

1. Run the kNN code with $k=2$. Display some of the misclassified digits alongside their nearest neighbours. Can you think of any reason why these digits might be more likely to be misclassified? Is there anything they have in common? Does this suggest any way in which the algorithm might be improved?
2. Try to display two of the classes (e.g. 8s and 3s) in 3D space using the 3D plotting code. Edit the code so that the misclassified digits stand out in the 3D space with respect to the other digits. You might have to use a different plotting symbol to do this.
Do the misclassified digits lie in a particular region? Is there anything different about their location with respect to the main clusters? Does this suggest any way in which the algorithm might be improved?
3. Find the accuracy of the kNN algorithm with different numbers of eigenvectors, where the number n could range from 2 up to 256 in steps of 5 or 10.
What is the shape of the curve? Does it increase steadily? Does it level off or does it have a peak? Can you explain the shape?
How does the execution time change as the number of eigenvectors changes? Is there an optimum number?
4. Try to run the code which selects digits within a box. Try to change the size of the box? How does the accuracy change? How does the execution time change?
5. Try to run the code which uses blurring to select digits within a box. Try to change the size of the box? How does the accuracy change? How does the execution time change?
Try changing the level of blurring? What effect does it have on accuracy? Can you come up with a multi-level strategy with different levels of blurring and different box sizes? Can you improve both the accuracy and execution time?