CS259 Quantitative Methods for Computer Science
Lab 6. Applying Linear Algebra: KNN classifier

Assessment
This lab is assessed only formatively: you don't receive a numeric mark. Feedback is provided only if submitted by the deadline. Please note however, that our quizzes may include questions based on the lab instructions, so the best way to prepare for the quiz is to do this lab (even if only partially or submitted late).

You are not required to attend lab sessions, but it is a good opportunity to work on your assignments and get help from the lecturer and the demonstrators.

You can work on this assignment in the groups of your choice no larger than 3 students. But make sure everyone retains the copy of the code and uploads the report to myPlace, since it will be marked individually. You don't need to state the names of the students in the group.

Submission
The report for this lab assignment with your screenshots needs to be uploaded to myPlace before 10am on Tuesday November 12th as a single PDF file.

Instructions

Task 1.
Apply the K-nearest neighbour classifier, as detailed in our lecture slides, to the provided training and testing datasets (with K set at 1). This can be done manually, either on paper or using software like MS Excel. First, apply our Model 1 to the test cases 1, 2 and 3 (as shown in our slides) and calculate their similarities with all the training cases. Determine the winners (nearest neighbours). Repeat that for the models 2 and 3 as well. Make sure your results match those in our slides. Compute the predictions for the fourth testing case from the testing dataset.

Task 2.
Utilize our provided template. Populate the missing sections based on your findings from Task 1, ensuring the assert statements run without issues. Capture screenshots that cover your entire code, as well as the corresponding outputs, and include them in your report. Upload the completed report to myPlace as per the "Submission" section above.
Hint: Java uses 0-based arrays, so remember to subtract 1 from indices referenced in the slides, which are 1-based. For instance, to refer to the 2nd test case in Java, you'd use index 1.