

Informatics Student Course Feedback 2017/18

<http://www.inf.ed.ac.uk/teaching/surveys/2017-18>

This report contains feedback from students about a course taught in the School of Informatics during the 2017/18 academic year, in response to the following questions:

- What would you say to students interested in taking this course?
- What did you find most valuable about the course?
- What improvements, if any, would you make to the course?
- Please add any other comments you have about workshops and tutors

Each course organiser receives this report as well as statistics on multiple-choice responses. All these reports, together with student feedback about individual members of teaching staff, are collected and sent to the Director of Learning and Teaching.

Please note that these are personal responses from individual students: some courses only have a few responses and a small sample can be unrepresentative.

Stereotyping and bias, especially unconscious bias, is a serious concern in any survey gathering personal responses. All students received the rubric below before completing the surveys, and you can read a brief introduction to issues of unconscious bias on the university web pages at <http://edin.ac/2iypZBv>

This information is provided for students and staff at the University of Edinburgh: **you may not redistribute or reuse it without permission**. If you would like the information in another format or want to use it in your own publication then please contact the Informatics Teaching Organisation at <http://www.inf.ed.ac.uk/teaching/contact>

Rubric given to all students taking the end-of-course feedback survey

We value your opinions on the courses you take here at the University, as they allow us to shape future delivery and development. We welcome constructive comments about your courses, whether positive or negative, and ask you to give details about any issues in order to help the course organiser to understand and address them.

We encourage you to be aware of the potential for bias in the completion of these questionnaires, so we have developed resources which may be helpful to you:

- Equality, Diversity and Unconscious Bias (<http://edin.ac/2iypZBv>)

You also have a responsibility to provide feedback in a manner which does not breach the University's Dignity and Respect Policy:

- University of Edinburgh Dignity and Respect Policy (<http://edin.ac/1Cq0VZY>)

The results of the questionnaires will never be analysed in a way that seeks to identify individual students from their responses. However, should you wish to remain anonymous, please do not identify yourself in your answers to the survey questionnaire implicitly or explicitly.

Comments Report

What advice would you give to a student taking this course in future?

- Be aware how much coding there is.
- Be aware of what you are getting yourself into - it is a highly demanding course in terms of programming and concepts related to computer science. If you have not taken any programming modules before, and you are uncertain about how much you would enjoy it, do think twice before you choose to take this course.
- Be familiar with computer programming, have a basic understanding how to do it or, have a qualification in computing.
- Be prepared for very difficult coding if you have never done any before. Look at the documentation online and try and get a head start before assignments are released. Go to every help session for assignments as you can to help with these. Do the readings regularly to keep up with the course content, as this also helps with assignments. Get on piazza, it was a life saver
- Be prepared to do independent learning, be prepared to step out of your comfort zone and tackle stuff you've never done before. Don't be scared of the assignments, they are marked kindly. Make study groups to understand lectures and go through them.
- Be prepared to spend lots of time in the lab if you don't have relevant prior experience; seek help in lab sessions.
- Do not take if you have no programming knowledge
- Do the readings and keep up with understanding the lectures - it's easy to fall behind
- Don't take it unless you have already taken computer science.
- Don't worry if you fall behind on the readings - this will happen for sure.
- Don't take this class - if you do prepare to do a lot of work which makes no sense with no support.
- Don't assume all the reading needs to be done thoroughly (please make this clear to students as the class believes there was 150 pages of reading a week). Be aware that the workload is obscenely high (3 lectures, 2 hour labs, 1 hour tutorial each week). The subject matter is very interesting, so engage with it while you can!
- Ensure that you know what you're putting yourself in for, particularly with regard to programming assessments
- If they are a non-informatics student, I would tell them to take a course in python first semester or complete the python course in codacademy.
- If you don't like maths or computation, think twice or even thrice about taking this course. It's extremely complex in the delicate things.
- It's neither for informatics student nor for psychology students. So it's useless.
- Know how to code already
- Make sure you are familiar with coding and computing, as although not a pre-requisite, this is clearly not a course targeted at beginners and there is no guidance for those struggling.
- Make sure you have knowledge on computer coding and jupyter
- Only take the course if you are interested in it, since it is very challenging and requires a lot of reading. The programming part might discourage non-informatics students but don't let it keep you from taking the course as it is very interesting and rewarding in the end.
- Please only take the course if you're genuinely interested
- Pre-existing knowledge of programming is beneficial, it is difficult to learn in such a short amount of time
- They should definitely know how to program before starting this course
- Think about it twice, you really need to be interested in cognitive science and understand more than basic maths (eg partial derivations)

Comments Report

What did you find most valuable about the course?

