STA 418 02, Joint with STA 518 02
Computing and Graphics with R
Course type: Face-to-Face

Evaluation Delivery: Online
Evaluation Form: A3
Responses: 16/24 (67% high)

Taught by: Bradford Dykes
Instructor Evaluated: Bradford Dykes-Assist Prof

Overall Summative Rating represents the combined responses of students to the four global summative items and is presented to provide an overall index of the class's quality:

Combined Median	Adjusted Combined Median
3.8	4.3

(0=lowest; 5=highest)

Challenge and Engagement Index (CEI) combines student responses to several *IASystem* items relating to how academically challenging students found the course to be and how engaged they were:

CEI: 5.7
(1=lowest; 7=highest)

SUMMATIVE ITEMS

	N	Excellent (5)	Very Good (4)	Good (3)	Fair (2)	Poor (1)	Very Poor (0)	Median	Adjusted Median
The course as a whole was:	16	12%	56%	19%	6%	6%		3.8	4.3
The course content was:	16	31%	44%	19%	6%			4.1	4.4
The instructor's contribution to the course was:	16	25%	31%	19%	12%	12%		3.7	4.1
The instructor's effectiveness in teaching the subject matter was:	16	19%	38%	19%	12%	6%	6%	3.7	4.2

STUDENT ENGAGEMENT

Relative to other college courses you have taken:	N	Much Higher (7)	(6)	(5)	Average (4)	(3)	(2)	Much Lower (1)	Median
Do you expect your grade in this course to be:	16	12%	12%	6%	50%	19%			4.1
The intellectual challenge presented was:	16	38%	25%	31%	6%				6.0
The amount of effort you put into this course was:	16	56%	25%	12%	6%				6.6
The amount of effort to succeed in this course was:	16	38%	38%	12%	12%				6.2
Your involvement in course (doing assignments, attending classes, etc.) was:	16	44%	19%	19%	12%	6%			6.2

On average, how many hours per week have you spent on this course, including attending classes, doing readings, reviewing notes, writing papers and any other course related work?

Class median: 11.5 Hours per credit: 3.8 (N=16)

Under 2	2-3	4-5	6-7	8-9	10-11	12-13	14-15	16-17	18-19	20-21	22 or more
		6%	19%	19%	6%	31%			12%	6%	

From the total average hours above, how many do you consider were valuable in advancing your education?

Class median: 6.5 Hours per credit: 2.2 (N=16)

Under 2	2-3	4-5	6-7	8-9	10-11	12-13	14-15	16-17	18-19	20-21	22 or more
6%	12%	25%	12%	6%	6%	25%		6%			

What grade do you expect in this course?

Class median: 3.8 (N=16)

A (3.9-4.0)	A- (3.5-3.8)	B+ (3.2-3.4)	B (2.9-3.1)	B- (2.5-2.8)	C+ (2.2-2.4)	C (1.9-2.1)	C- (1.5-1.8)	D+ (1.2-1.4)	D (0.9-1.1)	D- (0.7-0.8)	E (0.0)	Pass	Credit	No Credit
50%	12%	6%	25%	6%										

In regard to your academic program, is this course best described as:

(N=16)

In your major	A core/distribution requirement	An elective	In your minor	A program requirement	Other
75%	6%	6%		12%	

