【附件三】教育部教學實踐研究計畫成果報告格式(系統端上傳 PDF 檔)

教育部教學實踐研究計畫成果報告(封面)

Project Report for MOE Teaching Practice Research Program (Cover Page)

計畫編號/Project Number：PSL107021
學門分類/Division：社會(含法政)

執行期間/Funding Period：2018/08/01 ~ 2019/07/31

(計畫名稱/Title of the Project)：平衡內容和人際鷹架促進學生使用電腦內容分析方法解決研究問題的教學實踐/ Balancing Content and Interpersonal Scaffolding in Problem-based Learning of Computerized Content Analysis in a Master Course
(配合課程名稱/Course Name)：數位內容與社群網絡分析

計畫主持人(Principal Investigator)：譚躍

共同主持人(Co-Principal Investigator)：無

執行機構及系所(Institution/Department/Program)：國立中山大學行銷傳播管理研究所

繳交報告日期(Report Submission Date)：2019/8/15

平衡內容和人際鷹架促進學生使用電腦內容分析方法解決研究問題的教學實踐

一. 報告內文(Content)

1. 研究動機與目的(Research Motive and Purpose)

請描述所選擇研究議題的問題挑戰與背景、教學實務現場遇到之挑戰以及該議題的重要性與影響力。

Communication practitioners seek graduates who can create stories and derive valuable findings from quantitative data analyses. Text mining skills are becoming increasingly desirable and a growing rate of journalism and mass communication curriculums are introducing coding skills. However, instructors developing coding instructions for communications students are faced with a unique set of issues such as the lack of scholarship on how to teach coding (Treadwell, Ross, Lee & Lowenstein, 2016).

Students of mass communication are often indifferent and sometimes hostile to learning coding skills. The general perception is that learning a computer language warrants higher-order cognitive skills and thus, programing is considered one of the most difficult subjects even for students in technology fields (Fang, 2012; Korkmaz & Altun, 2014). Scholars of communication education, on the other hand, are exploring ways to incorporate courses on this challenging skill into an already full curriculum (Strugill, Hannam & Walsh, 2017).

2. 文獻探討(Literature Review)

請針對本教學實踐研究計畫主題進行國內外相關文獻、研究情況與發展或實作案例等之評析。

Programming can be defined as a problem-solving process that uses computer science concepts such as abstraction and decomposition. Lye and Koh (2014) propose a problem-solving learning environment with information processing, scaffolding, and reflection activities.

Scaffolding is a commonly used term in pedagogy in general and language teaching in particular (Wilson & Devereux, 2014). Scaffolding theory originates from Vygotsky’s (1978) theories of social learning, which argue that learning takes place in social environments through interactions with peers and experts. The theory has also been widely applied to computerized learning experiences (Pea, 2005). Wood, Bruner, and Ross (1976) theorize the following scaffolding functions that can be adopted when tutoring problem solving: stimulation of students’ interests, reduction in degrees of freedom (i.e., simplifying tasks by reducing the number of steps to reach a solution), direction maintenance, marking of critical features, frustration control, and demonstration. Mariani (1997) emphasizes the importance of both “high challenge” and “high support” in scaffolding student learning: that is, learners must be challenged to avoid boredom and high support can mitigate feelings of frustration.

In line with Collins's (1987) cognitive apprenticeship theory, scaffolding generally involves three stages: teachers and students work collaboratively to solve a problem, students work together to solve a similar problem, and students solve a problem without assistance (Collins, 1987; Wilson & Dvereux, 2014). The course examined in this study adopts this three-stage process for each class as follows.

First, the instructor provides the students with background knowledge on the academic field (i.e., basic concepts, theories, and application). The instructor shows them how to analyze the English text data using English codes directly copied from an online textbook. The students are then presented a published research paper using the same analytical methods to help them understand the operationalization of concepts and the application of the research method.

