Selecting and Training a Model

1. Which of these is a more accurate description of a data-centric approach to ML development?

   - Holding the neural network architecture fixed, work to improve the data to do well on the problem.
   - Holding the training data fixed, work to improve your neural network's architecture to do well on the problem.

   That's right! Data-centric means you focus your efforts on improving the data to raise the system's performance, while keeping the code fixed.

2. Say you have an algorithm that diagnoses illnesses from medical X-rays, and achieves high average test set accuracy. What can you now say with high confidence about this algorithm? Check all that apply.

   - It does well even on rare classes of diseases.
   - Its diagnoses are roughly equally accurate on all genders and ethnicities, so we are confident it is not biased against any gender or ethnicity.
   - The system can be safely deployed in a healthcare setting.
   - None of the above.

   High average test set accuracy is a great achievement, but there is more work to be done to ensure the algorithm works well on real-world data, is fair, and performs well on rare classes of diseases.

3. Which of these statements about establishing a baseline are accurate? Check all that apply.

   - Open-source software should not be used to establish a baseline, since the performance of a good open source implementation might be too good and thus too hard to beat.

   - For unstructured data problems, using human-level performance as the baseline can give an estimate of the irreducible error/Bayes error and what performance is reasonable to achieve.

     For most unstructured data problems, human-level performance is a great estimate of Bayes error - an upper limit to your system's potential.

   - It can be established based on an older ML system

     You can establish a baseline using an older system or via a literature or open source search.

- Human level performance (HLP) is generally more effective for establishing a baseline on unstructured data problems (such as images and audio) than structured data problems

  Humans perform well on unstructured data, like making sense of an image or a sound, but we aren't great at making sense of large amounts of structured data.

4. On a speech recognition problem, say you run the sanity-check test of trying to overfit a single training example. You pick a clearly articulated clip of someone saying "Today's weather", and the algorithm fails to fit even this single audio clip, and outputs "_____". What should you do?

   - Create a training set of this example repeated 100 times to force the algorithm to learn to fit this example well.
   - Debug the code/algorithm/hyperparameters to make it pass this sanity-check test first, before moving to larger datasets.
   - Train the algorithm on a larger dataset to help it to fit the data better.
   - Use data augmentation on this one audio clip to make sure the algorithm hears a variety of examples of "today's weather" to fit this phrase better.