© 2011–2018 IASystem, University of Washington

STANDARD FORMATIVE ITEMS

	N	Excellent (5)	Very Good (4)	Good (3)	Fair (2)	Poor (1)	Very Poor (0)	Median	Relative Rank
Course organization was:	16	12%	50%	12%	19%	6%		3.8	5
Explanations by instructor were:	16	6%	44%	19%	25%	6%		3.5	9
Instructor's ability to present alternative explanations when needed was:	16	12%	50%	19%		19%		3.8	6
Instructor's use of examples and illustrations was:	16	6%	31%	38%	12%	12%		3.2	16
Quality of questions or problems raised by the instructor was:	16	12%	25%	56%			6%	3.3	14
Student confidence in instructor's knowledge was:	16	25%	44%	25%		6%		3.9	11
Instructor's enthusiasm was:	16	50%	31%	12%	6%			4.5	3
Encouragement given students to express themselves was:	16	50%	38%	12%				4.5	1
Answers to student questions were:	16	19%	25%	31%	12%	12%		3.3	13
Availability of extra help when needed was:	16	31%	31%	25%	12%			3.9	7
Use of class time was:	16	31%	19%	25%	19%		6%	3.5	12
Instructor's interest in whether students learned was:	16	25%	38%	31%	6%			3.8	10
Amount you learned in the course was:	16	31%	44%	19%		6%		4.1	4
Relevance and usefulness of course content were:	16	44%	31%	19%		6%		4.3	2
Evaluative and grading techniques (tests, papers, projects, etc.) were:	16	31%	25%	25%	12%		6%	3.8	8
Reasonableness of assigned work was:	16	6%	19%	44%	12%	12%	6%	2.9	17
Clarity of student responsibilities and requirements was:	16	6%	31%	31%	19%	12%		3.1	15

© 2011–2018 IASystem, University of Washington

STA 418 02, Joint with STA 518 02
Computing and Graphics with R
Course type: Face-to-Face

Evaluation Delivery: Online
Evaluation Form: A3
Responses: 16/24 (67% high)

Taught by: Bradford Dykes
Instructor Evaluated: Bradford Dykes-Assist Prof

STANDARD OPEN-ENDED QUESTIONS

Was this class intellectually stimulating? Did it stretch your thinking? Why or why not?

1. The class content was very intellectually stimulating. It presented a challenge and covered many useful and interesting topics.

3. This class stretched my thinking by pushing me to solve problems in a new language. Beyond learning R, we practiced collaboration and learned to use tools like GitHub. Learning R in this context made me feel like I was learning it in its most powerful and realistic form. Now I have an idea of its strengths and weaknesses and could use effectively it in a workplace scenario as I see fit. I am very glad to have it in my data science toolkit.

4. It did in that it was a coding language that I have never used before. Also, the midterms and final were very challenging which required a lot of thought and research into alternative methods.

5. Very stimulating, challenged me to think about data, and programming data in a much different way than before. Tidy programming was very helpful to learn and made programming in this course and R way more intuitive.

6. This class was very interesting learning how to use the computer to manipulate our data, and do plenty of things with graphics and all that other cool stuff we learned with R. The professor was also great with helping us learn in class though. Hands down favorite stats professor at Grand valley.

8. The assigned text book was really useful and I learnt an enormous amount.

9. This class was very engaging. Since class time was so hands on it was definitely very stimulating.

10. Not really, the class was more to make us comfortable with R

11. As an R user, it is always good to go over the basics to pick up on topics you missed the first time. The new context of ideas after a little while can really be eye opening.

12. Yes, I started from scratch using RStudio and Github at the beginning of the class and it felt like sink or swim. I felt that learning R was both very challenging and rewarding.

13. Yeah, it introduced a variety of skills that I'll likely use in the future. There was a good selection of topics, from the Tidyverse stuff to web scraping to Shiny apps.

What aspects of this class contributed most to your learning?

1. The first midterm, and the homework assignments.

3. What contributed most to my learning was the dedication of class time to group work. The hands-on and collaborative nature of the tasks made the class very engaging, and it encouraged me to come prepared. I also liked the approach to grading. Even though it was kind of hard to understand, I think the mastery-based system was implemented very effectively. The feedback was very valuable, especially since we were able to use it to improve our submission. The feedback was also a big motivating factor for me because it demonstrated how much the professor cared about our learning.

4. I think the application tasks were helpful and most of the in class activities. I also liked that we were able to resubmit assignments.

5. The in class sessions were useful and engaging, finding the applicabilities of this course to other aspects was useful too. Really enjoyed that we got the ability to take this into the outside world too like I feel like I could start a blog and a solid professional base with the knowledge I have learned in this class.

