธ์ที่ได้จะใกล้เคียงกับใช้ข้อมูลดิบ ส่วนต่างเพียงเล็กน้อยเนื่องจากการปัดทศนิยม

6.3.2 การทดสอบสมมติฐานและการประมาณค่าแบบช่วงค่าเฉลี่ยของประชากรสองกลุ่ม

การทดสอบสมมติฐานเกี่ยวกับค่าเฉลี่ยสำหรับของประชากรสองกลุ่มเป็นการทดสอบเพื่อเปรียบเทียบ ว่าค่าเฉลี่ยของลักษณะใดลักษณะหนึ่ง ที่สนใจของประชากรทั้งสองกลุ่มแตกต่างกันหรือไม่อย่างไร มักแทนด้วยสัญลักษณ์ $\mu_1 - \mu_2$ เมื่อ μ_1 และ μ_2 แทนค่าเฉลี่ยของของประชากรกลุ่มที่ 1 และ กลุ่มที่ 2 ตามลำดับ

ตัวอย่าง 6.2 Birth weight data : ต้องการทดสอบว่าน้ำหนักของทารกแรกเกิดโดยเฉลี่ยของมารดาที่ไม่สูบบุหรี่ (μ_1) และสูบบุหรี่ (μ_2) ของประชากรศึกษา แตกต่างกันหรือไม่ กำหนดระดับนัยสำคัญ 0.01 (ระดับความเชื่อมั่น เท่ากับ 0.99) และ สมมติว่าความแปรปรวนของ birth weights ของทั้งสองกลุ่มมีค่าเท่ากัน

สมมติฐานของการศึกษาคือ น้ำหนักเฉลี่ยของทารกแรกเกิดที่มารดาไม่สูบบุหรี่ (μ_1) และที่สูบบุหรี่ (μ_2) ของประชากรศึกษา แตกต่างกัน

ใช้ z-test

$H_0 : \mu_1 - \mu_2 = 0$ vs $H_1 : \mu_1 - \mu_2 \neq 0$

```
> z.test(wt[smoke==0],wt[smoke==1], mu = 0, alternative = "two.sided",
var.equal=T, conf.level=0.99)
```

❶

```
        Two Sample z-test
```

❷ ❸

```
data:  wt[smoke == 0] and wt[smoke == 1]
z = 3.2565, p-value = 0.001128
alternative hypothesis: true difference in means is not equal to 0
99 percent confidence interval:
  2.005865 17.186766
sample estimates:
mean of x mean of y
 122.9844  113.3881
```

$\left(\bar{X}_1 - \bar{X}_2\right) \pm z_{0.995} s_{\bar{X}_1 - \bar{X}_2}$

ใช้ t-test

```
> t.test(wt[smoke==0],wt[smoke==1], mu = 0, alternative = "two.sided",
var.equal=T, conf.level=0.99)
 ❶

       Two Sample t-test

            ❷                ❸
data:  wt[smoke == 0] and wt[smoke == 1]
t = 3.2565, df = 129, p-value = 0.001441
alternative hypothesis: true difference in means is not equal to 0
99 percent confidence interval:
  1.891973 17.300658
sample estimates:
mean of x mean of y
 122.9844  113.3881
```

$$\left(\bar{X}_1 - \bar{X}_2\right) \pm t_{0.995,129} s_{\bar{X}_1 - \bar{X}_2}$$

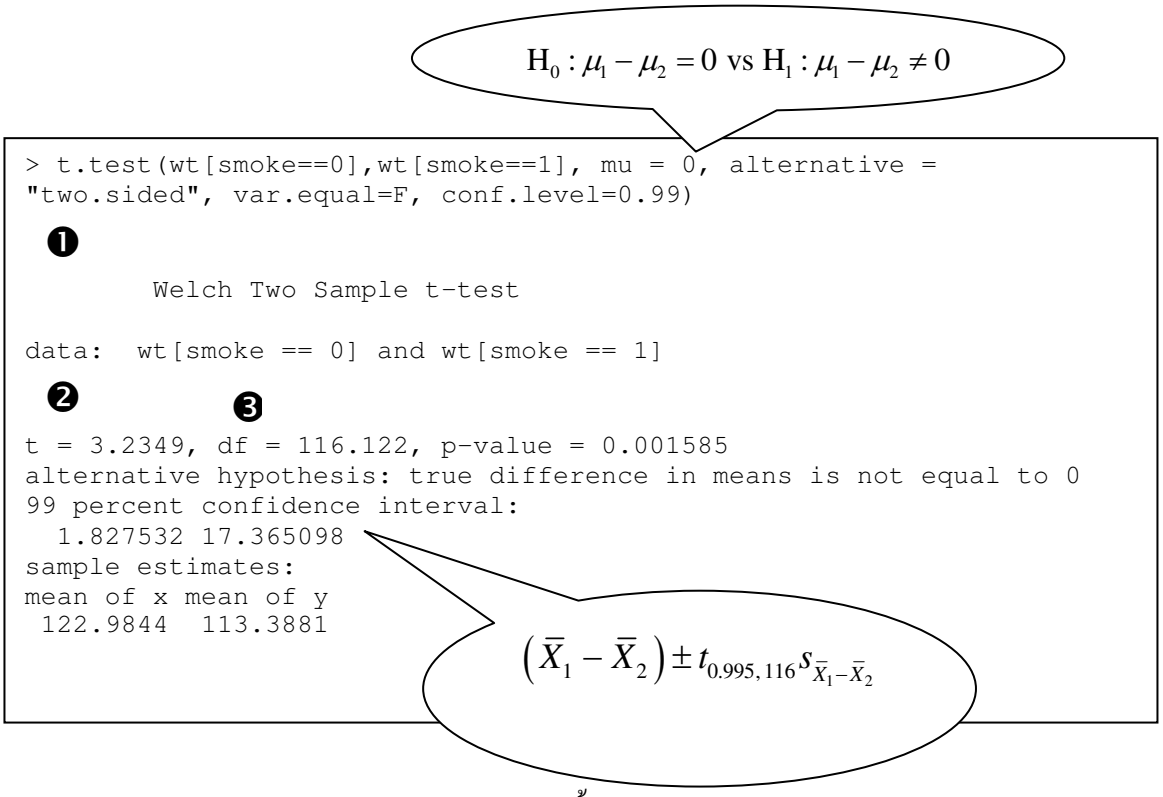
❶ ระบุว่าความแปรปรวนของ birth weights ของทั้งสองกลุ่มมีค่าเท่ากัน

❷ wt[smoke == 0] เป็นดัชนีที่ไปจับ birth weights ของกลุ่มที่มารดาไม่สูบบุหรี่ ซึ่ง R ระบุเป็น x

❸ wt[smoke == 1] เป็นดัชนีที่ไปจับ birth weights ของกลุ่มที่มารดาสูบบุหรี่ ซึ่ง R ระบุเป็น y

สรุป ทดสอบสมมติฐานโดยใช้แบบทดสอบทั้งสองแบบให้ข้อสรุปเช่นเดียวกัน คือ น้ำหนักเฉลี่ยของทารกแรกเกิดที่มารดาไม่สูบบุหรี่ (μ_1) และที่สูบบุหรี่ (μ_2) ของประชากรศึกษา แตกต่างกันอย่างมีนัยสำคัญยิ่งทางสถิติ (ระดับนัยสำคัญ 0.05)

ตัวอย่าง 6.3 Birth weight data : ต้องการทดสอบว่าน้ำหนักของทารกแรกเกิดโดยเฉลี่ยของมารดาที่ไม่สูบบุหรี่ (μ_1) และสูบบุหรี่ (μ_2) ของประชากรศึกษา แตกต่างกันหรือไม่โดยใช้สถิติ t กำหนดระดับนัยสำคัญ 0.01 (ระดับความเชื่อมั่น เท่ากับ 0.99) และ สมมติว่าความแปรปรวนของ birth weights ของทั้งสองกลุ่มมีค่าไม่เท่ากัน



❶ ระบุว่าความแปรปรวนของ birth weights ของทั้งสองกลุ่มมีค่าไม่เท่ากัน

❷ สถิติ $t = \dfrac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{\sqrt{\dfrac{S_1^2}{n_1} + \dfrac{S_2^2}{n_2}}}$

❸ Pooled degrees of freedom, $\mathrm{df} = \dfrac{\left(S_1^2/n_1 + S_2^2/n_2\right)^2}{\dfrac{\left(S_1^2/n_1\right)^2}{n_1 - 1} + \dfrac{\left(S_2^2/n_2\right)^2}{n_2 - 1}}$

หมายเหตุ ในทำนองเดียวกันเราสามารถเขียนฟังก์ชันขึ้นใช้เองทดสอบสมมติฐาน พร้อมคำนวณช่วงความเชื่อมั่น $100(1-\alpha)\%$ ในกรณีที่เรามีค่าเฉลี่ย ค่าความแปรปรวนของตัวอย่างและ ขนาดของตัวอย่างของทั้งสองกลุ่ม ใช้แนวคิดเดียวกับ `onemean.ztest` ถ้าใช้ z-test หรือ `onemean.ttest` ถ้าใช้ t-test

6.4 การทดสอบสมมติฐานและการประมาณค่าแบบช่วงค่าความแปรปรวนของประชากรสองกลุ่ม

การทดสอบสมมติฐานเกี่ยวกับค่าความแปรปรวนของประชากรสองกลุ่มเป็นการตรวจสอบว่าความแปรปรวนของประชากรสองกลุ่มมีค่าแตกต่างกัน หรือไม่ อย่างไร ฟังก์ชัน **R** ที่ใช้คำนวณค่าสถิติ F ที่ใช้ทดสอบ คือ var.test() ซึ่งมีรูปแบบของการใช้ ดังนี้

วิธีใช้

```
var.test(x, y, ratio = 1, alternative = c("two.sided", "less",
```

x, y : ตัวแปรหรือเวคเตอร์ของของตัวแปรสุ่มแบบต่อเนื่องจากประชากรกลุ่มที่ 1 และกลุ่มที่ 2 ตามลำดับ

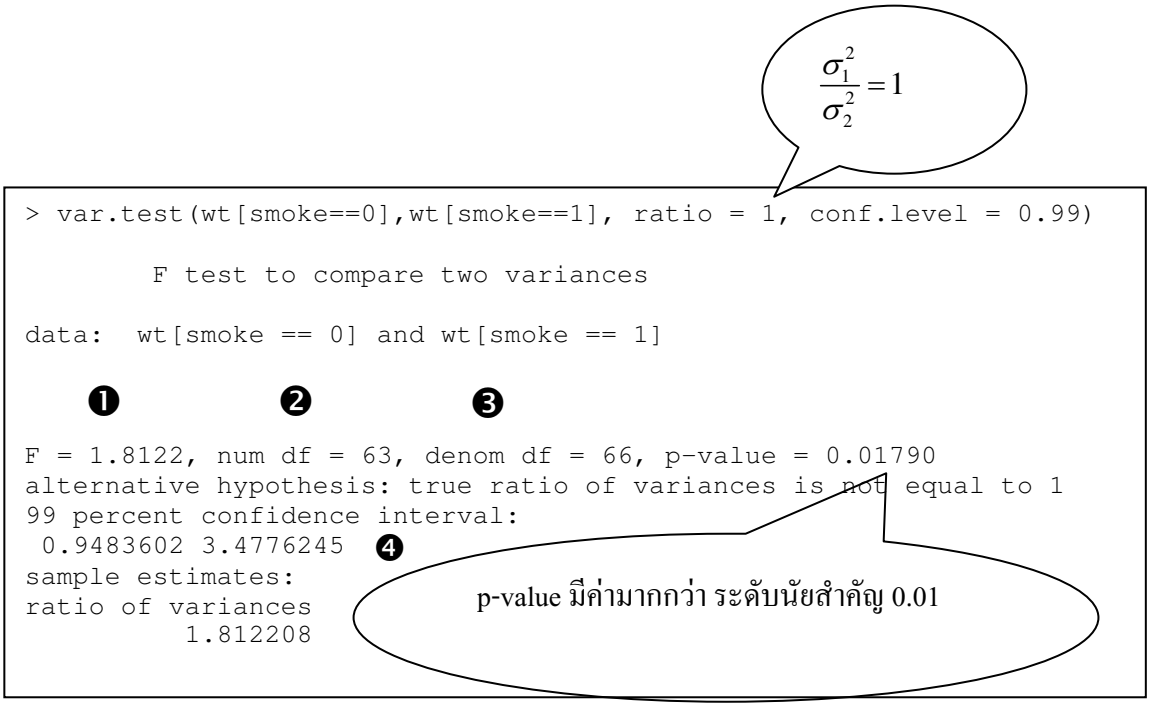
ratio: อัตราส่วนความแปรปรวนของ x และ y (ค่าปกติคือ ratio = 1)

alternative: ระบุสมมติฐานทางเลือก "two.sided" (ทดสอบแบบสองหาง), "greater" (ทดสอบแบบหางเดียวทางขวา) หรือ "less" (ทดสอบแบบหางเดียวทางซ้าย) อย่าง ใดอย่างหนึ่ง นอกจากนี้ผู้ใช้สามารถระบุโดยใช้เพียง อักษรตัวแรก คือ " t ", " g " หรือ " l " ก็ได้

conf.level : ระดับความเชื่อมั่นที่ใช้ทดสอบ ค่าปกติคือ 0.95

ตัวอย่าง 6.4 Birth weight data : จงตรวจสอบว่าความแปรปรวนของน้ำหนักของทารกแรกเกิดของมารดาที่ไม่สูบบุหรี่ (σ_1^2) และสูบบุหรี่ (σ_2^2) ของประชากรศึกษา แตกต่างกันหรือไม่ กำหนดระดับนัยสำคัญ 0.01 (ระดับความเชื่อมั่น เท่ากับ 0.99)

สมมติฐานของการศึกษาคือ ความแปรปรวนของน้ำหนักทารกแรกเกิดที่มารดาไม่สูบบุหรี่ (μ_1) และที่สูบบุหรี่ (μ_2) ของประชากรศึกษาไม่แตกต่างกัน $\frac{\sigma_1^2}{\sigma_2^2} = 1$