- Covered a wide range of topics, some of which were very interesting
- Feedback on coursework was very informative.
Tutorial answers were posted on learn very quickly which was beneficial.
Lectures recorded, alongside informative lecture slides.
- Lectures were interesting when accessible
- Lectures with Chris
- Linguistic ideas, inspiration for future research, nice but short introduction to neuronetworks
- Recorded lectures
- The ability it provided me to tackle completely unfamiliar things.
- The coding assignments helped me get a better understanding of how the theory is applied in the real life; moreover, helped us experience low key AI.
- The content is very interesting and often well presented. The introduction to Python (despite being badly taught) has been useful to me beyond uses within the course.
- The content was really interesting, and introduced us to important concepts which will be explored further later in the degree.
- The course content was interesting and valuable.
- The coverage of neural networks
- The fact that lectures have done research on the field
- The interdisciplinary aspect; I learned a lot about how to approach more theoretical subjects and read scientific papers, and it was nice to meet people outside of informatics.
- The lecturer's encouragement of students to approach the course material in a critical manner.
- The lectures in the second half of the course
- The skills in graphing with python are valuable interdisciplinary. The skills in reading papers with strategy, with doing research solo.
- To first get the basic concept of neural network
- it overlaps with psychology and linguistics which helps a lot
- wide range of topics covered; assignments were interesting; lab demonstrators were very keen to help

Comments Report

What improvements, if any, would you make to the course?

- - the course either needs prerequisites of programming background, needs to teach programming, or needs to get rid of programming coursework. Currently it is unrealistic for non-informatics students to be able to cope with the programming and maths parts of the course
 - By contrast there is a lot of support for informatics students for the essay writing part of the course
 - The labs and coursework instructions are extremely hard to understand and often are modified while coursework is going on.
 - There is too little of big picture explanation of topics and too much focus on very specific technical examples. This means different topics feel disjointed and unconnected and we miss a sense of where everything fits in.
- Assignment 1 and 2 were not 'beginner' coding level, especially for those from outside informatics who have no experience of pythons etc. before. The course is advertised in such a way on DRPS that, although it discusses coding, it invites those from out with informatics to enrol, and although it does state that this is a challenging course, the information given regarding the level of difficulty in the coding was not provided.
Either the course should give more information to students before enrolling regarding the level of difficult of the coding aspect of the course, or should give those from no coding experience the opportunity to learn from basics at a reasonable pace instead of being 'thrown in the deep end' like this year.
- Do not advertise it as a course for all students, programming proficiency required cannot be learned in 3 simple lab sessions. For non-informatics students, this course was terribly organised.
- Either make it a prerequisite to have taken some programming module(s) before entering the course, or make the programming part of the course less demanding/provide more support for students who are completely new to programming. Also to make it more clear to students what is going to be expected from them throughout the course, in terms of course content and assignments.
- Everything needs to be improved. The way the classes are taught, I always go out of the lecture not understanding anything. The tutorial questions and the tutorial is absolutely useless because no learning is taken place. The prof need to be more interested and engaging the subject.
- I have no idea what we were supposed to learn. The course is so random it's actually amazing.
- I really disliked this course. The informatics section of the course was not taught and we had to teach ourselves through labs that were full with only two people to answer everyone's questions. I felt very disadvantaged in this course because I had never done any computer science before, and I think informatics students had a much better ability to succeed. Assignments were extremely frustrating because I had no idea what I was doing. After talking with other frustrated students, I think it would be best to teach basic computer science before the labs so there is a ground understanding of how it works. Students should not be graded on things that the professors did not teach at all.
If planning to give us assignments in python, giving us some instruction about how to code in python (other than the labs where the supporting students were not always helpful). Otherwise, have a coding requirement so that non-informatics students can be prepared before starting the course. In other words, making it more non-informatics student friendly if the course is going to be open to them.
- It might be a bit too challenging, and often I couldn't keep up with the readings.
- Make learning outcomes clearer, work on communication with students (like deadlines of assignments)
- Make the learning aims clear, including the relative difficulty of programming assessments for non-Informatics students. An instructions sheet and two hours of contact time with two postgraduate students shared between 50 people makes it very difficult for non-Informatics students to perform well on the first two assessments.
- Make the second half of the course a bit clearer, cover more of neural networks
- More labs on coding before the assignments and more help in the labs, particularly for those who haven't coded before as the assignments very difficult.
- More reaching in programming and better instructions for the assignments in programming.
- N/A
- Put a maths and physics prerequisite on everything.
- So many, I am going to be submitting a separate feedback essay to Vali and my Tutorial leader in hopes that it isn't lost in the spreadsheet.
- That when choosing the course for an elective that they tell students that you do need knowledge on coding when he the course does not say you do need but two of the assignments include coding and this is not explained well for those with no experience
- The course has no pre-requisites, but in reality is out of reach for those who lack a computing background, particularly the labs, assignments 1 and 2 (which are a significant proportion of the grading) and the material on perceptrons/neural networks - there is a complete lack of guidance for those unfamiliar with this area, as it is clearly expected that students have a considerable level of pre-existing knowledge, which is not necessarily the case.
- The course would have been significantly easier had there been an appropriate introduction to python (perhaps in the form of tutorials?). Even as an inf student, I found assignments 1 and 2 cryptic and very difficult to interpret. Notably, assignment 3 was very well structured and introduced.

