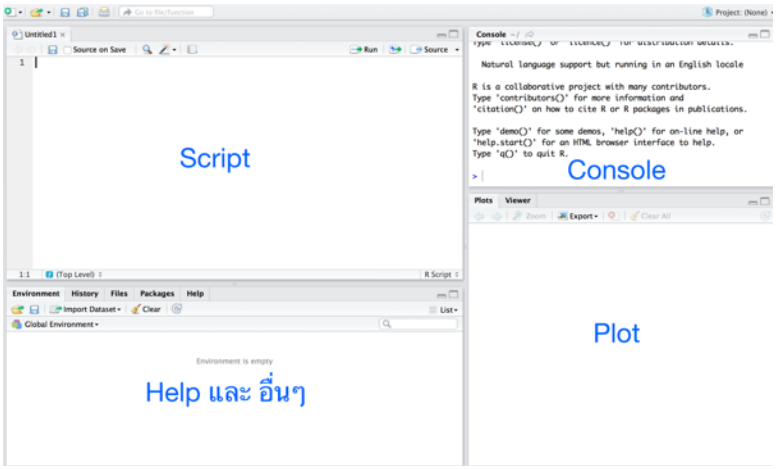
อ.ดร.เอกพันธ์ ไกรจักร์

เรียนรู้พื้นฐานการใช้ R

ก่อนที่จะใช้โปรแกรม R ในการวิเคราะห์ข้อมูลด้วยสถิติต่างๆ เราจะต้องมาเรียนรู้เกี่ยวกับพื้นฐานเกี่ยวกับภาษาและไวยากรณ์ (grammar) เบื้องต้นของ R เสียก่อน โดยให้เราปฏิบัติตามขั้นตอนในแบบฝึกหัดนี้ และเมื่อคำถามให้ปรึกษาเพื่อนและผู้สอน

*ก่อนเริ่มใช้งาน

- ให้เปิดโปรแกรม R Studio
- เลือก Session > Set Working Directory > Choose Directory ...
- ให้เลือกโฟลเดอร์ R_workshop ที่เราได้สร้างไว้บน Desktop
- ให้กดปุ่มหน้าตาแบบนี้ ที่มุมซ้ายบนของโปรแกรม
- แล้วคุณจะเห็นหน้าจอแบบนี้ แต่ละหน้าจะมีชื่อเรียกดังนี้



- ในแบบฝึกหัดนี้ให้พิมพ์คำสั่งลงในส่วนที่เป็น Script ลงไปเรื่อยๆ โดยคำสั่งที่จะให้พิมพ์นั้น จะปรากฏในรูปแบบนี้

```
xx <- seq(1, 100)
```

- เมื่อคุณพร้อมให้คำสั่งทำงาน (run) ให้เอา cursor ค้างไว้ที่บรรทัดนั้นของคำสั่งแล้วกด Ctrl และ R พร้อมกัน คำสั่งจะไปปรากฏที่หน้า Console ถ้าไม่มีปัญหาผิดพลาด จะไม่มีตัวสีแดงเกิดขึ้น และแปลว่าโปรแกรมได้ประมวลผลแล้ว
- คำสั่งที่จะลองให้คุณเขียนเอง จะกำกับด้วยคำว่า **ลองทำ** ให้พิมพ์คำสั่งลงไปด้วยตัวเอง จะใช้ในการส่งการบ้าน

*ลองเริ่มใช้งาน

1. เราสามารถ R เป็นเครื่องคิดเลขทั้งแบบธรรมดา และแบบยากได้ ให้พิมพ์

```
3+5
```

แล้ว run คำสั่งโดยกด Ctrl และ R คุณจะเห็นคำตอบใน Console หรือเราจะให้มันคำนวณอะไรที่ยากกว่านั้นก็ได้เช่น ให้คำนวน 3×4^2 หรือ $\log_{10}24$

```
3*4^2
log10(24)
```

เวลาที่เราพิมพ์ (หรือ “ โปรแกรม Rstudio จะเติม) และ “ ให้อัตโนมัติเอง

2. เราสามารถให้ R จำข้อมูลให้เราได้โดยการ “assign” ชื่อและค่าให้กับมันตัวอย่างเช่น เราอยากให้ x มีค่าเท่ากับ 10

```
x <- 10
```

เครื่องหมาย “<-“ คือการ assign ค่าให้กับตัวแปร ตอนนี้ตัว x ของคุณมีค่าเป็น 10 ทดสอบได้โดยการพิมพ์

```
x
```