❶ สถิติ $F=\frac{S_1^2\sigma_2^2}{S_2^2\sigma_1^2}$

❷ num df = ระดับขั้นความเสรีของตัวเศษของ ❶

❸ denom df = ระดับขั้นความเสรีของตัวส่วนของ ❶

❹ ช่วงความเชื่อมั่น 99% ของ $\frac{\sigma_1^2}{\sigma_2^2}$

สรุป จากข้อมูลตัวอย่างสามารถสรุปได้ว่าความแปรปรวนของน้ำหนักของทารกแรกเกิดของมารดาที่ไม่สูบบุหรี่ (σ_1^2) และสูบบุหรี่ (σ_2^2) ของประชากรศึกษาแตกต่างกันอย่างไม่มีนัยสำคัญ (0.01)

แบบฝึกหัด Birth weight data : จงทดสอบว่าน้ำหนักเฉลี่ยของทารกแรกเกิดจากมารดาที่ไม่สูบบุหรี่ (μ_1) มีค่ามากกว่าน้ำหนักเฉลี่ยของทารกแรกเกิดจากมารดาที่สูบบุหรี่ (μ_2) อยู่ 5 ounces หรือไม่ โดยใช้สถิติ t กำหนดระดับนัยสำคัญ 0.01 (ระดับความเชื่อมั่น เท่ากับ 0.99)

6.5 การทดสอบสมมติฐานและการประมาณค่าแบบช่วงค่าสัดส่วนของประชากร

6.5.1 การทดสอบสมมติฐานประมาณค่าแบบช่วงกับค่าสัดส่วนของประชากรหนึ่งกลุ่ม

การทดสอบสมมติฐานเกี่ยวกับค่าสัดส่วนของประชากรเป็นการตรวจสอบว่า สัดส่วน หรือความน่าจะเป็นของความสำเร็จ (p) ของประชากร มีค่าเท่ากับค่าใดค่าหนึ่งที่กำหนด หรือไม่ ฟังก์ชัน **R** ที่ใช้คำนวณค่าสถิติที่ใช้ทดสอบ จำแนกเป็น 3 กรณี คือ

(1) ใช้การวิเคราะห์ความถี่ **R** จะคำนวณค่าสถิติไคกำลังสอง $\left(\sum_i \frac{(O_i - E_i)^2}{E_i}\right)$ และคำสั่งที่ใช้คือ prop.test() ซึ่งมีรูปแบบของการใช้ ดังนี้

วิธีใช้

```
prop.test(x, n, p = NULL, alternative = c("two.sided",
"less", greater"), conf.level = 0.95, correct = TRUE)
```

x : เวกเตอร์ที่มีสมาชิกเป็นจำนวนของความสำเร็จ (success) หรือเมทริกซ์ที่มี 2 สดมภ์ ประกอบด้วย สดมภ์ของจำนวนของความสำเร็จ และสดมภ์ของจำนวนความล้มเหลว (failure)

n: เวกเตอร์ของจำนวนครั้งของการทดลอง ไม่ต้องมีถ้า x เป็นเมทริกซ์

p: เวกเตอร์ของความน่าจะเป็นของความสำเร็จ ขนาดของ p ต้องเท่ากับจำนวนกลุ่มที่จำแนกใน x สมาชิกแต่ละตัวของ p ต้องอยู่ในช่วง (0, 1)

correct : เงื่อนไขที่ ที่ต้องการ/ไม่ต้องการ continuity correction (Yates' approach)

ตัวอย่าง 6.5 Birth weight data : จงตรวจสอบว่าเปอร์เซ็นของมารดาที่สูบบุหรี่/เคยสูบ มีค่ามากกว่า 50 หรือไม่ กำหนดระดับนัยสำคัญ 0.05 (ระดับความเชื่อมั่น เท่ากับ 0.95)

```
> table(smoke>0, exclude = c(NA, NaN))

FALSE  TRUE
   56    92
```

ตาราง แสดงจำนวนมารดาจำแนกตามสถานะการสูบบุหรี่

	ไม่เคยสูบบุหรี่	ปัจจุบันสูบบุหรี่/เคยสูบ
จำนวน (ค่าสังเกต : O_i)	56	92
ค่าคาดหวัง (E_i)	$148\times0.50 = 74$	$148\times0.50 = 74$

$$H_0 : p \le 0.50 \quad vs \quad H_1 : p > 0.50$$

```
> prop.test(sum(smoke>=1, na.rm=T),148, p=.50, alternative = "g",
correct = T)

        1-sample proportions test with continuity correction

data:  sum(smoke >= 1, na.rm = T) out of 148, null probability 0.5
X-squared = 8.277, df = 1, p-value = 0.002007
alternative hypothesis: true p is greater than 0.5
95 percent confidence interval:
 0.5509927 1.0000000
sample estimates:
        p
0.6216216
```

❶ $\chi^2 = \dfrac{(|56-74|-0.5)^2}{74} + \dfrac{(|92-74|-0.5)^2}{74} = 8.277$

```
> prop.test(sum(smoke>=1, na.rm=T),148, p=.50, alternative = "g",
correct = F)

        1-sample proportions test without continuity correction

data:  sum(smoke >= 1, na.rm = T) out of 148, null probability 0.5
X-squared = 8.7568, df = 1, p-value = 0.001542
alternative hypothesis: true p is greater than 0.5
95 percent confidence interval:
 0.5544202 1.0000000
sample estimates:
        p
0.6216216
```

❶ $\chi^2 = \dfrac{(56-74)^2}{74} + \dfrac{(92-74)^2}{74} = 8.7568$

สรุป จากข้อมูลตัวอย่างสามารถสรุปได้ว่า ประชากรศึกษามีมารดาที่เคยสูบบุหรี่หรือปัจจุบันสูบบุหรี่มีมากกว่า 50% ($\alpha = 0.05$)

หมายเหตุ กรณีประมาณการแจกแจงของสัดส่วนของตัวอย่าง ($\hat{p}$) ด้วยการแจกแจงปกติค่าสถิติที่ใช้ทดสอบคือ $z = \dfrac{\hat{p} - p}{\sqrt{\dfrac{p(1-p)}{n}}} = \dfrac{0.6216 - 0.50}{\sqrt{\dfrac{0.50(1-0.50)}{148}}} = 2.9592$

และ $z^2 = (2.9592)^2 = \chi^2 = 8.7567$

ซึ่งฟังก์ชัน **oneprop.ztest** ที่แสดงข้างล่างถูกเขียนขึ้นสำหรับการทดสอบสมมติฐานและประมาณค่าแบบช่วงสำหรับค่าสัดส่วนของประชากร 1 กลุ่ม โดยใช้การแจกแจงปกติ โดยตัวอย่างการใช้ ฟังก์ชัน **oneprop.ztest()** ก็คงยังใช้ข้อมูลจากตัวอย่าง 6.5

```
oneprop.ztest <- function(p, n, prop = 0.5,
            alternative = c("two.sided", "less", "greater"),
            conf.level=0.95){

 # p: sample proportion of success;
 # n: sample size;
 # prop: true (population) proportion
 # alternative: a character string specifying the alternative
 # hypothesis,must be one of '"two.sided"', '"greater"' or '"less"'.

   stderr <- sqrt(prop*(1-prop)/n)
   zstat <- (p-prop)/stderr

   stderr.est <- sqrt(p*(1-p)/n)

     if (alternative == "less") {
           pval <- pnorm(zstat, lower.tail = T)
         }

     else if (alternative == "greater") {
           pval <- pnorm(zstat, lower.tail = F)
         }

    else {
      pval <- 2 * pnorm(abs(zstat), lower.tail=F)
         }

  cint <- cbind(p-qnorm((1-conf.level)/2,lower.tail=F)*stderr.est,
                p + qnorm((1-conf.level)/2,lower.tail = F)*stderr.est)
  colnames(cint) <- c("lwr", "uppr")


    rval <- list(statistic = zstat, Pvalue = pval,
             confidence.level=conf.level, confidence.interval=cint,
             estimate=p, null.value = prop, alternative=alternative)

    return(rval)
}
```

หมายเหตุ ชุดคำสั่งนี้ได้ถูกรวบรวมไว้ใน StatCan.txt แล้ว

ตัวอย่างการใช้ ฟังก์ชัน `oneprop.ztest()` โดยใช้ข้อมูลจากตัวอย่าง 6.5

```
> x <- 92; n<- 148;  p <- x/n

> oneprop.ztest(p, n,prop=0.5, alternative="greater")
$statistic
[1] 2.959182

$Pvalue
[1] 0.001542285

$confidence.level
[1] 0.95

$confidence.interval
          lwr      uppr
[1,] 0.543487 0.6997562

$estimate
[1] 0.6216216

$null.value
[1] 0.5

$alternative
[1] "greater" [1] "greater"
```

$$z = \frac{\hat{p} - p}{\sqrt{\frac{p(1-p)}{n}}} = \frac{0.6216 - 0.50}{\sqrt{\frac{0.50(1-0.50)}{148}}} = 2.9592$$

$$\hat{p} \mp z_{.05/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

$\hat{p}$

(2) ใช้การแจกแจงเบอร์นูลีหรือการแจกแจงทวินามในการคำนวณค่าสถิติที่ใช้ทดสอบ คำสั่งที่ใช้คือ binom.test() ซึ่งมีรูปแบบของการใช้ ดังนี้

วิธีใช้

```
binom.test(x, n, p = 0.5, alternative = c("two.sided",
           "less", "greater"), conf.level = 0.95)
```

x : จำนวนของความสำเร็จ (success) หรือเวกเตอร์ที่มีขนาด 2 ประกอบด้วย จำนวนของความสำเร็จ และจำนวนความ ล้มเหลว (failure)

n: จำนวนครั้งของการทดลอง ไม่ต้องมีถ้า x เป็นเมทริกซ์

p: ค่าความน่าจะเป็นของความสำเร็จ (p) ที่ต้องการทดสอบ p ต้องอยู่ในช่วง (0, 1) ค่าปกติคือ `p = 0.5`

ตัวอย่าง 6.6 Birth weight data : จงตรวจสอบว่าเปอร์เซ็นของมารดาที่สูบบุหรี่ หรือเคยสูบบุหรี่มีค่ามากกว่า 50 หรือไม่ กำหนดระดับนัยสำคัญ 0.05 (ระดับความเชื่อมั่น เท่ากับ 0.95) และใช้ **Binomial test**

```
> binom.test(sum(smoke>=1, na.rm=T),148, alternative = "g")

        Exact binomial test

data:  sum(smoke >= 1, na.rm = T) and 148
number of successes = 92, number of trials = 148, p-value = 0.001932
alternative hypothesis: true probability of success is greater than
0.5
95 percent confidence interval:
 0.5512597 1.0000000
sample estimates:
probability of success
             0.6216216
```

สรุป จากข้อมูลตัวอย่างสามารถสรุปได้ว่า ประชากรศึกษามีมารดาที่เคยสูบบุหรี่หรือปัจจุบันสูบบุหรี่มีมากกว่า 50% ($\alpha = 0.05$)

6.5.2 การทดสอบสมมติฐานเกี่ยวกับค่าสัดส่วนของประชากรตั้งแต่สองกลุ่มขึ้นไป

สมมมุติฐานของการทดสอบสำหรับประชากร k กลุ่ม คือ

$H_0 : p_1 = p_2 = \cdots = p_k$; H_1 : อย่างน้อย 1 กลุ่มมีค่าสัดส่วนแตกต่างจากกลุ่มอื่น

ตัวอย่าง 6.7 Birth weight data: จงตรวจสอบว่าสัดส่วนของมารดาที่ปัจจุบันสูบบุหรี่ในแต่ละสีผิว (ขาว, ดำ และ อื่นๆ) มีค่าแตกต่างกัน หรือไม่ กำหนดระดับนัยสำคัญ 0.05 (ระดับความเชื่อมั่นเท่ากับ 0.95)

แปลงรหัสผิวสีของมารดา

```
> race.group <- replace(race,race <=5, 1)
> race.group <- replace(race.group,race.group==7,2)
> race.group <- replace(race.group,race.group >2,3)
> # 1 : white, 2: black  3: others

> race.group <-factor(race.group,labels=c("White", "Black",
"Others"))
> table(race.group)
race.group
 White  Black Others
   102     29     19
```

← จำนวนมารดาจำแนกตามสีผิว

แปลงรหัสการสูบบุหรี่ของมารดา

```
> smoke.group <- replace(smoke, smoke!=1, 2)
> smoke.group <-factor(smoke.group,labels=c("Current
                  smoke","Stop/never smoke" ))
> table(smoke.group)
smoke.group
   Current smoke Stop/never smoke
             68               80
```

จำนวนมารดาจำแนกตามสถานภาพการสูบบุหรี่

```
> smoke.race <- table(race.group, smoke.group)
> smoke.race
          smoke.group
race.group Current smoke Stop/never smoke
    White             54               47
    Black              8               21
    Others             6               12
```

จำนวนมารดาจำแนกตามสีผิวและสถานภาพการสูบบุหรี่

```
> prop.test(smoke.race )

        3-sample test for equality of proportions without continuity
correction

data:  smoke.race
X-squared = 7.3883, df = 2, p-value = 0.02487
alternative hypothesis: two.sided
sample estimates:
   prop 1    prop 2    prop 3
0.5346535 0.2758621 0.3333333
```

เนื่องจากค่า p-value = 0.02487 มีค่าน้อยกว่าค่าระดับนัยสำคัญ 0.05 จึงสามารถสรุปได้ว่า สัดส่วนของมารดาที่ปัจจุบันสูบบุหรี่ในแต่ละสีผิว (ขาว, ดำ และ อื่นๆ) มีค่าแตกต่างกันอย่างมีนัยสำคัญ

