Homework #6
Due Su. 5/02

Be sure to show all your work for credit. You must turn in your code as well as output files (**code attached at the end of the report**).

Please generate a report that contains the code and output in a single readable format using Latex.

0. Getting Started

   Set up your work environment for deep learning. There are a number of options you can use to do this:

   - Follow Chapter 2 for machine setup (Get the Data section). You'll need to install Python along with additional modules. Be sure to create an isolated environment. Note that the book uses Keras with Tensorflow. If you have no preference then I would follow the book, however, you could use another framework like PyTorch if you prefer.
   - Use Anaconda Python for setting up your sandbox.
   - Google Colab is convenient and free service with GPU access which makes it highly recommended for most without GPU resources.

   For additional setup notes (Anaconda or Colab), see the Stanford CS321n course instructions. Additionally, if you have a Windows machine but would like to use Linux (without a virtual machine), look into the Windows Linux Subsystem.

   Note: Keras and Tensorflow installation instructions are in Chapter 10 - Implementing MLPS with Keras. You can find the Juptyer notebooks online at

   `https://github.com/ageron/handson-ml2`.

1. (Geron 10.1)

   For each letter sub-problem, take a screenshot to highlight your findings.

2. Building an Image Classifier Using the Sequential API (Ch 10)

   Follow the example classification using Keras with Fashion MNIST (page 297). You should generate the following:

   (a) A snapshot of the training from the `fit()` method as shown on page 303.
   (b) The learning curves as in Figure 10-12.
   (c) Generate the prediction probability, and class names for 3 random test examples (rather than the first three as done in chapter (page 306-307)).

3. Typical CNN Classifier (Ch 14)

   Follow the example to implement a simple CNN (page 461) to tackle the Fashion MNIST dataset. Like the previous problem, you should generate the following:

   (a) A snapshot of the training from the `fit()` method.
   (b) The learning curves as in Figure 10-12.
   (c) Generate the prediction probability, and class names for 3 random test examples (use the same random seed as in the previous problem).

4. Pretrained CNN Model and Transfer Learning (Ch 14)

   Follow the example to use a ImageNet pretrained model (ResNet-50) for the Fashion MNIST dataset.

   (a) Using the pretrained model, give the top-3 ImageNet prediction results on your 3 random test examples (same seed).

   (b) Perform transfer learning and give the test accuracy.

   (c) Give the top-3 prediction results on the 3 random test examples after transfer learning.