6. The best part for me was the primers I would say. I liked to be able to go through and look at examples to prepare for class.

7. My group, I don't think I would have learned much of anything in this course if it weren't for my group mates explaining terms and code. I had no experience of R coming into this class. I think this class is tailored to people who have experience with this language. Google also contributed to my learning some during the midterms and assignments.

8. I loved the flipped classroom approach.

9. Class activities!

10. The assignments

11. The take home applications. These really were thought provoking and allowed me time to walk away from the problem and revisit.

12. The individual assignments. This is where I had to spend the most time researching and learning new techniques of getting the code to work as I wanted. I also liked the group aspect of the class and the grading style.

13. Working through the activities in class.

What aspects of this class detracted from your learning?

1. Preview activities, 2nd midterm.

© 2011–2018 IASystem, University of Washington

3. What detracted from my learning was when we had too much to do for the activities in class. This pushed my team to divide up the work and rush through instead of taking our time to learn the content. I suppose this holds true for the individual assignments too - sometimes I simply didn't have the time in the day to get the full value this course had to offer. On one hand, you get out what you put in and this class can have a very high value; but on the other, it is frustrating that a three-credit class takes six credits of effort. If I were an undergrad, especially learning R for the first time, I don't think I'd be able to keep up.

4. I think the group activities were difficult when others in the group knew what was going on and how to do it but not everyone did because those that knew how to do it just kept going and those that did not were left behind. I also thought the work load was a lot. As a graduate level class I expected all of the work but it was still overwhelming at times. I think only having 2 weeks for the final project since we had the midterm to do too was difficult to put in the effort and quality that you were looking for in the final. I think better meeting preps would be more helpful as well. I also liked some aspects of the team based learning but I think some lecture and you talking about what it was that we were doing rather than just having us figure it out for ourselves would be beneficial.

5. Sometimes felt a bit rushed trying to get everything done in class. The first midterm I felt like too much time was forced on cleaning data and that wasn't really useful to me learning and felt more like busy work. I get why it is there though because data IRL will absolutely not be clean.

6. I didn't love the reverse classroom thing we did. I felt like I never had a good grasp on the content coming into class, even with pre-class assignments we had to prepare us. I'm not sure how else it could be taught though, but this felt like a hard way to learn it. I also felt that the graduate students made me feel stupid when they were blowing through assignments, or talking about how easy it was for them to do things. if possible maybe keep the undergrad and graduate students separate later in the future if they have enough students.

7. Class time usage, professor and assignments. This was a very frustrating class. I had no prior experience with R coming into the class. I didn't feel like anything was ever really taught to us. We sat in class and worked through activities, but there was no teaching/learning prior to doing the activities, so I had no knowledge base to work through the activities. I think the class was tailored to those who had experience with R prior to the class. I only learned based on my group teaching me things, which, as you can imagine, was quite frustrating (or googling code). Of note, I was very excited for this class and really wanted to gain a lot from it because I know use of R is growing in the biostats/research community. I've also taken other coding classes (CIS162 and STA318) in my undergrad and I excelled in both classes. We walked through examples and notes in class, then did activities on our own. So this was a very frustrating class as a whole for me - in terms of the teaching style, grading style, in-class work time, and level of difficulty of the assignments. The number of assignments we had as well as the amount of time it took me (someone with no experience prior to the class) was a little excessive as well. The first midterm took me over 25 hours to do. I definitely think this is related to my inexperience, but its difficult to manage an assignment that large.

8. The group work for in class activities did not promote individual activity or learning. It frequently ended with the most experienced coder doing all the work while other group members messed around with their own code.

9. Sometimes there were so many projects going on it was hard to prioritize.

10. All the Github emails

11. Some of the double task days. It felt like we blew through some of the topics.

12. I did not feel the primers were super helpful. Sometimes the group activities were less effective than they could've been. I felt if I took longer to understand something than some of my other group members did, then I was holding them back and slowing them down. But more often than not, I wasn't understanding something and they were.

13. Nothing to speak of.

What suggestions do you have for improving the class?