ตัวอย่าง 6.8 Birth weight data: จงตรวจสอบว่าเปอร์เซ็นของมารดาผิวขาว, ดำ และ ผิวสีอื่นๆ) ที่ปัจจุบันสูบบุหรี่ในแต่ละสีผิว เป็น 50, 25 และ 25 ตามลำดับ หรือไม่ กำหนดระดับนัยสำคัญ 0.05 (ระดับความเชื่อมั่น เท่ากับ 0.95)

H_0 : เปอร์เซ็นของมารดาผิวขาว, ดำ และ ผิวสีอื่นๆ) ที่ปัจจุบันสูบบุหรี่ในแต่ละสีผิว เป็น 50, 25 และ 25

```
> p <- c(0.5, 0.25, 0.25)
> prop.test(smoke.race, p= c(0.5,0.25,0.25),correct=F)

        3-sample test for given proportions without continuity correction

data:  smoke.race, null probabilities c(0.5, 0.25, 0.25)
X-squared = 1.2553, df = 3, p-value = 0.7398
alternative hypothesis: two.sided
null values:
prop 1 prop 2 prop 3
  0.50   0.25   0.25
sample estimates:
   prop 1    prop 2    prop 3
0.5346535 0.2758621 0.3333333

Warning message:
Chi-squared approximation may be incorrect in: prop.test(smoke.race, p =
c(0.5, 0.25, 0.25), correct = F)
```

Warning message: เป็นการเตือนเท่านั้น ไม่ได้บ่งบอกว่าทำผิด

สรุป เนื่องจากค่า มากกว่า 0.05 จึงสรุปได้ว่า เปอร์เซ็นของมารดาผิวขาว, ดำ และ ผิวสีอื่นๆ) ที่ปัจจุบันสูบบุหรี่ในแต่ละสีผิว ไม่พบความแตกต่างที่มีนัยสำคัญทางสถิติ

6.6 ฟังก์ชันสำหรับการทดสอบสมมติฐานและประมาณค่าแบบช่วงค่าเฉลี่ยและค่าสัดส่วนของประชากรสองกลุ่ม กรณีไม่มีข้อมูลดิบ

กรณีที่เรามีค่าเฉลี่ยและค่าความแปรปรวนของตัวอย่าง ค่าสัดส่วนของตัวอย่าง และขนาดของตัวอย่าง ถ้าเราต้องการทดสอบสมมติฐานและประมาณค่าแบบช่วง $100(1-\alpha)\%$ เราสามารถสร้างฟังก์ชันขึ้นใช้เอง ดังนี้

6.6.1 ฟังก์ชันสำหรับทดสอบสมมติฐานและประมาณค่าแบบช่วงของผลต่างค่าเฉลี่ยของประชากรสองกลุ่ม

ใช้ t-test

```
twomean.ttest <- function(mx, my, vx, vy, nx,ny,
      alternative = c("two.sided","less","greater"),
      mu = 0, var.equal = TRUE, conf.level = 0.95)
   {
     alternative <- match.arg(alternative)
     estimate <- c(mx,my)
     names(estimate) <- c("mean of x","mean of y")
       if(var.equal) {
            df <- nx+ny-2
            v <- ((nx-1)*vx + (ny-1)*vy)/df
            stderr <- sqrt(v*(1/nx+1/ny))
        } else {
            df <- (vx/nx + vy/ny)^2/((vx/nx)^2/(nx-1) +
(vy/ny)^2/(ny-1))
            stderrx <- sqrt(vx/nx)
            stderry <- sqrt(vy/ny)
            stderr <- sqrt(stderrx^2 + stderry^2)
        }
        tstat <- (mx - my - mu)/stderr
    if (alternative == "less") {
      pval <- pt(tstat, df)   }
    else if (alternative == "greater") {
      pval <- pt(tstat, df, lower = FALSE)}
    else {
      pval <- 2 * pt(-abs(tstat), df)}
      cint <- cbind(mx-my - qt((1-conf.level)/2, df,
                    lower.tail = F)*stderr ,
                 mx-my + qt((1-conf.level)/2, df,
                    lower.tail = F)*stderr )
      colnames(cint) <- c("lwr", "uppr")

    rval <- list(statistic = tstat, df = df, p.value = pval,
             confidence.level = conf.level, confidence.interval=cint,
             estimate=estimate, null.value = mu,
             alternative=alternative)
    return(rval)
}
```

หมายเหตุ 1) ชุดคำสั่งนี้ได้ถูกรวบรวมไว้ใน StatCan.txt แล้ว

2) ผลลัพธ์ของค่าประมาณแบบช่วงเหมือนกับใช้คำสั่ง t.test ที่ระบุ

alternative = "two.sided "

วิธีใช้

```
twomean.ttest(mx, my, vx, vy, nx,ny,
      alternative = c("two.sided","less","greater"),
      mu = 0, var.equal = TRUE,  conf.level = 0.95)
```

mx: ค่าเฉลี่ยตัวอย่างของกลุ่มที่หนึ่ง

vx: ค่าความแปรปรวนตัวอย่างของกลุ่มที่หนึ่ง

nx: ขนาดตัวอย่างของกลุ่มที่หนึ่ง

my, vy, ny: ค่าเฉลี่ยตัวอย่าง, ค่าความแปรปรวนตัวอย่าง, ขนาดตัวอย่าง, ของกลุ่มที่สอง ตามลำดับ

ตัวอย่าง 6.7 Birth weight data: จงคำนวน **(1)** ช่วงความเชื่อมั่น 99% ของผลต่างระหว่างน้ำหนักทารกโดยเฉลี่ยจากมารดาที่ไม่เคยสูบบุหรี่ (smoke= =0) และจากมารดาที่ปัจจุบันยังคงสูบบุหรี่ (smoke= =1) **(2)** จงตรวจสอบว่าน้ำหนักเฉลี่ยของทารกที่มารดาไม่เคยสูบบุหรี่มากกว่าน้ำหนักเฉลี่ยของทารกที่มารดาปัจจุบันยังคงสูบบุหรี่ สมมติว่าความแปรปรวนของน้ำหนักทารกทั้งสองกลุ่มไม่เท่ากัน

```
> mw.sm <- mean(wt[smoke==1],na.rm=T);
> mw.nsm <-mean(wt[smoke==0] ,na.rm=T)
> vw.sm <- var(wt[smoke==1],na.rm=T)
> vw.nsm <- var(wt[smoke==0],na.rm=T)
> nwnsm <- length(na.omit(wt[smoke==0]))
> nwsm <- length(na.omit(wt[smoke==1]))

> twomean.ttest(mw.nsm, mw.sm, vw.nsm, vw.sm, nwnsm, nwsm,
+ alternative = "greater", mu=0 ,var.equal = FALSE,conf.level = 0.99)
$statistic
[1] 2.723658

$df
[1] 95.80826

$p.value
[1] 0.003836664              ---------------(2)

$conf.int
[1] 0.3039884 17.0289831     ---------------(1)

$estimate
mean of x mean of y
 122.0545  113.3881

$null.value
[1] 0

$alternative
[1] "greater"
```

น้ำหนักเฉลี่ยของทารกที่มารดาไม่เคยสูบบุหรี่และน้ำหนักเฉลี่ยของทารกที่มารดาปัจจุบันยังคงสูบบุหรี่ ไม่ต่างกัน

6.6.2 ฟังก์ชันสำหรับทดสอบสมมติฐานและประมาณค่าแบบช่วงของผลต่างค่าสัดส่วนของประชากรสองกลุ่ม

```
twoprop.ztest <- function(p1, p2, n1, n2, prop = 0,
    alternative = c("two.sided", "less", "greater"),conf.level=0.95){

 # p1, p2: proportion of success of sample 1, sample 2;
 # n1, n2: sample size of sample 1, sample 2
 # prop: difference of 2 population proportions is 0
 # alternative: a character string specifying the alternative
 # hypothesis,must be one of '"two.sided"', '"greater"' or '"less"'.

   p <- (n1*p1 + n2*p2)/(n1+n2)
   stderr <- sqrt(p*(1-p)*(1/n1+1/n2))
   zstat <- (p1-p2)/stderr

   stderr.est <- sqrt(p1*(1-p1)/n1 + p2*(1-p2)/n2)

     if (alternative == "less") {
           pval <- pnorm(zstat, lower.tail = T)
         }

     else if (alternative == "greater") {
           pval <- pnorm(zstat, lower.tail = F)
         }

    else {
      pval <- 2 * pnorm(abs(zstat), lower.tail=F)
         }

      cint <- cbind((p1-p2)-qnorm((1-conf.level)/2,
                lower.tail = F)*stderr.est ,
                (p1-p2) + qnorm((1-conf.level)/2,
                lower.tail = F)*stderr.est )
        colnames(cint) <- c("lwr", "uppr")
        estimates <- cbine(p1, p2)


    rval <- list(statistic = zstat, Pvalue = pval,
            confidence.level=conf.level, confidence.interval=cint,
            estimates=estimates, null.value = prop,
            alternative=alternative)

    return(rval)
}
```

หมายเหตุ ชุดคำสั่งได้ถูกรวบรวมไว้ใน StatCan.txt แล้ว

วิธีใช้

```
twoprop.ztest(p1, p2, n1, n2, prop = 0,
     alternative = c("two.sided", "less", "greater"),conf.level=0.95)
```

p1, p2: ค่าสัดส่วนตัวอย่างของกลุ่มที่หนึ่งและของกลุ่มที่ 2

n1, n2: ขนาดตัวอย่างกลุ่มที่หนึ่งและของกลุ่มที่ 2

prop: ค่าของผลต่างระหว่างสัดส่วนของประชากรที่ 1 และของประชากรที่ 2 ภายใต้สมมติฐานว่าง จริง

ตัวอย่าง 6.8 Birth weight data: จงคำนวณช่วงความเชื่อมั่น 95% **(1)**ของผลต่างสัดส่วนของมารดาที่สูบบุหรี่ /เคยสูบบุหรี่ ผิวขาวและผิวสี และ **(2)** ทดสอบว่าสัดส่วนของมารดาผิวขาวและผิวสีที่ปัจจุบันสูบบุหรี่แตกต่างกัน หรือไม่

แปลงรหัสของ race จาก 0-5 เป็น 1 และ จาก 6-9 เป็น 2

```
> race.white <- ifelse(race <=5, 1,2)
> race.white <- factor(race.white, labels=c("White", "Others"))
> table( race.white)
race.white
  1   2
102  48

> table(race.white,smoke.group)

          smoke.group
race.white Current smoke Stop/never smoke
    White             54                47
    Others            14                33
```


```
> p.white <- 54/(54+47)
> p.others <- 14/(14+33)
> n.white <- 54+47; n.others <- 14+33
```

ทดสอบสมมติฐานและประมาณค่าแบบช่วง

```
> twoprop.ztest(p.white, p.others, n.white, n.others,
alternative="two.sided")
$statistic
[1] 2.690841

$Pvalue
[1] 0.007127207  ---------------(2)

$confidence.level
[1] 0.95

$confidence.interval
            lwr      uppr
[1,] 0.07381818 0.3997441        ---------------(1)

$estimates
            p1        p2
[1,] 0.5346535 0.2978723

$null.value
[1] 0

$alternative
[1] "two.sided"
```

H_0: สัดส่วนของมารดาผิวขาวและผิวสีที่ปัจจุบันสูบบุหรี่ไม่แตกต่างกัน

สรุป เนื่องค่า p-value = 0.0071 มีค่าน้อยกว่า 0.05 จึงสามารถสรุปได้ว่าสัดส่วนของมารดาผิวขาวและผิวสีที่ปัจจุบันสูบบุหรี่แตกต่างกัน อย่างมีนัยสำคัญ โดยมั่นใจได้ 95% ว่าผลต่างของสัดส่วนมีค่าระหว่าง 0.0738 และ 0.3997

แบบฝึกหัด

1. จงทดสอบว่าน้ำหนักของมารดาที่สูบบุหรี่หรือเพิ่งหยุดสูบเมื่อตอนตั้งครรภ์กับกับน้ำหนักของมารดาที่ไม่เคยสูบบุหรี่หรือเลิกสูบแล้วแตกต่างกัน หรือไม่
2. จงคำนวนช่วงความเชื่อมั่น 95 % ของระยะเวลาตั้งครรภ์เฉลี่ยของมารดาที่สูบบุหรี่
3. จงสร้างตารางการณ์จรแสดงจำนวนมารดา จำแนกตามผิวสี (ขาว และผิวสีอื่นๆ) และรายได้ของครัวเรือนซึ่งจำแนกเป็น 5 กลุ่ม: ต่ำกว่า 5,000 5,000 - 7,499 7,500 – 9,999 10,000 – 12,499 และตั้งแต่ 12,500 บาทขึ้นไป โดยใช้ **R**
4. จงใช้ผลที่ได้จากข้อ 3 ทดสอบว่าสัดส่วนของมารดาที่เป็นชนผิวขาวในแต่ละระดับรายได้เท่ากันหรือไม่ (ใช้สำหรับฝึกปฏิบัติการใช้ **R** แต่การทดสอบอาจไม่สมเหตุสมผล)

บทที่ 7
การทดสอบไคสแควร์ (Chi-Square Test)

การวิเคราะห์ความถี่หรือการทดสอบไคสแควร์ เป็นการทดสอบสมมติฐานสำหรับข้อมูลเชิงคุณภาพ หรือข้อมูลชนิดไม่ต่อเนื่อง ข้อมูลอาจแบ่งเป็นกลุ่มๆ ซึ่งอาจแสดงอยู่ในรูปของการแจงนับ (Counts) หรือ ความถี่ (frequency) หรือเป็นตารางการจร

7.1 ตัวอย่างตารางการจร

(1) **Birth weight data**: จำนวนมารดาจำแนกตามผิวสีและประวัติการสูบบุหรี่

ผิวสี	ประวัติการสูบบุหรี่		รวม
	ไม่สูบบุหรี่	เคยสูบ/ปัจจุบันสูบ	
ขาว	32	69	101
ผิวสีอื่น	24	23	47
รวม	56	92	148