In the second stage, the students must apply these skills to resolve new problems. The instructor invites the students to discuss the modification of original codes and replicate the analyses in the research papers using Chinese text data. Next, the students must elaborate on their ideas, interpret their designs, and reflect on the problem-solving process. The instructor and students together make notes of the discussion and use them to create new codes for the data analysis. During the discussion, the instructor offers contingent scaffolding by citing explicit links to students' prior knowledge and by indicating new directions. The instructor also encourages reflection by asking questions that deepen students' understanding through cued elicitation and increased prospectiveness (Hammond & Gibbons, 2005).

Finally, the instructor asks each student to analyze the Chinese text. This in-class exercise includes immediate interactions with the instructor and a teaching assistant. Moreover, students' problem-solving process and assessments are supported by scaffolding through "rich dialogic feedback."

The one-semester graduate course examined in this study also embodies Mariana's (1997) "high challenge and high support" pedagogy by offering several types of scaffolding support to simplify the coding process and problem-based learning. Following Sturgill et al. (2017), the scaffolding approach used in the course is categorized into content help (i.e., codes in textbooks, research papers, and windows-type software) and in-person help (e.g., tutorials during in-class exercises and group projects). In addition to codes in the textbooks and published research papers, the instructor introduces students to windows-type software to perform similar analyses without coding. By demonstrating its usage, the students learn not only an alternative approach to data but also to compare their coding-related strengths and weaknesses. Apart from their instructor and the teaching assistant, the students are encouraged to seek help from their peers.

Sturgill et al. (2017), however, find that a majority of support types, including in-person help (e.g., office and lab hours) and content help (e.g., textbook, videos, and blogs), are not related to student success measured in course grades. Video support, in fact, is negatively related with student success. The course examined in this study offers codes in textbooks, research papers, and windows-type software as content-based help and tutorials during in-class exercises and group projects as in-person help. We attempt to answer two research questions through this research.

RQ1: Are students satisfied with these support resources and why?

The perception of self-efficacy and attitude are the most important factors influencing the success of a learning process (Anastasiadou & Karakos, 2011; Korkmaz & Altun, 2014). Through his social cognitive theory, Bandura (1997) demonstrates that self-efficacy is a key determinant of learning motivations and learning performance. Bandura (1986) defines self-efficacy as “people’s judgements of their capabilities to organize and execute courses of action required to attain designated types of performances” (p.391); this can be exemplified as students’ belief in their ability to complete a learning goal.

Over the years, studies across research fields have examined students’ self-efficacy and indicated that domain-specific measures of self-efficacy offer more accurate performance predictions than general measures (Aesaert & van Braak, 2014; Saleem, Beaudury & Corteau, 2011). This study employs the concept and measures of computer self-efficacy (Compeau & Higgins, 1995) because they are preferred over those of general efficacy. Computer self-efficacy is defined as individuals’ judgment of their ability to apply computer skills to broader tasks in the future.

RQ2: How and why do support resources contribute to students’ efficacy and learning performance?

3. 研究方法(Research Methodology)

可包含實驗場域、研究對象、研究架構、資料蒐集方法與工具與分析方法等項目，但不限於列舉內容。

This study examines an interdisciplinary graduate course offered under a communications program held at a public university in Taiwan during fall 2018. The course offers three credit hours of computerized content analyses. Course students meet once a week for three hours throughout the 18-week semester. There is no perquisite to enroll in the course. A total of 11 students signed up and of these, six are communication majors, three are information management majors, and two are social sciences majors. However, three students who provisionally enrolled in the course dropped out in the first three weeks. The classroom set up is similar to that of an amphitheater with computers and related software.

The objective of the course is to help graduate students in the fields of communication, humanities, and social sciences master the basic skills of text mining. More specifically, the course aims to aid students in developing the ability to operationalize important theoretical concepts as well as gain exposure to existing theories, help discover phenomena, and explore new research questions.