1. Please lecture more, provide solutions, or show examples. Even paper handouts of solutions could work. It was very hard never seeing an example provided by the instructor. I had no idea if there was a better way to do things, or if what I was doing was correct. Also, the work load in this class was ridiculous. It was the hardest class I have ever taken in my college career. It basically ruined my life. I like the professor, and I enjoyed learning R. But having 4 or 5 things to turn in so weeks was too much. Don't give homework's during midterms. The preview activities are useless. This class would be excellent if the amount of work was cut down, consider less assignments and no preview activities, and if the professor lectured or provided more examples and structured learning. The group learning was fun, I liked it. But why not throw in lectures to help everyone learn in the way they are used to?

2. I spent too much time just to complete the homework and projects in-class or outside of the class, it was kind of stressful to get a good grade in this class.

3. (1) I think it would be beneficial to go over the midterms in class. From my other programming classes, I've learned that everybody codes differently and there is a lot of value in reading how others solved the same problem. Even though we code together for the activities, this would be different in that we are making direct comparisons between independently developed solutions. (2) We should have a central location/calendar for all the due dates instead of digging through Blackboard and GitHub. (3) We should maybe learn some base R before we jump into the tidyverse. Had I not known some R coming in, I think I would have needed that to get my footing. This foundation material could easily replace some of the later more special use case stuff like web scraping and resume building.

4. A little less group work time and a little more you talking. I feel like I learned mostly how to look things up online, which is helpful, but I feel like I should have learned a little more in class. I also do not think that we were well prepared for the difficulty that the midterms were. But it was a good experience.

5. Feels like the final project was kind of sprung on us last minute, would have been nice to have more time to plan or account for it, receiving the assignment towards the beginning of the semester etc. Maybe could build on the project as we go thorughout the course, e.g. for a shiny app, here is how you can start your analysis and some summary data, okay now here are some graphs, okay now try and put it in a shiny app and work up to it. Otherwise great class and loved learning in this class.

6. More primers!!!! I felt the primers were the best for me to try and understand the content, I also liked the fact that we can just google things for help on assignments and stuff like that. The only thing I would have liked to be improved was maybe throw in some vocab quizzes somehow?? so we are forced to study the important functions and whatnot that we need for the rest of the class. I spent a lot of time looking at things that weren't really necessary or that I didn't use again after one activity.

7. Teach during class time, walk through small activities and provide notes, then once students have a starting place, let them do activities and assignments on their own.

8. The organization of the course is problematic. Assignment deadlines are stored in different location to meeting prep requirements and neither of these are clearly listed on blackboard. If students are unfamiliar with github and do no know how to sign up to mailing lists they will miss out entirely on receiving notifications and meeting tasks. In addition, the deadlines to revise and resubmit work are only communicated in class on a whiteboard. This means students have to monitor at least 4 locations to find out requirements and deadlines. It makes it incredibly difficult to stay on top of what is due when. Most students are familliar with using blackboard as the central hub for assignment submission. While I appreciate the emphasis on learning github, it would be far more effective to issue class communications via blackboard, assign all tasks, meeting prep and assignments on blackboard and list all due dates on blackboard as well. This will vastly simplify the requirements for students to keep track of what's going on.

9. Applications every other week really would help - sometimes the workload was just intense. Also the rubric is a bit stressful in that getting an A hinges on the team project which is the only assignment that does not allow a resubmit and is not entirely my own. So it feels a bit like it doesn't matter how the rest of the class goes if I get a bad group or struggle in the last two weeks. I would recommend a caveat like received 95% Es on all assignments would still get an A even with an S on the team project.

10. Utilize blackboard more. Put all student responsibilities on a single page instead of spread between a few Github repos and blackboard.

11. I think this format is good, but it lends itself to some days from home. We were typically left to our own devices barring any questions, so maybe a hybrid style could be leveraged?