Comments Report

What improvements, if any, would you make to the course? (continued)

- The readings attached to each lecture were difficult to read and I struggled to have enthusiasm to read a scientific paper when I'm not very familiar with the discipline. Having a clearer understanding of what exactly is examinable.
- There should be maths and com sci pre reqs mandatory for this course. I, along with so many of different backgrounds, struggled a lot with completely foreign content. The first two weeks should be a good representation of the computational part of the course because only then is the student allowed to transfer between courses.
- As a non-informatics student I found this course quite hard, especially with the lab assignments and when the topics relate to the computational aspects. Therefore, I'd suggest making a note of the advanced level of computational programming involved in the course in the course description, so that non-informatics students would be mentally prepared for that. Also, I would appreciate it if more lab help sessions were organised for the lab assignments (like those arranged by Chris Lucas, but it'd be good to organise these sessions well beforehand so that more students could attend). Similarly, I think lab sessions should therefore not end until the 2nd lab assignment is completed, in order to allow time and opportunity for students who are struggling to seek help. As for the lecture content, although the range of topics covered was very wide, I feel that the lecturers were also trying very hard to delve deep into each topic, which made it quite difficult for us to catch up, especially when some topics were quite advanced. The amount of readings was also sometimes overwhelming, especially when the lab assignments were already quite demanding. The connection between lectures were not too clear at times, making it difficult for me to realize at the beginning of the lectures how this particular lecture relates (if it does) to the previous one. IN general, this occasionally made the understanding of lectures hard and confusing - like we had to jump in between different topics back and forth rapidly. I know that quite a few students including myself tend to have to replay the lecture recording for at least once after attending the lecture, as we often could not immediately grasp the ideas introduced in lectures - often times many advanced and specific terms were used that we did not know about before - hard to follow the entire lecture. The general content of lectures were especially difficult when there were computational/mathematical aspects involved, since there was often not enough time to explain those thoroughly during lectures. I would suggest adjusting the weighting of the assignments - perhaps increasing their weighting, as I personally devoted a lot of time into the lab assignments which only counted for 10% each.
- Cut it shorter in volume and in exchange go into more detail on a specific topic. I know it's an introduction course, but it would have been nice to get intellectually involved with some of the topics a bit more deeply. It's always a bit scratching on the surface. Maybe leave out the second part of the course and make the first part much more involving in exchange for that
- The organisation is very confusing, as well as the level required for it (lots of maths and programming for people with little background). The classes were very confusing and it was hard to follow how one class related to the other. The overall aim seemed unclear
- use simple words to explain the lecture

Comments Report

Please add any other comments you have about workshops and tutors

- I liked meeting psychology students and having discussions.
- Labs did not help at all with the second assignment.
- Labs were very dense and not really related to programming assignments
- Mathais is an excellent tutor. The 2 tutorials I had with him were fantastically engaging. Labs are as cryptic as assignments and I did not complete many of them to a high standard as they were unhelpful in grasping concepts.
- N/A
- Not enough tutors for the workshops, also more/better instructions would have been useful.
- See above. A lot of the time the supporting staff and tutors were unprepared or were unaware of what we were learning in the lectures. So more communication across staff.
- The people running the labs did their best to be helpful, but clearly felt out their depth explaining quite difficult material to beginner students who were struggling with fundamental aspects. I would stress that this is not the fault of the tutors, but a flaw in the course content which is often inaccessible.
- There was confusion over tutorial exercises in the second half of the course.
- These ranged from sort of good (running through perceptrons by hand without algorithms) to pointless (discussions about lectures we didn't understand or see). They weren't negative to my learning but they could've been a lot more informative. My tutor for some of the weeks, Mr Villagra, was great and this score does not reflect my opinion of his tutoring.
- Too few tutors in Labs
- Too much reading for a first year course - a whole book for one tutorial!?
- Tutorial arrangement got a bit messy towards the end of the course, making it hard for both tutors and students to be on the same page.
Lab sessions could be more structured, with more detailed and comprehensive instructions distributed to students. Often times other students and myself found it difficult to understand what we are supposed to do in the labs just by looking at the lab session slides ourselves.
- Tutors are always helpful and reply to emails quickly which is very helpful. Lab sessions were beneficial but could have been more structured to give students with less computing experience more teaching and tutoring
- Tutors seemed to be just as uncertain about the course content as the students, in terms of what level and depth of knowledge was expected from the students. Labs were a lot more demanding than may be expected from two scheduled hours each week - it took me, and many others, a lot more time than that to complete the exercise sets.
- Well aware tutors, please. Ours hardly knew what was happening in the course herself.
- While tutorials were helpful, I think the python labs were extremely disappointing and didn't really leave me with much understanding of the language
- the tutorial question could be more