ในที่นี้ ตัวแปร คือ ผิวสีและประวัติการสูบบุหรี่ของมารดา

(2) ผลจากการใช้น้ำตาล Moltose ในการขยายพันธุ์ส้มชนิดหนึ่งโดยวิธีเพาะเนื้อเยื่อ จากจำนวนจานทดลองเลี้ยงเนื้อเยื่อส้ม 280 จาน

เนื้อเยื่อ	ระดับน้ำตาล Maltose (μM)				รวม
	18	36	75	150	
ผลิต Embryo	36	30	21	100	187
ไม่ผลิต Embryo	4	10	19	60	93
รวม	40	40	40	160	280

ตัวอย่างที่นี้เราอาจจะต้องการทราบว่าอัตราการผลิต/ไม่ผลิตต้นอ่อนของเนื้อเยื่อส้มขึ้นอยู่กับระดับน้ำตาล Maltose หรือไม่

7.2 ฟังก์ชัน R สำหรับการทดสอบไคสแควร์

ฟังก์ชันหรือคำสั่งใน **R** สำหรับ การทดสอบไคสแควร์ในคลังคำสั่ง **stats** มี 2 คำสั่ง ได้แก่ chisq.test() และ fisher.test() นอกจากนี้ทีมวิจัยได้เขียนฟังก์ชันสำหรับสร้างตารางทางเดียวชื่อ oneway.tab() และตารางสองทาง ชื่อ twoway.tab() (สาขาสถิติ ภาควิชาคณิตศาสตร์) ซึ่งรวบรวมไว้ใน text file ชื่อ **StatCan.txt** ทั้งสองคำสั่งจะแสดงจำนวน ร้อยละทางแถว ร้อยละทางสดมภ์ ร้อยละคิดจากยอดรวม พร้อมกับค่าไคสแควร์สำหรับทดสอบ Goodness of fit สำหรับตารางทางเดียว และทดสอบความเป็นอิสระระหว่างตัวแปรสำหรับตารางสองทาง

chisq.test

วิธีใช้

```
chisq.test(x, y = NULL, correct = TRUE,p = rep(1/length(x),
length(x)), simulate.p.value = FALSE, B = 2000)
```

x : เวกเตอร์หรือเมทริกซ์ของข้อมูลแบบไม่ต่อเนื่อง หรือความถี่

y : เวกเตอร์ของข้อมูลแบบไม่ต่อเนื่อง หรือความถี่ ไม่ต้องมีถ้า x เป็นเมทริกซ์

correct: เงื่อนไขที่ ต้องการ/ไม่ต้องการ continuity correction (Yates' approach)

p: เวกเตอร์ของความน่าจะเป็นภายใต้ข้อสมมติว่า null hypothesis เป็นจริง ขนาดของ p ต้องเท่ากับขนาดของ x

fisher.test

วิธีใช้

```
fisher.test(x, y = NULL, workspace = 200000, hybrid = FALSE,
            control = list(), or = 1, alternative = "two.sided",
            conf.int = TRUE, conf.level = 0.95)
```

x : ตารางการณ์จรซึ่งเขียนในรูปของเมทริกซ์ มิติ 2 x 2 หรือเป็น object ที่เป็นแฟกเตอร์

y : object ที่เป็นแฟกเตอร์(ไม่ต้องมีถ้า x เป็นเมทริกซ์)

hybrid : เงื่อนไขที่ใช้กำหนดว่าจะให้คำนวณ exact probability (ค่าปกติ) หรือให้คำนวณค่าประมาณ ถ้า hybrid = TRUE ค่า probability ที่คำนวณเป็น asymptotic chi-squared probability สมมติฐานเกี่ยวกับ odds ratio จะมีการทดสอบกรณีข้อมูลนำเข้าเป็นตารางการณ์จร ขนาด 2 x 2

`oneway.tab`

วิธีใช้

```
oneway.tab(x, pct=FALSE, chi.test = FALSE, prop=NULL, data = NULL)
```

x: เวกเตอร์ที่มีสมาชิกเป็นค่าของตัวแปรเชิงคุณภาพที่ต้องการวิเคราะห์

pct : เลือกให้คำสั่งคำนวณค่าร้อยละ โดยที่ ค่าปกติคือ FALSE

chi.test: ถ้า TRUE ให้ทดสอบ Goodness of fit โดยใช้ chi-square test

prop: เวกเตอร์ความน่าจะเป็นที่ต้องการทดสอบที่มีขนาดเท่ากับของ x ถ้าไม่ระบุ สมาชิกของ p จะเท่ากับ 1/ขนาดของ x

data: กรอบข้อมูลที่เก็บตัวแปร x, y ที่ต้องการวิเคราะห์ หากไม่ได้ attach กรอบข้อมูลนั้น.

`twoway.tab`

วิธีใช้

```
twoway.tab(x, y, pct = c("row", "column", "total"),
                        chi.test = FALSE, data=NULL)
```

x, y: ตัวแปรเชิงคุณภาพ 2 ตัวที่ต้องการวิเคราะห์

pct : เลือกให้คำสั่งคำนวนค่าร้อยละตามแนว row, column หรือ total

chi.test: ถ้า TRUE ให้ทดสอบความสัมพันธ์ระหว่าง x ,y โดยใช้ chi-square test

data: กรอบข้อมูลที่เก็บตัวแปร x, y ที่ต้องการวิเคราะห์ หากไม่ได้ attach กรอบข้อมูลนั้น.

ตัวอย่าง 7.1 จากข้อมูล **Birth weight data**: จงทดสอบทารกที่ศึกษาถูกสุ่มมาจากกลุ่มครัวเรือนที่มีรายได้ ต่ำกว่า 5,000 5,000-<7,500 7,500-<10,000 10,000-<12,500 12,500-<15,000 และตั้งแต่ 15,000 ดอลลาร์ขึ้นไป ด้วยความน่าจะเป็นเท่ากัน หรือไม่

แปลงรหัสของ income ให้เป็นไปตามที่ต้องการศึกษา และเก็บไว้ในวัตถุชื่อ inc.group

```
> inc.group <- replace(income,income <=1,1)
> inc.group <- replace(inc.group,inc.group>=6,6)
```

แปลง `inc.group` ให้ factor พร้อมกับ labels และเก็บไว้ในวัตถุชื่อ income.group

```
> income.group <- factor(inc.group,
  labels=c("<5000","5000-7499","7500-9999",
 "10000-12499","12500+", "15000+"))
```

จำนวนทารกจำแนกตามรายได้ครัวเรือนต่อปี

```
> table(income.group)
income.group
      <5000  5000-7499   7500-9999 10000-12499 12500-14999    15000+
         33        18          26          16          14        29
```

ทดสอบ `Goodness of fit` โดยใช้ `chisq.test`

```
> chisq.test(table(income.group), p=c(1/6,1/6,1/6,1/6,1/6,1/6))

        Chi-squared test for given probabilities

data:  table(income.group)
X-squared = 13.2059, df = 5, p-value = 0.02152
```

ทดสอบ `Goodness of fit` โดยใช้ `oneway.tab` จะได้ตารางทางเดียวที่แสดงค่าจำนวนและร้อยละ ด้วย

```
> oneway.tab(income.group, pct=TRUE, chi.test=TRUE)
<5000   5000-7499   7500-9999 10000-12499 12500-14999  15000+  Total
  33        18          26          16          14     29       136

<percent>
<5000  5000-7499  7500-9999 10000-12499 12500-14999   15000+    Total
  24.26    13.24      19.12       11.76       10.29    21.32    99.99

         Goodness-of-fit test

Pearson's Chi-squared = 13.20588, df = 5 , p-value = 0.02152395
```

สรุป ทารกที่ศึกษาถูกสุ่มมาจากกลุ่มครัวเรือนที่มีรายได้ ต่ำกว่า 5,000 5,000-<7,500 7,500-<10,000 10,000-<12,500 12,500-<15,000 และตั้งแต่ 15,000 ดอลลาร์ขึ้นไป ด้วยความน่าจะเป็นต่างกันอย่างมีนัยสำคัญทางสถิติ (ระดับนัยสำคัญ0.05)

ตัวอย่าง 7.2 จากหัวข้อ 7.1 (1) **Birth weight data**: จงทดสอบว่าพฤติกรรมการสูบ (ปัจจุบันสูบ/เคยสูบ) หรือไม่สูบบุหรี่ของมารดาขึ้นอยู่กับผิวสีของมารดา หรือไม่

แปลงรหัสของ `smoke` ให้เป็นไปตามที่ต้องการศึกษา และเก็บไว้ในวัตถุชื่อ sm.group

```
> sm.group <- replace(smoke, smoke >0,1)
```

แปลง `sm.group` ให้ factor พร้อมกับ labels และเก็บไว้ในชื่อเดิม

```
> sm.group <- factor(sm.group, labels=c("Never smoke", "Smoke"))
```

```
> table(race.white, sm.group)  # Crosstabulation

          sm.group
race.white Never smoke Smoke
    White           32    69
    Others          24    23
```

ทดสอบความสัมพันธ์ระหว่างพฤติกรรมการสูบ (ปัจจุบันสูบ/เคยสูบ) หรือไม่สูบบุหรี่ของมารดาขึ้นผิวสีของมารดา โดยใช้ `chisq.test`

```
> chisq.test(race.white, sm.group)

        Pearson's Chi-squared test with Yates' continuity
correction

data:  race.white and sm.group
X-squared = 4.3312, df = 1, p-value = 0.03742
```

ทดสอบความสัมพันธ์ระหว่างพฤติกรรมการสูบ (ปัจจุบันสูบ/เคยสูบ) หรือไม่สูบบุหรี่ของมารดาขึ้นผิวสีของมารดา โดยใช้ `fisher.test`

```
> fisher.test(race.white, sm.group)

        Fisher's Exact Test for Count Data

data:  race.white and sm.group
p-value = 0.02929
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 0.2060893 0.9612545
sample estimates:
odds ratio
 0.4470144
```

ทดสอบความสัมพันธ์ระหว่างพฤติกรรมการสูบ (ปัจจุบันสูบ/เคยสูบ) และไม่สูบบุหรี่ของมารดาขึ้นผิวสีของมารดา โดยใช้ twoway.tab ซึ่งให้แสดงค่าร้อยละทางแถวด้วย

```
> twoway.tab(race.white, sm.group, pct="row",  chi.test = T)
< A two-way table >
       Never smoke Smoke Total
White           32    69   101
Others          24    23    47
Total           56    92   148
< Row percent >
       Never smoke  Smoke Total
White       31.683 68.317   100
Others      51.064 48.936   100
Total       37.838 62.162   100

Independent test
Pearson's Chi-squared = 4.331191 , df = 1 , p-value = 0.03742005

Fisher's exact test (2-sided) p-value = 0.0293
```

สรุป จากข้อมูลตัวอย่างอาจกล่าวได้ว่า พฤติกรรมการสูบ/ไม่สูบบุหรี่ของมารดาในประชากรศึกษามีความสัมพันธ์กับผิวสี อย่างมีนัยสำคัญ(ระดับนัยสำคัญ 0.05)

ตัวอย่าง 7.4 จากหัวข้อ 7.1 (2) ต้องการทดสอบว่า อัตราการผลิตต้นอ่อนของเนื้อเยื่อส้มขึ้นอยู่กับระดับน้ำตาล Maltose หรือไม่

ในกรณีที่ข้อมูลต้องการทดสอบอยู่ในรูปตารางสองทางเรียบร้อยแล้ว การวิเคราะห์ความถี่ใน **R** ทำได้โดยสร้างเมทริกซ์ที่เป็นตัวแทนของตารางนั้น ดังขั้นตอน ดังนี้

```
> # directly entry data to R using function c()
> embryo <- c(36,4,30,10,21,19,100,60)
> # create a 2 x 4 matrix 2 x 4

> embryo.tab <- matrix(embryo, ncol= 4)

> colnames(embryo.tab) <- c("moltose18",
"moltose36","moltose75","moltose150")

> rownames(embryo.tab) <- c("embryo", "no embryo")

> embryo.tab
          moltose18 moltose36 moltose75 moltose150
embryo           36        30        21        100
no embryo         4        10        19         60
```

ตั้งชื่อแถวและสดมภ์

```
> embryo.chi <- chisq.test(embryo.tab)
> embryo.chi

        Pearson's Chi-squared test

data:  embryo.tab
X-squared = 15.9393, df = 3, p-value = 0.001167

> attributes(embryo.chi)
$names
[1] "statistic" "parameter" "p.value"   "method"    "data.name"
"observed"
[7] "expected"  "residuals"

$class
[1] "htest"

> embryo.chi$expected # expected frequencies
          [,1]     [,2]     [,3]      [,4]
[1,] 26.71429 26.71429 26.71429 106.85714
[2,] 13.28571 13.28571 13.28571  53.14286
```

$\frac{187 \times 40}{280}$

$\frac{93 \times 40}{280}$

สรุป จากข้อมูลตัวอย่างและการใช้การทดสอบไคกำลังสอง ซึ่ง**ได้ค่า** **`X-squared = 15.9393, df = 3, p-value = 0.001167`** สรุปได้ว่า อัตราการผลิตต้นอ่อนของเนื้อเยื่อส้มขึ้นอยู่กับระดับน้ำตาล Maltose อย่างมีนัยสำคัญ (ระดับนัยสำคัญ 0.05)

แบบฝึกหัด

1. **Birth weight data**: จงทดสอบทารกที่ศึกษาถูกสุ่มมาจากมารดาที่มีอายุอยู่ในช่วง 15-20 21-25 26-35 และตั้งแต่ 36 ปีขึ้นไป ด้วยอัตราส่วน 1:2:3:1 หรือไม่

2**. Birth weight data:** จงทดสอบว่ารายได้ของครัวเรือนของประชากรศึกษาซึ่งจำแนกตามตัวอย่าง 7.2 กับสีผิว มีความสัมพันธ์ต่อกัน หรือไม่ (จัดกลุ่มตามความเหมาะสม)

ข้อแนะนำ ทดลองสร้างตารางทางเดียว หรือสองทางโดยใช้คำสั่งง่ายๆ เช่น table() ก่อน หรือถ้าต้องการแสดงจำนวนและร้อยละด้วย ก็ใช้คำสั่ง oneway.tab() สำหรับข้อ 1. หรือ twoway.tab() สำหรับข้อ 2.