โปรแกรมก็จะบอกว่า x มีค่าเป็น 10 เมื่อเป็นเช่นนี้แล้วคุณก็สามารถใช้ x ไปทำอย่างอื่นได้มากมายเช่น x + 22

```
x^2
x*x
```

คุณสามารถตั้งชื่ออะไรก็ได้ให้ที่ใช้ตัวอักษรอังกฤษ ขอให้ไม่ขึ้นด้วยตัวเลข ไม่มีช่องว่าง เช่น

```
my.height <- 165
My_STAT <- 4.00
```

ลองทำ ให้เขียนคำสั่งที่ assign รหัสนิสิตของคุณ โดยใช้ชื่อ my.id

*รู้จัก vector

3. การ assign ข้อมูลมากกว่าหนึ่งค่าก็สามารถทำได้โดยการใช้คำสั่ง c( ) และแยกแต่ละค่ากันด้วยเครื่องหมาย , อย่างเช่น เราจดบันทึกน้ำหนักของเพื่อน 6 คนในหน่วยกิโลกรัมดังนี้

```
weight <- c(60, 72, 57, 90, 95, 72)
weight
```

การกำหนดค่าข้อมูลที่มีค่ามากกว่าหนึ่งค่านี้ เราเรียกสิ่งนี้ว่า “vector” ซึ่งเราสามารถนำ vector ไปใช้คำนวณต่อได้ อย่างเช่น เราต้องการเปลี่ยนหน่วยจาก กิโลกรัม เป็น ปอนด์ ก็ทำได้โดยไปคูณ 2.2 กับทุกค่าแบบนี้

```
weight * 2.2
```

หรือเราสามารถเอาไปคำนวณกับ vector ตัวอื่นๆก็ได้ เช่น การหาค่าดัชนีมวลกาย (Body Mass Index: BMI) คือการเอาน้ำหนักเป็นกิโลกรัมหารด้วยส่วนสูงเป็นเมตรยกกำลังสอง

```
height <- c(1.75, 1.80, 1.65, 1.90, 1.75, 1.91)
bmi <- weight/height^2
bmi
```

4. เมื่อเรามีเวคเตอร์แล้ว เราสามารถหาค่าต่างๆ อย่างเช่น ค่าเฉลี่ย ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation: SD) ได้โดยการใช้คำสั่งที่มีอยู่แล้วใน R โดยไม่ต้องเขียนใหม่

- ค่าเฉลี่ย (x-bar)

```
mean(weight)
```

- ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation: SD)

```
sd(weight)
```

- ผลรวมของทุกค่าบวกกัน

```
sum(weight)
```

ลักษณะคำสั่งแบบนี้ที่มีคำภาษาอังกฤษตามด้วยวงเล็บ เราเรียกว่า ***function*** สิ่งที่เราใส่เข้าไปในวงเล็บนั้น เราเรียกว่า ***argument*** อย่างในกรณีของ sum(weight) คำว่า sum() เป็น function ในการหาค่าผลรวม ซึ่งเราก็ต้องบอกว่าให้หาผลรวมของอะไร เราก็เลยใส่ไปว่า weight ซึ่งเป็น argument ของ function นี้

ลองทำ ให้เขียนคำสั่งที่หาค่าเฉลี่ยและค่า SD ของคะแนนสอบของนักเรียนห้าคนที่ได้คะแนน 10, 12, 15, 17, 18

*หัดวาดกราฟ

5. เราสามารถนำข้อมูลที่เราได้อ่านบันทึกใน R แล้วมาวาดกราฟได้ ตัวอย่างเช่นเราจะวาดกราฟของความสูงอยู่ในแกนนอนและน้ำหนักอยู่บนแกนตั้ง

```
plot(height, weight)
```

เราสามารถควบคุมลักษณะเกือบทุกอย่างของกราฟนี้ได้ อย่างเช่น ขนาดของจุดข้อมูล ใช้ argument “cex”

```
plot(height, weight, cex = 2)
```

หรือสามารถเปลี่ยนสีของจุดได้โดยใช้ argument “col” ด้วยชื่อสีภาษาอังกฤษ หรือ รหัสสีแบบ HTML (http://html-color-codes.info)

```
plot(height, weight, col = “red”)
plot(height, weight, col = “#FFBF00”)
```