12. More examples of the types of things we did in R and in- class instruction. Also, only ONE midterm. I felt completely overloaded having both of those midterms, along with worrying about turning in other assignments for the course in the same weeks we were working on the midterms. It felt impossible at times and made me not even want to attempt it. Less structured assignments also would have been fun. The final project was great because we go tot be creative in what we did. I would have liked more of that throughout the class.

13. Maybe could've eased us into the GitHub stuff more gradually in the beginning, offered more explanation of the what and why of it all.

© 2011–2018 IASystem, University of Washington

IASystem Course Summary Reports summarize student ratings of a particular course or combination of courses. They provide a rich perspective on student views by reporting responses in three ways: as frequency distributions, average ratings, and either comparative or adjusted ratings. Remember in interpreting results that it is important to keep in mind the number of students who evaluated the course relative to the total course enrollment as shown on the upper right-hand corner of the report.

Frequency distributions. The percentage of students who selected each response choice is displayed for each item. Percentages are based on the number of students who answered the respective item rather than the number of students who evaluated the course because individual item response is optional.

Median ratings. *IASystem* reports average ratings in the form of item medians. Although means are a more familiar type of average than medians, they are less accurate in summarizing student ratings. This is because ratings distributions tend to be strongly skewed. That is, most of the ratings are at the high end of the scale and trail off to the low end.

The median indicates the point on the rating scale at which half of the students selected higher ratings, and half selected lower. Medians are computed to one decimal place by interpolation.[1] In general, higher medians reflect more favorable ratings. To interpret median ratings, compare the value of each median to the respective response scale: *Very Poor, Poor, Fair, Good, Very Good, Excellent (0-5); Never/None/Much Lower, About Half/Average, Always/Great/Much Higher (1-7); Slight, Moderate, Considerable, Extensive (1-4).*

Comparative ratings. *IASystem* provides a normative comparison for each item by reporting the decile rank of the item median. Decile ranks compare the median rating of a particular item to ratings of the same item over the previous two academic years in all classes at the institution and within the college, school, or division. Decile ranks are shown only for items with sufficient normative data.

Decile ranks range from 0 (lowest) to 9 (highest). For all items, higher medians yield higher decile ranks. The 0 decile rank indicates an item median in the lowest 10% of all scores. A decile rank of 1 indicates a median above the bottom 10% and below the top 80%. A decile rank of 9 indicates a median in the top 10% of all scores. Because average ratings tend to be high, a rating of "good" or "average" may have a low decile rank.

Adjusted ratings. Research has shown that student ratings may be somewhat influenced by factors such as class size, expected grade, and reason for enrollment. To correct for this, *IASystem* reports **adjusted medians** for summative items (items #1-4 and their combined global rating) based on regression analyses of ratings over the previous two academic years in all classes at the respective institution. If large classes at the institution tend to be rated lower than small classes, for example, the adjusted medians for large classes will be slightly higher than their unadjusted medians.

When adjusted ratings are displayed for summative items, **relative rank** is displayed for the more specific (formative) items. Rankings serve as a guide in directing instructional improvement efforts. The top ranked items (1, 2, 3, etc.) represent areas that are going well from a student perspective; whereas the bottom ranked items (18, 17, 16, etc.) represent areas in which the instructor may want to make changes. Relative ranks are computed by first standardizing each item (subtracting the overall institutional average from the item rating for the particular course, then dividing by the standard deviation of the ratings across all courses) and then ranking those standardized scores.

Challenge and Engagement Index (CEI). Several *IASystem* items ask students how academically challenging they found the course to be. *IASystem* calculates the average of these items and reports them as a single index. *The Challenge and Engagement Index (CEI)* correlates only modestly with the global rating (median of items 1-4).

Optional Items. Student responses to instructor-supplied items are summarized at the end of the evaluation report. Median responses should be interpreted in light of the specific item text and response scale used (response values 1-6 on paper evaluation forms).

[1] For the specific method, see, for example, Guilford, J.P. (1965). Fundamental statistics in psychology and education. New York: McGraw-Hill Book Company, pp. 49-53.

© 2011–2018 IASystem, University of Washington