บทที่ 8
การวิเคราะห์ความแปรปรวนทางเดียว
(One-Way Analysis of Variance)

การวิเคราะห์ความแปรปรวนทางเดียว เป็นวิธีการที่ใช้ทดสอบความแตกต่างระหว่างค่าเฉลี่ยของประชากรตั้งแต่ 2 กลุ่มขึ้นไปโดยที่มีองค์ประกอบ (ทรีทเมนต์) หรือตัวแปรอิสระเพียงตัวเดียวแต่ถูกแบ่งออกเป็นหลายระดับหรือหลายกลุ่ม ตัวอย่าง เช่น การทดลองเพื่อเปรียบเทียบอาหารเลี้ยงกุ้ง 4 ชนิด คือ A, B, C และ D ที่ใช้เลี้ยงกุ้งกุลาดำ หรือเปรียบเทียบน้ำหนักของทารกจากมารดาที่มีปริมาณการสูบบุหรี่ที่แตกต่างกัน

ตัวอย่าง 8.1

(1) **Birth weight data** : น้ำหนักเฉลี่ยของทารกจำแนกปริมาณการสูบบุหรี่ของมารดา

	จำนวนบุหรี่ที่สูบ (มวน)			รวม
	<10	10-19	20+	
น้ำหนักทารกโดยเฉลี่ย (ounces)	119.45	112.62	114.85	116.45
จำนวน cases	38	13	40	91

(2) ผู้จัดการโรงงานผลิตปลากระป๋องแห่งหนึ่งต้องการศึกษาว่าอัตราความเร็วของสายพานสำหรับขับเคลื่อนปลากระป๋องที่ผลิตได้จะมีผลต่อจำนวนกระป๋องที่มีตำหนิต่อล็อต (บรรจุล็อตละ 500 กระป๋อง) หรือไม่ เขาจึงเดินเครื่องจักรด้วยอัตราความเร็ว 4 ระดับ คือ ระดับ 1, 2, 3 และ ระดับ 4 ในแต่ละระดับเลือกตัวอย่างมา 4, 5, 3 และ 5 ล็อต ตามลำดับ แต่ละล็อตตัวอย่างนับจำนวนกระป๋องที่มีตำหนิ ได้ดังตาราง (ดัดแปลงจาก มัลลิกา บุญนาค 2536 สถิติเพื่อการตัดสินใจ หน้า 188)

ความเร็วขับเคลื่อน	ระดับ 1	ระดับ 2	ระดับ 3	ระดับ 4
	37	37	32	35
	35	35	36	27
	38	38	40	33
	36	39		31
		37		29

8.1 ฟังก์ชัน R สำหรับ ANOVA

ฟังก์ชันหรือคำสั่งใน **R** สำหรับ การวิเคราะห์ความแปรปรวน ได้แก่

ฟังก์ชัน/คำสั่งใน R	คำอธิบาย
aov()	สร้างตารางวิเคราะห์ความแปรปรวนภายใต้ Homogeneity of variance assumption
oneway.test()	สร้างตารางวิเคราะห์ความแปรปรวนโดยไม่ต้องคำนึงถึง Homogeneity of variance assumption
bartlett.test()	ทดสอบการเท่ากันของความแปรปรวนของประชากรตั้งแต่สองกลุ่มขึ้นไป
TukeyHSD()	Multiple comparisons โดยใช้ Tukey HSD Procedure

aov()

วิธีใช้

```
aov(formula, data = NULL, projections = FALSE, qr = TRUE,
contrasts = NULL, ...)
```

oneway.test()

วิธีใช้

```
oneway.test((formula, data, subset, na.action, var.equal = FALSE)
```

formula: สูตรที่ใช้นิยามตัวแบบเชิงเส้นของข้อมูลที่ต้องการวิเคราะห์ความแปรปรวน เช่น ตัวแปรตาม~ ตัวแปรอิสระ, ตัวแปรอิสระต้องเป็น แฟกเตอร์

data: กรอบข้อมูลที่เก็บตัวแปรที่ระบุใน formula ถ้าไม่ระบุ **R** จะค้นหาตัวแปรเหล่านั้น จาก working directory

bartlett.test()

วิธีใช้

```
   bartlett.test(x, g, ...)
หรือ
   bartlett.test(formula, data, subset, na.action, ...)
```

x: เวกเตอร์ของข้อมูลที่เป็นเชิงปริมาณ

g: เวกเตอร์ของข้อมูลที่เป็นแฟกเตอร์

formula: สูตรที่ใช้นิยามตัวแบบเชิงเส้นของข้อมูลที่ต้องการวิเคราะห์ความแปรปรวน เช่น ตัวแปรตาม~ ตัวแปรอิสระ, ตัวแปรอิสระต้องเป็น แฟกเตอร์

data: กรอบข้อมูลที่เก็บตัวแปรที่ระบุใน formula ถ้าไม่ระบุ R จะค้นหาตัวแปรเหล่านั้น จาก working directory

ตัวอย่าง 8.2 (1) **Birth weight data**: ปริมาณการสูบบุหรี่/วัน จำแนกเป็น น้อยกว่า 10 มวน 10-19 และ 20 มวนขึ้นไปของมารดามีอิทธิพลต่อน้ำหนักเฉลี่ยของทารก หรือไม่

```
> wt.sm <- wt[number!=0]
> num.sm <- number[number!=0]
```

น้ำหนักทารกที่มารดามีประวัติการสูบบุหรี่

จำนวนบุหรี่ที่สูบ/วันของมารดาที่มีประวัติ

```
> summary(wt.sm)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   86.0   104.5   115.0   116.5   127.5   157.0     3.0
> summary(num.sm)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  1.000   2.000   3.000   3.348   5.000   7.000   2.000
```

เนื่อง wt.sm และ num.sm มีจำนวน NA ไม่เท่ากันจึงต้อง remove cases ที่มี NA ออก

```
> wtnum.dat <- data.frame(wt.sm, num.sm)
> wtnum.dat <- na.omit(wtnum.dat)
> attach(wtnum.dat); rm(wt.sm, num.sm)
> wt.sm <- wtnum.dat$wt.sm
> num.sm <- wtnum.dat$num.sm
> num.sm <- replace(num.sm,num.sm==2, 1)
> num.sm <- replace(num.sm,num.sm==3| num.sm==4, 2)
> num.sm <- replace(num.sm,num.sm >=5, 3)
> num.sm <- factor(num.sm, labels =c("<10", "10-19","20+"))
```

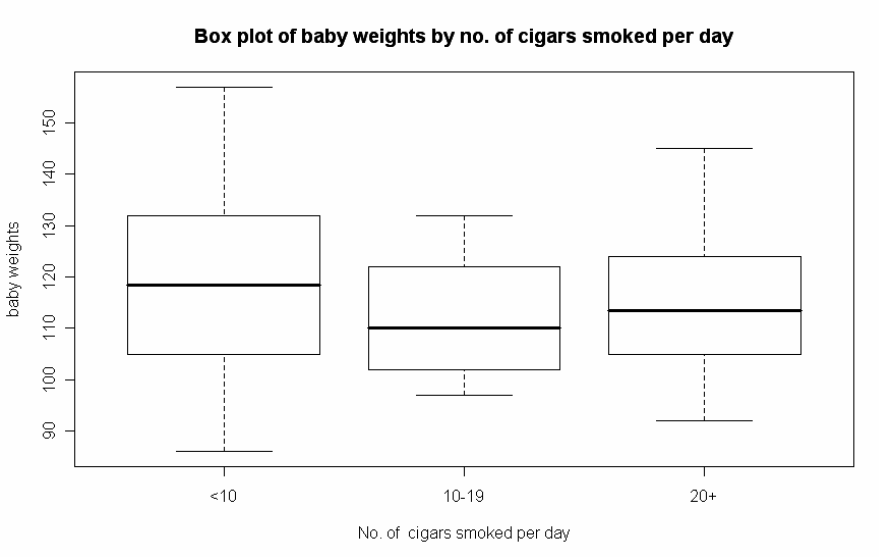
```
> tapply(wt.sm,num.sm,mean)
     <10    10-19      20+
119.4474 112.6154 114.8500
```

น้ำหนักเฉลี่ยของทารกที่มารดามีประวัติการสูบบุหรี่ตามที่ต้องการศึกษา

```
> table(num.sm)
num.sm
  <10 10-19   20+
   38    13    40
```

จำนวนมารดาที่มีประวัติการสูบบุหรี่ตามที่ต้องการศึกษา

```
> boxplot(wt.sm~num.sm, main="Box plot of baby weights by no. of
            cigars smoked per day")
> title(xlab="No. of  cigars smoked per day")
> title(ylab="baby weights")
```



รูป 8.1

```
>#  Test of homogeneity of variance  (ทดสอบการเท่ากันของความแปรปรวน)
>   bartlett.test(wt.sm~factor(num.sm))

        Bartlett test of homogeneity of variances

data:  wt.sm by factor(num.sm)
Bartlett's K-squared = 7.2921, df = 2, p-value = 0.02609
```

ความแปรปรวนของน้ำหนักเฉลี่ยของทารกของมารดาแต่ละกลุ่มต่างกันอย่างมีนัยสำคัญ

ดังนั้นจึงวิเคราะห์ความแปรปรวนโดยใช้คำสั่ง oneway.test()

```
> oneway.test(wt.sm~factor(num.sm))# (ทดสอบการเท่ากันของค่าเฉลี่ย)

        One-way analysis of means (not assuming equal variances)

data:  wt.sm and factor(num.sm)
F = 1.2016, num df = 2.000, denom df = 39.171, p-value = 0.3116
```

อิทธิพลของปริมาณการสูบบุหรี่/วันของมารดาที่มีต่อน้ำหนักเฉลี่ยของทารก ไม่มีนัยสำคัญ

ตัวอย่าง 8.3 จากตัวอย่าง 8.1 (2) ต้องการทดสอบว่า อัตราความเร็วของสายพานสำหรับขับเคลื่อนปลากระป๋องที่ผลิตได้จะมีผลต่อจำนวนกระป๋องที่มีตำหนิต่อล็อต (บรรจุล็อตละ 500 กระป๋อง) หรือไม่

ANOVA

```
> # directly entry data to using R function c()
> defect <- c(37,35,38,36,37,35,38,39,37,32,36,40,35,27,33,31,29)
> # create a vector of replication named speed
>  speed <- rep(c(1,2,3,4),c(4,5,3,5))

> tapply(defect, speed, sum)
  1   2   3   4
146 186 108 155
> tapply(defect, speed, mean)
   1    2    3    4
36.5 37.2 36.0 31.0
> tapply(defect, speed, var)
        1         2         3         4
 1.666667  2.200000 16.000000 10.000000

> # Test homogeneity of variance
> bartlett.test(defect~factor(speed))

        Bartlett test for homogeneity of variances

data:  defect by factor(speed)
Bartlett's K-squared = 4.4665, df = 3, p-value = 0.2153
```

ความแตกต่างระหว่างความแปรปรวนของจำนวนกระป๋องที่มีตำหนิต่อล็อตจากแต่ละอัตราความเร็วของสายพาน ไม่มีนัยสำคัญ

ดังนั้นจึงวิเคราะห์ความแปรปรวนโดยใช้คำสั่ง aov()

```
> # ตาราง ANOVA (ANOVA table)
> summary(aov(defect~factor(speed)))
             Df  Sum Sq Mean Sq F value  Pr(>F)
factor(speed)  3 116.200  38.733  5.8687 0.00924 **
Residuals     13  85.800   6.600
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

ความเร็วสายพานมีอิทธิพลต่อจำนวนกระป๋องที่มีตำหนิโดยเฉลี่ยอย่างมีนัยสำคัญ

Multiple Comparisons โดยใช้ ฟังก์ชัน TukeyHSD()

วิธีใช้

```
TukeyHSD(x, which, ordered = FALSE, conf.level = 0.95, ...)
```

: วัตถุที่เป็นตัวแบบ ซึ่งอยู่ใน class aov

hich : แฟกเตอร์ ใน x ที่ต้องการวิเคราะห์ Multiple Comparison ค่าปกติ คือ ทุกแฟกเตอร์

rdered : เงื่อนไขที่ระบุว่าต้องการเรียงลำดับ levels ของแฟกเตอร์ ที่ต้องการ วิเคราะห์ Multiple Comparison หรือไม่ ค่าปกติคือ FALSE

onf.level : ระดับความเชื่อมั่นที่ใช้ทดสอบ ค่าปกติคือ 0.95

ตัวอย่าง 8.4 จากตัวอย่าง 8.1 (2) Multiple comparisons โดย TukeyHSD

```
> TukeyHSD(aov(defect~factor(speed)))
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = defect ~ factor(speed))