หรือจะเปลี่ยนลักษณะของจุดได้ด้วย argument “pch” ซึ่งมีทั้งหมด 25 แบบ แบบที่ 15 นี้เป็นสี่เหลี่ยม

```
plot(height, weight, pch = 15)
```

เราสามารถใช้ argument หลายๆอันพร้อมกันได้ โดยใส่เพิ่มไปในวงเล็บแบบนี้

```
plot(height, weight, cex = 2, col = “red”, pch = 15)
```

ตัวเลือกของ argument ใน function “plot” นั้นมีเยอะมากมายถ้า คุณอยากรู้ทั้งหมดพิมพ์

```
?plot
```

เป็นการเรียกหน้าจอ Help ซึ่งจะปรากฏอยู่ด้านล่างของหน้าต่าง Script มันเป็นภาษาอังกฤษที่อ่านยากสักเล็กน้อย แต่ให้เลื่อนลงอ่านคร่าวๆ หรือ อ่าน Example ซึ่งจะมีตัวอย่างคำสั่งด้านล่างสุด

ลองทำ ลองวาดกราฟนี้ดูอีกทีแล้ว ให้กำหนด ขนาด (cex), สี (col), และรูปแบบจุด (pch) ที่คุณชอบ โดยลองเปลี่ยนตัวเลขหรือชื่อสีไปเรื่อยๆจนกว่าจะพอใจ เขียนคำสั่งที่จะใช้ไว้ในหน้า Script

*โหลดข้อมูลของตัวเอง

6. เราสามารถโหลดข้อมูลที่บันทึกไว้ใน Excel ได้แต่ไฟล์นั้นต้องบันทึกเป็น .csv โดยใช้คำสั่งดังนี้

```
sample <- read.csv(“sample_data.csv”)
sample
```

จะเห็นว่าไฟล์เราอาจจะใหญ่เกินที่ดูหมดบนหน้าต่าง console ได้เราสามารถเรียกดูได้แค่บางส่วนของไฟล์เช่น ด้านบนสุดหรือด้านล่างสุด

```
head(sample)
tail(sample)
```

7. เราสามารถวาดกราฟจากข้อมูลนี้ได้ ตัวอย่างเช่น จะวาดความสัมพันธ์ระหว่าง treatment กับ growthrate

```
plot(growthrate ~ treatment, data = sample)
```

วิธีอ่านคำสั่งนี้เป็นภาษาคนคือ ให้ วาด growthrate เป็นตัวแปรตามของ treatment โดยใช้ data ชื่อ sample

จะเห็นว่าคราวนี้ออกมาเป็นกราฟแท่งแบบ boxplot เพราะว่าแกนนอนของเราเป็น discrete variable ถ้าลองวาดตัวแปรที่เป็น continuous ทั้งคู่ ก็จะเป็นแบบที่เราได้เคยวาดไปแล้ว และก็แน่นอนว่า เราสามารถเปลี่ยนสีของกราฟนี้ ตามใจชอบได้

```
plot(growthrate ~ treatment, data = sample, col = "green")
```

และสามารถเปลี่ยน label ของแต่ละแกนได้ โดยคำที่ใช้นั้นจะต้องอยู่ใน " "

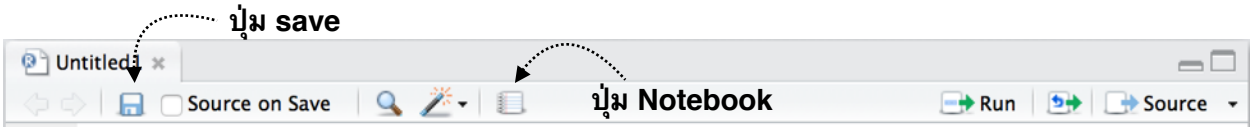
```
plot(growthrate ~ treatment, data = sample, xlab = "Experiment", ylab = "Growth Rate (cm)")
```

ลองทำ ให้วาดกราฟความสัมพันธ์ระหว่าง growthrate กับ species ในข้อมูล sample นี้ ให้เลือกสีเองตามใจชอบและให้กำหนด label แกนนอนว่า "Plant Species" และ การตั้งว่า "Growth Rate" เขียนคำสั่งที่จะใช้ไว้ในหน้า Script