From a technical perspective, the course involves cultivating three types of problem-solving abilities: R language coding, communication theories, and statistical analysis. R programing language is a major component of the course content because it is a fundamental skill needed for text mining.

The students enrolled in the course are subject to three assessments. First is a class presentation

(20% of the total grade) of a published research paper selected by the instructor. Students are required to discuss the research questions, literature review, methods, findings, and limitations of the paper. The in-class exercises and discussion participation account for 20% and 30% of the total grade.

Second is a proposal for a research paper (30%) that outlines research questions, identifies text data, and analyzes and visually reports the findings. In addition to the in-class presentation, students must write a 4,000-word report, including tables and references. It was observed that the students' data were largely derived from mass media content and Facebook posts.

Major concepts and data are measured and collected through panel questionnaire surveys, participant observations, and in-depth interviews. The questionnaire surveys include a self-efficacy scale that is applied five times (weeks 4, 7, 10, 14, and 18) and a leaning satisfaction scale used twice (weeks 10 and 18). Students' satisfaction with each type of support resource are evaluated for effectiveness, likeness, and easiness (Cronbach's $\alpha = 0.91$, see Table 1). The self-efficacy scale (Cronbach's $\alpha = 0.71$) is based on those used in previous research (Korkmaz & Altun, 2014; Yukselturk & Altiok, 2017) and modified according to the objective of this study (see Table 2). Both scales are evaluated on a five-point Likert scale.

Owing to the limited number of questionnaires, in-depth interviews were conducted at the end of the semester to acquire additional information on how various support resources contribute toward students' efficacy, satisfaction, and learning performance and why. The following four interview questions are based on students' learning experiences. (1) What was your main objective of enrolling in this course and have you achieved it? (2) What did you find most difficult in the course? (3) Which support resources are the most helpful? (4) Do you have any suggestion to improve the course?

The duration of each interview is about 20–30 minutes. Prior to each interview, the students are informed of its purpose by the instructor. The interviews are audio recorded with the participants' consent. The records are then transcribed and analyzed by the researcher. The data are grouped into main categories and interpreted as supplementary to the questionnaire data.

4. 教學暨研究成果(Teaching and Research Outcomes)

(1) 教學過程、成果與學生學習回饋

Table 1 presents the data obtained from the questionnaire surveys for student satisfaction with various support resources. Given the small sample size, this research reports the descriptive statistics and does not perform inferential tests. In addition to the quantitative analysis, in-depth interviews are performed to explore the process of various scaffolding resources and their effectiveness in improving students' learning of text mining skills. An analysis of the results and a discussion on this study's limitations highlight suggestions for teaching practices for coding.

Table 1.

Students' satisfactory scores for various support resources (mean and SD)

Type of support	Usefulness	Likeness	Easy-ness	Total

resources							
Week	W10	W 18	W10	W18	W10	W 18	
Original codes in the online textbook	4.1(.32)	4.1 (.6)	4.2(.63)	3.9(.6)	3.8(.79)	3.2(.67)	3.92(.32)
Modified codes for Chinese text-mining	4.3(.67)	4(.71)	4.4(.7)	3.9(.6)	3.8(.79)	3.4(.73)	3.96(.47)
Windows-type software	4.2(.79)	4(.5)	4.3(.67)	4(.5)	4(.67)	3.6(.53)	3.94(.42)
Sampled research papers	4.3(.67)	4.1(.93)	4.3(.67)	4.1(.78)	4.3(.67)	4.1(.6)	4.1(.56)
Tutorials during the in-class exercise	4.4(.52)	4.2(.67)	4.5(.53)	4.1(.6)	4.5(.53)	3.9(.6)	4.33(.43)
Team work with classmates	4.4(.7)	4.2(.83)	4.4(.7)	4.3(.71)	4.3(.67)	4(.71)	4.19(.65)
Total	4.3(.49)	4.1(.41)	4.4(.56)	4.1(.42)	4.1(.55)	3.7(.3)	4.07(.40)

N = 10. Cronbach's $\alpha = 0.91$. All items are rated on a five-point Likert scale, where 1 = "strongly disagree" and 5 = "strongly agree."