$`factor(speed)`
    diff        lwr        upr     p adj
2-1  0.7  -4.358270  5.7582696 0.9764120
3-1 -0.5  -6.259093  5.2590933 0.9939057
4-1 -5.5 -10.558270 -0.4417304 0.0315332   ⟶ มีนัยสำคัญ
3-2 -1.2  -6.706746  4.3067464 0.9172789
4-2 -6.2 -10.968982 -1.4310177 0.0100930   ⟶ มีนัยสำคัญ
4-3 -5.0 -10.506746  0.5067464 0.0804146
```

แบบฝึกหัด จงทดสอบว่าระยะเวลาที่เลิกสูบบุหรี่ของมารดาที่มีประวัติการสูบบุหรี่ (smoke) มีอิทธิพลต่อน้ำหนักของทารก (wt) อย่างมีนัยสำคัญ หรือไม่

บทที่ 9
การวิเคราะห์การถดถอยเชิงเดียว
(Simple Regression Analysis)

การวิเคราะห์สหสัมพันธ์และการถดถอยเป็นระเบียบวิธีทางสถิติอีกวิธีหนึ่งที่ใช้ศึกษาความสัมพันธ์ระหว่างตัวแปรตั้งแต่ 2 ตัวแปรขึ้นไป เช่น ความสัมพันธ์ระหว่างความสูง (Y) และน้ำหนัก (X) ความสัมพันธ์ระหว่างค่าใช้จ่าย (Y) กับรายได้ (X_1) และจำนวนสมาชิกในครัวเรือน (X_2) เป็นต้น **การวิเคราะห์สหสัมพันธ์** เป็นการวิเคราะห์เพื่อหาระดับและทิศทางของความสัมพันธ์แบบเชิงเส้นระหว่างตัวแปร **ส่วนการวิเคราะห์การถดถอย** เป็นการหารูปแบบความสัมพันธ์ในรูปสมการทางคณิตศาสตร์โดยมีวัตถุประสงค์เพื่อใช้ในการประมาณหรือทำนายค่าของตัวแปรหนึ่ง (ตัวแปรตาม : dependent variable) เมื่อกำหนดค่าของอีกตัวแปรหนึ่ง (ตัวแปรอิสระ: independent variable) อย่างไรก็ตามคู่มือการใช้โปรแกรม **R** เล่มนี้จะกล่าวถึงเฉพาะการวิเคราะห์การถดถอยเชิงเดียวซึ่งเป็นการวิเคราะห์การถดถอยที่มีตัวแปรตามและตัวแปรอิสระอย่างละหนึ่งตัวเท่านั้น

9.1 ฟังก์ชัน R สำหรับการวิเคราะห์การถดถอย

ฟังก์ชันหรือคำสั่งใน R สำหรับ การวิเคราะห์การถดถอยประกอบด้วย

ฟังก์ชัน/คำสั่งใน R	คำอธิบาย
plot()	สร้างแผนภาพการกระจาย (scatter plot) สำหรับตรวจสอบความสัมพันธ์ระหว่างตัวแปร 2 ตัว
cor()	คำนวนค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปร
cor.test()	ทดสอบสมมติฐานเกี่ยวกับสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปร 2
lm()	คำนวนสัมประสิทธิ์การถดถอย (Regression coefficients)
summary(lm())	ทดสอบสมมติฐานเกี่ยวกับสัมประสิทธิ์การถดถอยโดยใช้ สถิติ t
anova(lm())	ทดสอบสมมติฐานเกี่ยวกับสัมประสิทธิ์การถดถอยโดยใช้สถิติ F หรือ สร้างตาราง ANOVA
predict.lm()	คำนวนค่าของตัวแปรตามเมื่อกำหนดค่าของตัวแปรอิสระ
qqnorm(); qqline()	ตรวจสอบการแจกแจงปกติของเศษตกค้าง

9.2 แผนภาพการกระจาย (Scatter plots)

plot()

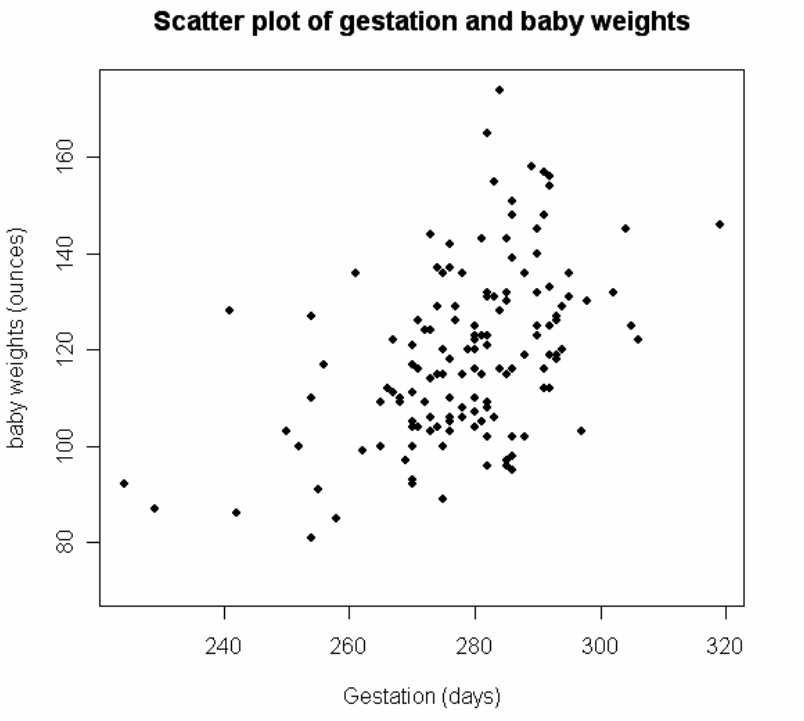
วิธีใช้

```
plot(x, y, ...)
```

x , y : เวกเตอร์ของคู่ลำดับที่ต้องการสร้าง scatter plot

ตัวอย่าง 9.1 Birth weight data : จงสร้าง scatter plot สำหรับตัวแปร wt, gestation

```
> plot(gestation,wt, xlab="Gestation (days)",
       ylab="baby weights (ounces)", pch = 19)

> title("Scatter plot of gestation and baby weights")
```



รูป 9.1

9.3 สัมประสิทธิ์สหสัมพันธ์ (Correlation Coefficients)

cor() เป็นฟังก์ชันสำหรับคำนวณค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปร

วิธีใช้

```
cor(x, y = NULL, use = "all.obs", method = c("pearson",
    "kendall", "spearman"))
```

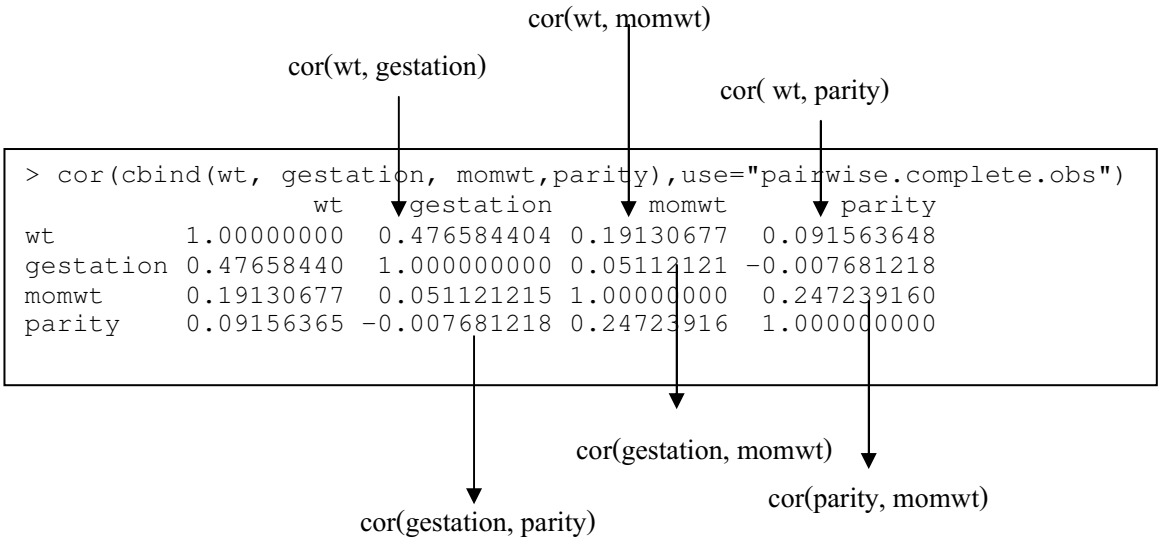
x: vector หรือ matrix หรือ data frame ที่มีข้อมูลเชิงปริมาณ

y: vector หรือ matrix หรือ data frame ที่มีมิติเหมือนกับ x ค่า ค่าปกติ คือ 'NULL' ซึ่งมีความหมายว่า 'y = x'

use: ตัวเลือกที่บอกวิธีคำนวณค่าความแปรปรวนร่วมเมื่อมีค่าสูญหายในชุดข้อมูล ค่าที่จะกำหนดได้ใน use คือ "all.obs", "complete.obs" และ "pairwise.complete.obs"

method: ตัวเลือกที่บอกว่าจะคำนวณค่าด้วยวิธีใด "pearson" (ค่าปกติ), หรือ "kendall" หรือ "spearman"

ตัวอย่าง 9.2 Birth weight data : จงคำนวน multiple correlation ระหว่างตัวแปร wt, gestation, momwt และ parity



cor.test() เป็นฟังก์ชันสำหรับใช้ทดสอบสมมติฐานเกี่ยวกับสัมประสิทธิ์สหสัมพันธ์ระหว่าง ตัวแปรสองตัว

วิธีใช้

```
cor.test(x, y, alternative = c("two.sided", "less", "greater"),
        method = c("pearson", "kendall", "spearman"),
         exact = NULL, conf.level = 0.95, ...)
```

x, y : ตัวแปรหรือเวคเตอร์ของตัวแปรสุ่มแบบต่อเนื่อง x และ y ต้องมีขนาดเท่ากัน

exact : เงื่อนไขที่ระบุว่า ต้องการ/ไม่ต้องการให้คำนวณค่า exact p-value ใช้กับกรณี method = "kendall" เท่านั้น

ตัวอย่าง 9.3 Birth weight data : จงทดสอบค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปร wt และ gestation

```
> cor.test(wt, gestation, data=baby23.dat)

        Pearson's product-moment correlation

data:  wt and gestation
t = 6.5053, df = 144, p-value = 1.196e-09
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.3404975 0.5931137
sample estimates:
      cor
0.4765844
```

```
> cor.test(wt, gestation, method="kendall",data=baby23.dat)

        Kendall's rank correlation tau

data:  wt and gestation
z = 5.9982, p-value = 1.995e-09
alternative hypothesis: true tau is not equal to 0
sample estimates:
      tau
0.3349161

> cor.test(wt, gestation, method="spearman",data=baby23.dat)

        Spearman's rank correlation rho

data:  wt and gestation
S = 271839, p-value = 1.956e-09
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.4758864
```

สรุป ทั้งวิธีของ Pearson's, Spearman's และวิธีของ Kendall's ต่างให้ผลสรุปเหมือนกัน คือ น้ำหนักทารกและระยะเวลาที่อยู่ในครรภ์มารดามีความสัมพันธ์เชิงเส้นต่อกัน (ระดับนัยสำคัญ 0.05)

9.4 การวิเคราะห์การถดถอยเชิงเดียว (Simple regression analysis)

9.4.1 การประมาณค่าสัมประสิทธิ์การถดถอยและการทดสอบสมมติฐาน

ถ้าให้สมการการถดถอยของข้อมูลจากประชากร ที่มี y เป็นตัวแปรตามและ x เป็นตัวแปรอิสระ นิยามด้วย

$$\mu_y = \beta_0 + \beta_1 x$$

สมการการถดถอยของข้อมูลจากตัวอย่างขนาด n นิยามด้วย

$$\hat{y} = b_0 + b_1 x$$

สัมประสิทธิ์การถดถอย b_0, b_1 เป็นค่าประมาณของ β_0, β_1 ตามลำดับ ค่าของ b_0, b_1 และการทดสอบสมมติฐานเกี่ยวกับการถดถอยเชิงเส้น β_0, β_1 โดยใช้โปรแกรม **R** ดำเนินการ ให้ใช้คำสั่ง lm()

วิธีใช้

```
lm(formula, data, subset, weights, na.action,
   method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,
   singular.ok = TRUE, contrasts = NULL, offset = NULL, ...)
```

ข้อมูลนำเข้าที่ใช้โดยทั่วไป คือ

formular : สูตรสำหรับการ fit ตัวแบบซึ่งประกอบด้วย ตัวแปร ตามและตัวแปรอิสระ เช่น y ~ x

data : กรอบข้อมูลที่เก็บตัวแปรที่ระบุใน formula

Tip

fit ตัวแบบเชิงเส้น (linear models)ที่มีตัวแปรตอบสนอง (y) และตัวแปรอธิบาย สมมติเป็น x1, x2 โดยให้ x2 เป็นแฟกเตอร์หรือตัวแปรหุ่น (dummy variable)

ตัวอย่าง formula

(1) the null model : y~1;

(2) a main effect model : y~x1+x2;

(3) the interaction model : y~x1+x2+x1:x2 หรือ y~x1*x2

ตัวอย่าง 9.4 Birth weight data: จงเขียนสมการถดถอยอย่างง่าย (Simple linear regression) แสดงความสัมพันธ์ระหว่าง wt และ, gestation

สมการถดถอยของประชากรคือ $\mu_{wt} = \beta_0 + \beta_1(\text{gestation})$

ดำเนินการโดยใช้คำสั่ง lm() ในโปรแกรม **R** และเก็บ fitted model ในชื่อของ wt.reg ดังนี้

```
> wt.reg1 <- lm(wt~gestation)
> wt.reg1

Call:
lm(formula = wt ~ gestation)

Coefficients:
(Intercept)    gestation
   -51.2886       0.6107
```

$$\text{wt} = -51.286 + 0.6107(\text{gestation})$$

Tip

ถ้าต้องการทดสอบสมมติฐานสัมประสิทธิ์การถดถอยแต่ละตัว ให้ใช้คำสั่ง summary(lm())

```
> summary(wt.reg1)

Call:
lm(formula = wt ~ gestation)

Residuals:
    Min      1Q  Median      3Q     Max
-28.381 -10.480  -1.498   8.448  51.841

