1.What was the hardest assignment for you and why? Did you learn anything from the experience? Is there anything you would change about the assignment to help your learning?

To me, the hardest assessment for the course is Project 2, and the reason for that is because it has lots of SVD analysis needed to be done to the given dataset. Since it's a totally new concept for me. I was expecting it to be challenging. However, the most difficult part came out to be the second half of the project since it's an open-ended questions. I was having trouble to come up with a convincing answer to my proposed questions. And it seems to always be a bit off. In the end, it came out to be just an okay grade received for the assignment. However, at least I tried my best to come up with a proper analysis :D.

I did learn a lot from that experience. Since group project requires more than one person's effort to tackle problems. We were having trouble using Git to communicate with each other. Hence, we just use message to talk with each other and update our code. It's not that efficient. And we had different versions of code in the end that we had to manually edit and fix. So next time when I started a group project with others. I will ensure that our way of communication is efficient and up to date.

Lastly, I would say probably giving some hints about how to come up with questions would be helpful for the second project. And everything else for the course has been fantastic. The teaching staff, grader, and most importantly, the lecturer are all super nice to communicate with.

2.What are two areas in which you think you did well this quarter? What are two areas in which you could have improved?

If I had to say something that is good for me for this quarter. I would say is that I did better in the commenting part. I used to take CSE 142/3 and CSE 163. And I use to had terrible commenting in my code. It often confuses the grader. And for this course, since we have a group project that we “have” to work with our classmates. I would say it’s the single factor that encouraged me to fix my commenting issue. I had to do well-explained comment on what has been changed and what are the things that still needs improvement. And the other area that I did well is answering questions posted on Piazza’s discussion board. I often get confused with many problems, and found out that Piazza/Edstem to be a perfect place to disguise issues and obstacles we had. And it’s always helpful to answer other students’ questions too!

For the areas that I can improve. The first thing would be stop trying to finish homework just before the deadline. For this quarter, my productivity has a negative correlation with the number of hours before the deadline. I don’t think it’s a good practice. Another thing that I didn’t do well enough is attending classes on a weekly basis. I came to class in the first couple of weeks and stop doing so later. I found myself to be a bit lazy to attend lectures, and that indeed hurts my understanding of the class content. Hence, I would like to address these issues for upcoming courses in the future.

3.Is there anything you are glad we covered this quarter? Is there anything you wish had been covered in this course? What programming and computing skills would you like to learn in the future?

Yes! The whole class is super helpful to me. It teaches a lot about R and how to manipulate data. I will be taking other courses like STAT 435 and STAT 391 in the future. Since both course uses R, it will be a smooth transition for me to transfer ideas into actual code that computers could understand.

For the things that I wish we could cover this quarter. Probably more on the ML side, more than just a simple intro to the KNN and regression. Though it's a bit biased since I was aware of the difficulty of taking STAT 435. And I hope to gain more knowledge in this area beforehand.

I would love to learn languages like C/C++ to get substantial programming experience in languages that expose machine characteristics and low-level data representation; explicit memory management; interacting with operating-system services; and cache-aware programing.