Immediate face-to-face interactions are necessary

This study assesses students' efficacy and satisfaction with six support resources: original codes in online textbooks, modified codes for Chinese text mining, windows-type software, sampled research papers, tutorials during in-class exercises, and classroom collaboration. As shown in Table 2, students are the most confident about their ability to complete text mining exercises when they receive help from their instructor and the teaching assistant (mean = 4.29, SD = .62). Among the six types of learning supports, the students prefer tutorials during the in-class exercises. In particular, they believe the tutorials are the most useful (mean = 4.38, SD = .44), likable (mean = 4.31, SD = .70), and easy to use (mean = 4.31, SD = .70).

Table 2.

Students' self-efficacy scores for text mining skills (mean and SD).

	week4	w7	w10	w14	w18	total
I can comprehend the text-mining codes in the textbook.	3.54 (1.05)	4.2 (.63)	4.3 (.67)	4.1 (.57)	3.8 (.44)	3.95 (.38)
I can understand the modified codes and apply them into Chinese text-mining.	3.44 (.88)	4.4 (.52)	4.1 (.32)	4.1 (.57)	3.7 (.71)	3.77 (.27)
I can finish the text-mining exercise with the help of the instructor and the teaching assistant.	4 (1)	4.5 (.71)	4.2 (.92)	4.5 (.71)	4.3 (.87)	4.29 (.62)

I can understand the research methods in the sampled research.	4.11 (.78)	4.3 (.67)	4.1 (.57)	4.2 (.42)	3.9 (.33)	4.0 (.20)
I can apply what I learned in this class into my future research.	3.67 (1.12)	4.1 (.88)	4.2 (.92)	4.1 (.74)	3.7 (.5)	3.77 (.56)
Total	3.78 (.66)	4.3 (.59)	4.2 (.51)	4.2 (.48)	3.87 (.41)	3.94 (.29)

N = 10. Cronbach's $\alpha = 0.71$. All items are rated on a five-point Likert scale, where 1 = "strongly disagree" and 5 = "strongly agree."

Thus, immediate face-to-face interactions help students resolve specific questions. Observations reveal that when the students faced minor errors when coding without immediate help, they would lose patience, stop working, and eventually give up. Most of the students recommend that the instructor incorporate more in-class exercises for future instructions.

Students have differing prior knowledge given their varying majors. It is necessary to resolve these knowledge gaps through extensive discussions with the students. An information management major notes, "In the beginning, I could not understand how to relate the terms and concepts from communication and marketing (e.g., media frames) to text mining skills. They are too abstract. The instructor's explanation clarified this connection."

Importance of teamwork in a final project

The students appreciate working collaboratively on a research project that mirrors the published journal papers. They believe this learning method is likeable (mean = 4.31, SD = .70), useful (mean = 4.19, SD = .70), and easy to use (mean = 4.06, SD = .62). A student mentions, "My team members helped me solve many problems." Another student adds, "When we wrote the final project, I had the opportunity to understand what I really understood." Some participants highlight that working in a team allows them to develop interpersonal and organizational communication skills.

Permanent scaffolding for programing codes

While both types of programming codes are rated as less important than interpersonal support, they are the most preferred by the students. The low satisfaction scores for the programming codes may be attributed to the high difficulty levels. Ease of use (original codes: mean = 3.5, SD = .46; modified codes: mean = 3.63, SD = .52) is rated the lowest among the six methods. Students report the lowest self-efficacy in understanding and applying the modified codes to Chinese text mining (mean = 3.77, SD = .27). In addition, they are the least confident when applying text mining skills in their future research (mean = 3.77, SD = .56).