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -51.28862   26.19843  -1.958   0.0522 .    ❶
gestation     0.61073    0.09388   6.505 1.20e-09 ***  ❷
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 15.41 on 144 degrees of freedom
Multiple R-Squared: 0.2271, ❸   Adjusted R-squared: 0.2218
F-statistic: 42.32 on 1 and 144 DF,  p-value: 1.196e-09
                ❹
```

❶ $H_0 : \beta_0 = 0$

❷ $H_0 : \beta_1 | \beta_0 = 0$; **สรุป** wt เป็นฟังก์ชันเชิงเส้นของ gestation

❸ Multiple R-Squared: 0.2271 แสดงว่า gestation อธิบายการเปลี่ยนแปลงของ wt ได้ 22.71%

❹ สถิติ F ที่ใช้ทดสอบว่า ค่าเฉลี่ยของ wt เป็นฟังก์ชันเชิงเส้นของ gestation หรือไม่ ซึ่งสอดคล้องกับทดสอบ $H_0 : \beta_1 = 0$; $H_1 : \beta_1 \neq 0$

จาก ❶ ได้ว่า $\beta_0 = 0$ ดังนั้นสรุปได้ว่าถ้าไม่มีการตั้งครรภ์ คือ gestation = 0 แล้วน้ำหนักทารกมีค่าเป็นศูนย์ ดังนั้นสมการถดถอยจึงไม่มีค่าคงที่ หรือไม่มี intercept ซึ่ง fitted model ที่ไม่มี intercept สามารถ fit โดยใช้โปรแกรม **R** ทำได้ ดังนี้

```
> wt.reg0 <- lm(wt~gestation-1)
> summary(wt.reg0)

Call:
lm(formula = wt ~ gestation - 1)

Residuals:
    Min      1Q  Median      3Q     Max
-28.468 -11.427  -2.541   8.226  52.687

Coefficients:
          Estimate Std. Error t value Pr(>|t|)
gestation 0.427158   0.004616   92.54   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 15.56 on 145 degrees of freedom
Multiple R-Squared: 0.9834,     Adjusted R-squared: 0.9832
F-statistic:  8564 on 1 and 145 DF,  p-value: < 2.2e-16
```

สมการถดถอยหรือ fitted model คือ wt = $0.4272(\text{gestation})$ โดยมีค่าMultiple R-Squared: 0.9834 แสดงว่า สมการถดถอยนี้สามารถอธิบายการเปลี่ยนแปลงของน้ำหนักทารกตัวอย่างได้ 98.34%

9.4.2 การตรวจสอบข้อสมมติพื้นฐานของสมการถดถอย (Model checking)

การตรวจสอบข้อสมมติพื้นฐานของสมการถดถอย (Model checking) ประกอบด้วย

(1) Constant variance นั่นคือตรวจสอบว่า ความแปรปรวนของตัวแปรตามหรือความแปรปรวนของเศษตกค้าง (Residuals) มีค่าคงที่

(2) Normality assumption นั่นคือตรวจสอบว่าตัวแปรตามหรือเศษตกค้าง (Residuals) มีการแจกแจงแบบปกติ

เราสามารถตรวจสอบข้อสมมติพื้นฐานทั้งสองข้อข้างต้นได้โดยใช้การวิเคราะห์เศษตกค้าง และวิธีที่นิยม คือ Graphical analysis โดย

- ตรวจสอบ (1)โดยลงจุดระหว่าง ค่าคาดคะเน (Fitted values) กับเศษตกค้าง ถ้าการกระจายของจุดเป็นไปอย่างสุ่ม แสดงว่าความแปรปรวนของตัวแปรตามหรือความแปรปรวนของเศษตกค้าง (Residuals) คงที่
- ตรวจสอบ (2) โดยวาด histogram หรือวาด Normal Q-Q plot ของเศษตกค้าง

```
> par(mfrow=c(2,2))

> plot(wt.reg0$fitted.values,wt.reg0$residuals)
> title(main = "Constant variance assumptiom")

> hist(wt.reg0$residuals, probability=T)
> lines(density(wt.reg0$residuals, bw = 15))

> qqnorm(wt.reg0$residuals)
> qqline(wt.reg0$residuals)
```

9.4.2 (1), รูป 9.2 (ก)

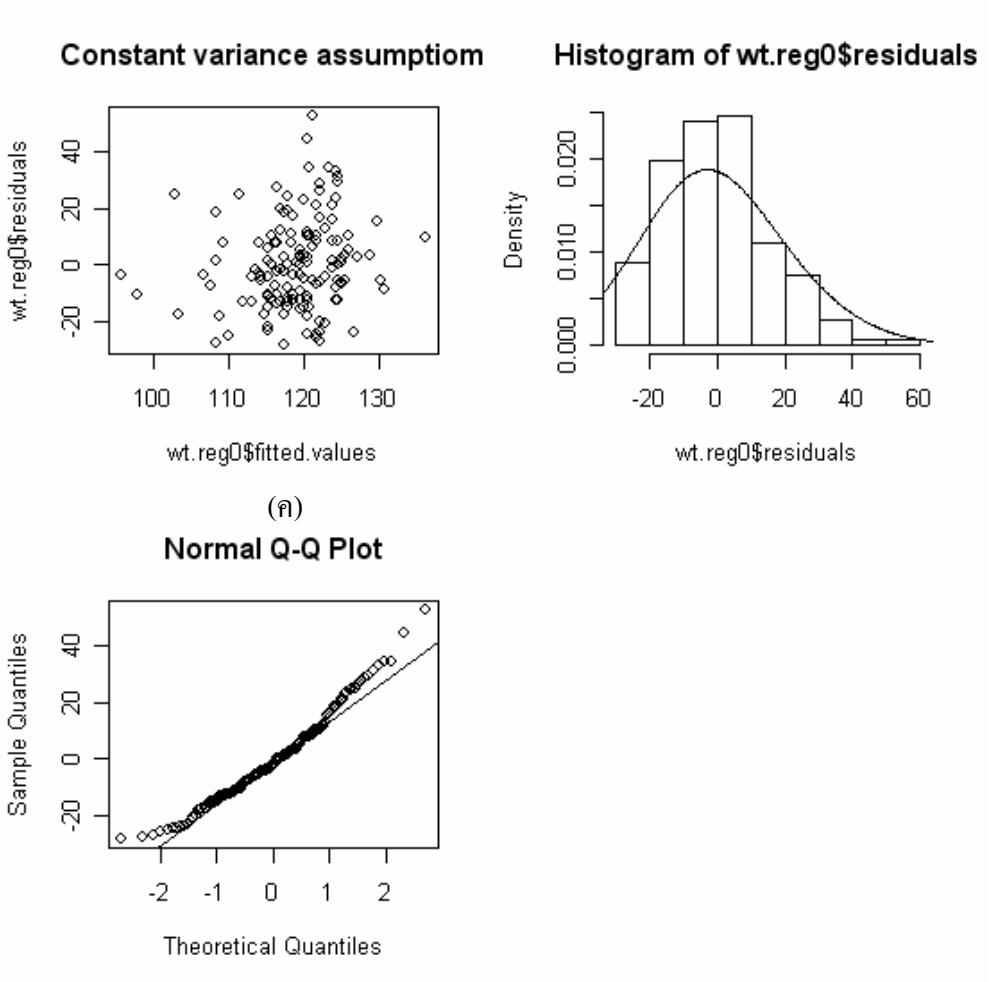
9.4.2 (2) , รูป 9.2 (ข) และ (ค)

Tip

ต้องการดูว่า wt.reg0 เก็บค่าอะไรไว้บ้างให้ใช้คำสั่ง attributes(wt.reg1) เช่น

```
> attributes(wt.reg0)
$names
 [1] "coefficients"  "residuals"     "effects"       "rank"
 [5] "fitted.values" "assign"        "qr"            "df.residual"
 [9] "xlevels"       "call"          "terms"         "model"

$class
[1] "lm"
```

(ก) (ข)



รูป 9.2

9.4.3 ค่าทำนาย (predicted values)

จากสมการถดถอยหรือ fitted model $\hat{y} = b_0 + b_1 x$ เราสามารถหาค่าทำนายได้ 2 ลักษณะ คือ ค่าทำนายสำหรับค่าเฉลี่ยของ y เมื่อกำหนดให้ $x = x_0$ ในทางสถิติอาจแทนด้วย $\hat{y}_0$ ที่มี

$$S^2_{\hat{Y}_0} = \text{MSE}\left[\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right]$$

และ ค่าทำนายสำหรับค่าเดี่ยวของ y เมื่อกำหนดให้ $x = x_0$ ในทางสถิติอาจแทนด้วย $\hat{y}_{\text{pred}}$ ที่มี

$$S^2_{\hat{Y}_{\text{pred}}} = \text{MSE}\left[1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\right]$$

คำสั่ง **predict()** ใน **R** สามารถคำนวนค่าทำนายทั้งสองชนิด รวมทั้งช่วงความเชื่อมั่น $100(1-\alpha)\%$ ของค่าทำนาย อย่างไรก็ตาม **R** จะแสดงค่าความคลาดเคลื่อนมาตรฐาน (Standard error) $S_{\hat{y}_0}$ ของ $\hat{y}_0$ เท่านั้น

วิธีใช้

```
predict(object, newdata, se.fit = FALSE, scale = NULL, df = Inf,
        interval = c("none", "confidence", "prediction"),
        level = 0.95, type = c("response", "terms"),
        terms = NULL, na.action = na.pass, ...)
```

object : วัตถุที่ได้จากการ fit ตัวแบบเชิงเส้นโดยใช้ฟังก์ชัน lm

newdata : กรอบข้อมูลที่เก็บค่าของตัวแปรอิสระที่ต้องการใช้ทำนาย

se.fit : ระบุว่าต้องการให้คำนวณและแสดงค่า standard error ของค่าทำนายหรือไม่ scale

interval : ระบุพารามิเตอร์ (ของค่าเฉลี่ย หรือ ค่าทำนาย) ที่ต้องการหาช่วงความเชื่อมั่น

level : ระดับความเชื่อมั่น

type : ระบุตัวแปรที่ต้องการการทำนายค่า (ตัวแปรตาม หรือ ตัวแปรอิสระ)

terms : ถ้า type="terms", ให้ระบุ ตัวแปรอิสระที่ต้องการ (ค่าปกติคือ ทุกเทอมในตัวแบบ)

ตัวอย่าง 9.5 Birth weight data :

1) จงประมาณค่าเฉลี่ยของ wt เมื่อทราบว่า gestation มีค่าเป็น 285 วัน พร้อมกับคำนวณค่าความคลาดเคลื่อนมาตรฐานและช่วงความเชื่อมั่น 95%

```
> newdata <- data.frame(gestation=285)
> predict(wt.reg0, newdata, se.fit = T, level= 0.95,
          interval = "confidence")

$fit
          fit      lwr    upr
[1,] 121.7399 119.1399 124.34

$se.fit
[1] 1.315512

$df
[1] 145

$residual.scale
[1] 15.56394
```

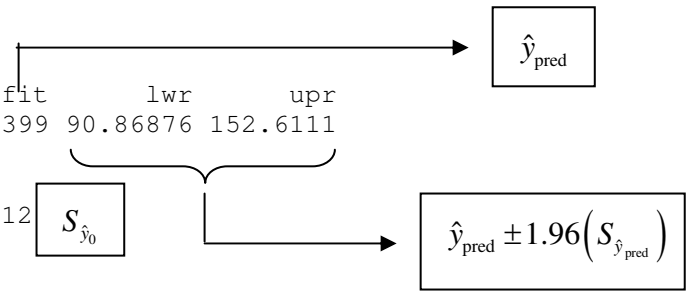
$\hat{y}_0$

$S_{\hat{y}_0}$

$\hat{y}_0 \mp 1.96(S_{\hat{y}_0})$

2) จงประมาณค่าของ wt เมื่อทราบว่า gestation มีค่าเป็น 285 วัน พร้อมกับคำนวณค่าความคลาดเคลื่อนมาตรฐานและช่วงความเชื่อมั่น 95%

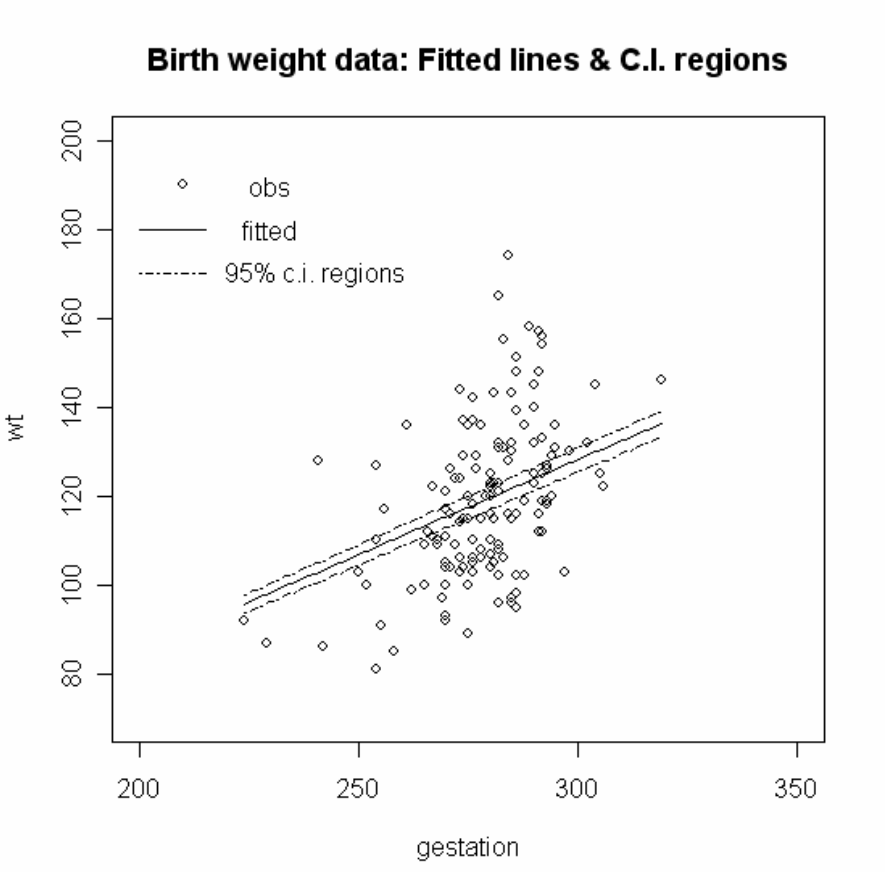
```
> newdata <- data.frame(gestation=285)
> predict(wt.reg0, newdata,level= 0.95, interval = "prediction")


