Hello everyone, my name is Soo and I will be giving a short video presentation that I worked on this summer vacation. Two teammates and I were developing short courses on machine learning. The courses aim to instruct general people with computational approaches to data-related problems. Which is, most of the contents are rather easy than difficult, but practical.

Our tasks were divided into 3 parts: making slides, building codes for Python and R. My duty was especially focused on constructing codes in R. The examples we used are cited from a book named “An Introductory to Statistical Learning”. It covers the well-known machine learning models and introduces the basic concepts on how to approach the data.

Among many examples in this book, we usually solved the advanced-level exercise problems, dealing with both the actual datasets and generated datasets through simulations. Since the course concentrates on providing hands-on experience for audience, we usually imported the packages to conduct an analysis. However, we also built up codes for relatively easy and essential methodologies. For example, linear regressions and bootstrapping methods are the most basic concepts underlying in this course. Therefore, we would tell how these computations can be done and suggest the algorithms.

The analyses are always followed by a comparisons step. We encourage the audience to fit several different models by tuning parameters and choosing different sets of variables. Then, we would use either cross-validations or graphics to view their performances. Finally, by these outputs, we will decide the best method.

So how are the problems solved? There are at least 2 exercise problems per each chapter. After the theories and inferences are taught from the instructor, we see how the algorithm works.

We would not let the audience make the functions all by themselves. Instead, we provide them with the “ready-made” code, which is nearly complete but missing a few arguments. I put blanks on the parts of the codes that are supposed to be crucial in the function. For instance, in linear regression models, I would leave a blank on where the model equation should be. Another blank would be for the name of the function when making box plots. The complete example codes are given first, and the students can see the result by compiling these by their own. These would be followed by question codes, but now they should solve the problems by filling in the holes. Since the example and the question codes are alike, it should not be too hard to guess what the answer should be. In this step, they can come up with different approaches and fit various models.

The models made from the analysis step must be compared to each other. In order to achieve this, we use cross-validation or data visualization to monitor overall performances. For example, when we are having a discrete variable as a response, we can use the cross-validation table to see how many predictions were made accurately using the validation dataset. The response variable is often continuous, though. Then we can make regressor lines with different options on the same plot. Sometimes, we make the models by arbitrarily choosing specific variables and parameters. We generally use iterations to select sets of predictors and the dimensions of the formulae to boost the process. This would yield at least hundreds of models, and it is nontrivial to check the approaches one by one. Instead, we use graphing devices to select our best option. Likewise, the criteria statistics can be visualized to make it easier for the audience to spot the differences and have luxurious outputs.

The last step is to choose the model. Among the numerous approaches we have tried, we should choose the most feasible and accurate one. For example, we can make regression lines with different degrees of polynomials. It is important that the model does not underfit the data, but also not overfit so that it does not follow that specific data exclusively. We can choose the one that is least complicated and that smoothly fits the trend. Next, they should give an interpretation of the model, and explain the mathematical formula if necessary.

These are the basic flow of fitting the model using R programming. We are excited that the courses would be delivered to many people who are interested in machine learning. Thank you for listening to my presentation, and I appreciate any feedback and questions.