Even students with a strong coding background prefer the codes because of their simplicity and usefulness. One student notes, "We usually find codes and debug advices on Google, but those codes are messy and it is difficult to understand their purpose…the instructor knows which parts are more important and useful…her codes are clean, logical, and workable."

Programing codes are critical because even when interpersonal scaffolds are withdrawn, they remain accessible to the students at all times. A majority of the students prefer programming codes when working on after-class exercises and the final project.

Connecting skills with research samples

Students can confidently understand the research methods presented in the sample papers (mean = 4.0, SD = .20). The sample papers are easier to use compared with the other resources (mean = 4.06, SD = .42). In general, the students believe that the sampled research papers are likeable (mean = 4.13, SD = .69) and useful learning resources (mean = 4.13, SD = .74). A student comments, "The sampled paper helps me understand the purpose of text mining skills." Another student mentions, "The connections provided in the sampled research papers helps me understand how these concepts are related with the programing codes." Another student adds, "When we worked on the final project, we repeatedly referenced Chang's paper, observing how she approached the research question and analyzed the data." Further, a student admits, "We don't know how to interpret the data…We don't know how to visually present our findings…We don't even know what to expect from forming the hypotheses….The sample papers helped a lot."

Windows-type software is an added bonus

Although windows-type software is regarded easier to use (mean = 3.75, SD = .46) than R codes, they are less useful (mean = 3.94, SD = .50) and less likable (mean = 4.13, SD = .52) than the modified codes. Students agree that windows-type software (e.g., Netvizz, HTML5, Wordsmith, NodeXL, Ucinest, and Gephi) is beneficial to learning during various text mining tasks. A student mentions, "The windows-type software is ready for use. You can obtain results by simply imputing data…It would have been impossible for me to find the free software by myself." However, many students are overwhelmed by the complicated functions. The most preferred software is Netvizz because of its simplicity. A student mentions, "I didn't use Wordsmith and Gephi. Without the exercises and review, I would forget how to use them. There are too many functions." Another student adds, "I am unfamiliar with the software. There are too many function and I feel fuzzy when using them."

Declined efficacy and satisfaction in final stage

It is noteworthy that students report an increase in self-efficacy in the seventh week; however, these levels drop in the final weeks (mean = 3.87, SD = .41 in week 18 compared with mean = 4.2, SD = .48 in week 14). Between week 14 and 18, the students are asked to collaborate with teammates to collect and analyze Chinese text for their research project. Unlike the third stage of each class, no tutorial is available and thus, the students must complete their task without support resources. This task presents the students with a significant challenge that they may not be ready to tackle. A student admits, "In class, every code seemed to work fine. However, I hit a roadblock when working on my own data. I did not know how to solve it."

The decline in self-efficacy is associated with reduced satisfaction with various support resources. In the final week, the students believe the resources are useful (mean = 4.1, SD = .41) and likable (mean = 4.1, SD = .42) but re-evaluate them as difficult to use (mean = 3.7, SD = .3 in week 18, mean = 4.1, SD = .55 in week 10). A student states, "I realized that coding is more difficult than I expected when I began working on my own project."

(2) 教師教學反思

This study is an initial effort to assess practices that can be adopted to teach text mining skills to graduates students with majors ranging from communication to information management. While the communication students had limited understanding of R code, the information management students had minimum understanding of communication theories. The results obtained from quantitative and qualitative data collected during the 18-week class may not be generalizable to students from other disciplines, particularly engineering and natural sciences.

The data suggest that among the different types of learning supports, tutorials during in-class exercises are the most preferred by the students. While the faculty considers it inefficient to work one-on-one with the students, the students believe that immediate face-to-face interactions are helpful. They suggest that the faculty can identify errors in codes as well as inefficiencies, oversights, and inconsistencies in the students' exercises. This support is important because the faculty offers the student with the context necessary for their learning process. During certain tasks, the students were unable to determine a solution or the online resources that could be useful to them. Repeated failures may significantly reduce students' self-efficacy. Moreover, the final research paper is the most challenging task in this class. Thus, higher support should accompany tasks to avoid frustration (Mariani, 1997).