*[ถ้าใครทำทั้งหมดนี้เสร็จก่อนเวลา อยากให้ลองท้าทายตัวเองด้วยการเขียนคำสั่งที่ยากขึ้นมาอีกนิด ก่อนจะที่จะบันทึกข้อมูลและการทำงานทั้งหมดในส่วนถัดไป ให้ข้ามไปส่วนสุดท้ายที่ใต้หัวข้อ **คำสั่งท้าทาย** แล้วค่อยกลับมาบันทึกข้อมูล]*

*บันทึกข้อมูลและผลการทำงาน

6. เมื่อพร้อมให้บันทึกคำสั่งที่เราได้ทำไปทั้งหมดในหน้า จอ script โดยการกดรูป floppy disk ด้านบน ใช้บันทึกในโฟลเดอร์ R_workshop โดยใช้ชื่อของตัวเองตามด้วย _tutorial1 ตัวอย่างเช่น "Ekaphan_tutorial1"



7. ข้อดีของ RStudio คือ เราสามารถบันทึกทุกอย่างที่ทำ (คำตอบการคำนวณและกราฟ) ได้โดยการกดปุ่ม notebook โดยให้เลือก notebook format เป็น HTML แล้วกด "Compile" ไฟล์ notebook ที่เป็น HTML จะไปปรากฏในโฟลเดอร์ของคุณเอง

การบ้านครั้งที่ 2: ส่งไฟล์ HTML นี้ไปที่ fsciepk@ku.ac.th ก่อนเวลา 20.00 น. ของวันที่ 21 เมษายน 2558

*คำสั่งท้าทาย

เลือกทำกี่ข้อก็ได้ ตามแต่เวลาที่คุณมีและความอยาก

1. แม้ว่าเราจะหาค่าเฉลี่ยได้ด้วย function สำเร็จรูปใน R เราก็สามารถเขียนคำสั่งหาค่าเฉลี่ย ได้เองโดยการเอาfunctionง่ายๆหลายๆอันมาประกอบกัน ให้ลองเขียนชุดคำสั่งเพื่อหาค่าเฉลี่ยโดยใช้เพียงแค่ function sum( ) และ length( ) เท่านั้น [*length( ) บอกจำนวนข้อมูลที่มีอยู่ใน vector*] ตรวจสอบความถูกต้องด้วย function mean( )
2. **[ยาก ท้าให้ลอง]** เขียนชุดคำสั่งที่หาค่า SD โดยfunctionพื้นฐาน sqrt( ) [*สำหรับถอดรูท*], mean( ), length ( ) และ sum ( ) เท่านั้น โดยสูตรจริงของ SD เป็นแบบนี้ $$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2}.$$
3. เขียนคำสั่งวาดกราฟ ความสัมพันธ์ระหว่าง weight และ height โดยให้มีลักษณะดังต่อไปนี้
   - จุดสี่เหลี่ยมทึบ สีเขียว, label แกน x เขียนว่า “Weight (kg)”, lable แกน y เขียนว่า “Height (m)”, ขนาดจุด 2.5
4. เขียนคำสั่งวาดกราฟ แสดงความสัมพันธ์ระหว่าง log ฐาน 10 ของ BMI กับ height เลือกสีและขนาดตามใจชอบ พร้อมเปลี่ยน label ของแกนให้เหมาะสม
5. ในข้อมูลของเราเองนั้น (sample) เราสามารถเลือกอ่านข้อมูลแต่ละคอลัมน์แยกกันได้ อย่างเช่น ถ้าอยากดูแค่ growthrate อย่างเดียว

   ```
   sample$growthrate
   ```

   สัญลักษณ์ “$” เป็นการเลือกชื่อคอลัมน์ที่มีอยู่ในข้อมูลของเรา เมื่อทราบอย่างนี้แล้ว ให้ลองหา ค่าเฉลี่ยและค่า SD ของ growthrate ในข้อมูลชุดนี้
6. เขียนคำสั่งโหลดข้อมูลจากอีกไฟล์ที่เรามี ที่ชื่อว่า “iris.csv”
7. แต่ละคอลัมน์ใน iris.csv คือ ความยาวและความกว้างของกลีบดอก (petal) และของกลีบเลี้ยง (sepal) และชื่อชนิดของพืชสกุล *Iris* ให้เขียนคำสั่งหาค่า SD ของ Sepal.Length, Sepal.Width, Petal.Length, Petal.Width
8. ให้เขียนคำสั่ง วาดกราฟแสดงความสัมพันธ์ระหว่าง Petal.Length และ Petal.Width ให้เลือกสีและขนาดตามใจชอบ และกำหนด label ของแกนให้เหมาะสม