$fit
           fit      lwr      upr
[1,] 121.7399 90.86876 152.6111

$se.fit
[1] 1.315512

$df
[1] 145

$residual.scale
[1] 15.56394
```



3) จงประมาณค่าเฉลี่ยของ wt สำหรับกลุ่มมารดาตัวอย่างที่ใช้เวลาตั้งครรภ์ gestation อยู่ในช่วง 224-319 วัน คำนวณช่วงความเชื่อมั่น 95% พร้อมกับเขียนกราฟ

```
> newdata <- data.frame(gestation=seq(224, 319, length=129))
> wt.pred1 <- predict(wt.reg0, newdata, level= 0.95,
              interval = "confidence")

> attributes(wt.pred1)
> plot(gestation, wt, type="n", xlim = c(200, 350), ylim=c(70,200))
> points(gestation, wt, pch =1)
> lines(newdata$gestation,wt.pred1[,1],lty=1)
> lines(newdata$gestation,wt.pred1[,2],lty=4)
> lines(newdata$gestation,wt.pred1[,3],lty=4)
> title(main = "Birth weight data: Fitted lines & C.I. regions")
> points(210,190, pch=1)
> text(230,190, "obs" )
> lines(c(200,215),c(180,180),lty=1)
> text(230,180, "fitted" )
> lines(c(200,215),c(170,170),lty=4)
> text(240,170, "95% c.i. regions" )
```

ดูรูป 9.3



รูป 9.3

แบบฝึกหัด

บริษัทน้ำอัดลมแห่งหนึ่งต้องการศึกษาว่า จำนวนเงินโฆษณาทางโทรทัศน์ (หน่วย : 10,000 บาท) ที่บริษัทจ่ายไปนั้นจะมีความสัมพันธ์กับยอดขาย (หน่วย : 10,000 บาท) หรือไม่ จึงรวบรวมข้อมูลในรอบ 12 เดือนที่ผ่านมา ได้ข้อมูลดังนี้

เดือน	ค่าใช้จ่าย	ยอดขาย	เดือน	ค่าใช้จ่าย	ยอดขาย
มกราคม	1.5	83	กรกฎาคม	0.85	45
กุมภาพันธ์	1.35	80	สิงหาคม	1.15	62
มีนาคม	0.40	35	กันยายน	0.95	43
เมษายน	0.70	41	ตุลาคม	0.95	40
พฤษภาคม	1.15	64	พฤศจิกายน	1.20	85
มิถุนายน	0.85	40	ธันวาคม	1.50	90

1. จงเขียนแผนภาพการขจายระหว่างยอดขาย กับจำนวนเงินโฆษณาทางโทรทัศน์
2. จงดำเนินการวิเคราะห์การถดถอย อย่างละเอียด ดังตัวอย่างข้างต้น

เอกสารอ้างอิง

1. จรัญ จันทลักขณา *สถิติวิธีวิเคราะห์และวางแผนงานวิจัย* บริษัทสำนักพิมพ์ไทยวัฒนาพานิช จำกัด กทม. 2540
2. มัลลิกา บุญนาค *สถิติเพื่อการตัดสินใจ* ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย 2536.
3. ศศิธร สุวิรัชวิทยกิจ. สถิติสำหรับนักวิทยาศาสตร์และนักวิทยาศาสตร์ประยุกต์ เล่ม1-2 สาขาสถิติ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร 2543.
4. อภิญญา วงศ์กิดาการ สถิติสำหรับชีววิทยา ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ สงขลา 2531.
5. Daniel W. *Biostatistics: A Foundation for Analysis in the Health Sciences, 7th edition*. John Wiley & Sons, INC. New York. 1999
6. Groebner D. F., Shannon P. W., Fry P. C. and Smith K. D. *Business Statistics (A Decision making Approach)6th edition.*Pearson Edition, Inc. New Jersey. 2005
7. http://stat-www.berkeley.edu/users/statlabs/lab.html.
8. R Development Core Team R: *A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing. Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org. 2006.

ดัชนี

ภาษาอังกฤษหรือสัญลักษณ์ตัวหนา หมายถึงคำสั่งหรือฟังก์ชัน **R**

การกำหนดชื่อของวัตถุ

ฟังก์ชัน

cbind() สร้างเมทริกซ์จากเวกเตอร์หรือกรอบข้อมูลโดยนำมาเรียงต่อกันในแนวคอลัมน์

rbind() สร้างเมทริกซ์จากเวกเตอร์หรือกรอบข้อมูลโดยนำมาเรียงต่อกันในแนวแถว

attach()

detach()

log(100) คำนวณ ln(100)

exp(5) คำนวณ e^5

sqrt(16) คำนวณรากที่สองของ 16

abs(-5) คำนวณค่าสัมบูรณ์ของ -5

สัญลักษณ์ทางคณิตศาสตร์

\+ บวก

- ลบ

***** คูณ

/ หาร

^ ยกกำลัง

สัญลักษณ์ทางตรรกศาสตร์

< น้อยกว่า

> มากกว่า

= = เท่ากับ

<= น้อยกว่าหรือเท่ากับ

>= มากกว่าหรือเท่ากับ

& และ

| หรือ

การใช้ตัวอักษรใน **R**

เรียกดูแถวหรือสดมภ์ในกรอบข้อมูล

ลบวัตถุออกจาก R console

rm()

การสร้างวัตถุใน **R**

<- Left assignment ให้นำค่าหรือผลจาการทำงานของฟังก์ชันที่ปลายลูกศร ไปเก็บในวัตถุที่เขียนไว้ที่หัวลูกศร (ด้านซ้าย)

-> Right assignment หมายถึง ให้นำค่าหรือผลจากการทำงานของฟังก์ชันที่ปลายลูกศร ไปเก็บในวัตถุที่เขียนไว้ที่หัวลูกศร (ด้านขวา)

= เป็นเครื่องหมาย assignment เช่นเดียวกับ <- แต่ไม่ควรใช้เพราะจะสับสนกับเครื่องหมาย = = ที่เป็นเครื่องหมายเท่ากับทางตรรกศาสตร์

การสร้างเวกเตอร์

นำข้อมูลหรือวัตถุที่ต้องการมาเรียงต่อกันโดยใช้ฟังก์ชัน **c()**

เป็นตัวเลขที่เป็นระบบ โดยใช้ฟังก์ชัน **seq()**

ที่มีค่าซ้ำ โดยใช้ฟังก์ชัน **rep()**

ที่ค่าเป็นตัวแปรเชิงกลุ่มหรือ factor โดยใช้ฟังก์ชัน **factor()**

ตัวอักษร (character vector)

การเปลี่ยนและการตรวจสอบชนิดของเวกเตอร์

as.integer() เปลี่ยนคุณสมบัติของวัตถุที่อยู่ใน () ให้เป็น integer

as.factor() เปลี่ยนคุณสมบัติของวัตถุที่อยู่ใน ()ให้เป็น factor

as.numeric() เปลี่ยนคุณสมบัติของวัตถุที่อยู่ใน ()ให้เป็น numeric

การสร้างแฟ้มข้อมูลหรือกรอบข้อมูล

สร้างกรอบข้อมูล (แฟ้มข้อมูล) โดยใช้ฟังก์ชัน **data.frame()**

เป็นเมทริกซ์โดยการนำเวกเตอร์มาเรียงต่อกันต่อกันในแนวคอลัมน์ โดยใช้ฟังก์ชัน **cbind()**

เป็นเมทริกซ์โดยการนำเวกเตอร์มาเรียงต่อกันต่อกันในทางแถว โดยใช้ฟังก์ชัน **rbind()**

การอ่านแฟ้มข้อมูล

library (foreign) เรียก library ชื่อ foreign เพื่อเตรียมอ่านแฟ้มข้อมูลที่สร้างขึ้นด้วยโปรแกรมอื่น เช่น SPSS STATA

อ่านแฟ้มข้อมูลจากไฟล์ SPSS (*.sav) เข้าใน **R** โดยใช้ฟังก์ชัน **read.spss()**

อ่านแฟ้มข้อมูลจาก text file (*.txt) เข้าใน **R** โดยใช้ฟังก์ชัน **read.table()**

อ่านแฟ้มข้อมูลจาก Excel (*.csv) เข้าใน **R** โดยใช้ฟังก์ชัน **read.csv()**

อ่านแฟ้มข้อมูลจาก STATA (*.dta) เข้าใน **R** โดยใช้ฟังก์ชัน **read.dta()**

อ่านแฟ้มข้อมูลจาก Minitab (*.mtp) เข้าใน **R** โดยใช้ฟังก์ชัน **read.mtp()**

การบันทึกผลการวิเคราะห์

บันทึกคำสั่ง และวัตถุที่เป็นผลจากการประมวลผล

บันทึกประวัติการใช้คำสั่งที่เคยทำมาแล้ว

บันทึกผลการวิเคราะห์ข้อมูลเพื่อนำไปประกอบรายงาน

บันทึกวัตถุที่ต้องการเก็บเป็น text file โดยใช้ฟังก์ชัน **write.table()**

การปรับเปลี่ยนข้อมูล

การจัดการกับค่าสูญหาย

การคำนวณทางคณิตศาสตร์

การปัดจุดทศนิยมโดยใช้ฟังก์ชัน **round()**

การเปลี่ยนค่าของตัวแปร โดยใช้ฟังก์ชัน

replace()

ifelse()

การจัดกลุ่มตัวแปรโดยใช้ฟังก์ชัน

cut()

อธิบายความหมายของรหัสโดยใช้ฟังก์ชัน **levels()**

การเลือกข้อมูลและสุ่มตัวอย่าง

การเลือกข้อมูลบางส่วนโดยใช้ฟังก์ชัน **subset()**

สุ่มตัวอย่างสุ่ม โดยใช้ฟังก์ชัน **sample()**

การสรุปข้อมูล

ด้วยค่าสถิติพื้นฐานโดยใช้ฟังก์ชัน

summary(…)

mean()

median()

max()

min()

quantile()

var()

sd()

tapply ( )

by()

table()

การสร้างกราฟ

การกำหนดสภาพแวดล้อมในการสร้างกราฟด้วยคำสั่ง **par()**

การกำหนดสัญญลักษณ์ในกราฟ

การกำหนดสี

การกำหนดแกน

การกำหนดชื่อ

High level plot

สร้าง high level plot ด้วยคำสั่ง **plot()**

Low level plot

การสร้างกราฟชนิดต่างๆ

สร้าง Scatter plot ด้วยคำสั่ง **plot()**

สร้าง Box plot ด้วยคำสั่ง **boxplot()**

สร้าง Histogram ด้วยคำสั่ง **hist()**

สร้าง Stem and leaf plot ด้วยคำสั่ง **stem()**

สร้าง Bar plot ด้วยคำสั่ง **barplot()**

สร้าง Dot chart ด้วยคำสั่ง **dotchart()**

สร้าง Pie chart ด้วยคำสั่ง **pie()**

การแจกแจงทวินาม

การคำนวณความน่าจะเป็นและความน่าจะเป็นสะสม

dbinom()

pbinom()

กราฟของการแจกแจงความน่าจะเป็นและความน่าจะเป็นสะสม

สร้างเวกเตอร์เลขสุ่ม

rbinom()

การแจกแจงปัวส์ซง

การคำนวณความน่าจะเป็นและความน่าจะเป็นสะสม

dpois()

ppois()

กราฟของการแจกแจงความน่าจะเป็นและความน่าจะเป็นสะสม

สร้างเวกเตอร์เลขสุ่ม

rpois()

การแจกแจงปกติ

การคำนวณความน่าจะเป็นและความน่าจะเป็นสะสม

dnorm()

pnorm()

กราฟของการแจกแจงปกติ

การสร้างเลขสุ่ม

rnom()

การแจกแจงแบบ student-t

การหาค่าตัวแปรสุ่มและความน่าจะเป็น

pt()

qt()

การแจกแจงของค่าเฉลี่ยตัวอย่าง

ที่สุ่มมาจากประชากรที่มีการแจกแจงปกติและทราบค่า σ

ที่สุ่มมาจากประชากรที่มีการแจกแจงปกติแต่ไม่ทราบค่า σ

ทดสอบสมมติฐานและการประมาณค่าแบบช่วงค่าเฉลี่ยของประชากร

หนึ่งกลุ่ม

z.test()

t.test()

onemean.ztest()

onemean.test()

สองกลุ่ม

z.test()

t.test()

twomean.ztest()

ทดสอบสมมติฐานและการประมาณค่าแบบช่วงค่าความแปรปรวนของประชากรสองกลุ่ม

var.test()

ทดสอบสมมติฐานและประมาณค่าแบบช่วงกับค่าสัดส่วนของประชากร

หนึ่งกลุ่ม

prop.test()

oneprop.ztest()

binom.test()

สองกลุ่ม

prop.test()

twoprop.ztest()

ทดสอบ Goodness of fit สำหรับตารางทางเดียว

ทดสอบความเป็นอิสระระหว่างตัวแปรสำหรับตารางสองทาง

chisq.test()

fisher.test()

oneway.tab()

twoway.tab()

วิเคราะห์ความแปรปรวนทางเดียว(One way ANOVA)

ทดสอบการเท่ากันของความแปรปรวนโดยใช้ฟังก์ชัน **bartlett.test()**

วิเคราะห์ความแปรปรวน

aov()

oneway.test()

summary(aov())

summary(oneway.test())

Multiple Comparisons โดยใช้ ฟังก์ชัน **TukeyHSD()**

แผนภาพการกระจาย

plot()

title()

สัมประสิทธิ์สหสัมพันธ์

cor()

ทดสอบสมมติฐานเกี่ยวกับสัมประสิทธิ์สหสัมพันธ์

cor.test()

ประมาณค่าสัมประสิทธิ์การถดถอยและการทดสอบสมมติฐาน

lm()

summary(lm())

การตรวจสอบข้อสมมติพื้นฐานของสมการถดถอย (Model checking)

hist()

qqnorm(); qqline()

ค่าทำนาย (predicted values)

predict()