The final project is divided into multiple simple tasks that are assigned to the students during the course. These assignments can replace in-class exercises, during which instructors can immediately address specific questions, correct mistakes, and guide studies through face-to-face interactions. Further, given the usefulness of peer tutorials, the students can be divided into small groups during the second class of the semester on the basis of their prior knowledge about computer languages. The instructor can also reduce the degrees of freedom by specifying text data sources and communication theories. In the final weeks, the instructor can set up a course blog, on which all students can post specific questions and access corresponding responses provided by the instructors.

Programing codes are important scaffolds that can be permanently available even after interpersonal scaffolds are withdrawn. They can be further integrated with data analyses in the sample research papers. When reporting such sample studies, instructors can ask students to propose alternative methods for text data analyses. Finally, instructors should filter windows-type software. While such software may seem easier to use, it requires time to learn and practice. Given the limited class time, instructors should inform students of the most useful and easiest software. Others can be briefly introduced with simple tutorial instructions.

Future research should further clarify the relationship between student learning and available resources. As Sturgill et al. (2017) highlights, balancing student desires and feasibility for instructors is key.

二. 參考文獻(References)

Anastasiadou, S. D., & Karakos, A. S. (2011). The beliefs of electrical and computer engineering students' regarding computer programming. *The International Journal of Technology, Knowledge and Society*, *7*(1), 37-51.

Azevedo, R., & Hadwin, A. F. (2005). Scaffolding self-regulated learning and metacognition–Implications for the design of computer-based scaffolds. *Instructional Science, 33*(5), 367-379.

Bandura, A. (1986). Social foundations of thought and action. *Englewood Cliffs, NJ*, *1986*.

Collins, A., et. al. (1987). *Cognitive apprentice- ship: teaching the craft of reading*, *writing*, *and mathematics.* Washington, D.C.: U S . Department of Education, Technical Report No. 403.

Fang, X. (2012). Application of the participatory method to the computer fundamentals course. In *Affective Computing and Intelligent Interaction* (pp. 185-189). Springer, Berlin, Heidelberg.

Hammond, J., & Gibbons, P. (2005). Putting scaffolding to work: The contribution of scaffolding in articulating ESL education.

Korkmaz, Ö., & Altun, H. (2014). Adapting Computer Programming Self-Efficacy Scale and Engineering Students' Self-Efficacy Perceptions. *Online Submission*, *1*(1), 20-31.

Pea, R. D. (2004). The social and technological dimensions of scaffolding and related theoretical concepts for learning, education, and human activity. *The journal of the learning sciences*, *13*(3), 423-451.

Vygotsky, L. S. (1978). Mind in society: The development of higher mental process. Cambridge, MA: Harvard University Press.

Wilson, K., & Devereux, L. (2014). Scaffolding theory: High challenge, high support in Academic Language and Learning (ALL) contexts. *Journal of Academic Language and Learning*, *8*(3), A91-A100.

Wood, D., Bruner, J. S., & Ross, G. (1976). The role of tutoring in problem solving. *Journal of child psychology and psychiatry*, *17*(2), 89-100.

三．附件(Appendix)

（1）學生問卷

1. 自我效能

		非常不同意	不同意	部分同意	同意	非常同意
1）	我可以理解課本上文字探勘的程式碼					
2）	我可以理解如何透過修改這段程式碼，而應用該程式碼分析中文					
3）	我可以在老師和助教的幫助下					

	完成文字探勘的練習題					
4）	我可以理解研究範例中的研究方法					
5）	我可以將本課學到的內容應用到我未來的研究中					

2. 學習滿意度

		非常不同意	不同意	部分同意	同意	非常同意
1.	針對**課本中的程式碼**					
1）	我滿意它對我學習的輔助效果					
2）	我可以接受這種學習模式					
3）	我認為這種學習模式是容易的					
2.	針對**修改過的程式碼**					
4）	我滿意它對我學習的輔助效果					
5）	我可以接受這種學習模式					
6）	我認為這種學習模式是容易的					

		非常不同意	不同意	部分同意	同意	非常同意
3.	針對**課程中的 Windows 軟體**					
7）	我滿意它對我學習的輔助效果					
8）	我可以接受這種學習模式					
9）	我認為這種學習模式是容易的					
4.	針對**課程中導讀的研究範例**					
10	我滿意它對我學習的輔助效果					
11	我可以接受這種學習模式					
12	我認為這種學習模式是容易的					
5.	針對**課堂練習中老師和助教的問題解答**					
13	我滿意它對我學習的輔助效果					
14	我可以接受這種學習模式					
15	我認為這種學習模式是容易的					
6.	針對**課堂練習中小組成員間的互相幫助**					
16	我滿意它對我學習的輔助效果					
17	我可以接受這種學習模式					
18	我認為這種學習模式是容易的					

（2） 訪談問題和 2 位學生的回答範例

访问者	受访者 A	受访者 B
1. 請問你最初選課的目的是什麼？為什麼？	“最初選課的目的是因為知道這堂課有教 R 語言為主，然後這次的計劃也是以內容分析為主要的模式，所以安授老師說建議我來修。我自己本身也是對 R 語言有點興趣，所以就要學這堂課。”	“最主選課的目的就是想要了解一下內容，研究方法，然後思考一下，看畢業論文這種可不可以參考。”

2. 請問您一開始上課的時候，覺得對自己最挑戰的是哪一部分內容？ 課上的內容對此有幫助嗎？	“最挑戰的部分應該是要像剛剛老師所講的在編碼上麵的程序，然後使用上，然後因為這東西，因為這東西隻有老師上課會使用到，對，然後因為比較少時間會聯係，所以每次重新上課的時候就有點……就看那個編碼會有點不知道幹嘛，然後就一直 run 一直 run 這種感覺，然後最挑戰的部分應該是每次 run 的時候會跑出那個錯誤。” “有幫助是有幫助的。”	“一開始比較挑戰的，也是程式放在眼前，就不太知道它們是哪一個部分。” “有，通過上課一些練習，然後導入範例。”
3. 在老師所提供的教學工具中，哪些對你比較有幫助？為什麼？哪些對你不太有幫助，為什麼？你覺得可以如何改進？	“我覺得第一個就是導讀的內容，因為導讀的內容就可以幫助你這些能快速的知道 R 語言主要是研究什麼東西的領域。 比如說我導讀的那邊是框架的東西……然後我覺得軟體很重要，就是老師看網上有介紹很多那種 R 之外的很多軟體，就像臉書裡面有 Graph，那我還想用它去搜集我自己的個人資料，最後會發現它不會收集自己個人資料。”	“都是導讀中的研究範例，然後修改過的程式碼就都能讓我更好的理解這些東西在做什麼，然後老師助教的解答，還有小組成員間互相幫助，也可以幫我解決一些遇到的問題。” “比較希望在課堂上能跳出現在那些程式碼，然後再有多一點練習，比如說想要做其他的 data 那種。然後大概需要哪些部分是改掉，然後可能在以後研究中的參考價值更大一些。 然後 PPT 的話也是希望稍微全面一點，然後這樣自己再想回顧的時候比較能找到。”
4. 請問你還可以提供一些其他關於本課程的	“建議的話，第一個是練習，覺得強製性練習	“沒有了。”

感想或者建議嗎？	會比較好一點。 然後我覺得老師可以提多一些有關 R 語言的研究主題可以給我們參考。因為其實這是我們在做最後論文的時候不知道到底哪種主題……就很難去想有關 R 語言的主題。”	